COMPUTER METHODS IN MATERIALS SCIENCE
Informatyka w Technologii Materiałów

Publishing House
A K A P I T

Vol. 19, 2019, No. 1

FRAMEWORK FOR TOPOGRAPHIC MESH GENERATION AND ITS APPLICATION TO THE POLLUTION SIMULATIONS IN KRAKÓW AREA

KRZYSZTOF PODSIADŁO[1], ALBERT OLIVER SERRA[2], MACIEJ PASZYŃSKI[1*]

[1] Department of Computer Science, Faculty of Electronics, Telecommunication and Computer Science,
AGH University of Science and Technology, Al. Mickiewicza 30, 30-059 Kraków, Poland
[2]Instituto Universitario SIANI, Edificio Central del Parque Científico y Tecnológico,
Campus Universitario de Tafira, 35017 Las Palmas de Gran Canaria, Spain
**Corresponding author: maciej.paszynski@agh.edu.pl*

Abstract

Air pollution is receiving a lot of interest nowadays. It is visible especially in the Kraków area, as this is one of the most polluted cities in Europe. People living there are more and more aware of the problem, what causes raising of various movements and NGOs that are trying to improve air quality. Unfortunately, this is not as simple as people usually think: air pollution grows because of multiple factors including traffic, climate, heating of buildings in winter, city's architecture, etc. In this paper, we simulate and predict pollution with high resolution, as air quality can vary significantly over a distance of even a few hundreds of meters. Air quality simulation is a multidisplinary endeavor, comprising physical models (meteorological and chemistry) as well as numerical methods (geometry discretization, time and space discretization, etc.). It has been found that there is no proper method for automatic terrain mesh generation, so an algorithm for doing it is presented in this document as its significant part.

Key words: Pollution simulations in Kraków, Advection-diffusion-reaction, Mesh generation

1. INTRODUCTION

Air pollution is receiving a lot of interest nowadays. It is visible especially in the Kraków area, as this is one of the most polluted cities in Europe, see (Air Quality in Europe report, 2017). People living there are more and more aware of the problem, what causes raising of various movements and NGOs that are trying to improve air quality. Unfortunately, this is not as simple as people usually think: air pollution grows because of multiple factors including traffic, climate, heating of buildings in winter, city's architecture, etc.

The full three-dimensional air pollution simulations are very important since they can predict in a realistic way the influence of different pollution sources to the distribution of pollution in the affected area, taking into account the topography of the area, distribution and intensity of the of pollution sources, as well as wind force and directions in the area over the simulated period. Its goal is to prove that it is possible to simulate and predict pollution with high resolution, as air quality can vary significantly over a distance of even a few hundreds of meters. To make a good three-dimensional simulator of the pollutant it requires several interconnected components, including the mesh generator that provides a good approximation to the topography of the area, the wind field simulator, based on some available point measurements, and finally the solver computing the advection-diffusion-reaction model

ISSN 1641-8581

including several components of the pollution, with e.g. stabilized finite element method.

In this paper we focus on the first important aspect of the air pollution simulation, namely we focus on algorithms for proper mesh generation with the application of the air pollution simulations. It has been found that there is no proper method for automatic terrain mesh generation, so an algorithm for doing it is presented in this document as its significant part.

The mesh generation is a very important field of computational science and is discussed for many years. It is hard to imagine for instance Finite Element Method without a proper meshing. Here, we focus on three-dimensional mesh, constructed from tetrahedral elements, a subset of three-dimensional cube, cut in the center to represent the terrain structure. The goal is to be able to generate the three-dimensional tetrahedral mesh in an automatic way. As an input, we use a solid which base is a terrain represented by a regular grid and the other walls are planes. When it comes to two dimensions, the situation is relatively simple, as this subject is well researched (Shewchuk, 2002). The situation with 3D is much more complicated. The most direct solution is to run a simple tetrahedralization algorithm such as Dealaunay triangulation on the terrain grid constrained from above, but it will end in producing a large number of elements. It should be avoided, since the computational cost of the pollution simulation is proportional to the number of elements in the mesh.

One of the more efficient methods to do so is called Meccano (Cascon et al., 2009). Generally, it consists of 3 steps:

1. Partition the surface of a genus-0 solid into 6 patches and parametrize it each one to a face of the cube.
2. Subdivide the cube into 6 tetrahedra and refine using Kossaszcy method according to a distance error in the physical volume.
3. Move the inner nodes in the physical space using the SUS mesh optimization.

Another one is presented in (Montenegro et al., 2002). It generates a cloud of points in the 3D-space according to the hierarchy of the 4-T Rivara refinement of the terrain. The author believes that another algorithm can provide better results. It would use Rivara's longest-edge refinement, (Rivara, 1997), to triangulate the terrain (since this mesh is a regular one, refinement procedure will be effective) and then Delaunay tetrahedralization will be run.

The mesh generation algorithm presented in this paper is available as an open access library, ready for download and usage.

2. MESH REFINEMENT ALGORITHM

The ultimate goal of the mesh refinement algorithm is to create a tetrahedral mesh of lower part of the troposphere over the simulated area. The process has two main steps which are: surface mesh generation and tetrahedralization of the entire volume. Such approach allows the resulting mesh to be precise on the surface, which is really important afterwards when it is used in wind and pollution simulation.

The first and most important step of the algorithm is creating a surface mesh (Algorithm 1). Given a set of points representing a rectangle surface (they will be called initial points) we approximated them by triangles. We apply the Longest-edge refinement algorithm (Rivara, 1997). We use a 3D projection of points during calculation of the longest edge. The Algorithm is the following. At first, it generates initial triangles and then it refines each of them as well as the ones created during the process as many times as is required to meet the condition.

```
1 function refine_mesh(T, tolerance)
1 T <- Generate initial triangles
2 Repeat
3 For each triangle t in T
4 If refinement condition in t is
   met (check_refinement(t, tolerance)==true)then
5    T' <- refine triangle t
                  (call refine(t))
6    T <- T+T'
7 Endif
8 Endfor
9 Until no refinements done
10 return T
```

***Algorithm 1.** Mesh refinement algorithm.*

The refinement condition in line 4 requires a call to the check refinement algorithm described in pseudo-code Algorithm 2. The tolerance of the refinement algorithm is a global variable controlling the global accuracy of the refined mesh.

```
function check_refinement(t, tolerance)
IP<- initial points of triangle t
e <- given tolerance
for each point P=(Px;Py;Pz)in IP do
    P0 <- projection of P onto OXY-plane
    t0 <-projection of t onto OXY-plane
    if P0 lies inside t0 then
        (k1;k2;k3) <- barycentric coordinates of P0
        in reference to t0
        if dist(k1Az+k2Bz+k3Cz,Pzj)> e then
            return true
        end if
    end if
end for
return false
```

***Algorithm 2.** Check refinement algorithm.*

Initial triangles should be, of course, possibly large and their corners' measures should be close to 45 or 90 degrees. This condition is to avoid elongated elements which may lead to Jacobians over these elements close to numerical zero, which in turn results in convergence problem during the pollution simulation algorithms. To achieve that, the greatest common divisor of surface's sides could be used as a length of their legs. However, if it is too small it is better to use bigger triangles with measures close to it.

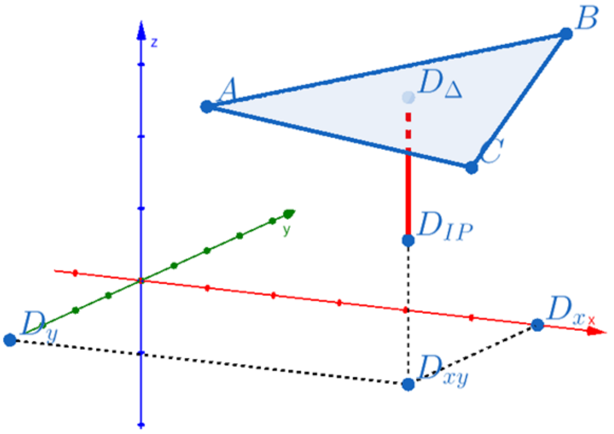

***Fig. 1.** 3D projection of points.*

The algorithm checking the refinement condition performs the projection of the triangles and the points from the initial data set into the planes of coordinates system 3D projection of points is shown in figure 1. Actually, it checks if the initial points are close enough to some generated triangle. If they are located inside the triangle on the projected planes, the triangle does not require a correction. However if they are outside the projected triangle, we need to refine the triangle to get closer.

The refinement itself is based on (Rivara, 1997) and presented as Algorithm 3. This is a recursive algorithm, that looks for the triangle which lies by the longest edge (calculation of this edge is the difference between 2D and 3D versions of the algorithm) of the given one and refines it at first.

```
function refine (t)
t <- given triangle
t0 <- triangle lying by the longest edge of t
if t0 does not exists then
    Split t by its median
    return
end if
t00 <- triangle lying by the longest edge of t0
if t00 = t then
    Split t by its median
    Split t0 by its median
    return
end if
call refine(t0)
t0 triangle lying by the longest edge of t
Split t by its median
Split t0 by its median
```

***Algorithm 3.** Triangle refinement algorithm.*

There are two stop conditions: If the neighbouring triangle shares its longest edge with the given triangle, then a point in the middle of this segment is created and connected to opposite vertices of both triangles. The recursion stops also when the longest edge is on the border of the domain. Then, similarly, the triangle is divided by its median.

A sample refinement is shown in the figure 2. We want to refine triangle t0. We look at its longest edge, which points to triangle t1, which longest edge points to triangle t2. Triangles t2 and t3 are pointing to each other, so the stop condition is met here. We split them by their medians. Then, triangles t1 and t0 have common longest edge. We also split them. The last step is to refine triangles t0 and t1. The result is shown in figure 2.

The mesh generation algorithm has been implemented in the TerGen, Automatic Terrain Mesh Generator, a stand-alone tool for adaptive mesh generation based on the data available from the topographical database Shuttle Radar Topography Mission, (Farr et al., 2005). The exemplary result of the mesh generation for the area of Kraków is presented in figure 3.

The algorithm has been implemented as a standing alone library available to download from githhub repository. The installation and usage manual is described in the Appendix.

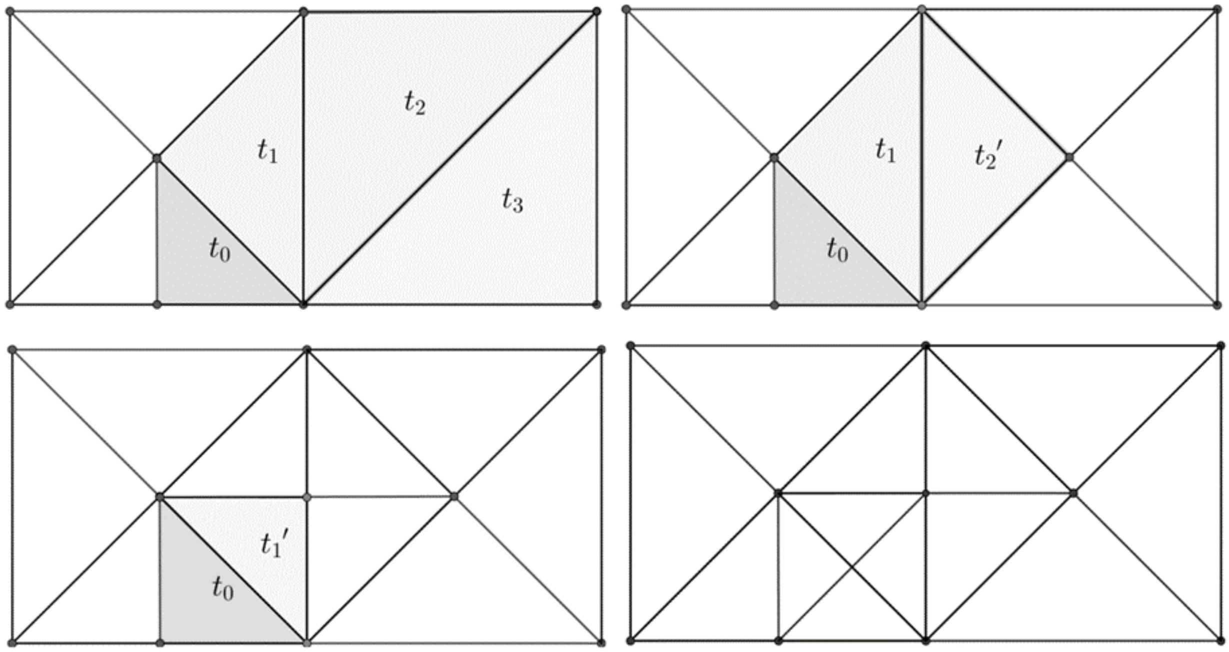

Fig. 2. *Iterations of the mesh correcting algorithm.*

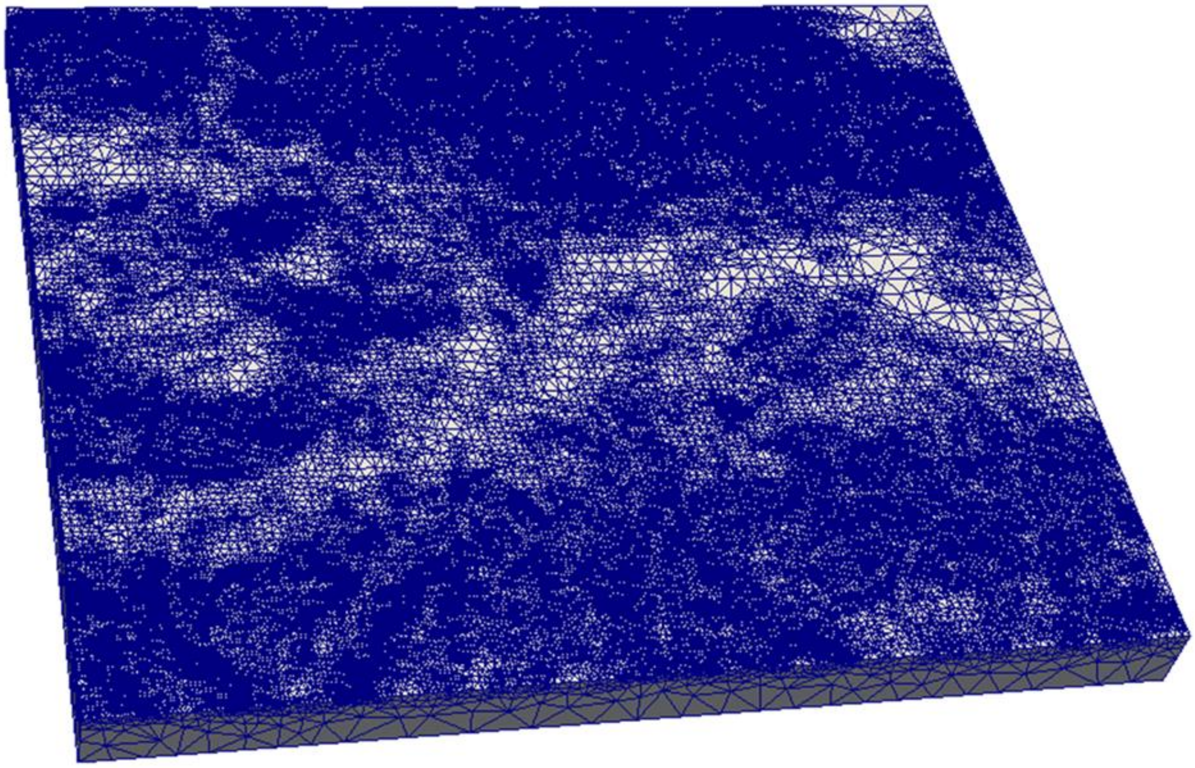

Fig. 3. *Three-dimensional mesh of Krakow area.*

3. THE POLLUTION EQUATIONS

Following Oliver et al., 2012, we focus on the advection-diffusion-reaction equations:

$$\frac{\partial c}{\partial t} + u^\circ \nabla c - \nabla^\circ \left(K \nabla c \right) = e + s\left(c \right) \tag{1}$$

where: $c(x,t)$ is the concentration of emissions (unknown) , namely the vector of 4 unknowns:
c1=[SO2], c2=[SO4], c3=[NO4], c4=[NO3],
$u(x,t)$ is the wind velocity (given), and $e(x,t)$ is the emission (given), $s(c)$ are the chemical vectors, $s(c) = Ac$, where A is the chemical reactions matrix:

$$A = \begin{bmatrix} -0.15 & 0 & 0 & 0 \\ 0.15 & 0 & 0 & 0 \\ 0 & 0 & -0.3 & 0 \\ 0 & 0 & 0.3 & 0 \end{bmatrix} \times 10^{-6} \tag{2}$$

and K diagonal diffusion matrix. The horizontal diffusion coefficient is eqaul to 8×10^{-6} m^2/s for the four species, and the vertical diffusion coefficient 4×10^{-6} m^2/s for the four species of the pollution.

We consider the following boundary conditions: At the top of the chimney:

$$c(x,t) = c^{emi}(x,t) \tag{3}$$

$$c_1^{emi} = c_3^{emi} = 6\frac{g}{m^3}, \quad c_2^{emi} = c_4^{emi} = 0$$

which means that the second and the fourth component are created by chemical reactions expressed by $s(c)=Ac$ in the PDE:

$$n^\circ \left(K \nabla c \right) = -V^d c \tag{4}$$

at the terrain level, where V^d = 1.3×10^{-3} m/s (so-called diagonal term of the deposition matrix):

$$c\left(x,t\right) = 0 \tag{5}$$

at wind outlet, and

$$n^\circ \nabla c = 0 \tag{6}$$

also at the wind outlet.

We consider time step size dt = 10 s, and e = 0, which means that the right-hand side is enforced by the emission from the top of the chimney, expressed by the Dirichlet boundary condition.

The strong equations are discretized with finite element method,:

$$\left(\frac{\partial c}{\partial t}, v \right) + \left(u^\circ \nabla c, v \right) + \left(K \nabla c, \nabla v \right) = \left(e, v \right) + \left(Ac, v \right) \tag{7}$$

We use the Crank-Nicolson time integration scheme, SUPG stabilization of every time step, and the GMRES solver in every time step.

4. NUMERICAL RESULTS

The simulation has been executed for the terrain data of the area of Kraków, with the wind assumed to blow in the east direction from the west boundary. The assume boundary wind condition has been extended to the entire domain by computing the wind velocity from divergence free equation. The resulting wind velocity map is presented on left panel in figure 4. Next, we solved the advection-diffusion-reaction problem (1) assuming the pollution coming from the west. The solution is the time advancing movie, as presented in figure 5.

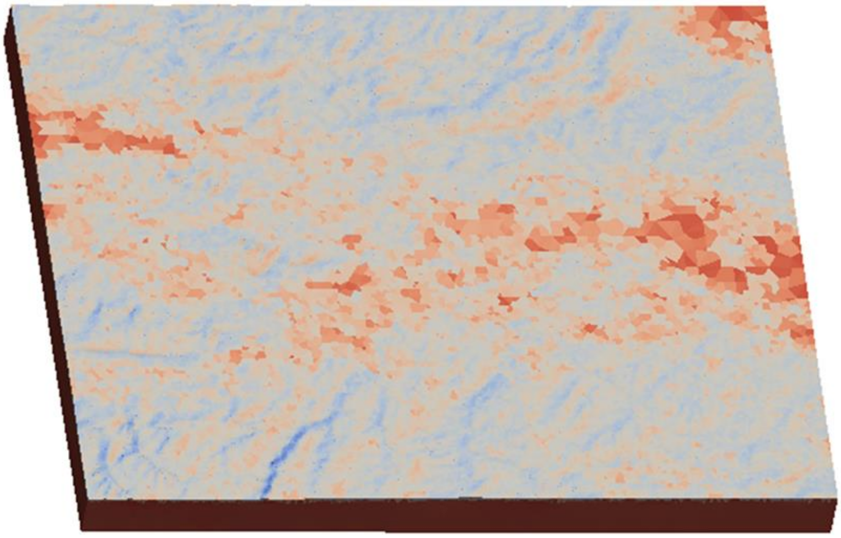

***Fig. 4.** Computed west wind distribution in the Krakow area.*

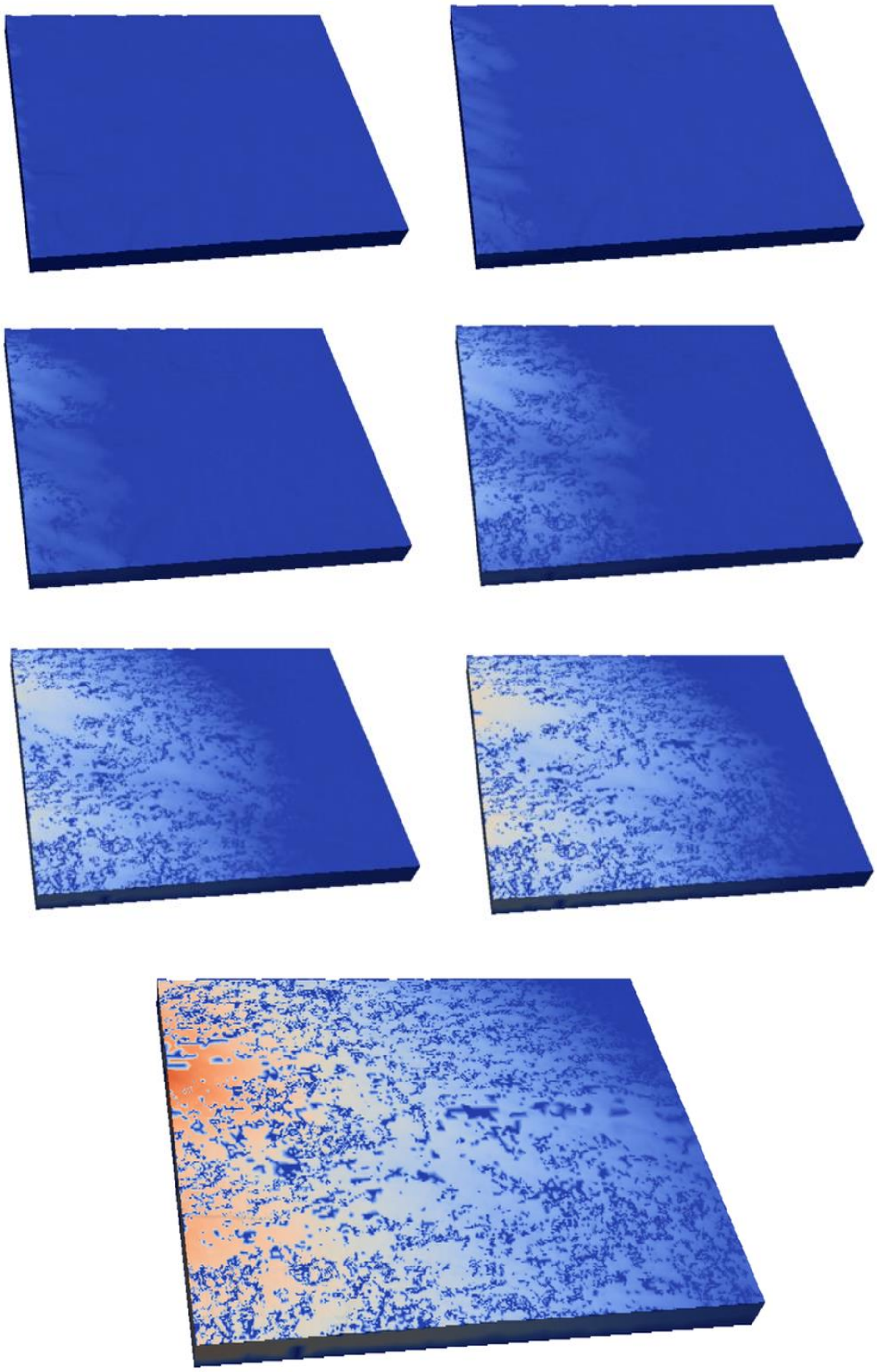

Fig. 5. Simulation of pollutant propagated by the west wind with velocity 3 m/s in Krakow.

The mesh generator produced a mesh representing 500 x 3000 meters domain of the Krakow area, starting from 24 elements on the initial mesh and adapting down to around 50,000 elements on the final 2D mesh. The 3D model has been obtained by adding around 5 levels of 3D elements, with the total number of half a million of tetrahdral elements. This quality of mesh allowed for stable simulation of time-dependent pollution problem, as illustrated in figures 4-5.

5. CONCLUSIONS

The paper describes the process of pollution simulation in Kraków area. The introduced stages include the topographic mesh generation, wind simulation, and the advection-diffusion-reaction solver. We focus here on a development of the mesh refinement algorithm, generating the approximation of a terrain, given as a set of points available from the Shuttle Radar Topography Mission. The library and the algorithm has been tested on a simulation of propagation of pollution in Krakow area.

APPENDIX

The terrain mesh generation tool TerGen is availabe at https://github.com/Sambard/meshgen

The installation requires CMake in version 3.1 or higher and C compiler.

After compilation you can simple run it. Arguments must specify input (i.e. -i or a set: -d, -N, -S, -E, -W), as well as output (-o and either -p or -m). All possible arguments are listed below:

Parameter	Meaning
-t <tolerance>	Sets the refinement tolerance in meters. Defaut option is 5.
-s <requsted_size>	Sets the lenght of initial triangles' catheti. It should be lower or equal than the shorter side of the domain.
-i <data_file>	Indicates the input file in ASCII format.
-o <output_file>	Base of filename in which results should be saved. The extension will be added depending on chosen output format.
-d <data_dir>	Sets a location of directory with data files in SRTM format.
-N <coordinate>	North border of domain.
-S <coordinate>	South border of domain.
-E <coordinate>	East border of domain.
-W <coordinate>	West border of domain.
-p	Write results into AVS UCP ASCII (.inp) format.
-m	Write results into .smesh format.
-g	Skip conversion into UTM coordinates.
-u	Convert to UTM coordinates before refinement
-U	Convert to UTM coordinates after refinement (default).
-h	Use 3D version of algorithm (2D is default).

ACKNOWLEDGMENTS

This work is partially supported by National Science Centre, Poland grant no. 2017/26/M/ST1/00281.

REFERENCES

Cascon, J. M., Montenegro, R., Escobar, J. M. , Rodriguez, E., Montero, G., 2009, The MeccanoMethod for Automatic Tetrahedral Mesh Generation of Complex Genus-Zero Solids, *Proc. 18th Int. Meshing Roundtable*, Salt Lake City, 463-480.

European Environment Agency. Air Quality in Europe - 2017 report. 13/2017.

Farr, T.G., Rosen, P.A., Caro, E., Crippen, R., Duren, R., Hensley, S., Kobrick, M., Paller, M., Rodriguez, E., Roth, L., Seal, D., Shaffer, S., Shimada, J., Umland, J., Werner, M., Oskin, M., Burbank, D., Alsdorf, D., 2005, The Shuttle Radar Topography Mission, *Reviews of Geophysics*, 45, 2, https://agupubs.onlinelibrary.wiley.com/ doi/pdf/10.1029/2005RG000183.

Montenegro, R., Montero, G., Escobar, J. M., Rodriguez, E., Gonzalez-Yuste, J. M., 2002, Tetrahedral Mesh Generation for Environmental Problems over Complex Terrains, *Lecture Notes in Computer Science, Computational Science*, 2239, 335-344.

Oliver, A., Montero, G., Montenegro, R., Rodríguez, E., Escobar, J.M., Pérez-Foguet, A., 2012, Adaptive finite element simulation of stack pollutant emissions over complex terrains, *Energy*, 49, 47-60.

Rivara, M.C., 1997, New longest-edge algorithms for the refinement and/or improvement of unstructured triangulations, *International Journal for Numerical Methods in Engineering*, 40(18), 3313-3324.

Shewchuk, J.R., 2002, Delaunay refinement algorithms for triangular mesh generation, *Computational Geometry*, 22(1-3), 21-74.

FRAMEWORK DO GENERACJI TOPOGRAFICZNYCH SIATEK OBLICZENIOWYCH I JEGO ZASTOSOWANIE DO SYMULACJI ZANIECZYSZCZEŃ W OKOLICY KRAKOWA

Streszczenie

Problem zanieczyszczenia powietrza staje się coraz bardziej zauważalny w czasach dzisiejszych. Dotyczy to szczególnie okolic Krakowa, będącego jednym z najbardziej zanieczyszczonych miast w Europie. Ludzie mieszkający w rejonie Krakowa stali się bardziej świadomi problemu zanieczyszczeń, co zmotywowało do powstawania różnych organizacji publicznych oraz zainicjowało różne działania władz na szczeblu lokalnym oraz globalnym w celu polepszenia jakości powietrza. Niestety problem powstawania i propagacji zanieczyszczeń jest bardzo skomplikowany, i łączy w sobie wiele czynników takich jak transport miejski, lokalny klimat, problem ogrzewania budynków w okresie zimowym, struktura architektury miejskiej, itp. W artykule tym zajmujemy się symulacjami i predykcją zanieczyszczeń z wysoką rozdzielczością, z uwagi na fakt, iż jakość powietrza może się istotnie zmieniać z jednego miejsca do drugiego na odległości kilkuset metrów. Symulacje zanieczyszczeń to zadanie wielodyscyplinarne, łączące ze sobą modele fizyczne (meteorologiczne i chemiczne), oraz modele numeryczne (dyskretyzacja geometrii, dyskretyzacja czasowo-przestrzenna). W szczególności zajęliśmy się skonstruowaniem nowego algorytmu automatycznej generacji topograficznych siatek obliczeniowych i jego wykorzystaniem w zagadnieniach symulacji zanieczyszczeń.

Received: January 19, 2019.
Received in a revised form: April 21, 2019.
Accepted: May 13, 2019.