An improved ant system algorithm for the vehicle routing problem

Bernd Bullnheimer, Richard F. Hartl and Christine Strauss

Institute of Management Science, University of Vienna,
Bruenner Str. 72, A - 1210 Vienna, Austria
E-mail: bernd.bullnheimer@univie.ac.at

The Ant System is a distributed metaheuristic that combines an adaptive memory with a local heuristic function to repeatedly construct solutions of hard combinatorial optimization problems. We present in this paper an improved ant system algorithm for the Vehicle Routing Problem with one central depot and identical vehicles. Computational results on fourteen benchmark problems from the literature are reported and a comparison with five other metaheuristic approaches to solve vehicle routing problems is made.

Keywords: ant system, adaptive memory, vehicle routing, metaheuritics

1. Introduction

The Ant System, introduced by Colorni, Dorigo and Maniezzo [6], [10], [12] is a new distributed meta-heuristic for hard combinatorial optimization problems and was first used on the well known Traveling Salesman Problem (TSP).

Observations on real ants searching for food were the inspiration to imitate the behaviour of ant colonies for solving combinatorial optimization problems. Real ants are able to communicate information concerning food sources via an aromatic essence, called pheromone. They mark the path they walk on by laying down pheromone in a quantity that depends on the length of the path and the quality of the discovered food source. Other ants can observe the pheromone trail and are attracted to follow it. Thus, the path will be marked again and will therefore attract more ants. The pheromone trail on paths leading to rich food sources close to the nest will be more frequented and will therefore grow faster.

The described behaviour of real ant colonies can be used to solve combinatorial optimization problems by simulation: artificial ants searching the solution space simulate real ants searching their environment, the objective values correspond to the quality of the food sources and an adaptive memory corresponds to the pheromone trails. In addition, the artificial ants are equiped with a local heuristic function to guide their search through the set of feasible solutions.

The ant system has been applied to the Job Shop Scheduling Problem in [7], to the Graph Colouring Problem in [8], to the Quadratic Assignment Problem in [18] and to the Vehicle Routing Problem in [2].

In this paper we present an improved ant system algorithm for the Vehicle Routing Problem (VRP) with one central depot and identical vehicles and show that the ant paradigm can be used to produce competitive results. The remainder of the paper is organized as follows. First we briefly describe the VRP and the two basic ant system phases construction of vehicle routes (§2.1) and trail update (§2.2), then we present our improved ant system algorithm (§2.3). In §3 we report on computational results and a comparison with other metaheuristics for the VRP before we conclude with a discussion of our findings in §4.

2. Ant System for VRPs

The Vehicle Routing Problem can be represented by a complete weighted directed graph $G = (V, A, d)$ where $V = \{v_0, v_1, v_2, \ldots, v_n\}$ is a set of vertices and $A = \{(v_i, v_j) : i \neq j\}$ is a set of arcs. The vertex v_0 denotes the *depot*, the other vertices of V represent *cities* or *customers*, and the nonnegative weights d_{ij}, which are associated with each arc (v_i, v_j), represent the distance (or the travel time or the travel cost) between v_i and v_j. For each customer v_i a nonnegative demand q_i and a nonnegative service time δ_i is given ($q_0 = 0, \delta_0 = 0$). The aim is to find minimum cost vehicle routes where

- every customer is visited exactly once by exactly one vehicle
- all vehicle routes begin and end at the depot
- for every vehicle route the total demand does not exceed the vehicle capacity Q
- for every vehicle route the total route length (incl. service times) does not exceed a given bound L.

The VRP is a very complicated combinatorial optimization problem that has been worked on since the late fifties, because of its central meaning in distribution management. Problem specific methods (e.g., [5], [15]) as well as meta-heuristics like tabu search (e.g., [24]), simulated annealing (e.g., [19]), genetic algorithms (e.g., [16]) and neural networks (e.g., [14]) have been proposed to solve it.

The VRP and the TSP are closely related. As soon as the customers of the VRP are assigned to vehicles, the VRP is reduced to several TSPs. For that reason our ant approach is highly influenced by the TSP ant system algorithm by Dorigo et al. [12].

2.1. Construction of vehicle routes

To solve the VRP (or the TSP), the artificial ants construct solutions by successively choosing cities to visit, until each city has been visited. Whenever the choice of another city would lead to an infeasible solution for reasons of vehicle capacity or total route length, the depot is chosen and a new tour is started. For the selection of a (not yet visited) city, two aspects are taken into account: how good *was* the choice of that city, an information that is stored in the pheromone trails τ_{ij} associated with each arc (v_i, v_j), and how promising *is* the choice of that city. This latter measure of desirability, called visibility and denoted by η_{ij}, is the local heuristic function mentioned above.

With $\Omega = \{v_j \in V : v_j \text{ is feasible to be visited}\} \cup \{v_0\}$, city v_j is selected to be visited after city v_i according to a *random-proportional rule* [11] that can be stated as follows:

$$p_{ij} = \begin{cases} \frac{[\tau_{ij}]^\alpha [\eta_{ij}]^\beta}{\sum_{h \in \Omega} [\tau_{ih}]^\alpha [\eta_{ih}]^\beta} & \text{if } v_j \in \Omega \\ \\ 0 & \text{otherwise} \end{cases} \tag{1}$$

This probability distribution is biased by the parameters α and β that determine the relative influence of the trails and the visibility, respectively. For the TSP Dorigo et al. [12] define the visibility as the reciprocal of the distance. The same is done for the VRP in [2] where the selection probability is then further extended by problem specific information. There, the inclusion of savings and capacity utilization both lead to better results. On the other hand, the latter is relative costly in terms of computation time (as it has to be calculated in each step of an iteration) and will therefore not be used in this paper. Thus, we introduce the parameters f and g, and use the following *parametrical saving function* [20] for the visibility:

$$\eta_{ij} = d_{i0} + d_{0j} - g \cdot d_{ij} + f \cdot |d_{i0} - d_{0j}|$$

2.2. Trail update

After an artificial ant has constructed a feasible solution, the pheromone trails are laid depending on the objective value of the solution. In early ant system approaches all ants contributed to the trail update (see [2], [12]). In more recent papers on the TSP, better results were obtained for update rules where only the best ant contributes to the pheromone trails (see [11], [23]). In another paper Bullnheimer et al. [1] suggest to rank the ants according to solution quality and to use only the best ranked ants as well as so-called elitist ants to update the pheromone trails. For the VRP this update rule is as follows

$$\tau_{ij}^{new} = \rho\tau_{ij}^{old} + \sum_{\mu=1}^{\sigma-1} \Delta\tau_{ij}^{\mu} + \sigma\Delta\tau_{ij}^{*} \tag{2}$$

where ρ is the trail persistence (with $0 \leq \rho \leq 1$), thus the trail evaporation is given by $(1-\rho)$. Only if an arc (v_i, v_j) was used by the μ-th best ant, the pheromone trail is increased by a quantity $\Delta\tau_{ij}^{\mu}$ which is then equal to $(\sigma-\mu)/L_{\mu}$, and zero otherwise (cf. second term in (2)). In addition to that, all arcs belonging to the so far best solution (objective value L^*) are emphasized as if σ elitist ants had used them. Thus, each elitist ant increases the trail intensity by an amount $\Delta\tau_{ij}^{*}$ that is equal to $1/L^*$ if arc (v_i, v_j) belongs to the so far best solution, and zero otherwise (cf. third term in (2)).

2.3. Ant system algorithm

After initializating the ant system, the two basic steps *construction of vehicle routes* and *trail update*, are repeated for a given number of iterations. Concerning the initial placement of the artificial ants it was found in [2] that the number of ants should be equal to the number of customers and that one ant should be placed at each customer at the beginning of an iteration.

To improve the performance of a metaheuristic it is common practice to include a local search procedure and use so-called *candidate lists*[1]. Bullnheimer et al. [2] show that using the *2-opt-heuristic* for the TSP [9] to shorten the vehicle

[1] The idea of candidate lists is to concentrate the search on promising candidates thus saving

routes generated by the artificial ants, considerably improves the solution quality[2]. In addition to this straight forward local search we also introduce candidate lists for the selection of customers which are determined in the initialization phase of the algorithm. For each location v_i we sort $V \setminus \{v_i\}$ according to increasing distances d_{ij} to obtain the candidate list. The proposed ant system for the VRP can be described by the schematic algorithm given in Figure 1.

I	Initialize	
II	For I^{max} iterations do:	
	(a)	For all ants generate a new solution using Formula (1) and the candidate lists
	(b)	Improve all vehicle routes using the 2-opt-heuristic
	(c)	Update the pheromone trails using Formula (2)

Figure 1. Ant System Algorithm for VRP

3. Computational experience

In this section we discuss the parameter settings for the proposed ant system algorithm and present computational results.

3.1. Benchmark problems

The ant system for VRPs was tested on fourteen benchmark problems described in [4]. These problems contain between 50 and 199 customers in addition to the depot. The customers in problems 1-10 are randomly distributed in the plane, while they are clustered in problems 11-14. Problems 1-5 and 6-10 are identical, except that for the latter the total length of each vehicle route is bounded, whereas for the former there is no route length restriction. The same is true for the clustered problems: problems 13-14 are the counterparts of problems 11-12 with additional route length constraint. For the problems with bounded route length all customers require the same service time $\delta = \delta_1 = \ldots = \delta_n$.

run time which can be better used for further iterations. Candidate lists were first used for an ant system approach in [11].

[2] The use of a more sophisticated heuristic such as Lin-Kernighan [17] could be reasonable, but only for problems where the number of customers per tour is large.

3.2. Parameter settings

Following our findings in [2] we used n artificial ants, initially placed at the customers $v_1, \ldots, v_n$ and set $\alpha = \beta = 5$ and $\rho = 0.75$. For the other parameters we found $f = g = 2$ as a good setting. For all problems $I^{max} = 2 \cdot n$ iterations were simulated with $\sigma = 6$ elitist ants. Thus, the five best ants of an iteration further contributed to the pheromone trail update, which is also proposed in [1]. The candidate list size was set to $\lfloor n/4 \rfloor$, i.e. only one fourth of the locations, namely the closest ones, were considered[3]. Figure 2 depicts the 50-customer problem (C1) and the corresponding candidate sets (each of size 12) for two selected customers. Customer v_{11} and its candidates are marked by a bold circle, customer v_{40} and its candidates by a square, and the depot is marked by a star.

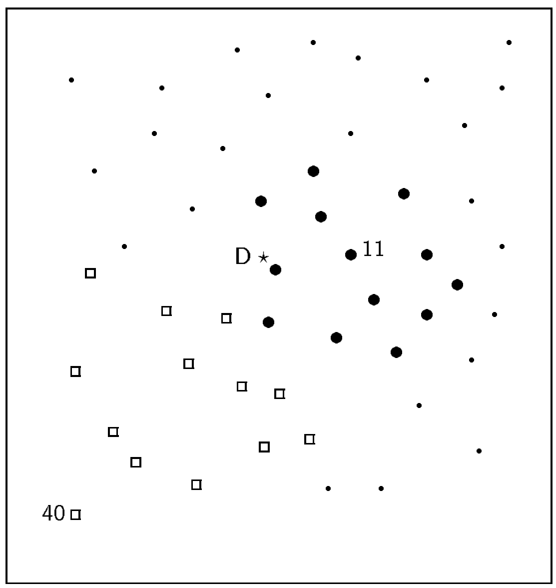

Figure 2. 50-Customer Problem (C1) with two Candidate Sets

The consequences of the use of the candidate lists were twofold: run times went down and solution quality improved. The former is obvious as time consuming calculations of p_{ij} are saved, the latter is due to the fact that selection probabilities are no longer watered down by 'very unlikely' customers. Suppose an artificial ant is located at customer v_{11} in Figure 2. Then calculating the

[3] If all *candidates* have been visited already the ant returns to the depot.

probability for selecting for example customer v_{40} to be visited next is obviously neither reasonable from a solution point of view nor from a run time point of view, especially as the corresponding pheromone trail value will be very low after a few iterations.

3.3. Ant system results

Table 1 gives the computational results for the fourteen test problems obtained by our ant system. For each problem the columns give the problem size n, the vehicle capacity Q, the maximal route length L and the service time δ. In the last three columns the best solutions obtained with our improved ant system (denoted by AS) are compared to previous ant system results (denoted by AS^{old}) from [2] and the best published solutions. Bold characters indicate that the best known solution was found by the ant system.

3.4. Comparison with other metaheuristics

In Table 2 we compare deviation from the best known solution, average deviation (denoted by Ø) as well as computation time of our ant system algorithm (denoted by AS) and five other metaheuristic approaches, where computation times were reported[4]. These are the *parallel tabu search algorithm* by Rego and Roucairol [21] (denoted by RR-PTS), *TABUROUTE*, a tabu search algorithm by Gendreau, Hertz and Laporte [13] (denoted by GHL-TS), the *first-best-admissible* variant of Osman's [19] tabu search (denoted by Osm-TS), his *simulated annealing* algorithm (denoted by Osm-SA) and a *neural network* implementation by Ghaziri [14] (denoted by Gha-NN).

The tabu search approaches seem to be superior in terms of solution quality with an average deviation of $\sim 0.5\% - 1\%$. This statement relies not only on this comparison but also on the fact that the best known solutions where optimality has not been shown were found using tabu search [22], [24]. Osman's simulated annealing is on average worse than the ant system (2.03% vs. 1.51%) but that is mainly due to its very bad performance on problem C11. Other than that, the performance of the two algorithms is fairly comparable. Only the neural network approach with an average deviation of 5.30%, four problems with deviations $> 8\%$ and only 12 problems solved, can not compete with the other metaheuristics.

[4] As there are differences in the simulation setups of the various methods a comparison on basis of execution times is hardly meaningful.

Table 1
Problem Characteristics and Solution Values for Ant System

Prob.	n	Q	L	δ	best publ.	AS	AS^{old}
Random Problems							
C1	50	160	∞	0	524.61[a]	**524.61**	**524.61**
C2	75	140	∞	0	835.26[a]	844.31	870.58
C3	100	200	∞	0	826.14[a]	832.32	879.43
C4	150	200	∞	0	1028.42[a]	1061.55	1147.41
C5	199	200	∞	0	1291.45[b]	1343.46	1473.40
C6	50	160	200	10	555.43[a]	560.24	562.93
C7	75	140	160	10	909.68[a]	916.21	948.16
C8	100	200	230	10	865.94[a]	866.74	886.17
C9	150	200	200	10	1162.55[a]	1195.99	1202.01
C10	199	200	200	10	1395.85[b]	1451.65	1504.79
Clustered Problems							
Prob.	n	Q	L	δ	best publ.	AS	AS^{old}
C11	120	200	∞	0	1042.11[a]	1065.21	1072.45
C12	100	200	∞	0	819.56[a]	**819.56**	819.96
C13	120	200	720	50	1541.14[a]	1559.92	1590.52
C14	100	200	1040	90	866.37[a]	867.07	869.86

[a] Taillard (1993)
[b] Rochat and Taillard (1995)

In the next section we briefly summarize and discuss our study and give an outlook on aspects that should be subject of further research.

4. Discussion and conclusion

In this paper we show the application and the improvement of an ant system algorithm to the VRP. The computational results confirm the positive experiences made with the ant system by applying it to the TSP [1], [11], [23]. Although some very good solutions for the VRP instances were obtained, the best known solutions for the fourteen test problems could not be improved. For practical purposes deviations up to 5% are more than acceptable as uncertainty about travel costs, demands, service times etc. makes perfect planning impossible. Therefore the presented ant system approach is a valid alternative to tackle VRPs.

A detailed investigation of parameter values (by extensive testing and/or

Table 2
Relative Percentage Deviation and Run Times for several Metaheuristic Approaches

	RR-PTS		GHL-TS		Osm-TS		Osm-SA		Gha-NN		AS	
Prob.	[%]	[min]	[%]	[min]	[%]	[min]	[%]	[min]	[%]	[min]	[%]	[min]
C1	0.00	*1.1*	0.00	*1.4*	0.00	*1.0*	0.65	*0.1*	2.78	*0.9*	0.00	*0.1*
C2	0.01	*43.4*	0.06	*39.2*	1.05	*0.8*	0.40	*59.4*	—	—	1.08	*1.3*
C3	0.17	*26.3*	0.40	*6.8*	1.44	*14.9*	0.37	*102.9*	8.14	*6.5*	0.75	*3.8*
C4	1.55	*48.5*	0.75	*54.5*	1.55	*29.4*	2.88	*71.6*	5.47	*13.2*	3.22	*18.4*
C5	3.34	*77.1*	2.42	*83.8*	3.31	*28.4*	6.55	*22.9*	8.51	*23.2*	4.03	*87.6*
C6	0.00	*2.4*	0.00	*7.8*	0.00	*1.0*	0.00	*11.6*	1.06	*4.3*	0.87	*0.1*
C7	0.00	*20.6*	0.39	*31.8*	0.15	*12.4*	0.00	*5.2*	—	—	0.72	*1.7*
C8	0.09	*18.9*	0.00	*5.9*	1.39	*32.7*	0.09	*6.1*	3.28	*18.4*	0.09	*4.8*
C9	0.14	*29.9*	1.31	*21.3*	1.85	*41.2*	0.14	*983.6*	8.73	*27.2*	2.88	*27.5*
C10	1.79	*42.7*	1.62	*44.1*	3.23	*67.1*	1.58	*40.3*	13.22	*52.4*	4.00	*81.8*
C11	0.00	*11.2*	3.01	*11.9*	0.09	*13.0*	12.85	*4.4*	5.79	*4.2*	2.22	*9.2*
C12	0.00	*1.6*	0.00	*1.7*	0.01	*5.7*	0.79	*0.8*	0.68	*1.7*	0.00	*5.0*
C13	0.59	*2.0*	2.12	*34.8*	0.31	*26.3*	0.31	*76.2*	4.37	*31.3*	1.22	*11.0*
C14	0.00	*24.7*	0.00	*29.7*	0.00	*9.7*	2.73	*5.0*	1.55	*8.5*	0.08	*5.8*
Ø	**0.55%**		**0.86%**		**1.03%**		**2.09%**		**5.30%**		**1.51%**	
	Sun Sparc 4		Silicon 4D/35		VAX 8600		VAX 8600		VAX 8600		Pentium 100	

thorough theoretical analysis) could yield better calibration and should allow further improvements. On top of that, applying a more sophisticated local search procedure such as Lin-Kernighan [17] to the generated vehicle routes, or even a VRP-specific local search that works on entire solutions, is another promising direction for future work. Finally, the algorithm seems to be well suited for parallel implementation [3]. Again, these improvements should lead to ant system algorithms that are able to find better solutions needing less run time.

Besides these methodological considerations, additional modifications of the algorithm to extensions of the VRP, e.g. multiple depots or problems with time windows are of interest.

Acknowledgements

The authors would like to thank Marco Dorigo and Herbert Dawid as well as three anonymous referees for valuable comments that helped to improve the quality of this paper.

References

[1] B. Bullnheimer, R.F. Hartl and C. Strauss, A new rank based version of the ant system: a computational study, Working Paper No.1, SFB Adaptive Information Systems and Modelling in Economics and Management Science, Vienna (1997).

[2] B. Bullnheimer, R.F. Hartl and C. Strauss, Applying the Ant System to the Vehicle Routing Problem, Paper presented at 2nd International Conference on Metaheuristics, Sophia-Antipolis, France (1997).

[3] B. Bullnheimer, G. Kotsis and C. Strauss, Parallelization Strategies for the Ant System, Paper presented at Conference on High Performance Software for Nonlinear Optimization: Status and Perspectives, Ischia, Italy (1997).

[4] N. Christofides, A. Mingozzi and P. Toth, The Vehicle Routing Problem, in: *Combinatorial Optimization,* ed. N. Christofides, A. Mingozzi, P. Toth and C. Sandi (Wiley, Chichester, 1979).

[5] G. Clarke and J.W. Wright, Scheduling of Vehicles from a Central Depot to a Number of Delivery Points, Oper. Res. 12(1964)568.

[6] A. Colorni, M. Dorigo and V. Maniezzo, Distributed Optimization by Ant Colonies, in: *Proc. Europ. Conf. Artificial Life,* ed. F. Varela and P. Bourgine, (Elsevier, Amsterdam, 1991).

[7] A. Colorni, M. Dorigo, V. Maniezzo and M. Trubian, Ant system for Job-Shop Scheduling, Belgian Journal of Operations Research, Statistics and Computer Science 34(1994)39.

[8] D. Costa and A. Hertz, Ants can colour graphs, J. Oper. Res. Soc. 48(1997)295.

[9] G.A. Croes, A Method for solving Traveling-Salesman Problems, Oper. Res. 6(1958)791.

[10] M. Dorigo, Optimization, Learning and Natural Algorithms, Doctoral Dissertation, Politecnico di Milano, Italy (1992), (in Italian).

[11] M. Dorigo and L.M. Gambardella, Ant Colony System: A Cooperative Learning Approach to the Traveling Salesman Problem, IEEE Trans. Evol. Comput. 1(1997)53.

[12] M. Dorigo, V. Maniezzo and A. Colorni, Ant System: Optimization by a Colony of Cooperating Agents, IEEE Trans. Sys., Man, Cybernetics 26(1996)29.

[13] M. Gendreau, A. Hertz and G. Laporte, A Tabu Search Heuristic for the Vehicle Routing Problem, Manag. Sci. 40(1994)1276.

[14] H. Ghaziri, Supervision in the Self-Organizing Feature Map: Application to the Vehicle Routing Problem, in: *Meta-Heuristics: Theory & Applications,* ed. I. Osman and J. Kelly (Kluwer, Boston, 1996).

[15] B.E. Gillett and L.R. Miller, A Heuristic Algorithm for the Vehicle Dispatch Problem, Oper. Res. 22(1974)340.

[16] H. Kopfer, G. Pankratz and E. Erkens, Entwicklung eines hybriden Genetischen Algorithmus zur Tourenplanung, Oper. Res. Spekt. 16(1994)21.

[17] S. Lin and B.W. Kernighan, An Effective Heuristic Algorithm for the Traveling Salesman Problem, Oper. Res. 21(1993)498.

[18] V. Maniezzo, A. Colorni and M. Dorigo, The Ant System applied to the Quadratic Assignment Problem, Technical Report IRIDIA / 94-28, Universite Libre de Bruxelles (1994).

[19] I. Osman, Metastrategy simulated annealing and tabu search algorithms for the vehicle routing problem, Ann. Oper. Res. 41(1993)421.

[20] H. Paessens, The savings algorithm for the vehicle routing problem, Eur. J. Oper. Res. 34(1988)336.

[21] C. Rego and C. Roucairol, A Parallel Tabu Search Algorithm Using Ejection Chains for the Vehicle Routing Problem, in: *Meta-Heuristics: Theory & Applications* , ed. I. Osman and J. Kelly (Kluwer, Boston, 1996).

[22] Y. Rochat and E.D. Taillard, Probabilistic Diversification and Intensification in Local Search for Vehicle Routing, J. Heuristics 1(1995)147.

[23] T. Stuetzle and H. Hoos, The MAX-MIN Ant System and Local Search for the Traveling Salesman Problem, *Proc. of ICEC'97 - IEEE 4th International Conference on Evolutionary Computation,* (1997)308.

[24] E. Taillard, Parallel Iterative Search Methods for Vehicle Routing Problems, Networks 23(1993)661.