

INT J COMPUT COMMUN, ISSN 1841-9836
9(3):340-347, June, 2014.

An Efficient Solution for the VRP by Using a Hybrid Elite Ant System

M. Yousefikhoshbakht, F. Didehvar, F. Rahmati

Majid Yousefikhoshbakht, Farzad Didehvar*, Farhad Rahmati

Department of Mathematics and Computer Science

Amirkabir University of Technology

No.424, Hafez Avenue, Tehran 15914, Iran

e-mail: khoshbakht@aut.ac.ir, frahmami@aut.ac.ir

*Corresponding author: didehvar@aut.ac.ir

Abstract: The vehicle routing problem (VRP) is a well-known NP-Hard problem in operation research which has drawn enormous interest from many researchers during the last decades because of its vital role in planning of distribution systems and logistics. This article presents a modified version of the elite ant system (EAS) algorithm called HEAS for solving the VRP. The new version mixed with insert and swap algorithms utilizes an effective criterion for escaping from the local optimum points. In contrast to the classical EAS, the proposed algorithm uses only a global updating which will increase pheromone on the edges of the best (i.e. the shortest) route and will at the same time decrease the amount of pheromone on the edges of the worst (i.e. the longest) route. The proposed algorithm was tested using fourteen instances available from the literature and their results were compared with other well-known meta-heuristic algorithms. Results show that the suggested approach is quite effective as it provides solutions which are competitive with the best known algorithms in the literature.

Keywords: Vehicle Routing Problem (VRP), elite ant system, , global and local updating, NP-hard problems.

1 Introduction

The vehicle routing problem (VRP) is one of the most important combinatorial optimization problems that was first defined by Dantzig and Ramser more than 50 years ago [1]. The VRP involves designing a set of vehicle routes each of which starts and ends at a depot. These routes are used for a fleet of vehicles which provide services for a set of customers with known demands. Each customer is visited by exactly one vehicle only once and the total demand of a route does not exceed the capacity of the vehicle type assigned to it. The objective is to minimize the total distance traveled by all the vehicles. To make VRP models more realistic and applicable, there are various forms of the VRP obtained by adding constraints to the basic model. Examples of such extensions are VRP with pickup and delivery (if the vehicles need to pick up and delivery) [2], VRP with time windows (if the services have time constraint) [3], heterogeneous fleet OVRP (if the capacity of vehicles in OVRP are different) [4], VRP with backhauls (if the customers with delivery demand have to be visited before by the customers with a pickup demand) [5] and generalized VRP (if the customers are partitioned into clusters with given demands such that exactly one customer from each cluster should be visited) [6].

The VRP solution methods fall into three main categories: exact methods, heuristic and meta-heuristic algorithms. Exact approaches for solving the VRP are successfully used only for relatively small problem sizes but they can guarantee optimality based on different techniques. These techniques use algorithms that generate both a lower and an upper bound on the true minimum value of the problem instance. If the upper and lower bound coincide, a proof of optimality is achieved. Branch-and-bound [7] and branch-and-cut [8] are two of exact methods which have

been proposed for VRP by researchers.

Since this problem is known to be NP-hard (Non-deterministic Polynomial-time Hard) problem [1], exact algorithms are not often suitable for real instances because of the computational time required to obtain an optimal solution. Therefore, in the past forty years, researchers have developed the heuristic algorithms using a permissible solution instead of the optimal solution. There are many celebrated algorithms in this class such as savings heuristic of Clarke and Wright [9] that gains a single route instead of two routes according to the savings obtained by this merger. The sweep algorithm is another famous construction heuristic that is proposed by Gillett and Miller [10].

A new kind of algorithm which basically tries to combine basic heuristic methods in higher level frameworks aimed at efficiently exploring a search space is meta-heuristics. Since the meta-heuristic approaches are very efficient for escaping from local optimum, they are one of the best group algorithms for solving combinatorial optimization problems. Some of the most popular metaheuristics applied to the VRP are simulated annealing (SA) [11], genetic algorithm (GA) [12], tabu search (TS) [11], Computational Intelligence Approach [13], large neighborhood search [3], ant colony optimization (ACO) [14], hybrid ant colony optimization [15] and particle swarm optimization [16].

The VRP is intrinsically a multiple objective optimization problem (MOP) in nature that has received much attention because of its practical application in industrial and service problems [17]. On the other hand, there are few research studies which have used the ACO in order to solve VRP. Furthermore, the elite ant system (EAS) is one of the most important and powerful versions of ACO that nowadays is applied on a lot of combinatorial optimization problems [18]. Therefore, this paper proposes a hybrid EAS (HEAS) mixed with insert and swap algorithms for solving the VRP.

The remaining parts of the paper are organized as follows. Section 2 describes the EAS, and the proposed algorithm. Section 3 describes computational experiments carried out to investigate the performance of the proposed algorithm. Finally, section 4 presents the results of the conclusions and future works.

2 Our Algorithm

In this section, first, the EAS are presented and then the proposed algorithm will be analyzed in more detail.

2.1 Elite Ant System

The first modification to ant system (AS) which was conducted by Dorigo and his colleagues in 1996 was using elite strategy in EAS [19]. The decision for choosing the unvisited N_i node by ant k located in node i is made based on formula (1) where τ_{ij} indicates the amount of pheromone on (i, j) edge while η_{ij} shows inverse distance between i and j . However, both are powered by α and β which can be changed by the user. Therefore, their relative importance can be altered.

$$P_{ij}^k = \begin{cases} \frac{\tau_{ij}^\alpha \eta_{ij}^\beta}{\sum_{j \in N_i} \tau_{ij}^\alpha \eta_{ij}^\beta} & \text{if } j \in N_i \\ 0 & \text{if } j \notin N_i \end{cases} \quad (1)$$

In the EAS, in addition to depositing pheromone on all local edges, in each iteration pheromone is released on the edges of the best path. Imagine T^{gb} is the best path gained after the algorithm is completed, T_k is total length of edges traversed by ant k and ρ is the rate of the pheromone evaporation in order to prevent rapid convergence of ants to a sub-optimal path.

While pheromone is being updated, the edges marched by the ant which have constructed the T^{gb} path absorb an additional amount of pheromone which is equal to $e/L^{gb}(t)$. The formula for pheromone updating can be written in (2). In this way, the edges of the shortest path up to the current iteration become more attractive and are updated based on the value of the best $L^{gb}(t)$ tour. As it is shown, $1/L^k(t)$ is the formula for the local trail updating. While traversing between nodes i and j , ants release pheromone on the respective edge which is equal to the inverse of the cost of the tour $L^k(t)$ taken by ants.

$$\tau_{ij}(t+1) = (1 - \rho) \cdot \tau_{ij}(t) + \sum_{k=1}^m 1/L^k(t) + e/L^{gb}(t) \quad (2)$$

2.2 The Proposed Algorithm

In EAS, local pheromone release is not a good guide for finding the best route because there might be some edges which belong to no best route in none of the iterations but pheromone is still deposited on them in each iteration. Therefore, local pheromone release must be abandoned so that the ants resort to global pheromone release in finding new solutions. This attracts the attention of ants to the edges which belong to the best route ever found. Furthermore, the accuracy of solutions in EAS is low at the beginning but it increases with the iterations of the algorithm and pheromone release. Therefore, constant e coefficient cannot be a suitable formula for encouraging the best path found ever because it is not important when and with what accuracy the best solution was found. To improve the mentioned shortcomings, the HEAS has made some changes to the EAS as follows:

Route Construction

In the original EAS, an ant, say k , moves from the present node i to the next node j according to the state transition rule given by formula (3). Here we define μ_{ij} as the savings of combining two nodes on one tour as opposed to serving them on two different tours. The savings of combining any two customers i and j are computed as $\mu_{ij} = d_{i0} + d_{0j} - d_{ij}$ where d_{ij} denotes the distance between nodes i and j , and node 0 is the depot. Furthermore, λ like α and β is control parameter.

$$P_{ij}^k = \begin{cases} \frac{\tau_{ij}^\alpha \eta_{ij}^\beta \mu_{ij}^\lambda}{\sum_{j \in N_i} \tau_{ij}^\alpha \eta_{ij}^\beta \mu_{ij}^\lambda} & \text{if } j \in N_i \\ 0 & \text{if } j \notin N_i \end{cases} \quad (3)$$

Pheromone Updating

In EAS, the pheromone of all edges belonging to the routes obtained by ants will be updated. The pheromone updating includes local and global updating rules. The pheromone updating formula was meant to simulate the change in the amount of pheromone due to both the addition of new pheromone deposited by ants on the visited edges and the pheromone evaporation. In HEAS, not only does the pheromone release increases pheromone on the edges of the best solution in each iteration but also the pheromone release reduction is used in order to distract ants from the edges of the worst solution. The idea of the elitist strategy in the context of the HEAS is to give extra emphasis to the best and the worst paths found so far after every iteration. The value of coefficient e in pheromone increase and pheromone decrease which is shown as $-e$ is a function. It was found out that using polynomial k^2 can generate better solutions for the algorithm. In k^2 , k is an integer number whose value is 1 at the beginning and increases one unit. Moreover, this polynomial is considered an appropriate function since it is not only an ascending function

but also has variable slope as well. In other words, the function increases and the best routes are encouraged and strengthened compared with the previous paths. It should be noted that the low value of the function at the beginning causes the pheromone released on the best route to have less impact on the route selection in the following iterations and ants demonstrate a kind of forgetting in their search. In other words, this helps the ants to forget the weak solutions found at the start of the algorithm. However, as the algorithm is iterated and the accuracy of the solutions increases, the value of the polynomial rises rapidly and the edges belonging to the best solution absorb more pheromone and the edges of the worst solution lose more pheromone. Finally, in order to prevent the edges of the worst solution from gaining a negative value, a minimum value which is equal to the half of the initial pheromone is considered for the edges so that when the amount of pheromone on the edges falls below this minimum, it is replaced with the minimum value.

Local Search

A local search approach starts with an initial solution and searches within neighborhoods for better solutions. Abundant literature on ACO indicates that a promising approach for obtaining high-quality solutions can only be obtained through coupling ACO with a local search algorithm. Therefore, in the HEAS, after the ants have constructed their solutions and before the pheromone is locally updated, each ant's solution is improved by applying a local search. Because local search is a time-consuming procedure, only a local search is applied to the iteration's best solution. The idea here is that better solutions may have better chance to find a global optimum. In the proposed algorithm, at first a local search based on insert move and then the swap move is applied to the ant. In insert algorithm a customer is moved to another route. However, in swap algorithm a customer in a certain route is swapped with another customer from a different route. It should be noted that the new solution will be accepted only if first, VRP constraints are not violated especially about each vehicle's capacity and second, a novel solution will gain a better value for a problem than the previous solution.

3 Results

The HEAS was coded in Matlab 7. All the experiments were implemented on a PC with Pentium 4 at 2.4GHZ and 2GB RAM and Windows XP Home Basic Operating system. Because the proposed HEAS approach is a meta-heuristic algorithm, the results are reported for ten independent runs and in each run the algorithm was iterated n times. Furthermore, the pack of optional parameters obtained through several tests is $\alpha = 1, \beta = 4, \lambda = 3, \rho = 0.2$.

The performance of the proposed algorithm was tested on a set of 14 benchmark instances designed by Christofides et al. which have been widely used as benchmarks in order to compare the ability of proposed HEAS to the results of six meta-heuristic algorithms including SA and TS [11], genetic algorithm (GA) [12], scatter search algorithm combined by ACO (SS-ACO) [14], particle swarm intelligent (PSO) [16] and genetic algorithm combined with particle swarm intelligent (GAPSO) [20]. The information of the 14 instances is shown in Table 1. In this table, n , m , HEAS, mHEAS, BKS, and PD are the number of customers, the number of vehicles of best known solution (BKS), the best solutions found by HEAS algorithm, the used vehicles of HEAS, the best known solution, and the percentage deviation of HEAS compared to the best know solutions (BKSs) respectively. The PD is computed by formula (4) where $c(s^{**})$ is the best solution found by our algorithm for a given instance, and $c(s^*)$ is the overall BKS for the same

instance on the Web. A zero PD indicates that the BKS is found by the algorithm.

$$PD = \frac{c(s^*) - c(s^{**})}{c(s^*)} \times 100 \quad (4)$$

Table 1: Computational results for standard VRP problems

Instance	n	m	SA	TS	GA	SS-ACO	PSO	GAPSO	HEAS	mHEAS	PD	BKS
C1	50	5	528	524	524.61	524.61	524.61	524.61	524.61	5	0	524.61
C2	75	10	838	844	849.77	835.26	844.42	835.26	847.14	10	-1.42	835.26
C3	100	8	829	835	840.72	830.14	829.40	826.14	712.36	8	13.77	826.14
C4	150	12	1058	1052	1055.85	1038.20	1048.89	1028.42	1066.89	12	-3.74	1028.42
C5	199	17	1376	1354	1378.73	1307.18	1323.89	1294.21	1311.35	17	-1.54	1291.45
C6	50	6	555	555	560.29	559.12	555.43	555.43	555.43	6	0	555.43
C7	75	11	909	913	914.13	912.68	917.68	909.68	909.68	11	0	909.68
C8	100	9	866	866	872.82	869.34	867.01	865.94	865.94	9	0	865.94
C9	150	14	1164	1188	1193.05	1179.4	1181.14	1163.41	1162.89	14	-0.03	1162.55
C10	199	18	1418	1422	1483.06	1410.26	1428.46	1397.51	1404.75	18	-0.64	1395.85
C11	120	7	1176	1042	1060.24	1044.12	1051.87	1042.11	1042.11	7	0	1042.11
C12	100	10	826	819	877.8	824.31	819.56	819.56	840.64	10	-2.57	819.56
C13	120	11	1545	1547	1562.25	1556.52	1546.20	1544.57	1545.93	11	-0.31	1541.14
C14	100	11	890	866	872.34	870.26	866.37	866.37	866.37	11	0	866.37

As can be seen from this table, the proposed algorithm finds the optimal solution for 6 out of 14 problems that are published in the literature. The results indicate that HEAS is a competitive approach compared to the BKSs. Furthermore, one new best known solution of the benchmark problem including C3 is also improved by the proposed method. For instances C4 and C12, the gap is about as high as 3%. However, in most of the instances, the proposed algorithm finds nearly the BKSs and for overall the average difference is 0.25%.

As the results in table 1 indicate, the GA has not been able to find the best solutions in thirteen of the fourteen examples. Therefore, it is considered to be the weakest algorithm among the seven presented algorithms. However, SS-ACO has been able to find better solutions than the GA and has come up with the best solutions in 12 examples. Among remaining 5 algorithms, in 11 examples SA has not been almost capable of finding the BKS. PSO has failed in improving the solutions in 10 examples and has come up with solutions similar to the ones found by SA. Hence, it can be concluded that TS is more efficient than GA, SS-ACO, PSO and SA in finding better solutions. From the comparison between GAPSO and HEAS, it can be seen that GAPSO in four examples has been able to find solutions with a gap of less than 1 percent. Despite being able to find the best solution ever found for ten examples, it has failed to achieve these results in the remaining four examples. However, HEAS has found better solutions than GAPSO for the two examples.

A simple criterion to measure the efficiency and the quality of an algorithm is to compute the average of solutions on specific benchmark instances. In figure 1, the average of each algorithm's solution is reported. From this table we conclude that the HEAS method has the best average with 975.44 and has been able to escape local optimum points. The algorithms in terms of their performance from the worst to the best can be listed as: GA, SA, TS, PSO, SS-ACO, GAPSO and HEAS. In addition, in order to demonstrate the efficiency of the proposed algorithm, two of the solutions found for the examples in table 1 are presented in Figure 2. It should be noted

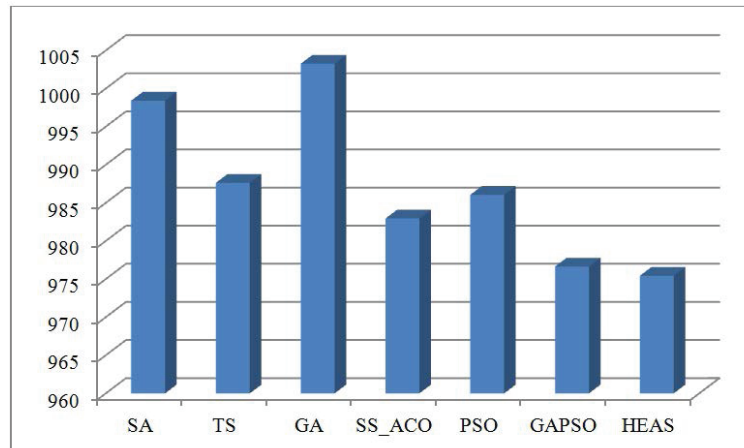


Figure 1: Comparison of mean for 14 instances between meta-heuristic algorithms in Table 1

that in 1 out of 2 examples presented in this figure, the HEAS has been able to improve the best solution ever found.

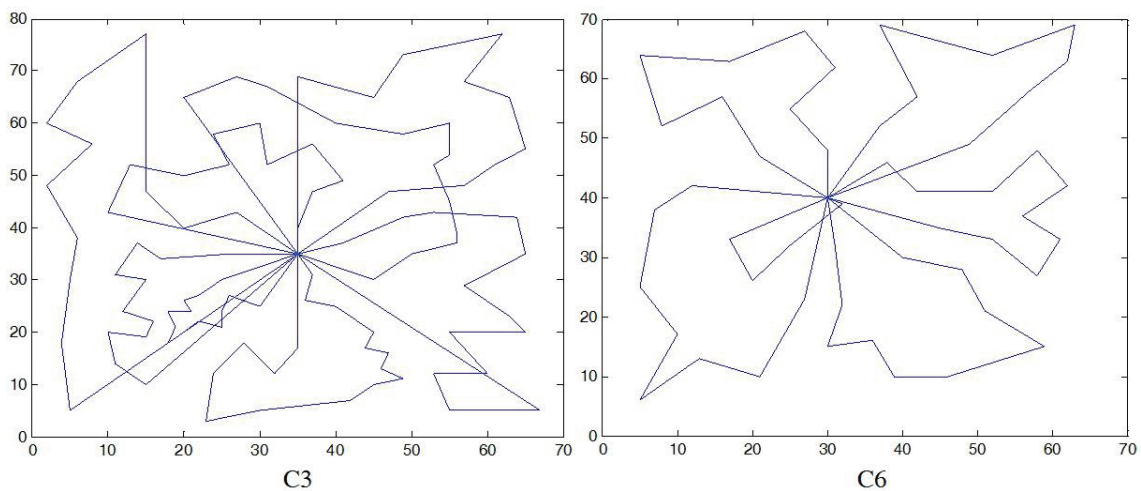


Figure 2: Some of the Solutions to the VRP Found by the proposed algorithm

4 Conclusion and Future Works

In this paper, a modified version of EAS which employs several effective modifications was presented. The modifications improved the performance of the classic EAS algorithm in escaping from local optimum points and finding better solutions in comparison with the other metaheuristic algorithms. It seems that combining the proposed algorithm with other metaheuristic algorithms like tabu search and making use of strong local algorithms like lin-kernigan algorithm can bring better results for the HEAS. Furthermore, HEAS can be used for other versions of VRP like OVRP and heterogeneous fixed fleet OVRP. Future projects will focus on working on such ideas and making them operational.

Bibliography

- [1] Yousefikhoshbakht, M. and Khorram, E. (2012). Solving the Vehicle Routing Problem by a Hybrid Meta-Heuristic Algorithm, *Journal of Industrial Engineering International*, 8(12):1-9.
- [2] Yousefikhoshbakht, M., Didehvar, F. and Rahmati, F. (2014). A Combination of Modified Tabu Search and Elite Ant System to Solve the Vehicle Routing Problem with Simultaneous Pickup and Delivery, *Journal of Industrial and Production Engineering*, 31(2): 65-75.
- [3] Hong, L. (2012). An improved LNS algorithm for real-time vehicle routing problem with time windows, *Computers and Operations Research*, 39(2):151-163.
- [4] Yousefikhoshbakht, M., F. Didehvar, and F. Rahmati, (2013). Solving the heterogeneous fixed fleet open vehicle routing problem by a combined metaheuristic algorithm, *International Journal of Production Research*, <http://dx.doi.org/10.1080/00207543.2013.855337>.
- [5] Toth, P. and Vigo, D. (1997). An exact algorithm for the vehicle routing problem with backhauls, *Transportation Science*, 31:372-385.
- [6] Pop P.C., Sitar C.P., Zelina I., Lupse V. and Chira C. (2011). ,Heuristic Algorithms for Solving the Generalized Vehicle Routing Problem, *International Journal of Computers Communications & Control*, ISSN 1841-9836, 6(1): 158-165.
- [7] Valle, C. A., Martinez, L. C. and Cunha, A. S., Mateus, G. R. (2011). Heuristic and exact algorithms for a min-max selective vehicle routing problem, *Computers and Operations Research*, 38(7):1054-1065.
- [8] Brad, J., Kontoravdis, G. and Yu, G. (2002). A Branch-and-Cut Procedure for the Vehicle Routing Problem with Time Windows. *Transportation Science*, 36(2):250-269.
- [9] Clarke, G. and Wright, J. W. (1964). Scheduling of vehicles from a central depot to a number of delivery points. *Operations Research*, 12: 568-581.
- [10] Gillett, B. E. and Miller, L. R. (1974). A heuristic algorithm for the vehicle dispatch problem. *Operations Research*, 22: 340-349.
- [11] Osman, I. (1993). Metastrategy simulated annealing and Tabu search algorithms for the vehicle routing problem, *Operations Research*, 41: 421-451.
- [12] Baker, B., and Ayechev, M. (2003). A genetic algorithm for the vehicle routing problem, *Computers and Operations Research*, 30: 787-800.
- [13] Hirota, K., Dong, F. and Chen, K. (2006), A Computational Intelligence Approach to VRSDP (Vehicle Routing, Scheduling, and Dispatching Problems), *International Journal of Computers Communications & Control*, ISSN 1841-9836, Suppl. issue, 1(S): 53-60,
- [14] Zhang, X. and Tang, L. (2009). A new hybrid ant colony optimization algorithm for the vehicle routing problem, *Pattern Recognition Letters*, 30(152): 848-855.
- [15] Negulescu S.C., Kifor C.V. and Oprean C. (2008). Ant Colony Solving Multiple Constraints Problem: Vehicle Route Allocation , *International Journal of Computers Communications & Control*, ISSN 1841-9836, 3(4):366-373.

-
- [16] Ai, J., Kachitvichyanukul, V. (2009). Particle swarm optimization and two solution representations for solving the capacitated vehicle routing problem, *Computers and Industrial Engineering*, 56:380-387.
- [17] Pinteá C.-M., Chira C., Dumitrescu D. and Pop P.C. (2011). Sensitive Ants in Solving the Generalized Vehicle Routing Problem, *International Journal of Computers Communications & Control*, ISSN 1841-9836, 6(4):734-741.
- [18] Yousefikhoshbakht, M., Didehvar, F. and Rahmati, F. (2013). Modification of the Ant Colony Optimization for Solving the Multiple Traveling Salesman Problem, *Romanian Journal of Information Science and Technology*, 16(1):65-80.
- [19] Yousefikhoshbakht, M. and Sedighpour, M. (2012). A Combination of Sweep Algorithm and Elite Ant Colony Optimization for Solving the Multiple Traveling Salesman Problem, *Proceedings of the Romanian academy, A*, 13(4):295-302.
- [20] Marinakis, Y. and Marinaki, M. (2010). A hybrid genetic-particle swarm optimization algorithm for the vehicle routing problem, *Expert Systems with Applications*, 37(33):1146-1455.