Parameter Tuning for the Ant Colony Optimization Algorithm used in ISR systems

P. Stodola, J. Mazal, and M. Podhorec

***Abstract*—**This paper deals with the Ant Colony Optimization (ACO) algorithm developed at University of Defence, Brno, Czech Republic. This algorithm is a metaheuristic algorithm designed for solving the Multi-Depot Vehicle Routing Problem (MDVRP). The algorithm has been integrated into the Tactical Decision Support System (TDSS) which is aimed at supporting commanders in their decision-making processes. TDSS contains several tactical models based on the MDVRP problem. This paper is aimed particularly at parameter tuning for the ACO algorithm.

***Keywords*—**Parameter tuning, ant colony optimization, multi-depot vehicle routing problem, ISR system.

I. Introduction

MULTI-DEPOT Vehicle Routing Problem (MDVRP) is a famous problem formulated in 1959 [1]. There are many real applications based on this problem, particularly in the areas of transportation, distribution and logistics.

The problem is based on computing optimal routes for a fleet of vehicles to drop off goods or services at multiple destinations (customers). The vehicles can start from multiple depots, each located in a different place. A simple illustration of the problem with 3 depots, 4 vehicles, and 10 customers (nodes) is shown in Fig. 1.

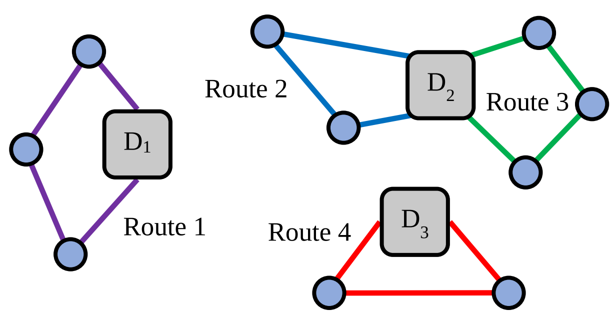

Fig. 1 Example of the MDVRP

MDVRP problem is an NP-hard problem, therefore polynomial-time algorithms are unlikely to exist [2]. Since the existence of this problem, a lot of methods has been proposed to search for solutions. Many of them are heuristic or metaheuristic algorithms based on stochastic and probabilistic approaches.

At University of Defence, Brno, Czech Republic, we proposed and developed an algorithm based on the Ant Colony Optimization (ACO) theory [3]. This algorithm is a probabilistic technique for solving computational problems.

The algorithm has been subsequently implemented and integrated into our Tactical Decision Support System (TDSS) which is designed to support commanders in their decision-making process on the tactical level.

Development of TDSS was started in 2006 and new functions and models are still created and implemented. TDSS contains several tactical models which are based on MDVRP problem. The models are solved by our ACO algorithm. Some of the models are as follows:

- Optimal distribution of unattended ground sensors (UGS) in the area of interest;
- Optimal logistics for units on the battlefield;
- Optimal reconnaissance of the area of interest via a fleet of unmanned aerial vehicles (UAVs);
- Optimal reconnaissance of the area of interest via ground elements (scouts or unmanned ground vehicles).

TDSS system is planned to be a part of ISR systems (Intelligence, Surveillance and Reconnaissance) currently used in the military intelligence to gather, analyze, evaluate and distribute intelligence information. TDDS serves for analysis of such information and decision support of commanders. More details about the TDSS system can be found in [4] and [5].

II. Objectives

This paper does not deal with the algorithm in detail. We have already published the principle of the algorithm which can be found in [5] or [6].

The main aim of this paper is to tune the parameters of the algorithm. To do this, we used empirical approach. The parameters discussed below are as follows:

- Mode of selecting depots;
- Mode of updating pheromone trails;
- Mode of selecting unvisited nodes;
- Number of ants in colonies (n_a);
- Total number of generations (n_g);
- Pheromone evaporation coefficient (ρ);
- Pheromone updating coefficient (σ);

P. Stodola is with University of Defence, Brno, Czech Republic (phone: +420 973 442 474; e-mail: petr.stodola@unob.cz).

J. Mazal is with University of Defence, Brno, Czech Republic (e-mail: jan.mazal@unob.cz).

M. Podhorec is with University of Defence, Brno, Czech Republic (e-mail: milan.podhorec@unob.cz).

- Pheromone repelling coefficient (ψ);
- Coefficients to control the influence of pheromone trails (α), distance between nodes (β), distance between the node and depot (γ), and load in the node (δ) when computing probabilities for visiting individual nodes.

More information about roles and purposes of the parameters mentioned above are in [6] along with possible values and their meaning.

III. Parameter Tuning

Empirical approach was used when searching for optimal values of parameters mentioned in Section II. We present some results here, although there is only room to show a minor part of them. All parameters were tested on 20 different problems (from tens to several hundred of nodes) and at least 100 executions of every set of values were processed to confirm the results statistically. We are also aware of the dependences between some parameters, yet we could not cover them all as the multidimensional space is too large.

A. *First Set of Parameters*

Fig. 2 compares the results of experiments for various values of parameters as follows: *mode of selecting depots* (red color), *mode of updating pheromone trails* (green color), and *mode of selecting unvisited nodes* in connection with the *pheromone repelling coefficient* (blue color; values of the repelling coefficient are written in brackets). All three sets of experiments were conducted independently on one another with constant values of other parameters.

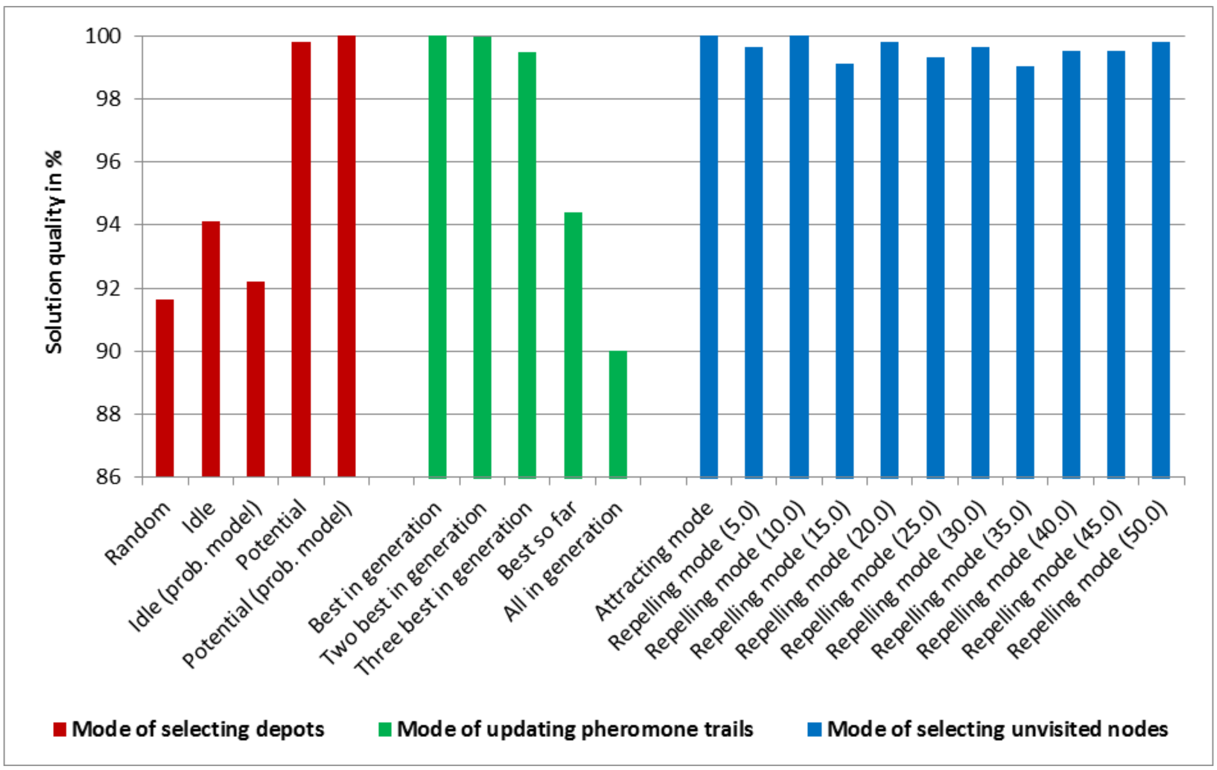

Fig. 2 Comparing results for three sets of experiments

All results are expressed as a solution quality in % (compared with the best result of a given set). When analyzing the graph, we can see that even the worst setting is able to achieve at least 90% of the best. From the first set (red color), we can conclude that there are substantial differences between individual methods; the ideal method of selecting depots is the *Selection of a depot with the greatest potential (probability model)*.

The second set of experiments covers the *mode of updating pheromone trails* (green color). The worse results when using *all solutions in a generation* are apparent. Also using only *the best solution found* so far is not ideal as it often causes getting stuck in a local optimum. The best variant is updating trails according to *the best solution found in a current generation* as this method ensures a diversity of solutions and thus prevents remaining in a local optimum.

The last set deals with the *mode of selecting unvisited nodes* (blue color). There are two options available: *attracting mode* and *repelling mode*. In the second case, the repelling coefficient is of importance (values are stated in brackets). The graph shows very similar results for all variants (the biggest error smaller than 1%). It follows that the best choice is the *attracting mode* as it is faster than *repelling mode*.

B. *Number of Ants and Generations*

Fig. 3 plots the solution quality as a function of the *number of ants* (n_a) and *generations* (n_g). It is apparent that the quality increases steadily with the number of ants and generations.

The graph also shows that the solution quality better than 80% can be achieved only with values $n_a \geq 30$ and $n_g \geq 180$ (it, however, applies only for the specific case where other parameters influencing the speed of convergence are constant).

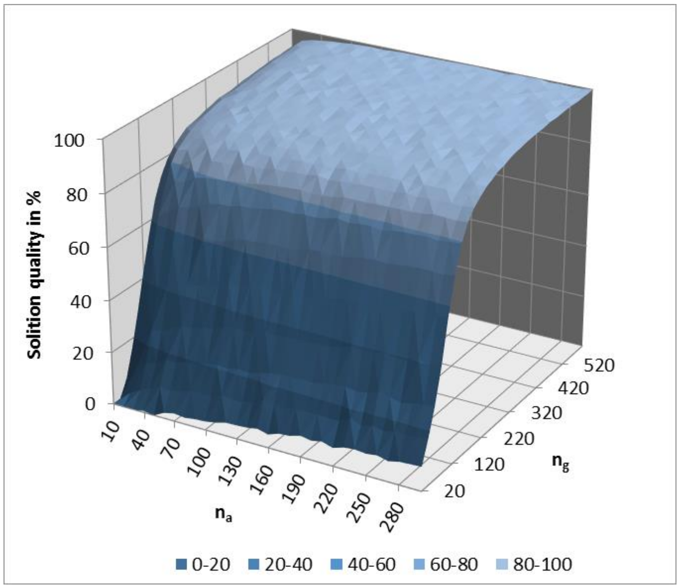

Fig. 3 Solution quality as a function of the number of ants and generations

The similar graph is shown in Fig. 4 where the solution quality is shown as a function of the number of generations for n_a = 10, n_a = 50, and n_a = 200, respectively. There we can see that solution converges quickly and the solution quality better than 90% can be achieved for n_g > 350 even for n_a = 10; then the quality is improving just slightly. The graph also shows that the influence of the number of ants is not substantial (average difference about 4.3% between n_a = 50 and n_a = 200, and about 11.3% between n_a = 10 and n_a = 200).

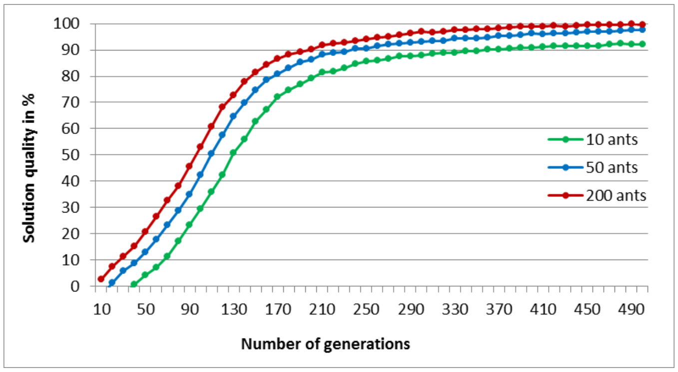

Fig. 4 Solution quality as a function of the number of generations

Fig. 5 presents the solution quality depending on the number of ants in connection with the number of nodes N (for N = 50, N = 100 and N = 200). From the graph, we can conclude that it is sufficient to set n_a = 3N (solution quality better than 98% of the best solution found).

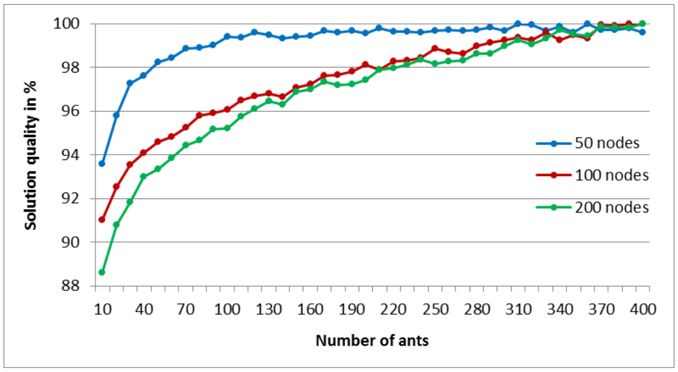

Fig. 5 Connecting the number of ants with the number of nodes

C. *Pheromone Evaporation and Updating Coefficients*

Next, the *pheromone evaporation coefficient* ρ and *updating coefficient* σ will be discussed. These two control pheromone trails belonging to individual colonies. The graph in Fig. 6 plots the solution quality as a function of these coefficients. The graph shows that the solution quality does not depend on the value of the pheromone updating coefficient. On the other hand, the value of pheromone evaporation coefficient is of importance there. The solution quality is improving quickly up to $\rho \leq 0.03$, then for bigger values it is decreasing again.

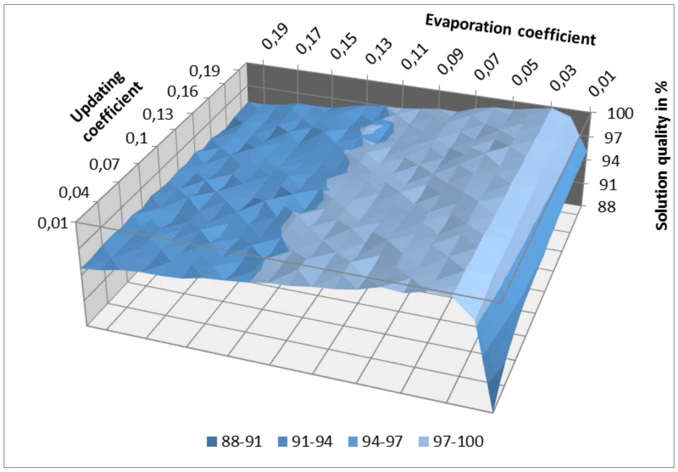

Fig. 6 Solution quality as a function of the pheromone evaporation and updating coefficients

D. *Probabilities coefficients α, β, γ, δ*

The last parameters to look at are *coefficients* α, β, γ, δ controlling probabilities of adding so far unvisited nodes on the route. The setting of these coefficients depends widely on the type of the task to be solved. Fig. 7 presents the solution quality as a function of coefficients α, β, and Fig. 8 as a function of coefficients γ, δ.

The first graph (Fig. 7) shows that both α and β needs to be at least 1. The bigger the values for both coefficients, the faster the convergence to some local optimum. It follows that with bigger values we are able to find a very good solution quickly; with smaller values, we need more generations to pass but the solution is a little bit better. It seems that in most cases the best values are $\alpha = 1$ and $\beta \in \langle 1; 3\rangle$.

When searching for the best values for coefficients γ and δ (Fig. 8), we find the situation different. Neither the influence of the distance between the node and depot (γ) nor the influence of ant's capacity (δ) when computing probabilities proved to affect the solution positively in our algorithm, thus the best values are $\gamma = 0$ and $\delta = 0$.

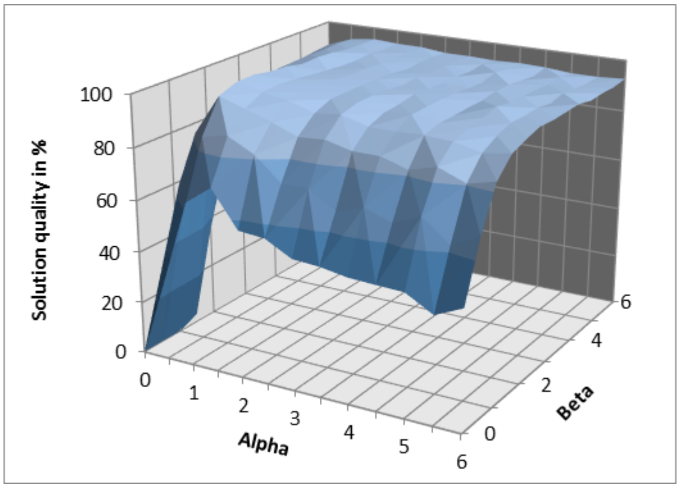

Fig. 7 Solution quality as a function of coefficients α, β

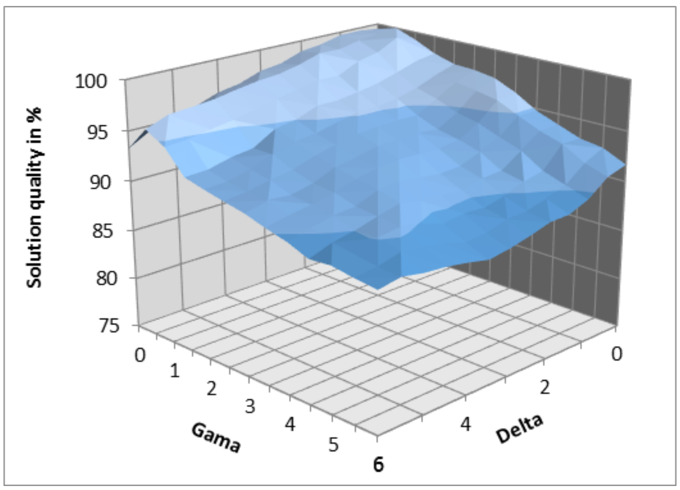

Fig. 8 Solution quality as a function of coefficients γ, δ

IV. Results and Conclusions

We verified the ACO algorithm on benchmarks problems consisting of Cordeau's MDVRP instances [7]. Values of parameters of the algorithm were set according to our experiences gained from the parameter tuning process described in Section III.

Table 1 presents the results for the benchmark instances. We conducted 100 tests on each instance and registered the best solution found, the mean along with the standard deviation. The best known solutions are received from [7].

The results show that error (the difference between our solution and the best known solution) is in no case bigger than 3%. In two cases (p01 and p12), we managed to find the best known solution.

Table 1 Results for MDVRP benchmark instances

Inst.	NoN	NoD	BKS	OBS	Mean	Stdev	Error
p01	50	4	576.87	576.87	583.15	6.50	0.00%
p02	50	4	473.53	475.86	482.86	3.44	0.49%
p03	75	5	641.19	644.46	650.04	4.12	0.51%
p04	100	2	1001.59	1018.49	1035.39	5.69	1.69%
p05	100	2	750.03	755.71	763.09	3.68	0.76%
p06	100	3	876.50	885.84	899.51	4.89	1.07%
p07	100	4	885.80	895.53	912.48	5.62	1.10%
p08	249	2	4420.95	4445.51	4572.23	66.75	0.56%
p09	249	3	3900.22	3990.19	4145.33	96.89	2.31%
p10	249	4	3663.02	3751.50	3864.92	50.21	2.42%
p11	249	5	3554.18	3657.16	3760.60	38.94	2.90%
p12	80	2	1318.95	1318.95	1320.48	1.90	0.00%
p15	160	4	2505.42	2510.11	2576.27	18.46	0.19%
p18	240	6	3702.85	3741.80	3812.25	37.22	1.05%
p21	360	9	5474.84	5631.12	5788.19	46.64	2.85%

NoN – number of nodes, NoD – number of depots
BKS – best known solution, OBS – our best solution

References

[1] G. B. Dantzig, and J. H. Ramser, "The Truck Dispatching Problem," in *Management Science*, vol 6, no 1, pp. 80-91, 1959.

[2] M. R. Garey, and D. S. Johnson, *Computers and Intractability: A Guide to the Theory of NP-Completeness*. New York: W. H. Freeman & Co., 1990.

[3] M. Dorigo, "Optimization, Learning and Natural Algorithms," Ph.D. dissertation, Milan: Politecnico di Milano, 1992.

[4] P. Stodola, and J. Mazal. "Tactical and Operational Software Library," in *International Conference on Military Technologies 2015*, Brno: University of Defence, pp. 347-350, 2015.

[5] P. Stodola, and J. Mazal. "Tactical Models based on a Multi-Depot Vehicle Routing Problem Using the Ant Colony Optimization Algorithm," in *International Journal of Mathematical Models and Methods in Applied Sciences*, vol. 9, no. 1, pp. 330-337, 2015.

[6] P. Stodola, and J. Mazal, "Applying the Ant Colony Optimisation Algorithm to the Capacitated Multi-Depot Vehicle Routing Problem," in *Int. J. Bio-Inspired Computation* [approved for publication 2015].

[7] NEO Web, *Networking and Emerging Optimization*. Malaga: University of Malaga, 2015, http://neo.lcc.uma.es/vrp/vrp-instances/multiple-depot-vrp-instances/ [Accessed on 30 August 2015].

[8] A. Krzyzak, and D. Valis, "Selected Reliability Measures of Composites with Natural Fibres Tested in Climatic Environment," in *International Conference on Military Technologies 2015*, Brno: University of Defence, pp. 81-87, 2015.

[9] M. Rybansky, "Soil Trafficability Analysis," in *International Conference on Military Technologies 2015*, Brno: University of Defence, pp. 295-299, 2015.

[10] M. Blaha, and K. Silinger, "Task Formation Commander`s Operation Order for Fire Support," in *Applied Mathematics, Computational Science and Engineering*, Varna: Europment, pp. 380-388, 2014.

[11] M. Blaha, K. Silinger, "Setting a Method of Determination of 'Fire for Effect' Firing Data and Conversion of the METCM into the METEO-11," in *International Journal of Circuits, Systems and Signal Processing*, vol. 9, pp. 306-313, 2015.