Sniper Line-of Sight Calculations for Route Planning in Asymmetric Military Environments

Ove Kreison(✉) and Toomas Ruuben

Department of Radio and Communication Engineering,
Tallinn University of Technology, Tallinn, Estonia
ove.kreison@gmail.com, toomas.ruuben@ttu.ee

Abstract. Situation aware route planning plays a key role in modern urban warfare. While planning routes to military convoys decision support systems have to take into account multiple environmental conditions to find safe routes and minimize the risk of convoy being attacked during mission. Considering that nowadays battles are fought in asymmetric conditions where red forces almost always have an upper hand then all systems aiding soldiers must try to take into account as much of those conditions as possible. This paper proposes a way how snipers locations and their line-of-sight can be added to dynamic threat assessment which in turn is an input for route planning. Risk minimization is done by using well known A* route planning algorithm where threat is presented as one of graph edge parameters that is in added to other parameters describing the surrounding environment.

Keywords: Multi-objective optimization · Military route planning · Military environment risk assessment · Route planning · A* algorithm · Situation aware route planning · Sniper calculation · Haversine formula · Situational awareness · Environment orientation · 3D risk assesment

1 Introduction

During urban military operations military commanders and team leaders are faced with complicated and time critical decisions. For example, when a military convoy is travelling through a previously planned route it might be stopped by the occurrence of an asymmetric threat. Urban military environments are filled with different types of asymmetric threats (for example improvised explosive devices, snipers, military booby traps) and finding routes in a military environment in a way that would help to avoid previously mentioned threats is defined as Military Unit Path Finding Problem (MUPFP) [1]. Therefore all Decision Support Systems (DSS) that aim to help commanders and team leaders have to solve MUPFP. Route planning algorithms are usually studied from the viewpoint of algorithmic optimality and efficiency, but the success in solving a MUPFP also relies heavily on modelling the surrounding environment and its constraints accurately enough so that they describe the actual environment as closely as possible. In addition to describing the environment effectively enough one also has to consider military units own parameters (such as vehicles width

J. Hodicky (Ed.): MESAS 2016, LNCS 9991, pp. 359–370, 2016.
DOI: 10.1007/978-3-319-47605-6_29

and weight) because their interaction with environmental parameters defines MUPFP solution. For example if it is known that convoy contains a vehicle weighing 5000 kilograms then roads that do not support such weight should be eliminated from possible routes.

Describing the surrounding environment using a graph in a way that would be adequate enough for solving MUPFP is an extremely complicated task because one needs to transform a variety of information (such as units hostility and road type) into a numerical representation and combine all that information together in a meaningful way. In the context of route planning these bits of information are called parameters. As route planning algorithms expect that all graph edges are described numerically then all different parameters can be combined using weighted sums, but as their magnitudes usually differ then firstly they should be normalized to a range [0, 1] and then their importance can be adjusted using weights. In [2] we found that for solving MUPFP it is possible to describe every graph edge and therefore model the surrounding environment using the following ten parameters: length, width, ground type, environment, road type, road infrastructure, maximum speed, maximum bearing capacity, units hostility and threat for describing graph edges representing the digital terrain. By combining these parameters under three properties categories: shortness, fastness and safety one is able to use well known A* search algorithm with Manhattan distance as a heuristic function or genetic algorithm for finding routes in a graph [2]. Test results presented in [2] showed that our model of the environment using already mentioned ten parameters was able to model the surrounding environment with sufficient precision for finding routes in a graph.

Most parameters described in [2] are static and therefore do not change that often. Only units hostility and threat values depend on whether there are any blue or red units located nearby. This in turn means that previously mentioned values can change in minutes and therefore DSS has to update its underlying parameters and aggregations describing the environment accordingly. For example, the threat value takes into account gatherings of crowds, IED's current and historical locations, sniper locations, locations of schools/hospitals/police stations/gas stations etc. Evaluating the risk to soldiers emanating from different threats is filled with uncertainties because all of its components could change in time and these changes could be unpredictable. For example a sniper on the rooftop may start to look on the other direction and therefore its line-of-sight will change and also the threat emanating from it will change and therefore the underlying graph parameters will change. But sniper changing its position is one thing, its surroundings can also change and they also in turn affect line-of-sight. As an example a large lorry might block its view. This means that in addition to taking into account objects the current state of an object, one also has to simulate interactions of a collection of objects to create actual situation awareness. On the other hand snipers line-if-sight depends on the altitude where sniper is located and the height of buildings surrounding it. Sniper on the top floor of a high building can in this context do much more damage than a sniper on the middle floor of medium sized building surrounded with high buildings. The example demonstrates effectively that in order to understand and to evaluate situations numerically we have to solve MUPFP taking into account spatiotemporal properties and their changes in real-time.

This paper proposes an approach how threats emanating from snipers in asymmetric warfare situations can be taken into account in a way that would allow to embed dynamic risk assessment into MUPFP. Firstly an overview of different parameters that are used for modelling the environment using a graph is given based on the results of [2]. Then a theoretical solution for assessing the threat from snipers and its coverage is given which is succeeded by the testing results. The result is achieved by using a separate process which constantly polls data about sniper locations, calculates the area size affected by them and modifies the road graph values so that route planning algorithm could find routes to avoid known sniper locations.

2 Routing Graph

As part of European Defence Agencies project CARDINAL (CApability study to investigate the essential man-machine Relationship for improved Decision making IN urbAn miLitary operations) we developed an approach which uses ten parameters to describe one graph edge. Since some of those parameters can vary in a large range which in turn affects the resulting route, the graph has to be constructed in a way, where most of these parameters are homogenous for the entire length of the edge.

Parameters describing one graph edge are following:

1. Length – Length describes the physical distance between two graph vertices and is presented in meters. Usually when graphs of urban networks are constructed, vertices are but on intersections and therefore length describes the distance between two intersections. Length is the only value that doesn’t have to be homogenous and consequently its values can range from a few dozen meters to thousands of meters. The latter case usually occurs outside urban areas, because other parameters don’t change so often out there.
2. Width – Width describes the physical width of the road segment that is represented as a graph edge. It is a very dispersive parameter and can range from less than a meter to a few dozen meters. Width should be kept homogenous for the entire length of the edge, but if for some reason that should prove to be impossible then the smallest value should be chosen to represent width because in the context of military route planning wider roads are preferred. For one reason, choosing the smaller width helps to eliminate unsuitable graph edges before they are even given to the route planning algorithm. For instance if the widest vehicle in the convoy is 2.5 m wide then it would be reasonable to eliminate edges whose width is smaller than that to speed up route planning algorithms work. Narrow roads can also be very dangerous because if an improvised explosive device (IED) is planted beside the road then the convoy would be much closer to the epicenter of explosion than in the case of a wider road.
3. Ground type – Ground type describes the ground on which the convoy is driving on. This parameter describes if the road segment is even passable and if it is, then how fast the convoy can drive on it. In this study seven values are used to describe ground type: unknown value, tarmac, gravel, soil, water, swamp and impassable. If road segments ground type is set equal to swamp, water or impassable then the route planning algorithm will know that this segment is very hardly passably, in the

case of swamp, or totally impassable, in the case of water and the impassable value. The reason, why ground type influences convoys speed is due to the fact that different ground types have different friction coefficients. Because of this, machines can drive faster on tarmac than on gravel or soil and therefore choosing slower road segments into the route will affect total travel time.

4. Environment – Term is used to describe the environment surrounding road segments. This study uses four different values for describing it: unknown value, urban, flat and forest. Urban is used to describe an area with higher building and population density. Flat on the other hand describes an area with low or non-existent building and population density (for instance a desert or a steppe). Finally forest is used to describe an area with plenty of trees, but low or non-existent human density. In general, the probability of being attacked in an urban area or forest is much larger than the probability of being attacked in a flat land because the first two offer more opportunities for an unexpected attack and an easy getaway. Since this study is focused on improving decision making in urban military operations, it is impossible to avoid urban areas or forests, if they are surrounding the one road between two cities, and therefore to ensure convoys safety, other parameters have to be considered in addition to environment. More thorough definition of environment types and on topic of how to measure them can be found in [10].
5. Road type – Road type describes the road based on its type. Seven different parameters are used to describe road type: unknown value, highway, road, lane, track, drive and causeway. Different types are mainly distinguished by their travelling speed, but choosing the right road type can also have an effect on routes safety. On the one hand using the highway or road can be safer than crossing a causeway because the latter could be rigged with explosives, but on the other hand, one can join or exit the highway only at certain points. Furthermore, highway lanes are usually separated from each other, which in turn means that if a convoy is attacked on a highway, its chances of getting to an alternative route or retreating are very limited. To prevent previously mentioned incidents, other parameters have to be considered together with road type.
6. Road infrastructure – Road infrastructure is used to describe the road segments based on their structure. Four values are used to describe it: unknown value, open structure, bridge and tunnel. Open structure is used to describe all sorts of different road types that are already mentioned in the previous point. Although the word open might suggest that besides the structure, the convoy would be open to different sorts of threats that suggestion is not completely true. Using a bridge or a tunnel is far more dangerous because they both have only one entrance and one exit and if both of them are blocked then the convoy is trapped. In addition, by blowing up the bridge or a tunnel it is possible to destroy the entire convoy at once. Due to historical reasons a lot of cities are located at riverbanks or in mountainous areas and therefore it might be impossible to avoid using bridges or tunnels because the only way in or out of an urban area can be via bridge or tunnel.
7. Maximum speed – Maximum speed describes how fast a convoy can travel on specific road segments and is measured in kilometers per hour or miles per hour. This value can be the legal speed limit or a value which is specifically calculated by military personnel.

8. Maximum bearing capacity – Maximum bearing capacity describes the maximum weight that the road segment supports and is usually presented in tons. Considering maximum bearing capacity in route planning comes even more important when there is at least one heavy machine in the convoy because if this machine collapses through the road, a roadblock problem is automatically created. This is an extremely difficult situation which could break the machine and make the entire convoy an easily attackable standing target. It follows that considering maximum bearing capacity is of couse more important for heavier machines than for jeeps but it has to be considered in the case of them as well because with equipment those jeeps can weigh more than 3 tons.
9. Units hostility – Units in military operations are usually divided into two categories which are denoted by different colors: red denotes enemy units and blue allies and own units. For route planning it is wise to use more than two categories. Authors found that for better performance it is suitable to use five values: unknown, no units, uncertain (used to mark civilians), blue and red. It is important to force the route planning algorithms to find routes that pass through areas filled with blue units or with no units at all. The probability of an armed conflict becomes extremely large when the convoy is directed close to red units and therefore these situations have to be avoided. Unfortunately it is impossible to plan routes in a city without passing by civilians. The reason why this is unfortunate is that it is practically impossible to assess the hostility of a single person. However to some extent it is possible to probabilistically evaluate the threat emanating from civilians by assessing the number of attacks made against soldiers in a specific region during a certain time interval. This means that units hostility, when not dealing with blue or red units, should always assessed together with the tenth parameter.
10. Threat – Threat is a numerical value, which is based on military intelligence and relevant information. Information for evaluating the threat of an area could be collected using unmanned aerial vehicles (UAV) and unmanned ground vehicles (UGV), but one should also consider information coming from blue units based on area of interest, previously known information about red units (for example coordinates, direction and speed of movement), historical data about attacks against soldiers in that region, the number of IEDs found in that area and other information that could be used to form an objective evaluation of threat in a specific area. Thales Groups MYRIAD system [7] was used to to evaluate the threat level in range from 0 to 10, where 10 stands for highest threat possible and 0 on the other hand denotes that the area is safe. Sniper calculations proposed in this article have a direct effect on the threat parameter value which is set to 10 if a graph edge or vertex is in direct line-of-sight from the sniper.
11. Altitude – Altitude describes graph edges, vertices or objects height from sea level in meters and is used to calculate line of site between different objects and graph vertices and nodes.

As some of the parameters are not numerical then they have to be factorized and normalized before they can be added together. The complete methodology can be found in [2], but for clarity the numerical representation of previous parameters is given in Table 1.

Table 1. Numerical representation of parameters

Parameter	Numerical representation
Length	Value in meters
Width	Value in meters
Ground type	0 – unknown value 1 – tarmac 2 – gravel 3 – soil 4 – swamp 5 – water 6 – impassable
Environment	0 – unknown value 1 – flat 2 – forest 3 – urban
Road type	0 – unknown value 1 – highway 2 – road 3 – lane 4 – track 5 – drive 6 – causeway
Road infrastructure	0 – unknown value 1 – open structure 2 – bridge 3 – tunnel
Maximum speed	Value in kilometers per hour
Maximum bearing capacity	Value in tons
Units hostility	0 – unknown value 1 – no units 2 – uncertain (civilians) 3 – blue 4 – red
Threat	Value in range from 0 to 10
Altitude	Value in meters

As one can see from Table 1 then when the parameters are added together, some of them will dominate in the sum because their values are much larger (for example length, width, threat, altitude). In order to add the parameters together using a weighted sum they have to be normalized to a range [0, 1] and then added together using weights to spotlight the parameters that affect certain properties. Only parameters that is not normalized is altitude which is used to calculate threat parameter values. The process of snipers threat calculations is given in the succeeding paragraphs. The properties that are used for finding routes can be found in Table 2.

Table 2. Parameters relationship to route properties

Properties	Parameters
Shortness	• Length
Fastnss	• Length • Ground type • Road type • Maximum speed
Safety	• Width • Environment • Road infrastructure • Maximum bearing capacity • Units hostility • Threat

As every graph edge can be described by only one numerical values then all ten values are added together using a weighted sum: calculated using methodology presented in [4]:

$$a_{i,j} = \sum\nolimits_{n=1}^{10} w_n x_n, \tag{1}$$

where $a_{i,j}$ denotes graph edge value between vertices i and j, n denotes the specific parameter, w_n is the weight given to the parameter and x_n is the parameter value. Using (1) a N x N weight matrix is constructed for the entire domain that is in turn used by the A* algorithm for finding optimal routes. As threat and units hostility values are dynamic then the matrix itself has to be cached so that it could be updated in real-time. As current papers aim is to demonstrate how snipers locations affect route planning results then we will always be finding the safest route.

Current paper uses A* algorithm for finding routes because of its performance over other greedy best-search first algorithms and the fact that it can find the optimal path if an admissible heuristic is used [8]. A* is a best-first search algorithm in which evaluation function is of the form:

$$\mathrm{f(n) = g(n) + h(n)}, \tag{2}$$

where f(n) is distance-plus-cost heuristic function, which determines path cost through vertex n, g(n) is the cost from start vertex to vertex n and h(n) is the estimated minimum cost from vertex n to goal vertex [3]. Manhattan distance between geographical coordinates is used as a heuristic functions and it can be presented in the form:

$$\mathrm{h(n) = |n.lat - goal.lat| + |n.lon - goal.lon|}, \tag{3}$$

where n.lat and goal.lat are the latitude values of vertex n and goal vertex and n.lon and goal.lon are the longitude values of vertex n and goal vertex. Manhattan distance was chosen as a heuristic function because it is admissible and therefore it never overestimates the actual cost of the route, i.e. the cost it estimates to reach the goal is not higher than the lowest possible cost from the current vertex in path [9]. During our tests in [2], A* algorithm showed excellent performance and accuracy; however one major drawback of this algorithm is its memory requirement. Since two lists of graph vertices are used during the calculation: open list and closed list, the amount of memory required to store open lists (at initialization all graph vertices are stored at open list) for large areas, containing multiple urban areas, can be extremely large. Because the purpose of our application is to help soldiers in urban military situations, we only need to store a graph describing one city and its surrounding areas into the open list, which in turn means that memory requirements can be satisfied by most of the computers on the market today.

3 Sniper Calculation

Taking into account sniper line of sight in a battlefield is an extremely complicated task and can be accomplished using different methods. One can either calculate snipers actual line-of-sight very precisely by for example taking into account every small obstacle and every building feature or it can be done using more robust methods. This paper use a robust approach because predicting snipers movement on their locations is filled with too much uncertainty and therefore is not practical in real life situations. Therefore we choose a way where snipers line-of-sight is modelled as a circle and all graph edges and vertices that fall into these circle will have a maximum threat value [2]. Previously mentioned approach has been used and has also been proven to be useful in military practice. This is due to the fact that when dealing with soldiers lives in warfare situations then it is safer to avoid sniper shooting zones entirely rather than trying to cut corners and put the entire convoy at risk.

As our main aim is to calculate new threat values for graph elements using sniper locations then all elements must have longitude and latitude coordinates. This is true for graph vertices and segments that are used for route planning as well as for sniper locations. The approach is useful because it is universal and is not dependent on the underlying map. We also know that average sniper shooting range is usually between 600–1000 [m] although there have been occasions when shots have been made from as far as 2 [km]. The actual shooting range depends on the rifle used and weather conditions. If the sniper rifle type used is known then its shooting range can be loaded to a database and loaded from there. As the coordinates of snipers location and its average shooting range are known and the coordinates of graph vertexes are known then we can use Haversine formula to calculate great circle distances between two points.

We previously stated that every object has geographical and coordinates and therefore distances between them can be calculated using methodology presented in [4]:

$$\mathrm{dLon} = \mathrm{lon2} - \mathrm{lon1}, \tag{4}$$

$$\mathrm{dLat} = \mathrm{lat2} - \mathrm{lat1}, \tag{5}$$

$$\mathrm{a} = (\sin\left(\frac{\mathrm{dLat}}{2}\right))^2 + \cos(\mathrm{lat1}) * \cos(\mathrm{lat2}) * \left(\sin\left(\frac{\mathrm{dLon}}{2}\right)\right)^2, \tag{6}$$

$$c = 2 * \text{atan2}\left(\sqrt{a}, \sqrt{1-a}\right), \tag{7}$$

$$d = R^{*}c. \tag{8}$$

where dLon and dLat are longitude and latitude differences, c is earth radius 6367 km and atan2 is an arctangent function with two arguments [4]. To model the environment more accurately then we also added an altitude parameter to graph edges described in [2]. This way it is possible to extend the model to take into account higher objects and their ability to reduce snipers line-of-sight. Currently the algorithm checks if there are higher objects in direct sight from snipers location to graph edge. Graphs describing entire cities and countries can have millions of vertices and segments and if we want to model the surrounding environment as accurately as we can then our models can turn very complex which sets high requirements to computation and storage. As we are dealing with urban military situations then we have to find routes in real-time or near real-time conditions to have an effect on situational awareness. This in turn means that the solution to the problem lies in finding a suitable practical model that allows to model the surrounding environment with a suitable accuracy while following real-time constraints. Together with results presented in [2] we are moving in direction of finding the maximum practical model still allowing real-time constraints.

Our solution, which currently is only for demonstration purposes, follows classical service oriented architecture and consists of a user interface that uses Google Maps for map engine, an agent who communicates with user interface and different services. This services include route planning application introduced in [2] which uses A* algorithm and handles combining the ten parameters together and sniper calculation service introduced in this paper. Currently as the underlying map is not very large and does not contain that many objects then the steps of the algorithm can be described as following:

1. Find all graph vertices in range from sniper location using previously described Haversine formula where lon1 and lat1 describe snipers location and lon2 and lat2 describe graph vertices coordinates.
2. If graph vertex is in snipers shooting range then its altitude is compared to snipers altitude and road infrastructure parameter is checked to determine if it is not a bridge or tunnel [2].
3. Find all graph segments that are connected to the vertex and change their threat value to maximum which is 10 [2].
4. Save the results and send them back to broker agent.

In the future all the calculations should be done in parallel to reduce time spent on calculations. One way to achieve that is to use cluster computing frameworks that are based on MapReduce algorithms [5]. For example we could use Apache Spark framework to distribute the map graph as resilient distributed datasets over a cluster and find vertices affected by snipers. To speed the process even more up then vertices can be sent to driving node which can forward the results to other worker nodes that can be used in parallel to find segments connected to vertices [6].

4 Testing

For testing purposes the sniper range is read from a properties file, but in productions environments the values are stored in database for different sorts of weapons. According to [2] the aim is always to find the safest route meaning that A* tries to find a route by minimizing threat value. As mentioned previously then all graph vertices that are in direct sight from snipers locations have a threat value 10 and also all segments that are connected to these vertices have a maximum threat value.

All tests are done in the area surrounding Tallinn University of Technology campus. Scenario is that a safe route is planned for a military convoy, but after gathering some intelligence information two snipers are sighted at region. Snipers line-of-sight is calculated and a new route is found for the convoy. As previously stated then snipers range is actually much longer, but for testing purposes it was set to 250 m. Figure 1. illustrates the route found when there is only one blue force object and no red forces objects.

Figures 1 and 2 demonstrate that sniper ranges are effectively taken into account. The route found on Fig. 1 is now in snipers line-of-sight and is therefore too dangerous. From Fig. 2 we can see that A* managed to find a route that avoids both snipers. As we are dealing with a greedy algorithm then it tries to minimize all the route values which in case of Fig. 2 leads to a situation where convoy can take an unnecessary risk by driving too close to the danger zone. This can be avoided by adding an extra buffer zone to the calculation. For example one could make the sniper danger zone wider than it actually is. Another interesting research opportunity comes from the situation described in Fig. 2 where there is an overlapping area between two snipers. For example if there is a situation where snipers cannot be avoided then algorithms should try to find a route that passes only the line-of-sight of one sniper rather than two. This means that data model has to be changed by adding a new value which describes how many threats and which affect certain vertices and segments. This approach also helps to improve risk analysis because then you can for example score different threat sources and make route planning more situation aware [7].

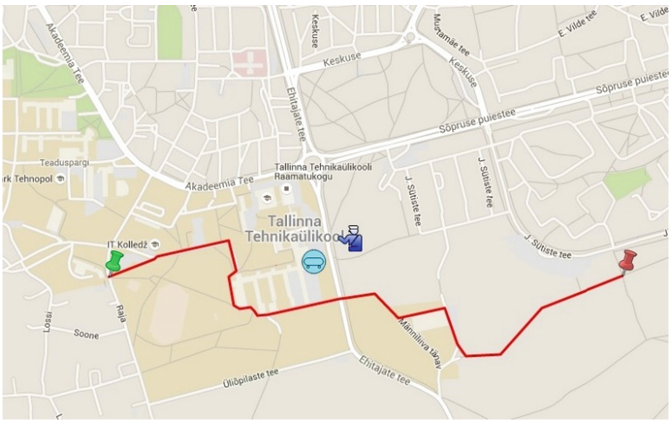

Fig. 1. Route without red forces objects (Color figure online)

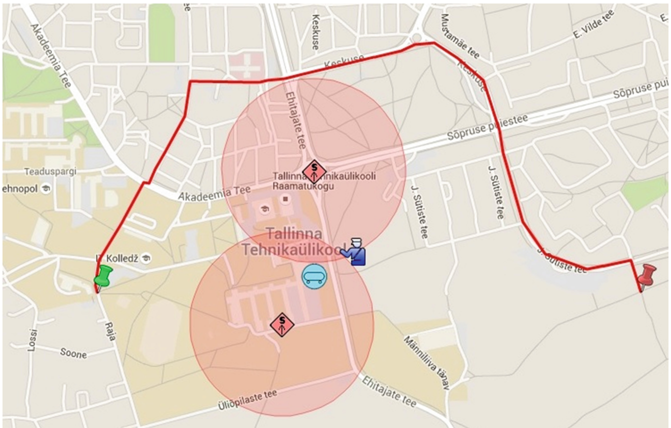

Fig. 2. Route with two red forces snipers (Color figure online)

Next research topics for us will be how radio communication parameters can be taken into account in route planning. For example modelling radio spectrum usage in real-time might give an indication whether snipers might have gotten a hint about convoy movement and might try to change their location. Also radio propagation modelling can be used in predictive analysis to avoid areas where convoy might not get any new information from decision support systems.

5 Conclusion

This paper proposes a new approach for including snipers line-of-sight modelling into route planning tasks in in asymmetric urban warfare situations. A Java application was developed which can be interfaced to already existing Battlefield Management Systems (BMS) for modelling and evaluating snipers threat. Developed solution was integrated to the agent oriented system built in [2] and route planning tests were done using popular A* algorithm. In the near future we will move our demonstrating platform to an actual BMS engine. As sniper calculation is done in a separate daemon process then all calculations can be done in real-time. Tests show that snipers line-of-sight is taken into account correctly but as the algorithm itself is greedy then it tries to minimize the route which in turn can lead to a situation where military convoys can pass the snipers too closely. To avoid that a special buffer zone can be simply added to the algorithm.

The DSS-s described before are heavily dependent on radio communication and therefore the next parameters that we will add to our model will describe radio wave propagation such as signal strength, interference etc. This will also allow to solve jammer avoidance and radio path problems. In the case of snipers we choose a robust approach, but in the case of radio wave propagation it cannot be done because for example vehicles in the convoy itself can block radio signals to other vehicles. Our current research is mainly focused on describing the surrounding environment with a data model that is practical enough for asymmetric warfare situations in a way that would allow all optimization to be done by route planning algorithms. Future research will also be more focused on predictive analysis so that possible emerging risks can already be included into route planning.

References

1. Mora, A.M., Merello, J.J., Millan, C., Torrecillas, J., Laredo, J.L.T.: CHAC, a MOACO algorithm for computation of bi-criteria military unit path in the battlefield. In: Proceedings of the Workshop on Nature Inspired Cooperative Strategies for Optimization (2006)
2. Ruuben, T., Kreison, O.: Route planning in asymmetric military environments. In: Second International Conference on Future Generation Communication Technology (FGCT) (2013)
3. Hoverd, T., Stepney, S.: Environment orientation: a structured simulation approach for agent-based complex systems. Natural Comput. **14**(1), 83–97 (2014)
4. Sinnott, R.W.: Virtues of the Haversine. Sky Telescope **68**(2), 158–159 (1984)
5. Lämmel, R.: Google's MapReduce programming model—Revisited. Sci. Comput. Program. **70**(1), 1–30 (2008)
6. Zaharia, M., Chowdhury, M., Franklin, M.J., Shenker, S., Stoica, I.: Spark: cluster computing with working sets. In: Proceedings of the 2nd USENIX Conference on Hot Topics in Cloud Computing, vol. 10, p. 10 (2010)
7. Labreuche, C., Le Huédé, F.: MYRIAD: a tool suite for MCDA. In: EUSFLAT, vol. 5, pp. 204–209 (2005
8. Nilsson, N.: Principles of Artificial Intelligence. Morgan Kaufmann Publishers, Burlington (1982). 476 p.
9. Russel, S.J., Norwig, P.: Artificial Intelligence: A Modern Approach, 3rd edn. Prentice Hall, New Jersey (2009). 1152 p.
10. Hodicky, J., Frantis, P., Litvaj, O.: Validation of simulator supporting movement of small group or individuals in different terrains. In: 2012 15th International Symposium, MECHATRONIKA, Prague, pp. 1–3 (2012)