Ground visibility analyses, algorithms and performance

Dana Kristalova[1], Vlastimil Neumann[2], PavlaZakova[3], Stepan Konecny[4], Jiri Kralicek[5], Pavel Foltin[6], Libor Kutej[7], Jan Zezula[8], Radomir Scurek[9]

[1] University of Defence, Brno, Czech Republic, email: dana.kristalova@unob.cz
[2] University of Defence, Brno, Czech Republic, email: vlastimil.neumann@unob.cz
[3] VŠB, Ostrava, Czech Republic, email: pavla.zaky@seznam.cz
[4] University of Defence, Brno, Czech Republic, email: stepan.konecky@unob.cz
[5] University of Defence, Brno, Czech Republic email: kralicek.jirka@gmail.com
[6] University of Defence, Brno, Czech Republic, email: pavel.foltin@unob.cz
[7] University of Defence, Brno, Czech Republic, email: libor.kutej@unob.cz
[8] University of Defence, Brno, Czech Republic, email: jan.zezula@unob.cz
[9] VŠB, Ostrava, Czech Republic, email: radomir.scurek@gmail.com

Abstract:

The paper deals with the topic of efficient visibility analyses, its algorithms and performance in context of area shape and its distance. Visibility analyses are fundamental component of almost any tactical estimations and has to be executed in very large numbers, usually in "point to area" version. Thus its effective execution and convenient algorithm selection is important and has to be solved in wider context and relations.

Keywords: *Tactical Path Optimization, Decision support, Traffic optimization, Visibility Analyses, Surface Analyses, Visibility Algorithm*

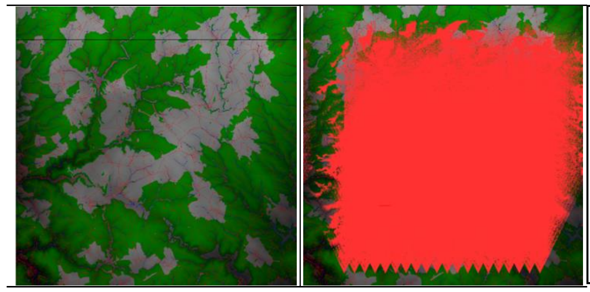
1. Problem statement:

- The problem is defined as a search for the decision to distinguish which algorithm should be used for particular area visibility calculation (analyses). Based on the previous practice, the Ray-Cast and Floating Horizon algorithms seems to be the most effective algorithms for altitude matrix defined areas.
- For operational calculations, the terrain square resolution usually varies between 5-10 meters, as a good compromise between data precision, computation time and final operational benefit of particular analyses (in our case we use a 5 m resolution). Each of mentioned algorithms has some performance advantage in specific area shape processing and the question is, if there exists a generic math model or algorithm, which will point to the concrete solution (Ray-Cast vs Floating Horizon) with the best calculation performance (time).
- When we look on both algorithms, there are some significant features that could intuitively lead to a performance dilemma, but without proper statistical evaluation, it is just an empirical estimation. Because the search for the existing appropriate C/C++ algorithmic solution was not successful (some algorithms were found, most of them were performance ineffective, using goniometric functions, for instance those from NVIDIA CUDA SDK, some of them were generally described but no-code available), we develop two algorithms fulfilling the initial conditions. These algorithms are part of broader framework, ensuring its functionality, what is not included, like variable initialization and final results storage and further operations.

2. Approach to the solution:

Which algorithm should be used for particular area visibility, is identified on the grounds of the following steps:

1. Development and implementation of two competing terrain visibility algorithms within a real terrain model. (See picture bellow: Terrain model 20x20 km used for the evaluation, resolution 5x5m.)
2. Measuring the performance of both algorithms on a variety of areas
3. Performance analyses and comparison in context of the area shape and distance.

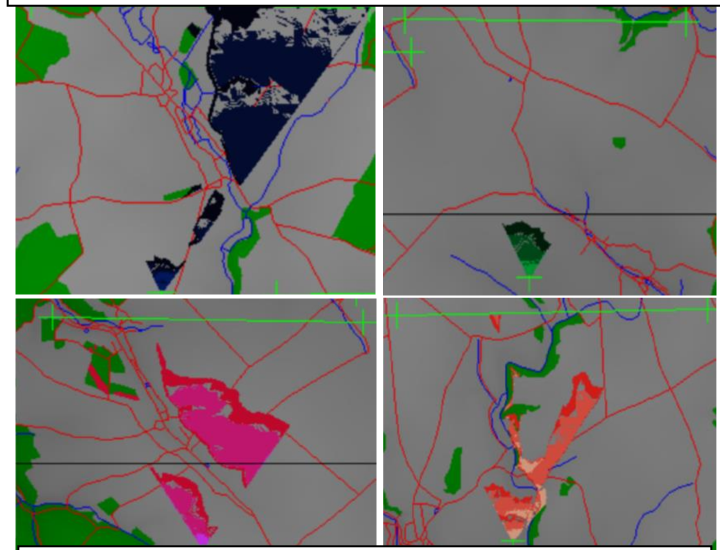


Terrain model 20x20 km used for the evaluation, resolution 5x5m (on the left). On the right there is a picture representing the ground observation situation. There are 400 (20x20) observation posts (starting with red peaks at the bottom). From each position, all particular visibility calculation was carried out, it ensures comprehensive coverage of the test area and sufficient algorithm

Some of key characteristics of used algorithms was already mentioned, both has a different approach and differs in the potential precision. Both methods are point to area (multipoint) processes and follow the search from front to back (from observer to far horizon) to the floating-point version creates particular horizons and which has an important impact on a solution performance and precision in various cases. In any case, for operational purposes, both of them are acceptable, because, lot of analyses takes a statistical format and usually large terrain models are about several months old, thus small precision handicap is still acceptable. Both algorithms could also count with potential visibility or hidden areas, which could be or could not be visible in a real situation, because these areas are close to the theoretical edge of visibility, which distinguish between visible or not.

Both algorithms are convenient for the fast terrain visibility processing. Due to the slightly different approach, differences in processing performance/efficiency can be identified. Both algorithms were evaluated for the visibility performance of the real territory in two steps. The first step is the angular calculation range, where each step was about 1° incremented (Graph - Angle Range Axis). The second step just repeat the first one with incrementing the distance (Graph - Visibility Distance Axis). The interval of this step was set to 100 m.

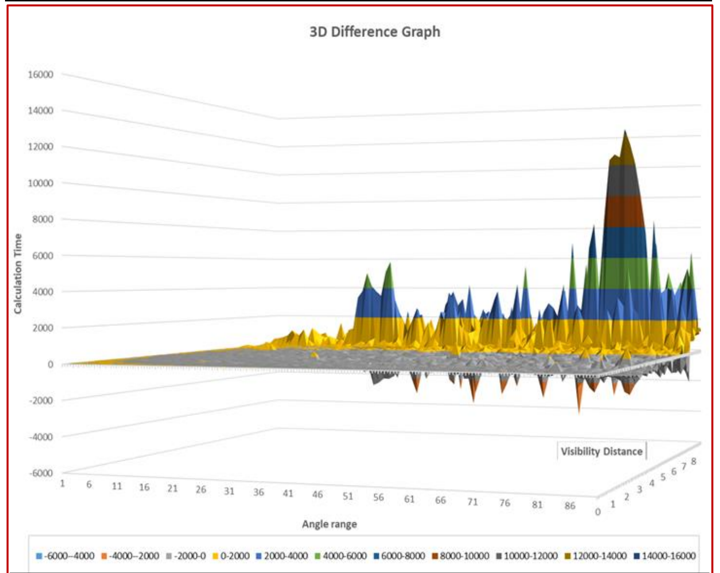
3. Algorithms and their Solution of Visualization:



Example of graphical visualization of potentially hidden areas. The difference of the particular horizon height and height of the terrain point, determines whether the area is "potentially" visible or not. Absolute value of the small difference (this threshold, could be derived from the resolution, precision and other attributes) indicates an uncertain situation that cannot be precisely decided, whether or not is the assessed area visible. The different colour shows the area where the visibility is "tricky", so it implies an unclear solution in practice. The dark (under the horizon) and light (above the horizon) colour indicates the areas close to particular horizon edge and this space

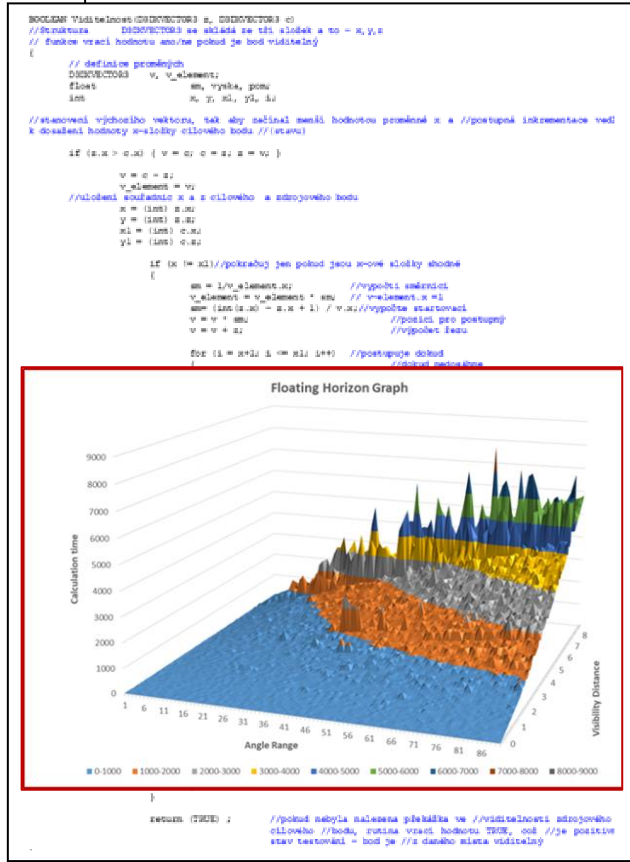
4. Comparative Analysis:

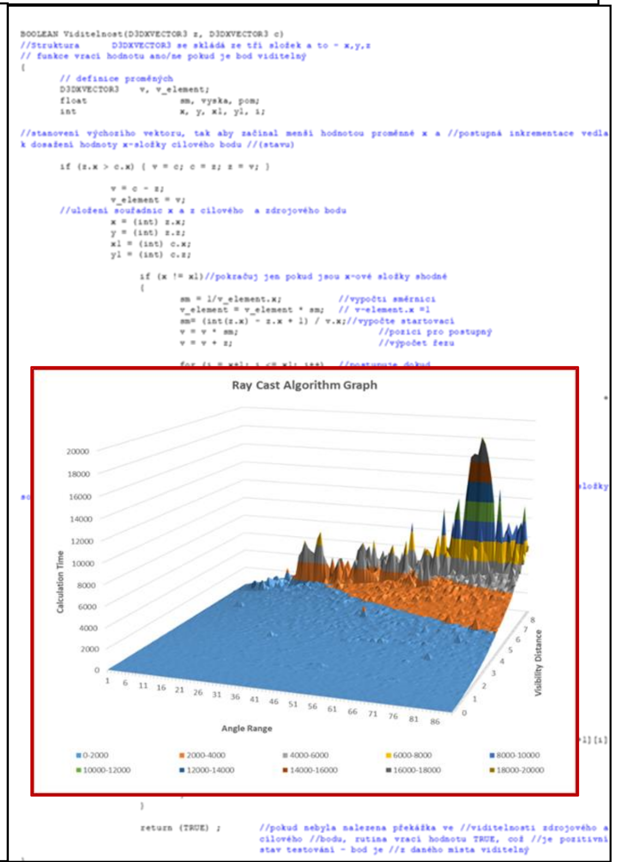
The third step of ground visibility analysis is a comparison of a performance and an effectivity of both algorithms and the comparative analysis in context of the area shape and distance give a solution which algorithm is it convenient to use.



Graphical expression of comparative analysis. The comparative analysis clearly shows an increase in time calculation values with the increasing angular range of the evaluated area as well as the length of the intentional beam. As mentioned in the article, the decisive limit distance is about 6 km, for a closer area it is advisable to use Ray-Cast algorithm, for more distance and large areas the Floating Horizon Algorithm is significantly faster.

Finally, the computation time of each iteration was collected and 3D graphs was generated, as presented on following figures. From graphs, it is possible to come to a following statement: Ray-Cast is faster for a closer area - approximately up to 6 km, but from this distance its performance decreases rapidly, and the time required for calculation increases significantly. From this distance, it would be more efficient to use the Floating Horizon Algorithm. More detailed investigation and math surface regression analyses will be a subject of further research.





3D graphical evaluation performance of the Floating Horizon Algorithm (above) and the Ray Cast Algorithm (bellow). The space is gradually evaluated in terms of visibility. From the observation point, the angular range first increases, then the distance (indicated in km). The calculation time is given in milliseconds. Computational algorithms are listed in the background of the graphical evaluation.

5. Discussion of other alternative approaches:

There exist other alternative approaches improving the speed or other factors of solution. One of the terrain visibility calculation approach is transformation of the depth map from the 3D rendering process, which could successfully utilize well established and highly parallel rendering pipeline of common graphic cards. For testing purposes, the sample application was developed, but performance do not reach the previous solution and further optimization and improvement could be a subject of further research. Performance of this approach is more than 100 times slower, comparing the mentioned algorithms, but visibility results are much more detailed and convenient for specific

A depth map is obtained by reverse transformation, where the visual rendering transforms each individual map pixel, even with objects. This approach brings superior results, millions of pixels are calculated. (on the left) Parallel execution of Floating Horizon Algorithm on CUDA platform, solution performance is about 150 times faster, comparing to the modern CPU processor (average time 2832ms, observer source area 1x2Km, visibility destination area 1.4x 2.5Km, and terrain square resolution 5m.) (on the right)