Available online at www.sciencedirect.com

SCIENCE DIRECT®

Advanced Engineering Informatics 18 (2004) 41–48

ADVANCED ENGINEERING INFORMATICS

www.elsevier.com/locate/aei

Ant colony optimization techniques for the vehicle routing problem

John E. Bell[a,*], Patrick R. McMullen[b]

[a]Department of Operational Sciences, Air Force Institute of Technology, Wright-Patterson AFB, OH, USA
[b]Babcock Graduate School of Management, Wake Forest University, Winston-Salem, NC, USA

Received 9 July 2004; accepted 9 July 2004

Abstract

This research applies the meta-heuristic method of ant colony optimization (ACO) to an established set of vehicle routing problems (VRP). The procedure simulates the decision-making processes of ant colonies as they forage for food and is similar to other adaptive learning and artificial intelligence techniques such as Tabu Search, Simulated Annealing and Genetic Algorithms. Modifications are made to the ACO algorithm used to solve the traditional traveling salesman problem in order to allow the search of the multiple routes of the VRP. Experimentation shows that the algorithm is successful in finding solutions within 1% of known optimal solutions and the use of multiple ant colonies is found to provide a comparatively competitive solution technique especially for larger problems. Additionally, the size of the candidate lists used within the algorithm is a significant factor in finding improved solutions, and the computational times for the algorithm compare favorably with other solution methods.

Keywords: Ant colony optimization; Vehicle routing; Vehicle scheduling; Combinatorial optimization; Heuristics; Optimization

1. Introduction

Finding efficient vehicle routes is an important logistics problem which has been studied for several decades. When a firm is able to reduce the length of its delivery routes or is able to decrease its number of vehicles, it is able to provide better service to its customers, operate in a more efficient manner and possibly increase its market share. A typical vehicle routing problem includes simultaneously determining the routes for several vehicles from a central supply depot to a number of customers and returning to the depot without exceeding the capacity constraints of each vehicle. 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

* Corresponding author. Tel.: +1-937-255-3636; fax: +1-937-986-4943.

E-mail address: john.bell@afit.edu (J.E. Bell).

1474-0346/$ - see front matter
doi:10.1016/j.aei.2004.07.001

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.

2. 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 $G=(V, A,d)$ where the vertices are represented by $V=\{v_0,v_1,\ldots,v_n\}$, and the arcs are represented by $A=\{(v_i, v_j): i\neq j\}$. A central depot where each vehicle starts its route is located at v_0 and each of the other vertices represents the n customers. The distances associated with each arc are represented by the variable d_{ij} which is measured using Euclidean computations. Each customer is assigned a non-negative demand q_i, 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, L_m, which each vehicle may not exceed. This maximum route length includes a service distance δ (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 $m=3$, $n=10$. The VRP studied here is symmetrical where $d_{ij}=d_{ji}$ 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.

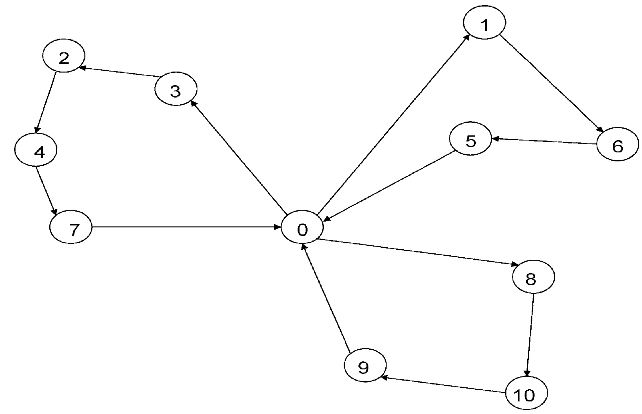

Fig. 1. Example solution—vehicle routing problem.

3. Ant colony optimization

Ant colony optimization is a part of the larger field of swarm intelligence in which scientists study the behavior patterns of bees, termites, ants and other social insects in order to simulate processes.

The ability of insect swarms to thrive in nature and solve complex survival tasks appeals to scientists developing computer algorithms needed to solve similarly complex problems. Artificial intelligence algorithms such as ant colony optimization are applied to large combinatorial optimization problems and are used to create self-organizing methods for such problems.

Ant colony optimization is a meta-heuristic technique that uses artificial ants to find solutions to combinatorial optimization problems. ACO is based on the behavior of real ants and possesses enhanced abilities such as memory of past actions and knowledge about the distance to other locations. In nature, an individual ant is unable to communicate or effectively hunt for food, but as a group, ants possess the ability to solve complex problems and successfully find and collect food for their colony. Ants communicate using a chemical substance called pheromone. As an ant travels, it deposits a constant amount of pheromone that other ants can follow. Each ant moves in a somewhat random fashion, but when an ant encounters a pheromone trail, it must decide whether to follow it. If it follows the trail, the ant's own pheromone reinforces the existing trail, and the increase in pheromone increases the probability of the next ant selecting the path. Therefore, the more ants that travel on a path, the more attractive the path becomes for subsequent ants. Additionally, an ant using a short route to a food source will return to the nest sooner and therefore, mark its path twice, before other ants return. This directly influences the selection probability for the next ant leaving the nest. 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 evaporation of pheromone also makes less desirable routes more difficult to detect and further decreases their use. However, the continued random selection of paths by individual ants helps the colony discover alternate routes and insures successful navigation around obstacles that interrupt a route. Trail selection by ants is a pseudo-random proportional process and is a key element of the simulation algorithm of ant colony optimization [10]. Detailed descriptions of ant behavior relating to ACO are found in [11,12].

The use of ant colonies was first applied to the traveling salesman problem and the quadratic assignment problem [13] and has since been applied to other problems such as the space planning problem [14], the machine tool tardiness problem [15] and the multiple objective JIT sequencing problem [16].

4. ACO heuristic for vehicle routing

4.1. Route construction

Using ACO, an individual ant simulates a vehicle, and its route is constructed by incrementally selecting customers until all customers have been visited. Initially, each ant starts at the depot and the set of customers included in its tour is empty. The ant selects the next customer to visit from the list of feasible locations and the storage capacity of the vehicle is updated before another customer is selected. The ant returns to the depot when the capacity constraint of the vehicle is met or when all customers are visited. The total distance L is computed as the objective function value for the complete route of the artificial ant. The ACO algorithm constructs a complete tour for the first ant prior to the second ant starting its tour. This continues until a predetermined number of ants m each construct a feasible route.

Using ACO, each ant must construct a vehicle route that visits each customer. To select the next customer j, the ant uses the following probabilistic formula [10]

$$j = \underset{\text{otherwise } S}{\arg\max}\{(\tau_{\text{iu}})(\eta_{\text{iu}})^{\beta}\} \quad \text{for } u \notin M_k, \quad \text{if } q \leq q_0, \tag{1}$$

where τ_{iu} is equal to the amount of pheromone on the path between the current location i and possible locations u. The value η_{iu} is defined as the inverse of the distance between the two customer locations and the parameter β establishes the importance of distance in comparison to pheromone quantity in the selection algorithm ($\beta > 0$). Locations already visited by an ant are stored in the ants working memory M_k and are not considered for selection. The value q is a random uniform variable [0,1] and the value q_0 is a parameter. When each selection decision is made, the ant selects the arc with the highest value from Eq. (1) unless q is greater than q_0. In this case, the ant selects a random variable (S) to be the next customer to visit based on the probability distribution of p_{ij}, which favors short paths with high levels of pheromone:

$$p_{ij} = \frac{(\tau_{ij})(\eta_{ij})^{\beta}}{\sum_{u \notin M_k} (\tau_{iu})(\eta_{iu})^{\beta}} \quad \text{if } j \notin M_k \text{ otherwise } 0 \tag{2}$$

Using formulas (1) and (2) each ant may either follow the most favorable path already established or may randomly select a path to follow based on a probability distribution established by distance and pheromone accumulation. If the vehicle capacity constraint is met, the ant will return to the depot before selecting the next customer. This selection process continues until each customer is visited and the tour is complete.

4.2. Trail updating

In order to improve future solutions, the pheromone trails of the ants must be updated to reflect the ant's performance and the quality of the solutions found This updating is a key element to the adaptive learning technique of ACO and helps to ensure improvement of subsequent solutions. Trail updating includes local updating of trails after individual solutions have been generated and global updating of the best solution route after a predetermined number of solutions m has been accomplished.

First, local updating is conducted by reducing the amount of pheromone on all visited arcs in order to simulate the natural evaporation of pheromone and to ensure that no one path becomes too dominant. This is done with the following local trail updating equation,

$$\tau_{ij} = (1 - \alpha)\tau_{ij} + (\alpha)\tau_0 \tag{3}$$

where α is a parameter that controls the speed of evaporation and τ_0 is equal to an initial pheromone value assigned to all arcs in graph G. For this study, τ_0, is equal to the inverse of the best known route distances found for the particular problem listed in Table 1.

After a predetermined number of ants m construct a feasible route, global trail updating is performed by adding pheromone to all of the arcs included in the best route found

Table 1
Problem characteristics and previous solution values

Problem	N	Q	L_m	δ	Best known	Single ACO [8]	Ranked ACO [9]
C1	50	160	∞	0	524.61	524.61	524.61
C3	100	200	∞	0	826.14	879.43	832.32
C4	150	200	∞	0	1028.42	1147.41	1061.55

by one of m ants. Global trail updating is accomplished according to the following relationship,

$$\tau_{ij} = (1-\alpha)\tau_{ij} + \alpha(L)^{-1} \tag{4}$$

This updating encourages the use of shorter routes and increases the probability that future routes will use the arcs contained in the best solutions. This process is repeated for a predetermined number of iterations and the best solution from all of the iterations is presented as an output of the model and should represent a good approximation of the optimal solution for the problem.

4.3. Route improvement strategies

The route construction procedures and pheromone updating processes described above are typical for ACO as it is applied to the traveling salesman problem [10] However, research shows that the attainment of improved solutions to the VRP is dependent on route improvement strategies in the algorithm [8]. The first of these strategies involves the inclusion of a local exchange procedure to act as an improvement heuristic within the routes found by individual vehicles. The technique used for this purpose is the common 2-opt heuristic [17] where all possible pairwise exchanges of customer locations visited by individual vehicles are tested to see if an overall improvement in the objective function can be attained. For example in Fig. 1, the first vehicle travels in order from the depot to locations {1,6,5}. The heuristic will also calculate the distances for all pairwise permutations {{6,1,5}, {1,5,6}, {5,6,1}} of the set including these three customer locations. If any of these solutions is found to improve the objective function, then the solution is modified prior to saving it as the best solution and conducting pheromone updating for the route. This process adds greatly to the number of individual combinations that are explored by the search and can be thought of as solving several TSPs after assigning the customers to vehicles [9].

The second improvement strategy is the use of a candidate list for determining the next location selected in a vehicle route. Each individual location is assigned a candidate list based on the distance to all other locations in the location set. Only the closest locations are included in the candidate list for the current location and are made available for selection as the next location to be visited in the route. The size of the candidate list has been determined in the past by restricting its size to a fraction of the total number of customers in the problem. For example, previous research sets the candidate list size equal to one forth of the total number of customers regardless of the number of customers [9]. For problems with fifty customers the candidate list is restricted to the rounded integer value of twelve (n/4). Fig. 2 depicts the twelve closest candidate locations for the current location during the construction of a vehicle route. It is believed that this restriction prevents the algorithm from wasting effort considering moves to locations that are a great distance from the current location and have very little chance of creating an improved solution to the problem. Problems with one hundred or even two hundred locations are quite possible in versions of the VRP and therefore candidate list sizes might reach as high as fifty locations using the simple expression (n/4) which has been used in past research [8]. This research explores the impact of selecting different candidate list sizes and further explores how this selection may be important for different problem sizes.

4.4. Multiple ant colonies

The use of specialized groups or families of ants has been suggested in the literature [10,11], but has not been implemented in previous research for the VRP The concept of using separate groups of ants is based on observations that ants in nature sometimes have special job classifications, such as sorting, foraging or defending the nest. Similarly, it is hypothesized that problem solving for the VRP might be more effective if a separate colony of ants with unique pheromone deposits is used for each vehicle. This separation is intended to differentiate paths typically used in the first vehicle route from those used by subsequent vehicles, and it is believed that this technique might be more effective as the size of the problem and the number of vehicles required increases. For example, in a problem with 50 customer locations, a good solution might contain five or more separate vehicles each with a separate route to and from the depot location. Using multiple ant colonies the first vehicle's route is always marked with pheromone from ant colony #1, and these deposits are not used to help determine the route for the second vehicle. Instead, the route for the second colony will only depend on the pheromone deposits made by ant colony #2 and so on. Although this technique puts a slight limit on communication, it is desired that the stratification will add to the concentrated efforts to simultaneously find multiple paths by increasing the probability that a vehicle continues to use routes it has

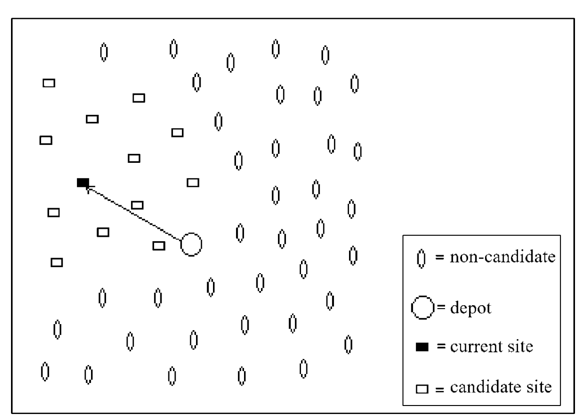

Fig. 2. Route selection with candidate list.

found to be successful and is not distracted by the pheromone paths of ants assigned to other vehicles.

5. 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 $2\times4=8$ 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 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 C++coding 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: $\alpha=0.1$, $\beta=2.3$, $q_0=0.9$, and $m=25$. 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 (n=50), 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 ($n=100$), 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 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 ($m=5$ for C1, $m=8$ for C3, $m=12$ 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 ($F=105.29$, $p<=0.001$) for the different candidate list sizes and for optimization methods used

Table 2
ACO results

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

(F=50.35, p<=0.001). 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 alpha=0.05, p<=0.001 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 (alpha=0.05, p<=0.001). These differences are represented in Fig. 3.

The second ANOVA for problem C3 again found a significant difference between the different candidate list sizes (F=93.63, p<=0.001). However, no significant difference exists between the two optimization approaches for this problem (F=1.42, p<=0.235). 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 α 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. 4. The ANOVA results for problem C4 again indicate a significant difference for candidate list size (F=281.53, p=0.001). Tukey's pairwise comparisons show that using a denominator of nine for the fractional approach produces improved results (family α=0.05). 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

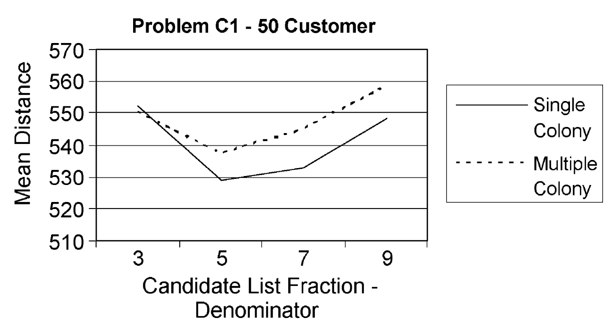

Fig. 3. Plot of means for problem C1.

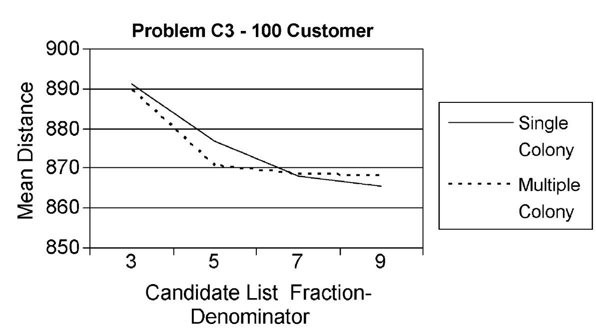

Fig. 4. Plot of means for problem C3.

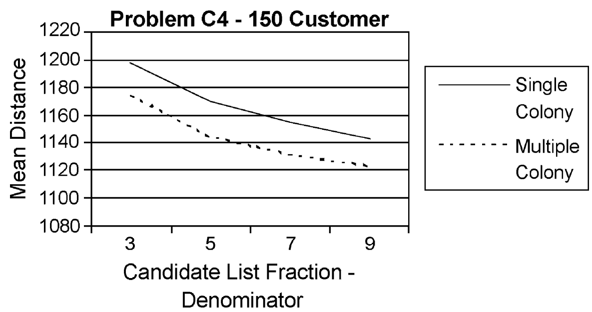

Fig. 5. Plot of means for problem C4.

approach ($F=296.86$, $p=0.001$). The plot of the means for problem C4 is shown in Fig. 5.

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 3. 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 3 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.

6. Discussion and concluding remarks

Ant colony optimization clearly has the ability to find good results within 1% of the known optimum for small problems. However, consistent with past research, the ACO methods used in this research are not as efficient in finding solutions for larger problems. The two ACO methods used here were able to find improved solutions to problems C3 and C4 in comparison to a single colony approach used in previous research that uses capacity utilization and savings components in the algorithm [8]. The two methods were not able to find improved solutions in comparison to a single ACO approach using ranking procedures [9], but the results found here were competitive with this method. This is significant because the multiple colony approach provides an alternative to using ant ranking, and a more advanced algorithm might be produced from this new technique. Additionally, the use of the mulitple ant colony algorithm also provides comparative performance versus traditional methods, and this research provides evidence that a multiple colony approach is beneficial for larger problems sizes as seen in the results for problems C3 and C4. It is believed that as the number of individual vehicles m increases, there is a benefit in having separate ant colonies with separate pheromone trails for each vehicle. It is speculated that the benefit of using separate pheromone trails results from the need to separate the most likely routes for different vehicles. Further research is warranted to understand the relationship between the number of vehicles needed to solve a vehicle routing problem and the need for multiple ant colonies. Testing of larger problem sizes and vehicles with different capacities might provide further insight about when to use a multiple colony approach to solve vehicle routing problems. Overall, it is accurate to say that the results of this research, while not providing a dramatic breakthrough at this time, do provide a new method with comparative abilities and the potential for further development.

Furthermore, the experiment shows that candidate list size and the method for determining candidate list sizes is important for finding good solutions to the VRP. This study uses a fractional approach [9] with several different denominator values to determine the candidate list size for each problem. The results indicate that as problem size n increases, the denominator value also had to increase to find the best solutions. This result seems to indicate that a fractional approach used in previous research may not be the best approach for determining candidate list size. For all three problem sizes, the best results were found when the candidate list size ranged between ten and twenty. Therefore, it is believed that candidate list sizes should be kept in this range and should be determined independently from the size of the problem, as has been suggested in past research for solving the traveling salesman problem [10]. Further research on candidate list sizes for the vehicle routing

Table 3
Comparison of CPU times (in seconds)

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

problem should attempt to further understand this phenomenon and attempt to develop a mathematical expression for determining candidate list sizes.

Future research should focus on improving the ACO algorithm for solving larger vehicle routing problems. This might be accomplished by combining the ranking methods of Bullnheimer, Hartl and Strauss [9] with the multiple colony approach tested in this research or by applying additional concepts to the algorithm such as a penalty function for less desirable routes as suggested in recent swarm intelligence research [11]. Additionally, experimentation with different parameter values in the algorithm is also thought to offer an area for improvement for larger problems. A method that varies the search parameter q_0 from a low value in the initial phases of a search to higher values towards the end of the search could improve the algorithms ability to not only select a variety of routes, but also concentrate on routes with higher pheromone values later in the search. Finally, the application of ACO techniques should be applied to other routing problems with unique clustering features such as the logistics problem set seen in the research of Ballou [20].

References

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

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

[3] Christofides N, Mingozzi A, Toth P. The vehicle routing problem. In: Christofides N, Mingozzi A, Toth P, Sandi C, editors. Combinatorial Optimization. Chichester: Wiley; 1979. p. 315–38.

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

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

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

[7] Osman LH. Metastrategy simulated annealing and tabu search algorithms for the vehicle routing problem. Ann Oper Res 1993;41: 421–51.

[8] 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.

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

[10] Dorigo M, Gambardella LM. Ant colonies for the traveling salesman problem. BioSystem 1997;43(1):73–81.

[11] Tarasewich P, McMullen PR. Swarm intelligence: power in numbers. Commun ACM 2002;45(8):62–7.

[12] Dorigo M, Di Caro G, Gambardella LM. Ant algorithms for discrete optimization. Artif Life 1999;5:137–72.

[13] Dorigo M. Optimization, learning and natural algorithms, Unpublished Doctoral Dissertation Politecnico di Milano, Dipartimento di Elettronica, Italy 1992 [In Italian].

[14] Bland JA. Space planning by ant colony optimization. Int J Comput Appl Technol 1999;(6):320–8.

[15] Bauer A, Bullnheimer B, Hartl RF, Strauss C. An ant colony optimization approach for the single machine tool tardiness problem. In: Proceedings of the 1999 Congress on Evolutionary Computation, 1999. p. 1445–50.

[16] McMullen PR. An ant colony optimization approach to addressing a JIT sequencing problem with multiple objectives. Artif Intell Eng 2001;15:309–17.

[17] Lin S, Kernighan BW. An effective heuristic algorithm for the TSP. Oper Res 1973;21:498–516.

[18] Fisher MA. Optimal solution of vehicle routing problems using minimum k-trees. Oper Res 1994;42:626–42.

[19] Xu J, Kelly JP. A new network flow-based tabu search heuristic for the vehicle routing problem. Transport Sci 1996;30:379–93.

[20] Ballou RH. A continued comparison of several popular algorithms for vehicle routing and scheduling. J Business Logistics 1990;11(1): 111–26.