Defence Science Journal, Vol. 67, No. 5, September 2017, pp. 529-535, DOI : 10.14429/dsj.67.10530

Model of Optimal Cooperative Reconnaissance and its Solution using Metaheuristic Methods

Petr Stodola* and Jan Mazal

University of Defence, Brno, Czech Republic
**E-mail: Petr.Stodola@unob.cz*

ABSTRACT

The model of optimal cooperative reconnaissance as a part of the tactical decision support system to aid commanders in their decision-making processes is presented. The model represents one of the models of military tactics implemented in the system to plan the ground reconnaissance operation for the commander optimally. The main goal of the model is to explore the area of interest by multiple military elements (scouts, UAVs, UGVs) as quickly as possible. A metaheuristic solution to this problem which combines two probabilistic methods: simulated annealing and the ant colony optimisation algorithm is proposed. In the first part of this study, the optimal cooperative reconnaissance problem is formulated. Then, metaheuristic solution, which is composed of three independent steps, is presented. Finally, experiments are conducted to verify the approach to this problem.

Keywords: Optimal cooperative reconnaissance; Decision support system; Metaheuristic solution; Simulated annealing; Ant colony optimisation; Multi-depot vehicle routing problem

1. INTRODUCTION

Decision support systems (DSS) are powerful tools to support people in decision making. They are aimed at decisions about an effective and correct solution of a problem at hand. DSS can be defined as an interactive, flexible, and adaptable computer-based information system that aids the process of decision making[1].

These systems have also become trends in the military. Modern armies all over the world started research and development projects of such systems, especially for supporting a commander with his/her decision making process.

The concept of such a system is to find a solution to a specific problem (military operation) and present it to the commander. It consists of models of military tactics. Each model can solve the corresponding task.

One of many important models in military tactics is presented. It is the optimal cooperative reconnaissance which is a problem when an area of interest needs to be searched (observed) by multiple military elements (scouts, UAVs, UGVs) as quickly as possible. Searching can be conducted from multiple positions (waypoints). In these waypoints, each element is capable to observe some portion of the area of interest; observed area is the area where the element has a visual line of sight.

2. LITERATURE REVIEW

Research publications concerning decision support systems used in the military to control small units is limited. Although it is evident that modern armies invest into research on computer systems to support their commanders on the battlefield, most research projects are confidential, thus detailed information is not available.

One of the projects where there is some information on the subject, is Deep Green. The project started in 2008 and main goal was ‘to using simulation to support ongoing military operations while they are being conducted’[2]. Assumed users of the system are commanders of the US Army. We have however no information about the models implemented in the system and the current state of the project.

Optimal cooperative reconnaissance problem is a new problem formulated. The proposed solution to this problem combines two well-known metaheuristic methods: simulated annealing and the ant colony optimisation (ACO) algorithm. Former is used to optimise positions of a number of waypoints in the area of interest, latter to optimise routes between waypoints to be visited by ground elements.

We can find a number of publications which deal with position optimisation using simulated annealing[3-6]. We can also find a lot of publications concerning routes optimisation using ACO theory to find optimal path in a graph[7,8], or to solve the travelling salesman problem[9,10], or the vehicle routing problem[11,12].

3. PROBLEM FORMULATION

Let $S=\{S_1,S_2,...,S_M\}$ be a set of homogeneous elements and $W=\{W_1,W_2,...,W_N\}$ be a nonempty set of waypoints. The number M of elements is known as well as their initial positions; there is no information, however, about the number and positions of waypoints.

Let A be an area of operations and $A_I \subseteq A$ be an area of interest. As elements are homogeneous, each one from the set S located in a waypoint $W_i \in W$ is capable to observe an

Received : 26 September 2016, Revised : 12 July 2017
Accepted : 28 July 2017, Online published : 19 September 2017

area $A_i \subseteq A_I$ for all $i = 1$ to N. The total area observed from all waypoints A_O is defined according to Eqn. (1).

$$A_O = \bigcup_{i=1}^{N} A_i \tag{1}$$

Let α be a coefficient representing the proportion of the observed area A_O to the whole area of interest A_I according to Eqn. (2).

$$\alpha = \frac{A_O}{A_I} \tag{2}$$

Let $G = (V, E)$ be a graph where $V = S \cup W$ is a set of vertices (nodes) and E is a set of edges between all nodes in a graph. For every edge E_{ij} between existing nodes V_i and V_j $(i \neq j)$, a non-negative cost c_{ij} is associated which can be interpreted as a travel cost or as a travel time.

To observe the entire area A_O, every waypoint $W_i \in W$ has to be visited just once by any element, i.e. there is a tour through the graph G which connects all waypoints. Equation (3) presents the main goal of the task which is to minimise the total cost spent by all elements.

$$\min \sum_{i \in V} \sum_{j \in V} c_{ij} \cdot x_{ij}, \tag{3}$$

where $x_{ij} = \begin{cases} 1 \text{ if } i \text{ preceeds } j \text{ on the tour,} \\ 0 \text{ otherwise.} \end{cases}$

Moreover, several constraints exist. Equation (4) indicates that at least a minimum portion of the area of interest has to be observed. Equation (5) illustrates that each waypoint has to be visited only once, Eqn (6) forces each element to leave its initial position no more than once, and Eqn (7) indicates that each element has to return back to its initial position at the end of the task.

$$\alpha \geq \alpha_{\min}, \tag{4}$$

$$\sum_{i \in V} x_{ij} = 1 \quad \text{for all } j \in W, \tag{5}$$

$$\sum_{j \in W} x_{ij} \leq 1 \quad \text{for all } i \in S, \tag{6}$$

$$\sum_{j \in W} x_{ij} = \sum_{k \in W} x_{ki} \quad \text{for all } i \in S. \tag{7}$$

4. METAHEURISTIC SOLUTION

The proposed metaheuristic solution to this problem employs two probabilistic methods: Simulated Annealing and Ant Colony Optimisation algorithm. The solution is found in three successive steps as follows:

- Determining the total number of waypoints N
- Computing positions of waypoints W
- Computing the tour of elements S between waypoints W.

4.1 Determining the Number of Waypoints

In this first step, the total number of waypoints N is determined. This number is constant for the rest of the process. For determining this number, two conditions are applied:

- The number should be as small as possible to accomplish the main goal of the task, i.e. to minimise the total cost spent by all elements. This assumption is true unless the new waypoints are inserted and lie on at least one of the routes of elements. Since these routes are not known in this step, the assumption is considered valid.
- Condition (4) must not be violated, i.e. the proportion of the observed area A_O to the whole area of interest A_I must be equal to or greater than the value of coefficient $\alpha_{\min}$ (preset according to user's requirements).

In this paragraph, a straightforward method of determining the number of waypoints is introduced. Let K be a set of coefficients $K = \{\alpha_1, \alpha_2, ...\}$, where α_k is defined according to Eqn (2) as a proportion of an observed area A_O to the whole area of interest A_I when exactly k waypoints are used for observation. From this set K, we need to select the coefficient α_k which keeps both above mentioned conditions. This can be done easily by picking up the nearest coefficient α_n which is equal to or greater than the coefficient $\alpha_{\min}$. Then, the number of waypoints is set as $N = n$.

Although this method is clear and easy to implement, it is not feasible in practical applications. It requires computation of coefficients α_k for $k = 1$ to N (the process is stopped when α_k is equal to or greater than $\alpha_{\min}$; this is when $k = N$). Computation of each coefficient α_k requires the optimisation of k waypoints' positions in the area of operations in order to maximise the observed area A_O.

To increase the efficiency of the presented principle, we developed an improved algorithm. It is as shown in Fig. 1.

```
1. Compute N_min
2. Set k_min = k_max = N_min
3. While A*_O(k_max) = 0
     Set k_min = k_max + 1
     Set k_max = 2 · k_max
4. While k_min ≠ k_max
     Set m = (k_min + k_max) / 2
       If A*_O(m) = 0 then
         Set k_min = m + 1
       else
         Set k_max = m
5. Return k_min
```

Figure 1. Algorithm of determining the number of waypoints in pseudocode.

In point 1, the minimum number of waypoints $N_{\min}$ is computed according to Eqn. (8).

$$N_{\min} = \left\lceil \frac{A_I}{A_w} \right\rceil, \tag{8}$$

where A_I is the whole area of interest, A_w is the maximum possible area observed by elements from any waypoint.

The area A_w is computed according to Eqn. (9) as a circular area with radius v representing the range of visibility of elements.

530

$$A_w = \pi \cdot v^2 \tag{9}$$

Variables $k_{\min}$ and $k_{\max}$ (introduced in point 2) represents the lower and higher limits for number of waypoints. These limits are determined in point 3. There is a loop where the function $A_O^*(k_{\max})$ is evaluated; this function is defined according to Equation (10).

$$A_O^*(k) = \begin{cases} A_O(k) & \text{for } \alpha_k \geq \alpha_{\min} \\ 0 & \text{otherwise} \end{cases}, \tag{10}$$

where $A_O(k)$ is the observed area from waypoints, α_k is the proportion of $A_O(k)$ to the area of interest A_I.

In point 4, both limits are narrowed iteratively via the interval halving (bisection) method. The loop ends when both limits $k_{\min}$ and $k_{\max}$ are equal and this value is returned in point 5.

Computation of the observed area $A_O(k)$ (and coefficient α_k respectively), is the most challenging problem as it requires optimisation of positions of k waypoints. It is done in the same way as is introduced in the next section. Values of input parameters are, however, set in order to maximally accelerate the whole process; the result is then only a rough approximation of optimal waypoints' positions. It is acceptable since the goal of this step is just to determine the number of waypoint necessary, not to optimise their positions (which is the goal of the next step introduced below).

4.2 Computing Positions of Waypoints

When the total number of waypoints is known, their positions need to be optimised in order to maximise the observed area A_O. To do this, we choose simulated annealing which is a generic probabilistic metaheuristic method. The reason for choosing this algorithm is that the method proved to be very suitable in the similar problem which we dealt with in the past. The problem was about the optimal deployment of sensors of a monitoring system[13] in the area of interest to guard the area against intruders.Besides simulated annealing, we implemented and verified two other metaheuristic methods for this problem: the genetic algorithm and self-organising migrating algorithm (SOMA). Simulated annealing provided high-quality solutions to the problem; moreover, the quality was higher when compared with both the genetic algorithm and SOMA. Comparison of results of these methods applied on the problem of deploying the sensors can be found in[14].

The simulated annealing algorithm in peudocode is as shown in Fig. 2. A solution x is represented by a vector of $2N$ independent variables; each pair corresponds to the position of one waypoint in two-dimensional space.

The most important part of the algorithm is the process of transformation of a current solution x to a new solution x' (see step 5). It is based on addition or subtraction of random values with normal distribution from individual variables. The range of those random values is controlled by the value of temperature T, i.e. the greater temperature the greater changes.

The transformed solution x' is then evaluated, i.e. the observed area A'_O is computed for this newly created solution. In step 6, the original solution x is replaced by the new solution x' with the probability based on the Metropolis criterion according to Eqn. (11).

1. Generate a random solution x
2. Set $T = T_{\max}$
3. While $T > T_{\min}$
4.   Set $k = 0$
5.   While $k < k_{\max}$
6.   Transform $x \rightarrow x'$
7.   Replace x with x' according to the Metropolis criterion
8.   Save the best solution if found
9.   Set $k = k + 1$
10.   Decrease T
11. Return the best solution found

Figure 2. Simulated annealing in pseudocode.

$$p(x \rightarrow x') = \begin{cases} 1 & \text{for } A'_O \geq A_O \\ e^{-\frac{A_O - A'_O}{T}} & \text{otherwise} \end{cases} \tag{11}$$

The process of evaluation of a given solution consists in computing the observed area A_O according to Eqn. (1). In order to accelerate this process. Two approaches have been implemented:

- The area of operations is rasterised with the specified step. Observed area is computed as a sum of squares where a VLOS exists from an arbitrary waypoint.
- The visibility is computed via the floating horizon algorithm[15].

4.3 Computing the Tour of Elements

When positions of waypoints W in the area of operations are found, the last step of computing the tour of elements is conducted. The goal is to visit all of the waypoints in an optimal manner in accordance with Eqn. (3). As elements are homogeneous, each waypoint can be visited just once by whichever element.

This task can be transformed to a well-known problem which is known as the Multi-Depot Vehicle Routing Problem[16]. MDVRP is an NP-hard problem, therefore polynomial-time algorithms are unlikely to exist[17].

MDVRP consists in computing optimal routes for a fleet of vehicles to drop off goods or services at multiple destinations (customers). The vehicles might start from multiple depots, each located in a different place. The analogy to our problem is straightforward: initial positions of elements can be seen as depots, and positions of waypoints represent customers.

Solution to this problem based on the Ant Colony Optimisation (ACO) theory is developed. The proposed ACO algorithm is a probabilistic metaheuristic method[18] inspired by nature where ants explore their environment to find food.

The reason why only the ACO algorithm was considered as a method to handle this step is in the very high quality results which has been achieved on benchmark MDVRP instances. We managed to compete and even outperform the other state-of-the-art methods published. More information and details about the algorithm and conducted experiments are available[19,22].

Travelling costs between all elements and waypoints are computed by our own model of element movement. This model takes into account features of the environment of the operation divided into 8 independent layers as follows:

(i) Terrain relief layer
(ii) Soil layer
(iii) Vegetation layer
(iv) Water surface layer
(v) Infrastructure layer
(vi) Build-up area layer
(vii) Meteorological conditions layer
(viii) Threat layer.

All of these layers influence the movement of elements in the area of operations. At each location at this area, the cost of the element to cross over this location is computed according to Eqn. (12).

$$c(x,y)=\sum_{i=1}^{8} w_i \cdot f_i(x,y) \tag{12}$$

where $c(x,y)$ is the cost of an element to cross over the location given by coordinates x,y,

w_i is a weight reflecting the influence of i-th layer on the model.

$f_i(x,y)$ is a function of i-th layer. The function $f_i(x,y)$ incorporates features and properties of i-th layer and their relation to the cost value at location x,y. Thus, there are 8 different functions, one for each layer. The values of weights, i.e. the extent to which each layer participates in the model, depends on the predefined requirements and the type of operation at hand.

As costs are computed for all points of the grid placed over the area of operations, the cost matrix is created. This matrix is as shown in Fig. 3 on the right as a grey scale image (the darker colour the higher cost). The cost matrix can be transformed into a non-oriented graph in which the shortest path from the initial to the destination position is computed via the modified Floyd-Warsh all algorithm[20]. Example of such a path is also as shown in Fig. 3 as a red line.

Optimal paths are computed for every pair of waypoints and elements; thus, there needs to be computed $\frac{1}{2}\cdot(N+M)\cdot(N+M-1)$ optimal paths. The total cost of an element to move between two locations is computed as a sum of costs in all locations along the optimal path.

Finally, we have enough information to execute our ACO algorithm. As a result, we obtain the list of waypoints for each element to be visited in the correct order; elements conduct the observation in each waypoint along their tour. In this way, at the end the whole process, the area A_O is observed in accordance with the main goal of the task given by Eqn. (3).

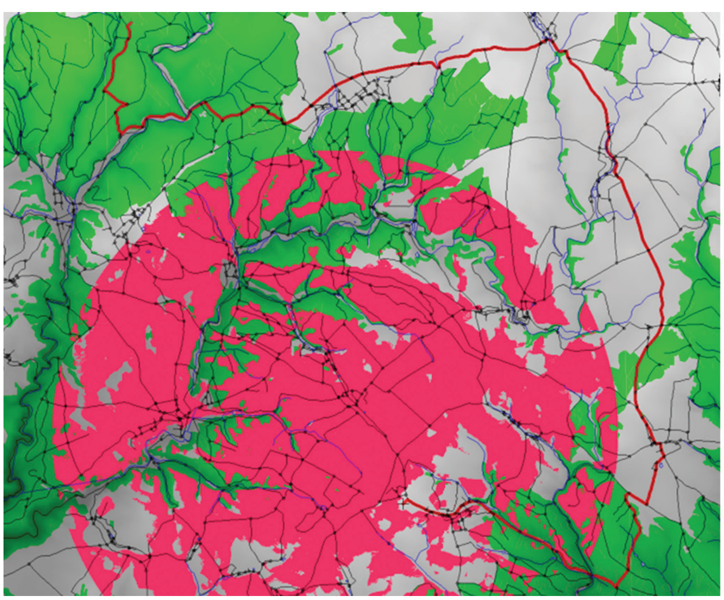
5. EXPERIMENTS

The proposed solution into the tactical decision support system (TDSS) was implemented. This system has been in development at University of Defence since 2006 to support commanders in their decision-making process[19]. At present, it incorporates several models of military tactics such as optimal maneuver, optimal supply chain, and optimal aerial reconnaissance.

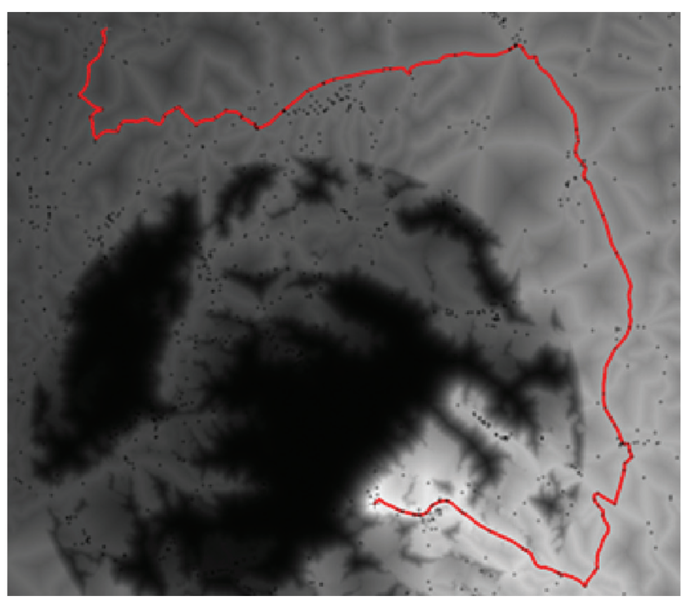
To verify our metaheuristic solution, we carried out a set of experiments. The experiments are as shown in Table 1 ordered according to their computational complexity. We set up the parameters of experiments as the problem of cooperative reconnaissance has been formulated for the first time, therefore no benchmark instances have been created yet.

Table 2 contains solutions to the proposed experiments. Several parameters are included: number of waypoints N, observed area A_O, coefficient α, total distance travelled by all elements, and total time necessary to conduct the task. As optimality criterion, we selected the last parameter, i.e. total time of the whole reconnaissance operation.

Table 3 presents the verification of solutions from Table 2. The two steps of the algorithm (finding positions of waypoints and computing the tour of elements between waypoints)



(a)



(b)

Figure 3. (a) Area of operations and (b) cost matrix.

were verified independently. At first, we verified the quality of waypoints deployed to observe the area of interest – see coefficient α in Table 3. In this case, for verification we used the brute force by evaluating all possible solutions in the state space. Due to this reason, we were able to conduct this verification only for experiments cr1-1, cr1-2 and cr1-3 with at most two waypoints (4 variables) – see coefficient α_{optim}. The brute force solution for more than two waypoints is not feasible. The error when comparing our solution with the optimal solution is below 0.3 per cent.

Next, we compared the tours of elements between waypoints found by our algorithm (see column d in Table 3) and the algorithm FIND (Fast improvement, intensification, and Diversification)[21] which is one of the best tabu search based algorithms for MDVPR problems (see column d_{FIND}). The solutions are the same in all cases. We can suppose that the solutions are optimal. There are 53 nodes in the most complex experiment (cr4-4); both algorithms are able to deal with problems with several hundred nodes[22].

The next part of this section shows experiment cr4-4 in detail. The area of interest is shown in Fig. 4(a) along with topographic objects of the area of operations. Figure 4(b) presents positions of 6 elements (marked by letters) and 53 waypoints (green circles); the number and positions of waypoints were found via the metaheuristic solution described above. In Fig. 4(c), the area observed from all waypoints is coloured in green. Finally, Fig. 4(d) presents the solution found by our algorithm, i.e. the order of waypoints to be visited by each element along with precise routes between successive waypoints (see the red lines).

The covered (observed) area of interest is 91.02 per cent. Total distance travelled by all elements is 22.62 km and the whole operation of reconnaissance can be done in 47.5 min (average speed of each elements was set to 5 km/h). Figure 5 shows experiment cr4-4 as it looks like in TDSS.

Table 1. Specification of experiments carried out to verify the proposed method.

Experiment	Area of interest A_I	Number of elements M	Coefficient a_{min}	Range of visibility v (m)
cr1-1	745,311 m^2	1	0.80	1,000
cr1-2		1	0.90	1,000
cr1-3		2	0.90	1,000
cr1-4		2	0.98	1,000
cr2-1	1,572,741 m^2	4	0.80	250
cr2-2		4	0.95	500
cr2-3		8	0.80	250
cr2-4		8	0.95	500
cr3-1	4,853,097 m^2	4	0.95	1,000
cr3-2		8	0.95	1,000
cr3-3		12	0.95	1,000
cr3-4		16	0.95	1,000
cr4-1	5,675,033 m^2	6	0.60	250
cr4-2		6	0.70	250
cr4-3		6	0.80	250
cr4-4		6	0.90	250

6. CONCLUSIONS

The article deals with the metaheuristic solution proposed by the authors to the optimal cooperative reconnaissance problem. This problem has been implemented as one of the models in the tactical decision support system being developed at University of Defence, Czech Republic. The TDDS is proposed to aid commanders with their decision-making process.

Using metaheuristic approach is a feasible way to obtain a high quality solution in the acceptable time. The

Table 2. Results of experiments achieved by the proposed method

Experiment	Number of waypoints N	Observed area A_O	Coefficient α	Total distance travelled (km)	Total time consumed (min)
cr1-1	1	620,546 m^2	0.8326	0.92	11.0
cr1-2	2	722,206 m^2	0.9690	1.18	14.2
cr1-3	2	722,206 m^2	0.9690	1.91	12.3
cr1-4	3	733,535 m^2	0.9842	1.65	13.3
cr2-1	10	1,296,725 m^2	0.8245	5.98	20.8
cr2-2	8	1,505,428 m^2	0.9572	5.91	23.1
cr2-3	10	1,296,725 m^2	0.8245	5.73	12.7
cr2-4	8	1,505,428 m^2	0.9572	3.90	11.4
cr3-1	16	4,639,561 m^2	0.9560	16.32	52.6
cr3-2	16	4,639,561 m^2	0.9560	14.35	31.1
cr3-3	16	4,639,561 m^2	0.9560	15.88	26.8
cr3-4	16	4,639,561 m^2	0.9560	9.55	17.7
cr4-1	22	3,442,475 m^2	0.6066	15.22	37.6
cr4-2	28	4,045,731 m^2	0.7129	17.90	38.2
cr4-3	38	4,630,827 m^2	0.8160	19.88	43.1
cr4-4	53	5,165,415 m^2	0.9102	22.62	47.5

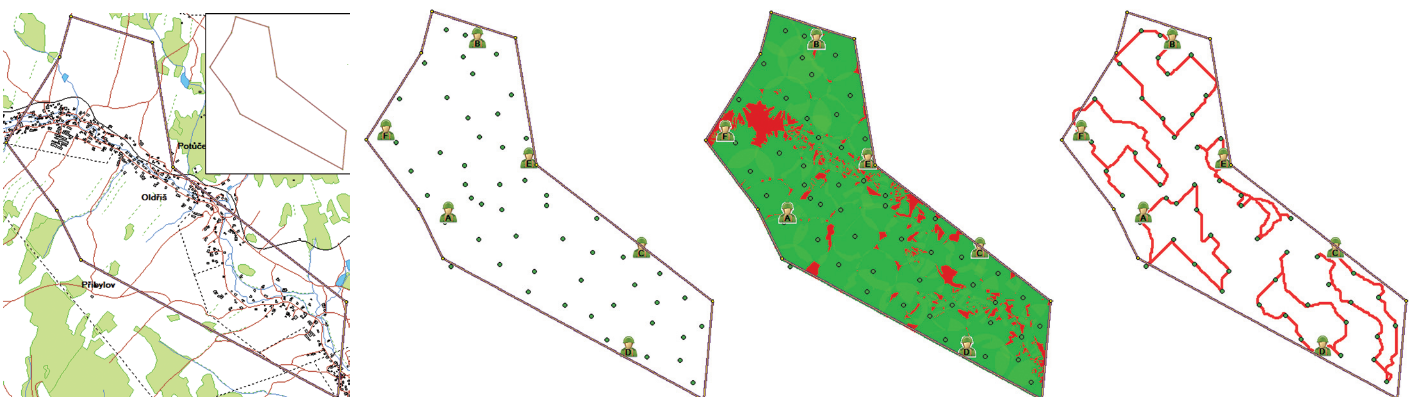
Table 3. Verification of the proposed method

Experiment	α	α_{optim}	Error	d (km)	d_{FIND} (km)	Error (%)
cr1-1	0.8326	0.8346	0.24%	0.92	0.92	0
cr1-2	0.9690	0.9715	0.26%	1.18	1.18	0
cr1-3	0.9690	0.9715	0.26%	1.91	1.91	0
cr1-4	0.9842	-	-	1.65	1.65	0
cr2-1	0.8245	-	-	5.98	5.98	0
cr2-2	0.9572	-	-	5.91	5.91	0
cr2-3	0.8245	-	-	5.73	5.73	0
cr2-4	0.9572	-	-	3.90	3.90	0
cr3-1	0.9560	-	-	16.32	16.32	0
cr3-2	0.9560	-	-	14.35	14.35	0
cr3-3	0.9560	-	-	15.88	15.88	0
cr3-4	0.9560	-	-	9.55	9.55	0
cr4-1	0.6066	-	-	15.22	15.22	0
cr4-2	0.7129	-	-	17.90	17.90	0
cr4-3	0.8160	-	-	19.88	19.88	0
cr4-4	0.9102	-	-	22.62	22.62	0

total time consumed by the optimisation process depends on the complexity of the experiment at hand. Simpler experiments were solved in several seconds, the most complex experiments in several minutes (under 5 minutes in all cases).

Our metaheuristic approach provides a solution to the problem which is close to optimal. We cannot, however, compare our results with results of other rival methods as we formulated the problem for the first time and there are no such methods yet. Nevertheless, we have compared the results of our ACO algorithm applied to Cordeaus's MDRVP benchmark instances. In all cases, the difference between our solutions and the best known solutions were under 3 per cent.

The reconnaissance in our model was proposed to be done only in waypoints positioned in the area of operations. Our future work will be to improve the model so that the reconnaissance will be done not only in waypoints but also en route between them. This new model will be called optimal vigilant cooperative reconnaissance.



(a) (b) (c) (d)

Figure 4. Experiment cr4-4: (a) Area of interest, (b) Elements and waypoints, (c) Observed area, and (d) Possible solution.

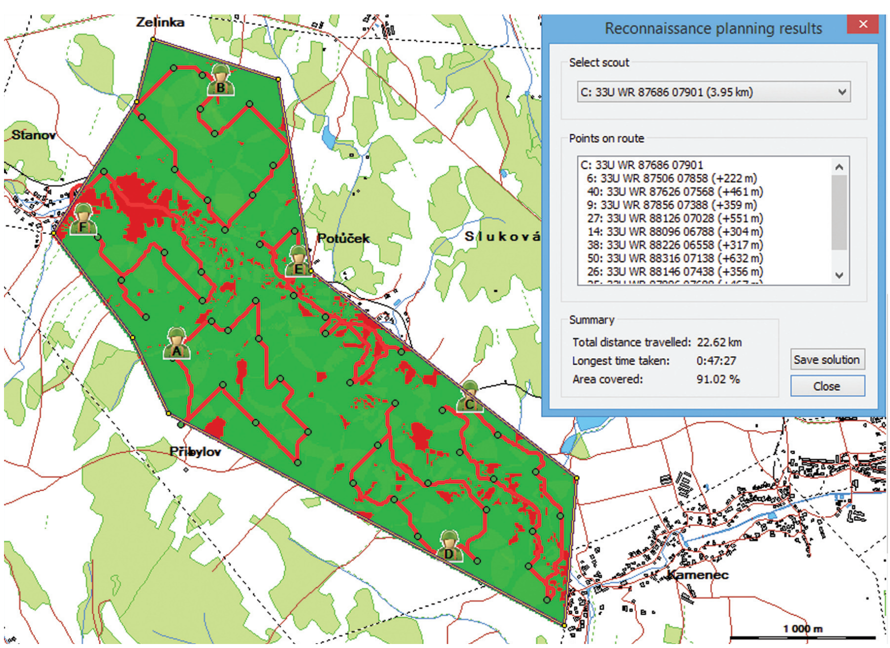


Figure 5. Experiment cr4-4 in TDSS.

REFERENCES

1. Finlay, P.N. Introducing decision support systems. Blackwell Pub, 1994.
2. Surdu, J.R. & Kittka, K. The deep green concept. *In* Proceedings of the 2008 Spring simulation multi-conference Spring Sim ’08. San Diego, 2008, pp. 623-631.
3. Rivas, R.A.; Clausen, J.; Hansen, K.S. & Jensen, L.E. Solving the turbine positioning problem for large off shore wind farms by simulated annealing. *Wind Engineering,* 2009,**33**(3), 287-298. doi:10.1260/0309-524X.33.3.287
4. Kolish, R. & Dahlmann, A. The dynamic replica placement problem with service levels in content delivery networks : A model and a simulated annealing heuristic. *OR Spectrum,* 2015, **37**(1), 217-242. doi:10.1007/s00291-013-0358-z
5. Tong, K. h.; Bakhary N.; Kueh, A.B.H. & Mohd Yassin, A.Y. Optimal sensor placement for mode shapes using improved simulated annealing. *Smart Struct. Sys.*, 2014, **13**(3), 389-406. doi:10.12989/sss.2014.13.3.389
6. Zih-Ping, Ho & Chien-Sheng, Wu. Application of simulated annealing algorithm to optimization deployment of mobile wireless base stations. *Adv. Comput. Sci. Info. Eng.,* 2012, **169**, 665-670. doi:10.1007/978-3-642-30223-7_105
7. Mann, M. & Sangwan, O.P. Generating and prioritizing optimal pathsusing ant colony optimization. *Comput. Ecology Soft.,* 2015, **5**(1), 1-15.
8. Sudholt, D. & Thyssen, Ch. Running time analysis of ant colony optimization for shortest path problems. *J. Discrete Algo.,* 2012, **10**, 165-180.
9. Mahia, M.; Baykanb, Ö.K. & Kodazb, H. A new hybrid method based on particle swarm optimization, ant colony optimization and 3-opt algorithms for traveling salesman problem. *Applied Soft Comput.,* 2015, **30**, 484-490. doi:10.1016/j.asoc.2015.01.068
10. Kan, Jun-Man & Zhang, Yi. Application of an improved ant colony optimization on generalized traveling salesman problem. *Energy Procedia,* 2012, **17**(A), 319-325. doi:10.1016/j.egypro.2012.02.101
11. Zaza, T. & Richards, A. Ant colony optimization for routing and tasking problems for teams of UAVs. *In* International Conference on Control. *Loughborough: IEEE,* 2014, pp. 652-655. doi:10.1109/CONTROL.2014.6915216
12. Reed, M.; Yiannakou, A. & Evering, R. An ant colony algorithm for the multi-compartment vehicle routing problem. *Applied Soft Comput.,* 2014, **15**, 169-176. doi:10.1016/j.asoc.2013.10.017
13. Stodola, P. Optimization methods for position of unattended ground monitoring systems. *In* Proceedings of International Conference Transport Means 2006. Kaunas, 2006, pp. 79-82.
14. Stodola, P. Position optimization of cooperative arrays of adaptive ground unattended sensors. *Adv. Military Technol.*, 2007, **2**(2), 13-32.
15. Stodola, P. & Mazal, J. Tactical and operational software library. *In* International Conference on Military Technologies 2015. Brno: University of Defence, 2015, pp. 347-350. doi:10.1109/MILTECHS.2015.7153645
16. Dantzig, G.B. & Ramser, J.H. The truck dispatching problem. *Management Science*, 1959, **6**(1), 80–91. doi:10.1287/mnsc.6.1.80
17. Bovet, D.P. & Crescenzi, P. Introduction to the theory of complexity. Prentice Hall, 1994.
18. Dorigo, M. Optimization, learning and natural algorithms, Politecnico di Milano, Italy, 1992. (Thesis)
19. Stodola, P. & Mazal, J. Tactical models based on a multi-depot vehicle routing problem using the ant colony optimization algorithm. *Int. J. Math. Models Methods Applied Sci.,* 2015, **9**, 330-337.
20. Stodola, P. & Mazal, J. Planning algorithm and its modifications for tactical decision support systems. *Int. J. Math. Comput. Simulation*, 2012, **6**(1), 99-106.
21. Renaud, J.; Laporte, G. & Boctor, F. F. A tabu search heuristic for the multi-depot vehicle routing problem. *Comput. Operations Res.*, 1996, **23**(3), 229-235. doi:10.1002/(SICI)1097-0037(199709)30:2<105::AID-NET5>3.0.CO;2-G
22. Stodola, P. & Mazal, J. Applying the ant colony optimisation algorithm tothe capacitated multi-depot vehicle routing problem. *Int. J. Bio-Inspired Comput.,* 2016, **8**(4), 228-233. doi:10.1504/IJBIC.2016.078639

CONTRIBUTORS

Dr Petr Stodola received his PhD in the field of military technology at University of Defence, in 2006. Currently working as an Associated Professor at University of Defence in Brno, Czech Republic. His research interests include : Optimisation, combat modelling and simulation, command and control decision support systems and C4ISR systems. He is the author or co-author of more than 100 scientific papers, and more than 10 functional samples and application software.

His contribution in the current study is the proposal of the metaheuristic solution, application of the simulated annealing and ant colony optimisation algorithm and verification of the proposed method.

Mr Jan Mazal is a graduate of the Faculty of Military Systems Management, Military College of Ground Forces, Vyskov, Czech Republic. Since 2013 he is an associated professor in the field of military management and C4ISR systems. Currently he works as Doctrine Education and Training Branch Chief at NATO Modelling and Simulation Centre of Excellence in Rome. He is focused on the issues of military intelligence and reconnaissance, C4ISR systems, and implementation of information and model support to the commander’s decision-making process at the tactical level.

His contribution in the current study is the proposal and implementation of the shortest path algorithm to find optimal paths between waypoints including the influence of the tactical situation on the movement of units.