

# Application of ACO to Vehicle Routing Problems Using Three Strategies

Yunpeng Wu

College of Computer Science and Technology  
Jilin University  
Changchun, China  
E-mail: 2412466779@qq.com

Ying Xin

College of Software  
Jilin University  
Changchun, China  
E-mail: 292793238@qq.com

Yonggang Zhang \*

College of Computer Science and Technology  
Jilin University  
Changchun, China  
E-mail: zhangyg@jlu.edu.cn  
\* Corresponding Author

**Abstract**—Ant Colony Optimization algorithm (ACO) is a viable method for attacking hard combinatorial optimization problems. In this paper, researchers applied adaptive method to the Max-Min ant system (MMAS). In every cycle of the search, researchers can determine the number of elite ants adaptively. In the search process, as long as the ant reached a certain requirement, researchers can think of them as the elite ants, and their pheromone is maintained. When updating the pheromone of path, researchers considered the distance between two points, rather than a uniform release of the pheromone in the path. Punishment and reward system was proposed. Based on the comparison the current path of the ants walked and the optimal path of last circle, the ants were grading. Ant pheromone on the first paths were strengthen, and those on the last paths and weaken. Experiments showed that, in most cases, our algorithm could find the better solution.

**Keywords**—Ant Colony Optimization; Maximum and Minimum Ant System; Combinatorial Optimization Problems; Adaptive; Elite Ants

## I. INTRODUCTION

Ant Colony Optimization (ACO) is a recently developed and populated-based approach which has been successfully applied to several NP-hard combinatorial optimization problems. ACO has been inspired by the behavior of real ant colonies [1,2,3].

The first ACO algorithm, called Ant System (AS) [4,5], which was applied to the Traveling Salesman Problem (TSP). It gave encouraging results, yet its performance was not competitive with state-of-the-art algorithms for the TSP. Therefore, one important focus of research on ACO algorithms has been the introduction of algorithmic improvements to achieve a much better performance. Typically, these improved algorithms have been tested again on the TSP. While they differ mainly in specific aspects of the search control, all these ACO algorithms are based on a stronger exploitation of the search history to direct the ants' search process. Recent research on the

search space characteristics of some combinatorial optimization problems has shown that for many problems there exists a correlation between the solution quality and the distance from very good or optimal solutions. Hence, it seems reasonable to assume that the concentration of the search around the best solutions found during the search is the key aspect that led to the improved performance shown by the modified ACO algorithms.

In the year of 2000, Stutzle presented the Max-Min Ant System (MMAS) algorithm [6]; it achieved a strong exploitation of the search history by allowing only the best solutions to add pheromone during the pheromone trail update. Also, the use of a rather simple mechanism for limiting the strengths of the pheromone trails effectively avoids premature convergence of the search. Finally, MMAS can easily be extended by adding local search algorithms. In fact, the best performing ACO algorithms for many different combinatorial optimization problems improve the solutions. MMAS differed from Ant System in several important aspects, whose usefulness researchers demonstrate by means of an experimental study. Additionally, researchers relate one of the characteristics specific to MMAS—that of using a greedier search than Ant System—to results from the search space analysis of the combinatorial optimization problems attacked.

The vehicle routing problem (VRP) is a combinatorial optimization and integer programming problem which asks "What is the optimal set of routes for a fleet of vehicles to traverse in order to deliver to a given set of customers?". It generalizes the well-known Travelling Salesman Problem (TSP) [7,8]. It first appeared in a paper by George Dantzig and John Ramser in 1959, in which first algorithmic approach was written and was applied to petrol deliveries. Often, the context is that of delivering goods located at a central depot to customers who have placed orders for such goods. The objective of the VRP is to minimize the total route cost. In 1964, Clarke and Wright improved on Dantzig and Ramser's approach using an effective greedy

approach called the savings algorithm.

The remainder of this paper is structured as follows. In Section 2, researchers introduced the background of the ACO algorithms and Vehicle Routing Problems. In Section 3, researchers introduced the improved MMAS with 3 strategies. In Section 4, researchers gave the results and analysis of the experiments. In Section 5, researchers brief summarized the whole paper and pointed out the future work.

## II. ANT COLONY OPTIMIZATION

### A. ACO algorithms

ACO algorithms make use of simple agents called ants which iteratively construct candidate solution to a combinatorial optimization problem. The ants' solution construction is guided by (artificial) pheromone trails and problem-dependent heuristic information. In principle, ACO algorithms can be applied to any combinatorial optimization problem by defining solution components which the ants use to iteratively construct candidate solutions and on which they may deposit pheromone [9]. An individual ant constructs candidate solutions by starting with an empty solution and then iteratively adding solution components until a complete candidate solution is generated. Researchers will call each point at which an ant has to decide which solution component to add to its current partial solution a choice point. After the solution construction is completed, the ants give feedback on the solutions they have constructed by depositing pheromone on solution components which they have used in their solution. Typically, solution components which are part of better solutions or are used by many ants will receive a higher amount of pheromone and, hence, will more likely be used by the ants in future iterations of the algorithm. To avoid the search getting stuck, typically before the pheromone trails get reinforced, all pheromone trails are decreased by a factor.

The ants' solutions are not guaranteed to be optimal with respect to local changes and hence may be further improved using local search methods. Based on this observation, the best performing ACO algorithms for many NP-hard static combinatorial problems are in fact hybrid algorithms combining probabilistic solution construction by a colony of ants with local search algorithms [10]. In such hybrid algorithms, the ants can be seen as guiding the local search by constructing promising initial solutions, because ants preferably use solution components which, earlier in the search, have been contained in good locally optimal solutions.

In general, all ACO algorithms for static combinatorial problems follow a specific algorithmic scheme outlined in Fig. 1. After the initialization of the pheromone trails and some parameters, a main loop is repeated until a termination condition-which may which may be a certain number of solution constructions or a given CPU-time limit — is met. In the main loop, first, the ants construct feasible solutions, and then the generated solutions are possibly improved by applying local search, finally the pheromone trails are updated. It should be noted that the ACO metaheuristic [10] is more general than the algorithmic scheme given here.

### ACO algorithm for static combinatorial problems

```

1. Begin Set parameters, initialize pheromone trails
2. While (termination condition not met) Do
3.   ConstructSolutions
4.   UpdateTrails
5. End
6. End

```

Figure 1. Algorithmic skeleton for ACO algorithms applied to static combinatorial problems.

### B. Vehicle Routing Problems

The VRP concerns the service of a delivery company. How things are delivered from one or more depots which have a given set of home vehicles and operated by a set of drivers who can move on a given road network to a set of customers. It asks for a determination of a set of routes,  $S$ , (one route for each vehicle that must start and finish at its own depot) such that all customers' requirements and operational constraints are satisfied and the global transportation cost is minimized. This cost may be monetary, distance or otherwise.

The road network can be described using a graph where the arcs are roads and vertices are junctions between them. The arcs may be directed or undirected due to the possible presence of one way streets or different costs in each direction. Each arc has an associated cost which is generally its length or travel time which may be dependent on vehicle type.

To know the global cost of each route, the travel cost and the travel time between each customer and the depot must be known. To do this our original graph is transformed into one where the vertices are the customers and depot and the arcs are the roads between them. The cost on each arc is the lowest cost between the two points on the original road network. This is easy to do as shortest path problems are relatively easy to solve. This transforms the sparse original graph into a complete graph. For each pair of vertices  $i$  and  $j$ , there exists an arc  $(i,j)$  of the complete graph whose cost is written as  $c_{ij}$  and is defined to be the cost of shortest path from  $i$  to  $j$ . The travel time is the sum of the travel times of the arcs on the shortest path from  $i$  to  $j$  on the original road graph.

Sometimes it is impossible to satisfy all of a customer's demands and in such cases solvers may reduce some customers' demands or leave some customers not served. To deal with these situations, a priority variable for each customer can be introduced or associated penalties for the partial or lack of service for each customer given [2]

The objective function of a VRP can be very different depending on the particular application of the result but a few of the more common objectives are:

- Minimize the global transportation cost based on the global distance travelled as well as the fixed costs associated with the used vehicles and drivers
- Minimize the number of vehicles needed to serve all customers Least variation in travel time and vehicle load.
- Minimize penalties for low quality service

### III. THE IMPROVED MMAS

In the past of ant colony algorithm study and MMAS algorithm is one of the most outstanding an algorithm, which can effectively balance the ant colony search solution path efficiency and expansion of the unknown path between the conflicts. However, it still has many problems, for example, in each cycle, only the performance of the best of the elitist ants to their pheromone release in path, for the ants search provides pheromone; in the elite ant pheromone release when, between any two points in the current optimal path on the release of the information was consistent, and do not consider differences in the length of the distance between two points; in a cyclic search and pheromone update before, information on all paths each will be volatile, and not for the search of the worst path do not treat.

The idea of self adaptation is to distinguish the different attributes of the problem, rather than to rely on a strategy in the process of solving the problem. For example, it can be solved by using different methods to solve the problem of difficult parts and simple parts, which can greatly improve the efficiency of solving problems. Based on the idea, researchers make the following three improvements to the MMAS algorithm:

**Strategy 1:** the number of adaptive determination of elite ants. The related work has tried to increase the number of ants. However, in the whole process of searching, the number of the elite ants is fixed, and it is set by human. In our experiments, before a cycle does not given the elite ant's number, but according to the performance of the ants to decide the ant whether as elitist ant pheromone update to the path.

**Strategy 2:** in the process of releasing pheromone in the elite ants, according to the distance between two points on the current optimal path, the pheromone is released to different two nodes. Ants in reality may not have such a capability, but this does not prevent us from using this strategy to speed up the process.

**Strategy 3:** in each cycle of the search, according to the performance of each ant's performance advantages and disadvantages. Researchers can mark the path of a certain number of ants in the loop, for example, in a limited time to the end point of the ant, or to the end of the ant colony. Among them, the number of ants is also based on the performance of ants to decide. The path is at the top of the list of several ants in the global update of the pheromone to reward system, and the path of the ants is passed by the path of punishment system.

### IV. EXPERIMENTS

In this Section, researchers will show the experimental evaluation of the proposed algorithm. All the problem instances are come from the Website (<http://neo.lcc.uma.es/vrp/>). All algorithms have been implemented in C++ and embedded into the constraint solving platform designed by ourselves. In this platform, researchers use dom/deg variable ordering because it is

easy to implement and very efficient. The experiments were carried out on a DELL Intel i7-3770 3.40GHz CPU/4GB RAM with Windows 7.0/Visual C++6.0.

#### A. Adaptive determination of the number of elite ants

In the last cycle, the optimal path length is  $L$ . In the process of searching for the optimal path, the path length of the ant  $k$  is  $L_k$ , and it is satisfied

$$\frac{L_k}{L} \leq \rho \quad (1)$$

The ant is chosen as the elite ant, the path traversed by a certain amount of information. In our experiments, researchers assign  $\rho = 1.2$ . Table 1 is the experimental data comparison of the using MMAS method to solve the vehicle routing problem. In table 1, the best data is represented by bold font. The best performance data is also used in the same instance. Through the experimental data, researchers can see that, in the three examples, the strategy is the best experimental results obtained by the two problems. Even though the best data is 10 times, the MMAS with strategy 1 is better than the basic MMAS.

#### B. Pheromone release accoring the distance

In an elitist ant pheromone release process, a release of information element content is certain, but after this elitist ant path continuous two-point release pheromones are not the same, the shorter the distance between two points, the release of information elements are also more. Researchers use the following formula to determine the ratio of  $L_i$  to the release of pheromone between two consecutive points:

$$\frac{\sum_j L_j}{\sum_k L_k} \quad (2)$$

Table 2 are the experimental data comparison of the vehicle routing planning problem of using MMAS with strategy 2. In the table 2, fixed elitist ants were a number of MMAS algorithm significantly than the other two MMAS solution effect good, but in this instance of solving A-32-k5, although the average cycle number and other little difference, but on the optimal number of cycles to far inferior to the other two MMAS. Applying the MMAS algorithm, the algorithm is better than the basic MMAS algorithm, and the MMAS algorithm is better than the fixed number of the elite ants in A-32-k5.

#### C. MMAS with Strategy 3

After ending a search cycle, each one finds the solution for ants. The results with the previous optimal solution in comparison, judging the grade, pheromone according to specific proportion add to the path. Table 3 is the experimental data comparison of the vehicle routing planning problem of using MMAS with Strategy 3.

TABLE I. COMPARISON OF BETWEEN THE MMAS WITH STRATEGY 1 AND THE TWO MMAS ALGORITHM

Instances	Algorithms	1	2	3	4	5	6	7	8	9	10	Mean
E-n22-k4	MMAS	220	398	77	15	54	721	209	561	89	<b>14</b>	235.8
	MMAS+Elite ants	28	17	14	163	<b>12</b>	17	29	33	107	114	<b>53.4</b>
	MMAS+(1)	<b>10</b>	37	156	94	164	17	203	23	41	52	79.7
A-n32-k5	MMAS	<b>88</b>	632	178	603	617	705	363	821	198	869	507.4
	MMAS+ Elite ants	511	676	593	<b>346</b>	369	771	384	553	487	347	503.7
	MMAS+(1)	591	127	987	447	<b>89</b>	754	311	284	971	389	<b>495</b>
A-n33-k5	MMAS	323	770	391	679	166	<b>66</b>	945	776	831	865	581.2
	MMAS+ Elite ants	683	543	168	387	563	103	114	602	163	<b>29</b>	335.5
	MMAS+(1)	123	<b>62</b>	120	800	290	248	523	387	106	277	<b>293.6</b>

TABLE II. COMPARISON OF BETWEEN THE MMAS WITH STRATEGY 2 AND THE TWO MMAS ALGORITHM

Instances	Algorithms	1	2	3	4	5	6	7	8	9	10	Mean
E-n22-k4	MMAS	220	398	77	15	54	721	209	561	89	<b>14</b>	235.8
	MMAS+ Elite ants	28	17	14	163	<b>12</b>	17	29	33	107	114	<b>53.4</b>
	MMAS+(2)	287	68	178	<b>44</b>	80	72	125	174	159	469	165.6
A-n32-k5	MMAS	<b>88</b>	632	178	603	617	705	363	821	198	869	507.4
	MMAS+ Elite ants	511	676	593	<b>346</b>	369	771	384	553	487	347	503.7
	MMAS+(2)	138	543	843	438	450	326	343	221	<b>114</b>	590	<b>400.6</b>
A-n33-k5	MMAS	323	770	391	679	166	<b>66</b>	945	776	831	865	581.2
	MMAS+ Elite ants	683	543	168	387	563	103	114	602	163	<b>29</b>	<b>335.5</b>
	MMAS+(2)	310	469	278	879	433	174	386	322	<b>74</b>	369	369.4

TABLE III. COMPARISON OF BETWEEN THE MMAS WITH STRATEGY 3 AND THE TWO MMAS ALGORITHM

Instances	Algorithms	1	2	3	4	5	6	7	8	9	10	Mean
E-n22-k4	MMAS	220	398	77	15	54	721	209	561	89	<b>14</b>	235.8
	MMAS+ Elite ants	28	17	14	163	<b>12</b>	17	29	33	107	114	<b>53.4</b>
	MMAS+(3)	466	67	802	<b>27</b>	399	605	55	743	192	50	340.6
A-n32-k5	MMAS	<b>88</b>	632	178	603	617	705	363	821	198	869	507.4
	MMAS+ Elite ants	511	676	593	<b>346</b>	369	771	384	553	487	347	503.7
	MMAS+(3)	689	338	848	362	931	707	652	<b>106</b>	175	173	<b>498.1</b>
A-n33-k5	MMAS	323	770	391	679	166	<b>66</b>	945	776	831	865	581.2
	MMAS+ Elite ants	683	543	168	387	563	103	114	602	163	<b>29</b>	<b>335.5</b>
	MMAS+(3)	237	459	997	997	<b>55</b>	323	827	426	473	263	505.7

## V. CONCLUSIONS

As the ant colony algorithm is more suitable for solving the problem of Vehicle Routing Problems, at present, the ant colony algorithm has been used as a kind of heuristic algorithm to solve the VRP problem. In this paper, researchers have carried out the transformation of three strategies through the MMAS ant colony algorithm, and have been verified in the problem instance. Experiments show that after the strategies and tactics of the second transformation of MMAS algorithm in finding the optimal solution in the process, whether it is the average number of cycles or at least cycles, are more efficient than the original MMAS algorithm. After the algorithm of three transformations, it is proved that the algorithm is more efficient than the original MMAS algorithm, and some examples are less than the number of examples. However, in the three kinds of strategies, the efficiency of the solution has been greatly improved.

## ACKNOWLEDGMENT

The authors of this paper express sincere gratitude to all the anonymous reviewers for their hard work. This work was supported in part by NSFC under Grant No. 61170314, No. 61373052.

## REFERENCES

- [1] M. Dorigo. "Optimization, learning, and natural algorithms" (in Italian). PhD thesis, Dip. Elettronica, Politecnico di Milano, Italy, 1992.
- [2] M. Dorigo, G. Di Caro, and L. M. Gambardella, "Ant algorithms for distributed discrete optimization". *Artificial Life*, volume 5, May 1999, pp.137–172..
- [3] M. Dorigo, V. Maniezzo, and A. Colomi, "Positive feedback as a search strategy". Technical Report 91-016, Dip. Elettronica, Politecnico di Milano, Italy, 1991.
- [4] M. Dorigo, V. Maniezzo, and A. Colomi, "The ant system: Optimization by a colony of cooperating agents", *IEEE Transactions on Systems, Man, and Cybernetics – Part B*, volume 26, June 1996, pp. 29–42.
- [5] M. Dorigo and L. M. Gambardella. "Ant colony system: A cooperative learning approach to the traveling salesman problem", *IEEE Transactions on Evolutionary Computation*, volume 1, Jan. 1997, pp. 53–66.
- [6] T. Stützle, H. H. Hoos, "Max-Min ant System", *Future Generation Computer Systems*, volume, April 2000, pp.889-914.
- [7] Y. Wang, "Hybrid Max-Min ant system with four vertices and three lines inequality for traveling salesman problem", *Soft Comput*, Volume 19, March 2015, pp. 585-596, doi: 10.1007/s00500-014-1279-8
- [8] A. Vlachos, I. Trichas, "Max-min ant system algorithm for dartboard design", *IJCIS studies*, volume 2, April 2013, pp.367-378, doi: 10.1504/IJCISSTUDIES.2013.057640.
- [9] X. Liu, C. Yang, "Optimization of Vehicle Routing Problem Based on Max-Min Ant System with Parameter Adaptation", *Proc. International Conference on Computational Intelligence and Security (CIS 2011)*, IEEE Press, Dec.2011, pp.305-307, doi:10.1109/CIS.2011.74.
- [10] B. Crawford, R. Soto, F. Johnson, E. Monfroy and F. Paredes, "A Max-Min Ant System algorithm to solve the Software Project Scheduling Problem", *Expert Syst. Appl.*, volume 41, Aug.2014, pp.6634-6645, doi: 10.1016/j.eswa.2014.05.003.