

## General Purpose Computation on a GPU: A Case Study of the Czech Helicopter Emergency Medical Service

Bruno Jezek, Jan Vanek and Andrea Vokalova  
Faculty of Informatics and Management, University of Hradec Kralove,  
Hradec Kralove, Czech Republic

---

**Abstract:** Graphics Processing Units (GPUs) are primarily designed and used for solving graphical tasks such as 3D scene rendering or image processing. General purpose computation on GPUs enables the utilization of high computational performance of GPUs to solve complex computational problems. This study deals with an algorithm for finding a possible optimization in the Czech Republic's Helicopter Emergency Medical Service. The goal is to select the optimal locations of airbases from which it is possible to access maximum of an area of interest in the defined response time which is prescribed by law. The criteria used for the optimization of deployment can be the number of airbases, total covered area, costs or risk probability. The implementation of calculations on an off-the-shelf programmable GPU provides a processing speedup of several orders of magnitude.

**Key words:** Graphics processing units, general purpose computation, helicopter emergency medical service, coverage, optimization, magnitude

---

### INTRODUCTION

Helicopter Emergency Medical Service (HEMS) has been an integral part of pre-hospital emergency care in the Czech Republic for several years. The main advantages of HEMS include:

- High speed in order to get rescuers to the requested place
- The ability to provide resuscitation and intensive care in an inaccessible terrain
- Easier and quicker transport of the patient to target and specialized medical facility or hospital

Basics disadvantages of the HEMS are:

- High economical costs compared with ground rescue service crew
- A limited usability in case of adverse weather conditions

For proper functioning of the HEMS system, it is necessary to cover the whole area of the Czech Republic by a network of many base stations. From these bases all possible places should be available in statutory time period which represents the limit time required by law. In general, this task is called as a planar geometric

covering problem, i.e., finding an optimal coverage points in a planar surface geometric objects such as circles. This problem belongs to the group of NP complete problems (Fowler *et al.*, 1981). The solution or result can be verified in polynomial time but it is not known whether it is possible to find this solution in polynomial time. For an unbounded coverage of a plane with circles of known radius, it is possible simply to position their centers in a regular hexagonal grid (Bannai, 1999). For a limited set of points in a plane, there exists a greedy algorithm seeking local minimum (Agarwal and Sharir, 1998; Gerla and Tsai, 1995; Huang and Tseng, 2005). However, the final solution may not be optimal. In this study, proposed algorithm based on brute-force approach finds the optimal coverage of an area. The procedure uses the high computing performance of current programmable graphics cards (Couturier, 2013).

The surface area covered by one airbase can be defined as a circular area whose radius is determined by a required maximum range time, a helicopter type and weather conditions. Range time determines the time from a take-off (from a base) to landing (to the place of rescue action) and includes the time necessary for crew and helicopter preparation. This range time limit for response is set by current law for 20 min. Average flight speed of a helicopter (Kessler, 2015) is 240 km/h. Take-off time for one flight is set at 2.5 min. According to these indicators,

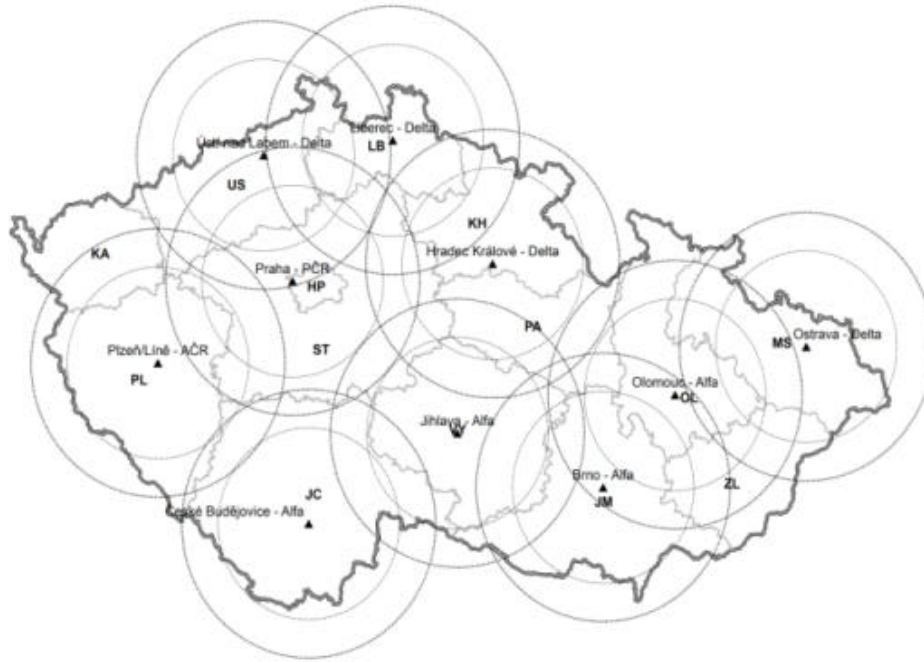


Fig. 1: Current locations of the Czech HEMS bases with 50 km (15 min) and 70 km (20 min) limits

it can be determined the radius of the circular area of 70 km. In the previously applicable 15 min time limit which was 12.5 min of flight was considered radius 50 km. Current coverage of the Czech Republic range of individual bases in 15 and 20 min limit is shown in the Fig. 1. Defined radius does not necessarily mean that all points lying within the area will always and under all weather conditions available in declared time limit. Real range may be affected by an actual situation such as weather conditions.

**Complexity of the subject:** The goal of this study is to find an optimal coverage area of HEMS bases in which the area lying outside all circle range parts will be minimal. We are looking for an optimal selection of the  $k$  bases from the total number  $n$  possible locations with a maximal total coverage area. Algorithms using brute-force solutions are based on testing all possible options that meet the specified conditions and choosing the one that best satisfies the optimization criteria. The number of all possible options is determined as the number of combinations without repetition which corresponds to a combination number:

$$\binom{n}{k} = \frac{n!}{k!(n-k)!} \quad (1)$$

Where:

$k$  = The number of airbases to be deployed in selected territory

$n$  = The number of possible locations

At the moment, in the Czech Republic is  $k = 10$  but there is a reduction effort using optimization to  $k = 9$ . Generally, the number of possible locations is infinite. However, in practice, we can restrict the possible locations of bases only in selected areas (such as in individual regions, towns or municipalities) which can be on the order of tens of thousands, depending on other parameters or in a regular grid. This grid can be for example  $10 \times 10$  km which corresponds to the order of thousands possible locations. In both cases, the calculation still counts with many combinations. Even with the computational power of today's computers, the calculation time is still too long.

In practice, deployment of HEMS bases is associated with conditions necessary for technical operation of the helicopter (such as the airport or heliport) but primarily airbase must be close to an appropriate medical facility or hospital. Otherwise, there are a lot of returns with empty helicopters from a medical facility which leads to unnecessarily increase the number of flight hours and total costs. Therefore, as potential locations of bases are considered only towns with population over 50,000 inhabitants and the municipalities in whose territory a

hospital with emergency or trauma center unit is located. The total number of possible locations is reduced to  $n = 21$  and the number of combinations for  $k = 9$  is 293,000 and for  $k = 10$  it is 352,000.

## MATERIALS AND METHODS

**Developing an algorithm and data preparation:** The total calculation time, i.e., in our case the time for finding an optimal deployment solution of HEMS bases is affected by the computational complexity of used algorithm and the time for of a single solution calculation.

Calculating the size of the covered area is defined as the union of all individual circular areas. This calculation is performed by using the geometry of circle and for a higher number of differently deployed circles is time-consuming. In this study, a solution using the geographical software ArcGIS was tested first which was implemented using a script in Python. Calculation for one single deployment took several seconds which implied that the calculation of all possible solutions would take several weeks.

The proposed solution uses the high computational performance of programmable graphics cards with GPU cores which enable parallel processing of computation methods and provide a theoretical computing power on the order of thousands GFLOPS (Billions of Operations in Floating Points per Second). If the calculating task is transformed to a graphical one applicable on the GPU, it is possible to achieve a solution in far less computational time. GPU enables programming the so-called visualization pipeline at the vertex level that processes geometry and at the pixel level of already rasterized graphical object. Coding is processed in both vertex and fragment shader processors whose code is performed in parallel on many GPU cores simultaneously.

The developed algorithm consists of two operations. The first one is the process of rendering a map of the Czech Republic as one textured rectangle. This part of the algorithm is processed on a Central Processing Unit (CPU). The second one deals with the process of checking all possible deployment combinations for each rasterized pixel. If the pixel is located inside a circular area which is determined by the distance to the airbase, it will not be rendered in the final solution. Therefore, it displays only the parts of the area which are not available from any of the bases in declared time limit. The size of the uncovered area which is also the number of displayed pixels can be easily determined by a direct query on the GPU. When the algorithm is executed for all possible combinations, the variant with minimal size of all uncovered areas can be found. This case means that there

is a maximum coverage of the entire territory. At the rendering speed of hundreds of frames per second the optimal solution can be found quickly.

For the proper function of the algorithm it is necessary to map the raster image with a marked area of the Czech Republic on rectangle rendered across the screen. It is also necessary to forward information about the actual tested combination of base deployment to the GPU processor. Due to the limited number of potential bases it is possible to encode the combination to the binary form of a 32 bit integer. Each of the zero to thirty one bit positions corresponds to one possible position of the base. The bit value determines whether the corresponding base is calculated or not. All possible combinations without repetition are generated by the required number of embedded program cycles. Generated combination is coded into the bit sequence and through a uniform variable forwarded to GPU before the process of rendering a map occurs Algorithm 1.

### Algorithm 1; Pseudocode processed on CPU:

```
int n = numberOfPossibleAirbaseLocations
int k = numberOfAirbases
forEach c in combinationsWithoutRepetition(n, k) {
    int state = bitCode(c)
    glUniform1i(state); //send to GPU
    drawTexturedScreenQuad(); //render on GPU
    results[c] = queryNumberOfDisplayedPixels()
    //get number of displayed pixels
}
findMinimum(results)
```

**GPU calculation:** Virtually all GPU calculation takes place at the pixel level in the fragment shader. By setting the size of the rendered rectangle correctly any transformation in the vertex shader can be avoided. The whole capacity of GPU is thus, used just for pixel processing. In fragment shader each pixel of the map is analyzed on the basis of its color and its distance to any potential base determined by the currently used combination. If the pixel is colored with the background color, it means that it does not lie in the defined area of the Czech Republic. Such pixel will not be processed and rendered in further calculations. In the opposite situation distance to all selected air bases is calculated sequentially and if at least one base is within the defined limit distance, again, it will not be rendered. In both cases, the pixel rendering on the GPU is interrupted by the discard command in the fragment shader. The result shows only the pixels lying within the area of the Czech Republic which are not covered by any bases (Algorithm 2). After the rasterization process of the entire rectangle with the map the GPU is queried for the number of pixels that were actually rendered. The value of the number of pixels is recorded with a code corresponding to

the combination of bases deployment for finding the minimum. Subsequently, the image is erased and rendered again for next generated combination. After rendering all combinations we, get the one which has the lowest number of pixels.

**Algorithm 2; GLSL code processed for every pixel in fragment shader on GPU:**

```

in vec2 texCoord;           //xy position in quad
out vec4 outColor;
uniform sampler2D texture;  //mapped texture
uniform int state;         //current combination
vec2 LOCATIONS[21] = vec2[(...), (...)]
//possible locations as constant or uniform variable
float LIMIT_R = ...;      //limit radius
void main() {
outColor = texture2D(texture, texCoord)
if (outColor.r < 0.1)      //background color?
discard;                  //xy out of border
for (int i = 0; i < k; i++){ //for all locations
if (state % 2 == 1){      //selected location?
dist = distance(texCoord, LOCATIONS[i])
if (dist < LIMIT_R)
discard;                  //xy within circle
state = state >> 1;      //next location
}
}
}
}

```

The presented algorithm solves the problem of coverage area by using decomposition of the task in many elementary squares, each represented by individual pixel. Therefore, it is necessary to consider the question of the accuracy that can be achieved in the calculation. The common size of the frame buffer, i.e., the image into which the result is rendered is in the order of hundreds of pixels in each dimension. This also affects the size of the image mapped to the rendered rectangle. The size of the Czech Republic (measured from East to West) is about 500 km. Considering the width of the raster of image buffer to be 512 pixels, one pixel corresponds approximately to the size of a square area with the side of 1 km<sup>2</sup>. This is sufficient for the task accuracy. Using a half of the raster size will double the size of a square area side.

The second limitation is due to a character of the covered geographical area. In fact, it is not a plane or flat area but part of the Earth’s surface. Commonly used maps are only projections of the spherical surface to a plane. To achieve greatest precision the task should be treated as three-dimensional. But for small areas (in size of hundreds of kilometers) the distortion is negligible.

The last limitation is particularly important for rugged contours or for discrete areas. In such situation the distance to a specific position from the nearest base cannot be computed by Euclidean distance, as the shortest path could lead over the territory of a foreign country which in practice is unacceptable. This problem

can occur during the computation of the coverage of the Czech Republic as well, especially in border areas. A simple modification of the proposed algorithm which would keep the final path within the Czech territory is not possible. However, since, the influenced areas are usually small and also very sparsely populated, the problem was not considered in calculation.

**RESULTS AND DISCUSSION**

The algorithm was implemented using the OpenGL graphic library and its Java wrapper JOGL. Testing was carried out on a laptop with NVIDIA GeForce 630M GPU. Grid size of 256×256 pixels achieved the average number of over 600 rendered Frames Per Second (FPS). For the 9 air bases selected from 21 possible locations the calculation took 8 min. In the case of the 10 bases the calculation took approximately 9.5 min. By comparison, the original estimated time of calculation using ArcGIS was 2 weeks.

The optimal solution for the case of 9 air bases (Solution No. 1 in the following text) corresponds to a situation where the number of HEMS bases is reduced by one airbase from the current status. The proposed algorithm found the best coverage area with the location of bases in Brno, Plzen, Hradec Kralove, Zlin, Kladno, Opava, Ceske Budejovice, Jihlava and Decin (Fig. 2). This solution covers 92.4% of the territory of the Czech Republic.

The solution for the 10 air bases (Solution No. 2) can be compared with the existing deployment of the bases. The solutions match in six locations (i. e., Praha, Brno, Plzen, Hradec Kralove, Ceske Budejovice and Jihlava). On the other hand in four cases new locations of the bases were selected and they are Opava, Zlin, Karlovy Vary, Decin over existing bases in Ostrava, Olomouc, Liberec and Usti nad Labem (Fig. 1-3).

The solution No. 2 coverage value of 96.5% of the Czech Republic area can be compared with the value of 92.9% coverage for existing locations while keeping the same total number of air bases. The percentage coverage values also show that an appropriate deployment can be found with comparable coverage of the Czech Republic area but with a smaller number of HEMS bases (Table 1).

Table 1: Coverage comparison of calculated solutions to the existing deployment

Solution	Number of bases	Total area coverage (%)
No. 1 (calculated)	9	92.4
No. 2 (calculated)	10	96.5
Existing (current)	10	92.9

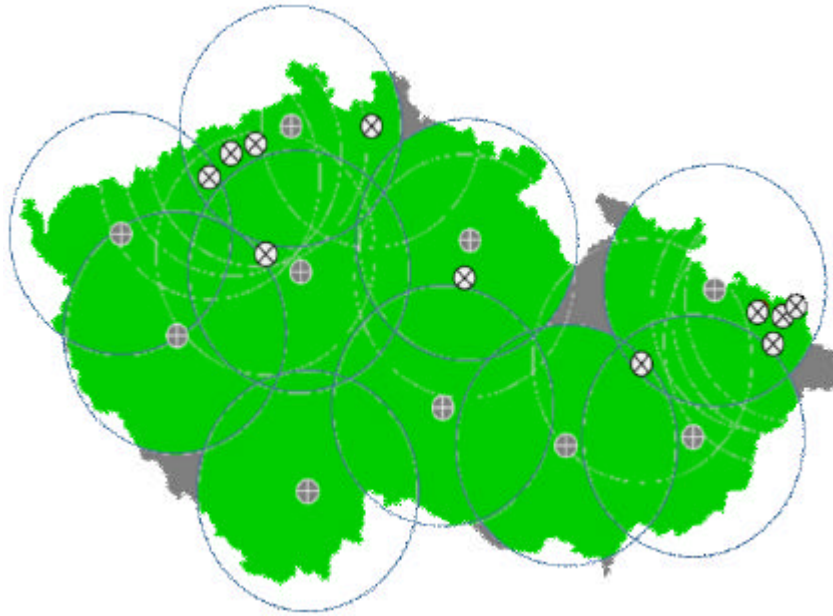


Fig. 2: Solution No. 1 for the 9 HEMS bases covering 92.4% of the Czech Republic area (• markers indicate the locations of selected bases, • markers refer to analyzed locations, green-colored areas show the range territory in the given time limit, pixels of the green area are discarded during computation task)

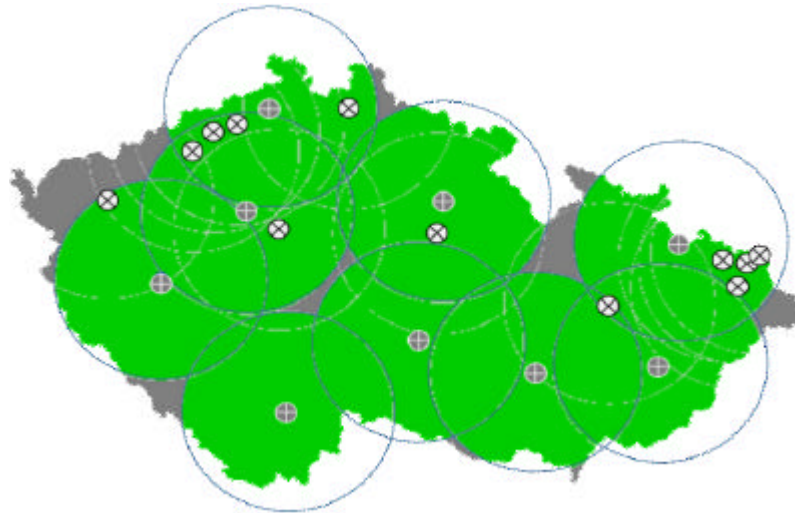


Fig. 3: Solution No. 2 for the 10 HEMS bases covering 96.5% of the Czech Republic area (• markers indicate the locations of selected bases, • markers refer to analyzed locations, green-colored areas show the range territory in the given time limit)

### CONCLUSION

The proposed algorithm and its implementation on GPU enabled the authors to find a solution for the specific case of HEMS bases coverage in the Czech Republic. The original and very time consuming solution has been

accelerated by using high performance GPU which is commonly available and programmable. The implementation accelerated the final calculation roughly a thousand times. With the usage of more powerful GPUs the calculation could be even more accelerated. The calculated optimal solution requires only 9 airbases and

covers 92.4% of the Czech Republic which is only 0.5% point less than the actual current deployment with 10 bases. Another solution counting with the total number of 10 air bases covers about 3.6 % point more of the total Czech Republic area.

### **RECOMMENDATIONS**

Future research and study can work with the upgraded algorithm in order to find more permissible solutions, not just the one with maximal coverage. In practice, it is necessary to include more information about the selection process. Decision-making process based just on one criterion may not be always correct. The consequent use of an optimized real network of HEMS would consider other metrics based on population density, road structure and quality or the probability of action (or accident).

### **ACKNOWLEDGEMENTS**

This study was supported by the research project SPEV, University of Hradec Kralove, Faculty of Informatics and Management, 2016.

### **REFERENCES**

- Agarwal, P.K. and M. Sharir, 1998. Efficient algorithms for geometric optimization. *ACM. Comput. Surv.*, 30: 412-458.
- Bannai, E., 1999. *Sphere Packings, Lattices and Groups*. Springer, Berlin, Germany.
- Couturier, R., 2013. *Designing Scientific Applications on GPUs*. Chapman and Hall/CRC, USA., Pages: 473.
- Fowler, R.J., M.S. Paterson and S.L. Tanimoto, 1981. Optimal packing and covering in the plane are NP-complete. *Inf. Process. Lett.*, 12: 133-137.
- Gerla, M. and J.T.C. Tsai, 1995. Multicluster, mobile, multimedia radio network. *Wireless Networks*, 1: 255-265.
- Huang, C.F. and Y.C. Tseng, 2005. The coverage problem in a wireless sensor network. *Mob. Networks Appl.*, 10: 519-528.
- Kessler, C., 2015. Helicopter emergency medical service: Motivation for focused research. *CEAS. Aeronaut. J.*, 6: 337-394.