March. 2015. Vol. 6. No. 3 ISSN2305-8269
International Journal of Engineering and Applied Sciences

www.eaas-journal.org

# ANT COLONY SYSTEM METHOD TO VEHICLE ROUTING PROBLEMS

**AMIN AHMADI[1]; AHMAD FOROOZANTABAR[2]**

[1] Department of Electrical Eng, Mehriz Branch, Islamic Azad University, Faculty of Engineering, Mehriz, Iran.
[2]Faculty of Electrical Engineering, Islamic Azad University, Fars Science and Research Branch, Fars, Iran.

m.a.ahmadi870@gmail.com[1]; a.forouzantabar@srbiau.ac.ir[2]

**ABSTRACT:**

*In this paper we want by use of ant Colony Optimization (ACO) that is one of the meta-heuristic methods that constructs solutions of hard combinatorial optimization problems. In this desertion, a suitable algorithm is presented to minimize logistic costs in supply chain downstream and in distribution network's chain. Suggested algorithm is based on Ant Colony System and searched two goals with finding paths with the minimum number of vehicles and minimum time of costumers, service. This algorithm has been implemented on a distribution network company producing dairy products with 75 clients. Initially vehicle routing problem of this company has been solved with used nearest neighborhood heuristic algorithm And its result is compared with the proposed algorithm. The results show that the response of proposed algorithm is more efficient from the response of nearest neighborhood algorithm. The proposed algorithm is fully applicable and it has this capability that is implemented on the supply chain network in our country and thereby to achieve reducing supply chain logistic cost and ultimately reducing product's final price.*

***Keywords: Ant colony optimization; Vehicle routing; Cyclic transfer; combinatorial optimization; Optimization.***

## INTRODUCTION

The problem of transportation of people, goods or information is commonly denoted as routing problem. As the routing problem has wide areas of application, optimization of the transportation has become an important issue.
The basic routing problem is the Traveling Salesman Problem (TSP). The TSP is the problem of finding a minimal length closed tour that visits all cities of a given set exactly once. The Vehicle Routing Problem (VRP) is the TSP with *m* vehicles where a demand is associated with each city and the system has various constraints. VRP was first formulated by Dantzig and Ramser in 1959.
The problem can be defined as the design of a set of minimum-cost vehicle routes, originating and terminating at a central depot, for a fleet of vehicles that services a set of customers with known demand [1].

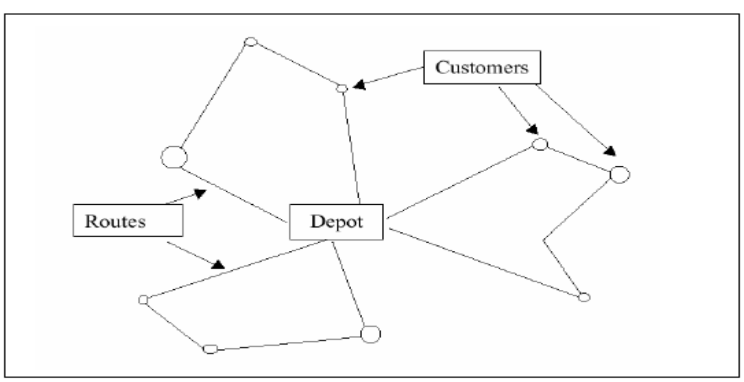


Fig. 1. General representation of the Vehicle Routing Problem

March. 2015. Vol. 6. No. 3 ISSN2305-8269
International Journal of Engineering and Applied Sciences

www.eaas-journal.org

The VRP can be defined as a complete graph G = (V,A) where V = {0, . . .,n} is a vertex set, and A = $\{(I,j) \mid I,j \in V\}$ is an edge set. Vertex 0 is a central depot, while each other vertex represents a customer with a known nonnegative demand to be delivered. The distance dij is associated with each edge (i, j) $\in$ A and represents the distance from customer i to customer j. Customers geographically scattered are visited by a homogeneous fleet of vehicles with limited capacity Q and initially located at the central depot, and routes are assumed to start and end at the depot. The objective is to minimize total traveled distance and the number of vehicles, such that each customer is visited only once by a single vehicle, and vehicle capacity and the total distance must not be violated. The VRP is clearly NP-hard combinatorial optimization problem and difficult to solve. There have been important advances in the development of exact and approximate algorithms. Exact solution methods can only be used for very small instances, so for real-world problems, researchers have to rely on and resort to approximate or heuristic methods in solving the problem.

There have been many papers proposing exact algorithms for solving the VRP. These algorithms are based on dynamic programming, Lagrangean relaxation, and column generation. On the other hand, as the VRP is known to be NP-hard, exact algorithms are not capable of solving problems for big numbers of customers.

This problem is of economic importance to businesses because of the time and costs associated with providing a fleet of delivery vehicles to transport products to a set of geographically dispersed customers. Additionally, such problems are also significant in the public sector where vehicle routes must be determined for bus systems, postal carriers, and other public service vehicles. In each of these instances, the problem typically involves finding the minimum cost of the combined routes for a number of vehicles in order to facilitate delivery from a supply location to a number of customer locations. Since cost is closely associated with distance, a company might attempt to find the minimum distance traveled by a number of vehicles in order to satisfy its customer demand. In doing so, the firm attempts to minimize costs while increasing or at least maintaining an expected level of customer service.

The process of selecting vehicle routes allows the selection of any combination of customers in determining the delivery route for each vehicle. Therefore, the vehicle routing problem is a combinatorial optimization problem where the number of feasible solutions for the problem increases exponentially with the number of customers to be serviced. In addition, the vehicle routing problem is closely related to the traveling salesman problem where an out and back tour from a central location is determined for each vehicle. Since there is no known polynomial algorithm that will find the optimal solution in every instance, the vehicle routing problem is considered NP-hard. For such problems, the use of heuristics is considered a reasonable approach in finding solutions and this paper uses an ant colony optimization (ACO) approach to find solutions to the vehicle routing problem (VRP). ACO simulates the behavior of ant colonies in nature as they forage for food and find the most efficient routes from their nests to food sources. The decision making processes of ants are embedded in the artificial intelligence algorithm of a group of virtual ants which are used to provide solutions to the vehicle routing problem. This approach is relevant because it provides solutions to an important problem in transportation science and the experimental results indicate that the performance of the technique is competitive with other techniques used to generate solutions to the VRP.

**Vehicle routing problem**

The vehicle routing problem has been an important problem in the field of distribution and logistics since at least the early 1960s [1]. It is described as finding the minimum distance or cost of the combined routes of a number of vehicles m that must service a number of customers n. Mathematically, this system is described as a weighted graph GZ(V, A,d) where the vertices are represented by VZ{v0,v1,.,vn}, and the arcs are represented by AZ{(vi, vj): i≠j}.A central depot where each vehicle starts its route is located at v0 and each of the other vertices represents the n customers. The distances associated with each arc are represented by the variable dij which is measured using Euclidean computations. Each customer is assigned a non-negative demand qi, and each vehicle is given a capacity constraint, Q. The problem is solved under the following constraints.

- Each customer is visited only once by a single vehicle.

- Each vehicle must start and end its route at the depot, $v_0$.
- Total demand serviced by each vehicle cannot exceed Q.

Additionally, the problem may be distance constrained by defining a maximum route length, Lm, which each vehicle may not exceed. This maximum route length includes a service distance d (translated from service time) for each customer on the route. An example of a single solution consisting of a set of routes constructed for a typical vehicle routing problem is presented in Fig. 1, where mZ3, nZ10. The VRP studied here is symmetrical where $d_{ij}Zd_{ji}$ for all i and j.

A vast amount of research has been accomplished on the vehicle routing problem [2,3] including advanced meta-heuristic approaches such as Tabu Search [4–6] and Simulated Annealing [7]. A limited amount of research addressing the vehicle routing problem has used ACO with candidate lists and ranking techniques to improve the ability of a single ant colony to solve the VRP [8,9]. The research in this paper uses multiple ant colonies and experiments with different candidate list sizes in order to improve the ability of ACO to solve known instances of the VRP.

**Ant colony optimization and scatter search**

Ant colony optimization (ACO) simulates the behavior of ant colonies in nature as they forage for food and find the most efficient routes from their nests to food sources. As some ants travel, they deposit a constant amount of pheromone trail that other ants are attracted to follow them. The increase in pheromone enhances the probability of the next ants selecting the path. Over time, as more ants are able to complete the shorter route, pheromone accumulates faster on shorter paths and longer paths are less reinforced. The ants are capable of not only finding the shortest path from a food source to the nest, but also adapting to changes in the environment once the old one is no longer feasible due to a new obstacle. This natural behavior of ants can be used to explain reason that they can find the shortest path [13]. applies ant colony optimization to an established set of vehicle routing problems. The ACO method includes the two basic steps of construction of solutions and pheromone updating. Scatter search (SS) is a novel evolutionary method. It is a population- based meta-heuristic that combines solutions from a reference set to create improved solutions. The reference set is a set of feasible solutions and is usually updated by combining these existing solutions to obtain new ones. In the reference set, scatter search maintains some high quality solutions during the search process and also ensures diversity to explore other regions when the process is trapped in a local minimum. The main purpose behind scatter search is that the newly combined solutions will explore various regions of the solution space where the better solution may possibly be found. Scatter search has regulations and strategies that are still not emulated by other evolutionary methods, and this method has been applied successfully to many combinatorial optimization problems [14,15]. Provide detailed descriptions and an implementation framework of the scatter search approach. Applied scatter search to the standard vehicle routing problem and combined solutions from the reference set through a common arc mechanism. The scatter search method is not restricted to a single uniform design, and it is very flexible and effective, since each of its elements can be implemented in variety of ways.

**The Ant Colony Optimization Heuristic**

In Ant Colony Optimization (ACO), a number of artificial ants with the described characteristics search for good quality solutions to the discrete optimization problem. If $G = (C, L)$ is assumed as the graph of a discrete optimization problem, ACO can be used to find to find a solution to the shortest path problem defined on the graph *G*. A solution is described in terms of paths through the states of the problem in accordance with the problems' constraints. For example, in the TSP, *C* is the set of cities, *L* is the set of arcs connecting cities, and a solution is a closed tour.

Each ant is assigned to an initial state based on problem criteria. The start state is usually defined as a unit length sequence. Artificial ants find solutions in parallel processes using an incremental constructive mechanism to search for a feasible solution.

It starts from the initial state and move to feasible neighbor states. Moves are made by applying a stochastic search policy guided by ants' memory, problem constraints, pheromone trail accumulated by all the ants from the beginning of the search process

International Journal of Engineering and Applied Sciences

www.eaas-journal.org

and problem-specific heuristic information (visibility). The ants' memory keeps information about the path it followed. It can be used to carry useful information to compute the goodness of the generated solution and/or the contribution of each executed move. It also provides the feasibility of the solutions. While building its own solution, each ant also collects information on the problem characteristics and its performance. It uses this information to modify the representation of the problem, as seen by the other ants. The information collected by ants is stored in pheromone trails. Visibility measures the attractiveness of the next node to be selected. Visibility value represents a priori information about the problem instance definition. A solution is constructed by moving through a sequence of neighbor states.

The decisions about when the ants should release pheromone on the environment and how much pheromone should be deposited depend on the problem. Ants can release pheromone while building the solution, or after a solution has been built, or both. In addition, pheromone trails can be associated with all problem arcs or some of them.

Probabilistic tables that are function of the pheromone trail and heuristic values guide the ants' search. The stochastic component of the decision policy and the pheromone evaporation mechanism prevents a rapid drift of all the ants towards the same part of the search space.

After building a solution the ant deposits additional pheromone information on the arcs of the solution. In general, the amount of pheromone deposited is proportional to the goodness of the solution. If a move generates a high-quality solution its pheromone will be increased proportionally to its contribution. After an ant constructs a solution and deposits pheromone information it dies.

Although a single ant can find a solution high quality solutions are only found as a result of the global cooperation among all ants. Communication among ants is mediated by information stored in pheromone trail values.

In brief, a colony of ants concurrently moves through feasible adjacent states of the problem by applying a stochastic decision process. By moving, ants incrementally build solutions to the optimization problem. During the solution construction process or/and after the solution is constructed, the ants evaluate the (partial) solution and update pheromone trail values.

## Ant Colony System

Dorigo and Gambardella [16] proposed the ACS which has two types. In the first type, after all the ants have built a solution, pheromone trails on the arcs used by the ant that found the best tour so far are updated. In the second, after all the ants have built a solution, a local search procedure based on 3-opt is applied to improve the solutions and then pheromone trails on the arcs used by the ant that found the best tour so far are updated. The pheromone trail update rule is as follows:

$$\tau_{ij} = (1-\rho)\cdot\tau_{ij} + \rho.\tau\Delta_{ij} \qquad (1)$$

where $\tau\Delta_{ij}$ = (length of the shortest tour)$^{-1}$

A different decision rule, called pseudo-random-proportional rule, is used in the ACS. The pseudo-random-proportional rule $p_{jk}$, used by ant k in node *I* to choose the next node *j* is the following:

$$P_{ij}^{k} = \begin{cases} \arg\max\limits_{j\in allowed_i}\left\{\tau_{ij}\cdot\left[\eta_{ij}\right]^{\beta}\right\} & , \quad \text{if } q \le q_0 \\ p_{ij}^{k} & , \quad \text{otherwise} \end{cases}$$

where *q* is a random variable uniformly distributed over [0, 1], and $q_0$ Є [0, 1] is a parameter.

While using the probabilistic choice of the components to construct a solution is called exploration, choosing the component that maximizes a blend of pheromone trail values and heuristic evaluations is called exploitation.

ISSN2305-8269
International Journal of Engineering and Applied Sciences

www.eaas-journal.org

$$p_{ij}^{k} = \begin{cases} \dfrac{[\tau_{ij}]\cdot[\eta_{ij}]^{\beta}}{\sum\limits_{k\in allowed_k}[\tau_{ik}]\cdot[\eta_{ik}]^{\beta}} & , j \in allowed_k \\ 0 & ,\text{otherwise} \end{cases}$$

An ant moving from city *i* to city *j* updates the pheromone trail on arc (*i*, *j*).

$$\tau_{ij} = (1-\varphi)\cdot\tau_{ij} + \varphi.\tau_0$$

Where $0<\varphi\leq1$

$$\tau_0 = (n.L_{nn})^{-1}$$

where $n$ is the number of cities and $L_{nn}$ is the length of a tour produced by the nearest neighbor heuristic.
Last, ACS exploits a data structure called candidate list which provides additional local heuristic information. A candidate list is a list of preferred cities to be visited from a given city. In ACS, when an ant is in city, instead of examining all the unvisited neighbors, it chooses the city to move to among those in the candidate list. Other cities are examined only if no candidate list city has unvisited status. The candidate list of a city contains *d* cities ordered by non-decreasing distance (*d* is a parameter) and the list is scanned sequentially and according to the ant tabu list to avoid already visited cities [16].
There are other versions of ACS. These differ from the ACS described above:

(i) in the way local pheromone update applied, such as setting $\tau_0=0$,

(ii) in the way the decision rule are made

(iii) in the type of solution used for global pheromone updating, such as adding the pheromone only to arcs belonging to the best solution found.

**Experimentation**

The single ant colony and multiple ant colony methodologies are applied to three problems found in previous research [3] in order to compare their ability to find solutions to the VRP for problems with varying numbers of customers. These three problems differ in size, and they were selected in order to establish the baseline performance of the new multiple ant colony technique and to rigorously analyze the results of the method prior to testing on larger problems. In addition, the fractional size of the candidate list is used as an experimental factor in order to determine if it plays a significant role in finding improved solutions to the VRP. Our ACO-based search results are compared to similar ant colony approaches to the VRP and to the best known solutions for each problem.

5.1. Design of experiment.
A description for each of the three problems used in the experiment is presented in Table 1 and the specific location coordinates on the graph for all locations and the depot are available in [2,3] The ant colony optimization results for this experiment are compared to the single ACO approach found in [8], the ranked ACO approach found in [9] and the best known solutions for each problem. The best solution for problem 1 has been proven to be optimal [18] and the best solutions for all three problems have been found using Tabu Search [5].

The experiment includes two different optimization methods (single and multiple ant colonies) and is applied to the three different problems. In addition, the candidate list size is set at four different levels for each problem by varying the fraction of sites available to be candidates. This is done by dividing the number of customers n by four different denominator values (n/3, n/5, n/7 and n/9). The resulting experimental design consists of 2!4Z8 different treatment cells for each of the three problems. Solutions for each of these cells have been generated 25 times in order to understand the central tendency and variances associated with the results of the experiment and in order to make meaningful statistical comparisons. The measures of performance for the model include mean route distance L, minimum route distance L, average CPU run-time, and the percentage inferiority of the minimum route

ISSN2305-8269
International Journal of Engineering and Applied Sciences

www.eaas-journal.org

distance L in comparison to the best known solution to the problem. These measures are evaluated in order to answer two research questions for this study:

1. Does the multiple ant colony method provide improved solutions to the problems in comparison to the single colony method?

2. Does the use of different candidate list sizes result in significantly different solutions to the problems.

Generation of all ACO solutions for each problem and its variations was done using CCCcoding on an Athlon AMD4 900 MHz processor. In all ACO solutions, the following search parameters were set to values that were
found to be robust in previous research and pilot-testing: aZ0.1, bZ2.3, q0Z0.9, and mZ25. Each run of the model consisted of 5000 iterations of the trail construction and trail updating processes.

5.2. Experimental results

The results of the experiment listed in Table 2, reveal that the ACO approaches used in this research are able to generate competitive solutions for the VRP for problem C1 (nZ50), both the single and multiple ant colony approaches are able to generate solutions within less than 1% of the optimum solution to the problem. In addition, the fractional size of the candidate list does not seem to play a great role in the quality of the solution for problem C1, and all candidate list sizes are able to find solutions within 3% of the known optimum. However, for problem C3 (nZ100), the best results are found using a candidate size equal to fourteen as determined by the fractional level n/7. Using this candidate list size, the single

Table 1. ACO results

colony approach was able to find its best solution within 3.9% of the best known result and the multiple colony approach was able to find a result within 1.7% of the best known solution. This pattern is apparent in the experimental results for Problem C4, as the single colony approach finds solutions within 10.06% of the best known result and multiple ant colony optimization is able to find solutions within 6.45% of the best known results. These initial observations tend to indicate that as the problem size (n) grows the multiple colony approach becomes more appealing and the candidate list size should be kept small by using a larger denominator in the fraction that determines the number of candidate sites. Additionally, in all instances the single and multiple colony versions of the algorithm were able to find solutions using the minimum number of vehicles m as established by previously best-known solutions to the problem (mZ5 for C1, mZ8 for C3, mZ12 for C4). In no instance did the algorithms need to add an additional vehicle in order to solve any of the three problems.

In order to further analyze the results and provide answers to the research questions, a two factor analysis of variance (ANOVA) was conducted for each of the three problems in order to determine if there are significant differences in the route distance L as a result of using different candidate list sizes and single versus multiple colony approaches.

The first ANOVA for problem C1 found significant differences (FZ105.29, p!Z0.001) for the different candidate list sizes and for optimization methods used.

International Journal of Engineering and Applied Sciences

www.eaas-journal.org

| Approach | Problem | CL Fraction | Min $L$ | Mean $L$ | Std. dev. | % inferior |
|---|---|---|---|---|---|---|
| Single ACO | C1 | 1/3 | 540.30 | 551.98 | 3.84 | 2.99 |
| | | 1/5 | 524.80 | 528.90 | 2.89 | .04 |
| | | 1/7 | 528.60 | 532.85 | 1.91 | .77 |
| | | 1/9 | 539.80 | 548.18 | 4.95 | 2.91 |
| | C3 | 1/3 | 871.30 | 891.21 | 9.40 | 5.47 |
| | | 1/5 | 856.80 | 876.71 | 7.24 | 3.72 |
| | | 1/7 | 854.10 | 868.19 | 6.53 | 3.39 |
| | | 1/9 | 854.20 | 865.65 | 5.31 | 3.41 |
| | C4 | 1/3 | 1178.61 | 1197.43 | 8.52 | 14.60 |
| | | 1/5 | 1154.15 | 1170.47 | 9.89 | 12.22 |
| | | 1/7 | 1136.16 | 1154.40 | 8.38 | 10.48 |
| | | 1/9 | 1131.83 | 1143.43 | 6.29 | 10.06 |
| Multiple ACO | C1 | 1/3 | 535.20 | 550.06 | 3.30 | 2.03 |
| | | 1/5 | 524.80 | 537.12 | 9.03 | .04 |
| | | 1/7 | 528.40 | 544.49 | 11.17 | .73 |
| | | 1/9 | 538.40 | 558.00 | 9.42 | 2.63 |
| | C3 | 1/3 | 881.40 | 889.49 | 4.96 | 6.69 |
| | | 1/5 | 848.30 | 870.68 | 10.14 | 2.69 |
| | | 1/7 | 840.20 | 868.37 | 9.42 | 1.71 |
| | | 1/9 | 852.40 | 867.91 | 8.37 | 3.18 |
| | C4 | 1/3 | 1151.31 | 1173.43 | 10.59 | 11.95 |
| | | 1/5 | 1119.88 | 1144.47 | 9.86 | 8.89 |
| | | 1/7 | 1094.80 | 1131.21 | 11.40 | 6.45 |
| | | 1/9 | 1104.21 | 1123.28 | 9.97 | 7.37 |

(FZ50.35, p!Z0.001). Tukey's pairwise comparisons of the means showed that a candidate list size of 10 (as determined by the expression n/5) had significantly lower route distances than routes found using the other three candidate list sizes (family alphaZ0.05, p!Z0.001 for all differences). In addition, solutions for the single ant colony method were 8.59–4.85 units smaller than solutions generated using multiple ant colony optimization (alphaZ0.05, p!Z0.001). These differences are represented in Fig. 2. The second ANOVA for problem C3 again found a significant difference between the different candidate list sizes (FZ93.63, p!Z0.001). However, no significant difference exists between the two optimization approaches for this problem (FZ1.42, p!Z0.235). Tukey's pairwise differences for this problem indicate that the candidate list of 11 (n/9) finds route distances 27.61–19.51 units less than the candidate list of 33 (n/3) and 10.96–2.86 units less than a candidate list of 20 (n/5) with a equal to .05. No significant difference is evident between the candidate list size of eleven and fourteen (n/7) for this problem. The results for this problem are shown in Fig. 3. The ANOVA results for problem C4 again indicate a significant difference for candidate list size (FZ281.53, pZ0.001). Tukey's pairwise comparisons show that using a denominator of nine for the fractional approach produces improved results (family aZ0.05). Route distances using a candidate list size of seventeen (n/9) are 57.00–47.16 units smaller than distances found with a candidate list size of fifty (n/3). They are 29.04–19.20 units smaller than solutions found with a candidate list size of thirty (n/5) and 14.37–4.53 units smaller than solutions with a candidate list size of twenty one (n/7). Comparison of the optimization methods indicates the multiple ant colony approach produces routes distances 20.66–26.01 units smaller than the single colony approach (FZ296.86, pZ0.001). The plot of the means for problem C4 is shown in Fig. 4.

ISSN2305-8269
International Journal of Engineering and Applied Sciences

www.eaas-journal.org

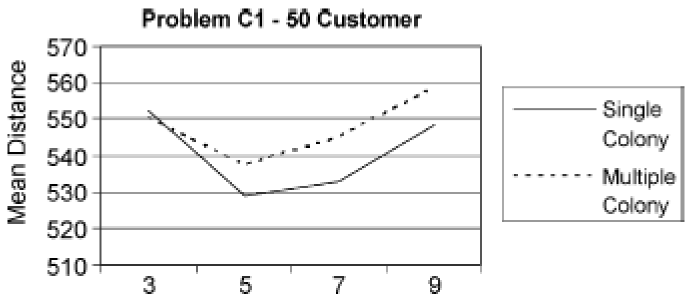

Fig. 2. Plot of means for problem C1.

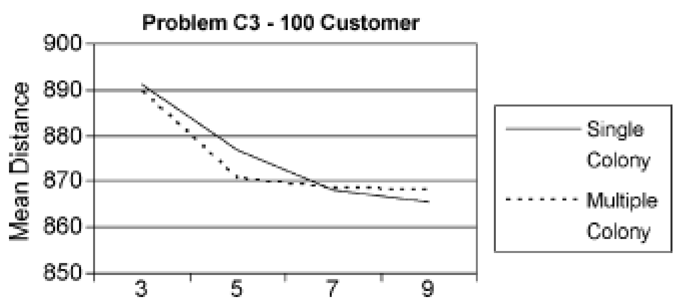

Fig. 3. Plot of means for problem C3.

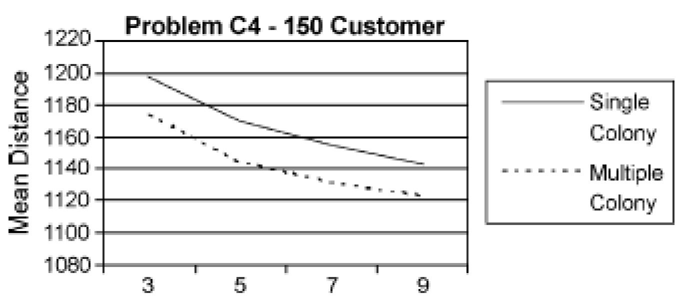

Fig. 4. Plot of means for problem C4.

The computational speed of an application is also an important means of measuring the ability of an algorithm.
Therefore, the two ACO methods used in this research are compared to the CPU times reported by other research for solving the three problems in Table 2. The methods of this study are compared to the results of Taillard (T) [5], Osman (O) [7], Gendreau, Hertz and Laporte (GHL) [4], Xiu and Kelly (XK) [19], and the savings generator (SG) solutions of Kelly and Xiu [6]. The times reported for the ACO methods are the average time the algorithm spent in finding the best solution for the problem. Since each of the research studies in the comparison in Table 2 use different computer platforms it may not be accurate to compare the run times from one study to the next. However, it can be seen that the two ACO methods used in this research are very competitive in terms of computational times, especially as problem size increases. The ACO times are smaller than other known techniques for problem C3 and only the recent Savings Generator approach of Kelly and Xiu is able to find the best solution faster than ACO. For all three problems, the multiple-colony ACO is faster than the single colony version of ACO.

Table 2. Comparison of CPU times (in seconds)

International Journal of Engineering and Applied Sciences

www.eaas-journal.org

| Problem | T | O | GHL | XK | SG | Single ACO | Mult. ACO |
|---|---|---|---|---|---|---|---|
| C1 | 49 | 61 | 360 | 1795 | 373 | 138 | 99 |
| C3 | 580 | 895 | 1104 | 4316 | 557 | 539 | 450 |
| C4 | 3800 | 1761 | 3528 | 8994 | 821 | 1457 | 1395 |

## CONCLUSION

The purpose of this study was to develop ant system based approach for solving VRPs. The proposed approach basically differ from the other ant system heuristics in the way that it forms the candidate list, and it calculates the initial pheromone values and the visibility function.

Most of the ant system based heuristics forms the candidate lists at the beginning, and do not update them. On the other hand, attractiveness of arcs depends on the pheromone values on them. As pheromone values on arcs are updated, some arcs that are not on the candidate lists may become attractive. Thus, in this study candidate lists are updated after global pheromone update procedure.

In most of the ant colony based algorithms to VRP, initial pheromone trails is calculated based on the best known route distances found for the particular problem. However, in this study it is calculated based on the feasible solution found.

Finally, visibility of an arc is calculated as a function of distance between two customers, customers' distance to the depot and the time window associated with the customer to whom the ant is considered to move.

## ACKNOWLEDGMENT

We acknowledge our friend, Mehdi fatemi, and the Associate Editor and anonymous reviewers for their valuable comments and suggestions that have helped us to improving the paper.

## REFERENCES

[1] Gökçe, Elif İlke. *A revised ant colony system approach to vehicle routing problems*. Diss. Sabancı University, 2004.

[2] Clark G, Wright JW. Scheduling of vehicles from a central depot to a number of delivery points. Oper Res 1964;12:568–81.

[3] Christofides N, Eilon S. An algorithm for the vehicle dispatching problem. Oper Res Quarterly 1969;20(3):309–18.

[4] Christofides N, Mingozzi A, Toth P. The vehicle routing problem. In: Christofides N, Mingozzi A, Toth P, Sandi C, editors. Combinatorial

[5] Optimization. Chichester: Wiley; 1979. p. 315–38.

[6] Gendreau M, Hertz A, Laporte G. A tabu search heuristic for the vehicle routing problem. Manage Sci 1994;40:1276–90.

[7] Taillard E. Parallel iterative search methods for vehicle routing problems. Networks 1993;23:661–73.

[8] Kelly JP, XU J. A set-partitioning based heuristic for the vehicle routing problem. INFORMS J Comput 1999;11(2):161–72.

[9] Osman LH. Metastrategy simulated annealing and tabu search algorithms for the vehicle routing problem. Ann Oper Res 1993;41:

[10] 421–51.

[11] Bullnheimer B, Hartl RF, Strauss C. Applying the ant system to the vehicle routing problem. In: Voss S, Martello S, Osman IH, Roucairol C, editors. Meta-Heuristics: Advances and Trends in Local Search Paradigms for Optimization. Boston: Kluwer; 1998.

[12] Bullnheimer B, Hartl RF, Strauss C. An improved ant system algorithm for the vehicle routing problem. Ann Oper Res 1999;89:319–28.

[13] Bell, J.E., McMullen, P.R., 2004. Ant colony optimization techniques for the vehicle routing problem. Adv. Eng. Inform. 18, 41–48.

[14] Marti, M., Laguna, M., Glover, F., 2006. Principles of scatter search. Eur. J. Oper. Res. 169, 359–372.

[15] Glover, F., 1998. A template for scatter search and path relinking. In: Hao, J.-K., Lutton, E., Ronald, E., Schoenauer, M., Snyers, D. (Eds.), Artificial Evolution.

[16] Dorigo, M., Maniezzo, V. Colorni, A., “The ant system: Optimization by a colony of cooperating agents,” *IEEE Transactions on Systems, Man, and Cybernetics-Part B*, vol.26, pp.29-41, 1996.

[17] Gholipour, Yousof, and Mahmood Mola. "Investigation stability of Rikitake system." Journal of Control Engineering and Technology 4, no. 1 (2014).

[18] Gholipour, Yousof, Amin Ramezani, and Mahmood Mola. "Illustrate the Butterfly Effect on the Chaos Rikitake system." Bulletin of Electrical Engineering and Informatics 3, no. 4 (2014): 273-276.

[19] Mehriz, Iran, and Iran Dariun. "Illustrate the effect of value of P, I, D in a PID controller for a four Tank process.

[20] Gholipour, Y., Mola, M. STABILIZATION OF CHAOS RIKITAKE SYSTEM BY USE OF FUZZY CONTROLLER. *SCIENCE INTERNATIONAL-LAHORE* 27 (1), 115-119. (2015).

[21] Gholipour, Y., Chavooshi Zade, M.; “REPLACEMENT UNSTABLE TRANSMISSION ZEROS FOR A NON MINIMUM PHASE QUADRUPLE-TANK PROCESS”. SCIENCE INTERNATIONAL-LAHORE 27 (2)1097-1100, 2015 (2015).