

# Numerical simulation of 2D electrodynamic problems with unstructured triangular meshes

D.A. Fadeev<sup>1</sup>

<sup>1</sup> Institute of Applied Physics RAS, Nizhny Novgorod, Russia

## Abstract

We present a generalization of standard leap-frog plus Yee mesh approach for Cauchy problem in electrodynamics simulations on unstructured triangulated mesh. The presented approach still inherits from finite-difference time-domain and do not use techniques developed in finite-volume time-domain approach. In the paper the whole flow from mesh creation to actual simulation is presented. The proposed computation flow is parallel ready and can be implemented for distributed systems (computation servers, graphical processing units, etc.). We studied the influence of non-regular triangulation on stability and dispersion properties of numerical solution.

**Keywords:** numerical approximation and analysis, mathematical methods in physics, computational electromagnetic methods.

**Citation:** Fadeev DA. Numerical simulation of 2D electrodynamic problems with unstructured triangular meshes. *Computer Optics* 2019; 43(3): 385-390. DOI: 10.18287/2412-6179-2019-43-3-385-390.

**Acknowledgements:** Author acknowledges the support from Federal Research Center Institute of Applied Physics of the Russian Academy of Sciences (project No.0035-2014-0020) and program of the Presidium of the Russian Academy of Sciences "Nonlinear dynamics in mathematical and physical sciences" (project No 0035-2018-0006).

## Introduction

Typically dynamics of electromagnetic fields in vacuum or media is calculated via advancing of electromagnetic field values set in the nodes of regular Manhattan Yee [1] mesh according to Maxwell equations plus equation for media response. In some cases the above approach is reduced to some scalar equations [2], lower order with respect to time derivative [3, 4, 5] etc. In this paper we will discuss those cases when vector nature of electrodynamic radiation matters and pure Maxwell equations should be solved [6, 7, 8]. In the case of free space or interaction of electromagnetic radiation with soft-bounded media e.g. air plasma [9] regular Manhattan meshes seems to work fine if resolution is enough for media gradients and wavelengths under consideration. In the cases when electromagnetic radiation interacts with condensed media the geometry starts to play a significant role. Even if boundaries are straight lines (e.g. triangle particle) but those lines does not match mesh constraints the local amplification of electromagnetic fields instantly occurs along whole particle surface. These amplifications can be treated in average and for linear problems does not play a significant role. In cases of non-linear problems such amplifications can lead to over estimation of nonlinear terms or/and result in some instabilities. In such cases

the general type of meshes has to be used to match the geometry of the objects in simulation area. The most natural mesh is triangulated mesh.

In the next section we will discuss the implementation of algorithm that can advance field values to simulate dynamics of electromagnetic radiation. Then we will study properties of the proposed method. In the last section we will describe how the mesh can be generated and optimized for parallel calculations.

Before getting to the details let us to present basic idea of switching from regular quad meshes to triangulated meshes. Standard Yee mesh shown in Fig. 1a assumes setting electrodynamic fields in some specially shifted positions. This allows using a simplest form of discrete approximation of partial derivatives. In Maxwell equations and linear hydrodynamic equations all partial derivatives are of the first order so the best approximation is achieved exactly in middle point with respect to the nodes of operator operands. Very similar thing can be naively achieved if we bind all  $\mathbf{E}$  and  $\mathbf{j}$  vectors to the edges of the triangle mesh, magnetic field  $\mathbf{B}$  vector will be then bound to some center point of triangles. For plasma like media we can attach charge density  $n$  to the nodes of the mesh (see Fig. 1).

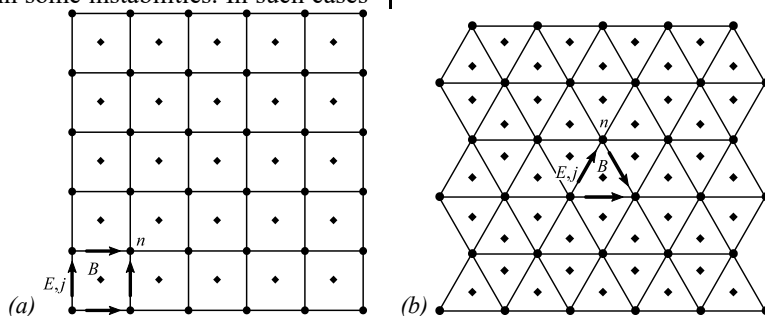


Fig. 1. Generalization of quad mesh (a) to triangle mesh (b) for electrodynamic simulations

Note that the above approach differs from another known approach utilizing benefits of unstructured meshes FVTD [10, 11]. In that approach both field values  $\mathbf{E}$  and  $\mathbf{B}$  per current cell are bound to same cell point (typically some center point) and both values are known at the same moments of time. In our methodology the field values are known in different time points shifted by  $\Delta t/2$  allowing us to adopt leap-frog approach. The stated approach was also studied in papers [12, 13, 14] and in [15] as well.

### 1. Numerical simulation

We will consider Maxwell equations in vacuum, generalization to plasma like media will be given in the end of this section. We will not consider dielectric media here. In case of dielectric media more accurate analysis should be done for “mass matrices” which represent discretization for dielectric and magnetic permittivities [12, 13]. The discrete formulation for unstructured mesh remains its general form:

$$\begin{aligned} \frac{\partial \mathbf{E}}{\partial t} &= \hat{\mathbf{L}}_B \mathbf{B}, \\ \frac{\partial \mathbf{B}}{\partial t} &= \hat{\mathbf{L}}_E \mathbf{E}. \end{aligned} \quad (1)$$

Here linear operators  $\hat{\mathbf{L}}_E$  and  $\hat{\mathbf{L}}_B$  are discrete formulations of curl operator. It is worth to introduce the following short notation for above system as well:

$$\frac{\partial \mathbf{V}}{\partial t} = \hat{\mathbf{M}} \mathbf{V}, \quad \mathbf{V} = \begin{pmatrix} \mathbf{B} \\ \mathbf{E} \end{pmatrix}, \quad \hat{\mathbf{M}} = \begin{pmatrix} 0 & \hat{\mathbf{L}}_B \\ \hat{\mathbf{L}}_E & 0 \end{pmatrix}. \quad (2)$$

Let us accurately write down the exact form of  $\hat{\mathbf{L}}$  operators on triangular mesh. In the Fig. 2 we present all necessary symbols to compose those operators.

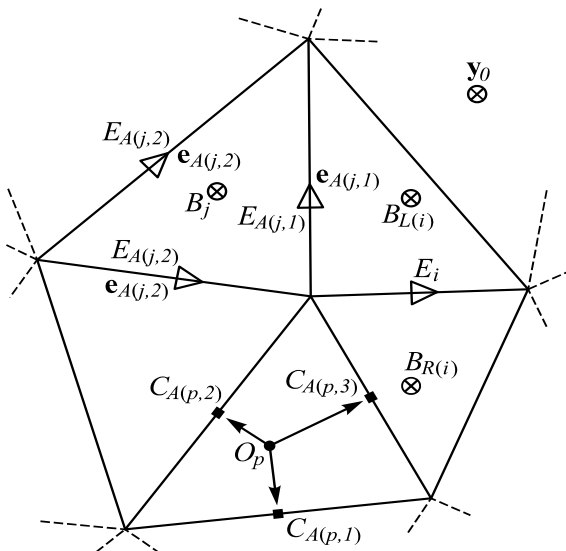


Fig. 2.  $E_i$  is the projection of electric field to  $i$ -th edge in the center of the edge,  $B_i$  is a single magnetic field component (along  $y_0$  orth) in the center of circumcircle of  $i$ -th triangle,  $e_i$  is an edge vector with arbitrary direction (since edges are shared we cannot apply any winding rule to choose some specific direction for edge vectors)

Using notation from Fig. 2 operators  $\hat{\mathbf{L}}_E$  and  $\hat{\mathbf{L}}_B$  will obtain the following form:

$$\frac{\partial E_i}{\partial t} = \frac{B_{R(i)} - B_{L(i)}}{O_{R(i)} O_{L(i)}}, \quad (3)$$

$$\begin{aligned} \frac{\partial B_j}{\partial t} &= \\ &= \frac{-1}{S_j} \sum_{q=1}^3 \text{sgn} \left( \mathbf{y}_0 \left( \overline{O_j C_{A(j,q)}} \times \mathbf{e}_{A(j,q)} \right) \right) |e_{A(j,q)}| E_{A(j,q)}. \end{aligned} \quad (4)$$

Here  $R(i)$  and  $L(i)$  are the indexes of right and left triangles correspondingly with respect to  $i$ -th edge.  $A(i, j)$  is the index of  $j$ -th edge adjacent to  $i$ -th triangle.  $O_i$  is a circumcircle center of  $i$ -th triangle and  $C_i$  is the center point of  $i$ -th edge.

It can be noted that at least  $\hat{\mathbf{L}}_B$  approximation does not achieve its best at the point where  $\mathbf{E}$  vector is set since  $\mathbf{E}$  is not necessary falls exactly between  $O_R$  and  $O_L$ . The best approximation is achieved for mesh of equilateral triangles. We will discuss meshing and quality criteria two sections later.

It is worth to note that like in structured quad mesh from (3) it follows that  $\text{div} \mathbf{E}$  remains constant over time. We will not prove this fact in details in order to not overload the scheme in Fig. 2 with additional notations. We will just notice that  $\text{div}$  can be calculated as a flux of  $\mathbf{E}$  through the boundary of the Voronoy [16] diagram cell.

To generalize such an approach to plasma like media one can add current vector  $\mathbf{j}$  to each center of mesh edge. In that case equations for currents become local and without any matrix formalism can be written in the following form:

$$\frac{\partial \mathbf{j}}{\partial t} = n \mathbf{E} - \nu \mathbf{j}, \quad (5)$$

where  $n$  and  $\nu$  are plasma density and collision frequency correspondingly. Since electromagnetic field  $\mathbf{E}$  and current density  $\mathbf{j}$  are set in the same point and are parallel so no extra interpolation is needed. According to Maxwell equations current should be added to Eq.(3) as:

$$\frac{\partial E_i}{\partial t} = \frac{B_{R(i)} - B_{L(i)}}{O_{R(i)} O_{L(i)}} + j_i. \quad (6)$$

Here we used dimensionless equations, thus no speed of light, electric charge or  $4\pi$  factor appears in above equations. This natural generalization of vacuum problem (3, 4) gives the system (4, 5, 6) that can be solved with leap-frog method [17]. In the Fig. 3 we present snapshots of solution achieved for a bounding box having plasma like triangle inside and  $\mathbf{B}_\Sigma = 0$  boundary condition. As seen from presented snapshots the quasi planar wave hits the triangle and reflects on it. Solution is stable with criteria for  $dt$  discussed below. Fig. 3a and 3c show the results achieved with unstructured mesh, Fig. 3b and 3d show the results for the same problem obtained with classic FDTD with Yee mesh (the full movie for modeling with unstructured mesh is available at [18]). The source code can be found in GitHub page (see the link in the end of the paper).

In the next chapters we will carefully study how switching from structured quad mesh to unstructured triangle mesh impacts dispersion and stability.

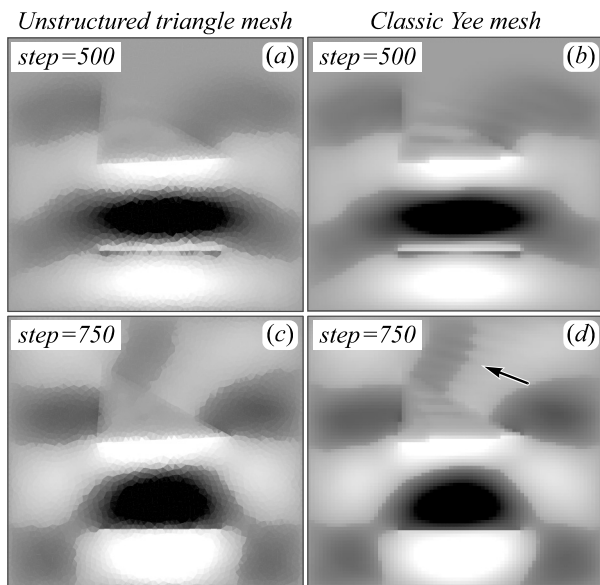


Fig. 3. Snapshots of magnetic part of  $p$ -polarized electromagnetic wave dynamics in the presence of plasma-like object (triangle in the center) calculated with unstructured grid – (a), (c) and with structured Yee-like grid (b), (d).

The source is located in the bottom and appears to be a radiating line. Each inset has step label indicating sequential index of iteration over time. Arrow in (d) inset shows a wavy field pattern not observed in (c)

### 2. Solution properties

In this section we will study how switching from classic Yee mesh [1] to triangle based unstructured mesh affects stability of solution and its dispersion properties. First we will discuss the dispersion properties for the discrete formulation of Maxwell PDEs on unstructured mesh. We will start from energy conservation law which is preserved for discrete approximation of operator  $\hat{\mathbf{M}}$ . Using this relation we will derive some properties of exact solution for discrete problem (i.e. with continuous time). Such an approach was also utilized in paper [14] where authors studied properties of similar operator  $\hat{\mathbf{M}}$  for orthogonal unstructured rectangular grid. It was shown that the operator preserves second order of approximation. Here we will focus mostly on dispersion properties of our spatial operator.

In the second part of the section we will discuss stability of leap-frog type of solution for Cauchy problem with finite time step  $dt$ .

#### Energy momentum

To calculate full electromagnetic energy  $W = \int (|E|^2 + |B|^2) dS$  one should choose the specific form of scalar product for vector  $\mathbf{V}$ . We will introduce the following scalar product:

$$\mathbf{V}^a \mathbf{V}^b = \sum_{i=1}^{n_e} E_i^a E_i^b (\mathbf{O}_{R(i)} \mathbf{O}_{L(i)}) |e_i| + \sum_{j=1}^{n_t} B_j^a B_j^b S_j, \quad (7)$$

where  $n_e$  and  $n_t$  are the number of edges and triangles correspondingly. From the chosen metrics it follows that electromagnetic energy momentum  $W$  can be calculated as:

$$W = \sum_{i=1}^{n_e} E_i^2 (\mathbf{O}_{R(i)} \mathbf{C}_i + \mathbf{O}_{L(i)} \mathbf{C}_i) |e_i| + \sum_{j=1}^{n_t} B_j^2 S_j. \quad (8)$$

Using equations (1, 2) for the first derivatives of  $E$  and  $B$  the first derivative of the above quantity can be rewritten as:

$$\begin{aligned} \frac{1}{2} \frac{\partial W}{\partial t} = & - \sum_{i=1}^{n_e} (B_{L(i)} - B_{R(i)}) E_i |e_i| + \\ & + \sum_{j=1}^{n_t} \sum_{q=1}^3 \text{sgn}(\mathbf{y}_0 (\overline{\mathbf{O}_j \mathbf{C}_{A(j,q)}} \times \mathbf{e}_{A(j,q)})) |e_{A(j,q)}| E_{A(j,q)} B_j. \end{aligned} \quad (9)$$

Last member of right hand side of the above equality can be rewritten in ‘per edge’ form instead of the above ‘per triangle’ notation. In such a new notation it is easier to find discrete analog of energy conservation law. We will also assume that electric field set for outer edge centers is zero which corresponds to ideal resonator case.

$$\begin{aligned} \frac{1}{2} \frac{\partial W}{\partial t} = & - \sum_{i=1}^{n_e} (B_{L(i)} - B_{R(i)}) E_i |e_i| - \\ & - \sum_{i=1}^{n_e} \left( \text{sgn}(\mathbf{y}_0 (\overline{\mathbf{O}_{R(i)} \mathbf{C}_i} \times \mathbf{e}_i)) B_{R(i)} + \right. \\ & \left. + \text{sgn}(\mathbf{y}_0 (\overline{\mathbf{O}_{L(i)} \mathbf{C}_i} \times \mathbf{e}_i)) B_{L(i)} \right) E_i |e_i| = \\ = & - \sum_{i=1}^{n_e} (B_{L(i)} - B_{R(i)}) E_i |e_i| - \\ & - \sum_{i=1}^{n_e} (B_{R(i)} - B_{L(i)}) E_i |e_i| = 0. \end{aligned} \quad (10)$$

More accurately the above equation can be rewritten in a divergent form for Poynting vector, but for the sake of simplicity we restricted ourselves with ideal resonator case with boundary conditions  $E_{\Sigma} = 0$ , where  $\Sigma$  is any boundary edge index.

From discrete energy conservation law(10) it follows that

- all eigenvalues  $\lambda$  of  $\hat{\mathbf{M}}$  operator are real;
- every two eigenvectors with different eigenvalues are orthogonal in the scope of metric (7).

To solve eigenvalues problem Galerkin method is used typically [8]. Here with the above conclusions we can use ‘complex time’ method to find eigenvectors and eigenvalues which does not require any zero approximation. The idea is based on the fact that if  $t$  is substituted by its new eigenvalues will become real, which does actually follow from energy conservation law (10). First eigenvector with maximal eigenvalue can be then found by updating new system with either Euler or leap-frog method. Being normalized at each step at some point any random initial vector will converge to first eigenvector since all other solutions will not survive due to lower eigenvalues. Next eigenvectors can be found using the fact that all eigenvectors with different eigenvalues are orthogonal in metric (7). So we can withdraw found eigenvector from any initial vector and start from it. By removing first eigenvector at each step we then will not allow this dominant eigenvector to rise from round-off errors, and finally will come to next eigen-

vector. Complete solution can be found in GitHub (see the link in the end of the paper).

In the Fig. 4 we plotted eigenvalues for all electromagnetic modes of rectangular resonator. The number of that eigenvalues equal to the number or cells in mesh or number of B vector components. Note that all eigenvalues has their copies with negative signe. We also have a number of zero eigenvalues describing eigen-subspace of static solutions. From Fig. 4 it is well seen that for lowest eigenvectors all three unstructured, structured and analytic solutions are quite close, and then analytic solution starts to show a difference while structured and unstructured ones remains quite close. This is a natural behavior for discrete problems related to so-called numerical dispersion. Then unstructured solution suddenly drops of the structured one. This happens due to completely different type of eigenvectors for structured and unstructured triangle meshes (compare colormap insets for mode number 68).

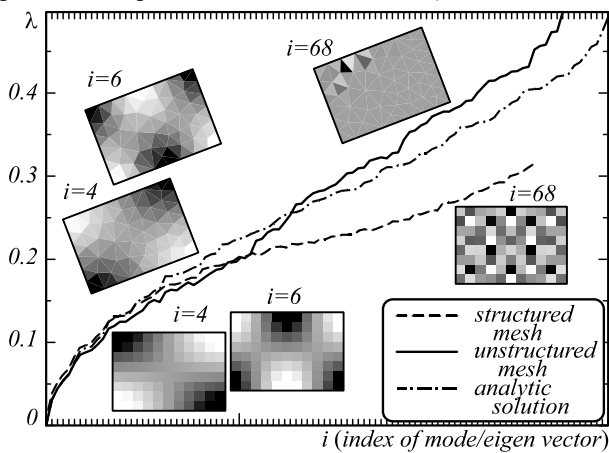


Fig. 4. Eigenvalues  $\lambda$  against mode (eigenvector) index  $i$  for rectangular resonator with dimensions  $70 \times 101$ . Comparison of structured quad mesh against unstructured triangle mesh and exact analytic solution (see markers in legend). Modes are shown as colormap insets plotted for magnetic field value. Tilted colormaps are for unstructured mesh, straight ones are for structured mesh; mode number is specified near each inset

The main conclusion here is that for eigenvectors that barely correspond to “physical” modes of ideal resonator the structured and unstructured discretization give very close results. We will get back to Fig. 4 right away after introducing a stability criterion for leap-frog scheme.

### Stability

To study stability of proposed numerical scheme we will follow Taflove [15] and investigate stability of solution for modes discussed in previous section. Briefly speaking we will generalize well known leap-frog method stability criteria for oscillatory motion [17]. First we will write the system of equations for  $i$ -th mode, having eigenvalue  $\lambda_i$ :

$$\frac{\partial \mathbf{V}_i}{\partial t} = \hat{\mathbf{M}} \mathbf{V}_i = \lambda_i \mathbf{V}_i, \quad (11)$$

$$\frac{\partial \mathbf{B}_i}{\partial t} = \hat{\mathbf{L}}_B \mathbf{B}_i = \lambda_i \mathbf{E}_i = i\omega_i \mathbf{E}_i, \quad (12)$$

$$\frac{\partial \mathbf{E}_i}{\partial t} = \hat{\mathbf{L}}_E \mathbf{E}_i = \lambda_i \mathbf{B}_i = i\omega_i \mathbf{B}_i. \quad (13)$$

Now by applying leap-frog method we will achieve the following (further  $i$  is omitted):

$$\begin{aligned} \mathbf{E}(t+dt) &= \mathbf{E}(t) + \hat{\mathbf{L}}_B \mathbf{B} \left( t + \frac{dt}{2} \right) dt = \\ &= \mathbf{E}(t) + i\omega \mathbf{E} \left( t + \frac{dt}{2} \right) dt, \end{aligned} \quad (14)$$

$$\begin{aligned} \mathbf{B} \left( t + \frac{3dt}{2} \right) &= \mathbf{B} \left( t + \frac{dt}{2} \right) + \hat{\mathbf{L}}_E \mathbf{E} (t+dt) dt = \\ &= \mathbf{B} \left( t + \frac{3dt}{2} \right) + i\omega \mathbf{B} (t+dt) dt. \end{aligned} \quad (15)$$

By substituting the equations to each other we are making sure that for both E and B sub-vectors we have the same equation. Now let us consider this equation for any vector  $\mathbf{X}$ :

$$\mathbf{X}(t+dt) = \mathbf{X}(t) + i\omega \mathbf{X} \left( t + \frac{dt}{2} \right) dt. \quad (16)$$

Let us study solution properties by substituting exponential solution  $\mathbf{A}_0 \exp(i\omega(t+dt))$ :

$$\begin{aligned} \mathbf{A}_0 \exp(i\omega(t+dt)) &= \mathbf{A}_0 \exp(i\omega t) + \\ &+ i\omega \mathbf{A}_0 \exp \left( i\omega \left( t + \frac{dt}{2} \right) \right) dt, \end{aligned} \quad (17)$$

$$\exp(i\omega dt) = 1 + i\omega \exp \left( \frac{i\omega dt}{2} \right) dt, \quad (18)$$

$$\exp \left( \frac{i\omega dt}{2} \right) = \frac{i\omega dt}{2} \pm \sqrt{1 - \frac{\omega^2 dt^2}{4}}. \quad (19)$$

Note that in above expression if the square root is real argument of exponent can be treated as pure complex number. So we come to stability criteria in the following form:

$$\omega \in \Re \quad \text{if} \quad \omega dt < 2. \quad (20)$$

Stability is achieved if for every mode the above criterion is fulfilled. So for whole scheme stability is achieved if  $dt < 2/\omega_{\max}$ , where  $\omega_{\max}$  is the cyclic frequency of highest mode. Obviously the criterion (20) is also true for structured mesh. Getting back to Fig. 4 it is well seen that unstructured mesh is a little more unstable than structured one and requites smaller time steps. But at the same time it has less electric field components: 3/2 per magnetic field component compared to 2 in case of quad structured mesh.

### 3. Mesh generation

In order to create a mesh for simulation one need to seed geometry i.e. add points to contours of boundary, objects and possibly some paths. Then using same seeding parameter one should add some volume(area in 2d) nodes to mesh. For this node cloud classical Delaunay method [19] can be used to make triangles covering the convex hull of node cloud.

The first problem of above scheme is that generated triangles can cross the geometry. As non-robust solution we use ‘flip’ refinement method that flips pair of triangles contacting the edge that crosses the geometry (see Fig. 5a). This method has a drawback when multiple adjacent triangles cross the geometry edge.

The next problem is seeding of volume. This is expected to create a uniform fill but having some predefined geometry we can produce very small triangles. According to our results from previous chapter this is not acceptable for leap-frog scheme. Each tiny triangle will produce a mode with very high oscillatory frequency forcing us to lower  $dt$  and loose performance. To overcome the above problem we can run an iterative process of mesh refinement which will shift nodes in a way to make nodes that are far from each other to become closer and vice versa. This can be done by solving a physical problem of nodes connected by springs along triangle edges (black springs in Fig. 5b) and along heights (gray springs in Fig. 5b). Both types of springs are needed to make triangle elastic enough since black springs only may allow for collapsing of the triangle. The initial spring length can be calculated in assumption that we magically generated mesh of perfect triangles covering whole volume. This will make uniform meshing that is needed in most cases for Maxwell equations simulations. If dense meshing is needed in some area springs length should become a field variable over space. Since we want a static solution we need to introduce quite big dumping to nodes motion. By neglecting second order derivatives over time we also can simplify the problem of relaxation. This will lead us to jello model which can be easily solved either with Euler method. After some iterations of the above jello model we need to remesh the node cloud because some triangles may become significantly deformed and cancel further relaxation in their neighboring area. Plus we need to anchor seeding nodes of input geometry.

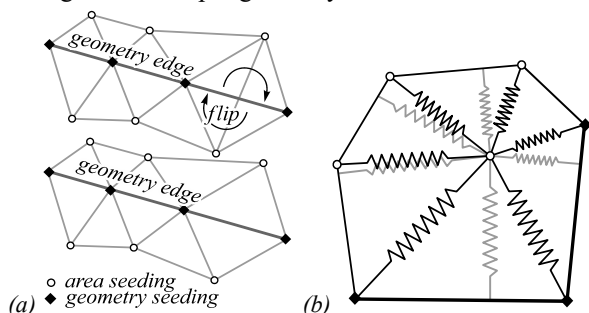


Fig. 5. Meshing algorithms: flip operation (a), jello model for area seeding (b)

Later on some triangles of convex hull can be removed to cover just needed area. We typically add bounding box around input geometry and then remove all elements that are out of input geometry. The above algorithm was implemented and can be found in GitHub (see the link in the end of article).

*Divide and conqueror*

It is well known that classic leap-frog plus Yee mesh approach is highly local so it can be easily divided among

calculating units and processed in parallel. The same is also true for an approach with unstructured meshes. The simplest idea of partitioning is to use KD tree idea. So having the depth of partitioning at each step one divide each part along x or y direction by two parts with same amount of elements. This works well for areas with rectangular boundary and produce partitioning with optimal shared boundary length. At the same time by the algorithm each part has nearly identical number of elements i.e. number of elements deviation is small.

We have to fulfill both deviation and shared boundary criteria. Since we cannot use queue due to data transfer overhead we need to balance the load for each compute unit. To reduce the amount of data shared between compute units at each step we need to minimize the size of shared boundary. In order to do that the bigger partitions should be created.

For tricky geometries ‘KD’ partitioning may cross boundary contour some bad way and produce uncontrollably small partitions. This can be solved with ‘greedy’ partitioning method. This method takes any element in the mesh and grows the area by adding neighboring elements at each step. When number of elements reaches desired number the algorithm switches to another partition by choosing another root node that was not ‘eaten’ before. It was found that such an algorithm performs best if each partition is started from the ‘sharpest’ node i.e. the node on existing boundary that has smallest number of neighbors.

After the partitioning is done with either KD or ‘greedy’ the result can be refined. We used two types of refinement:

- Join very small(smaller than threshold) parts with neighboring parts. We join to smallest part above with size the threshold.
- Join ‘hanging’ triangles to neighboring part. ‘Hanging’ are the triangles that contact own part by only one edge and with other two edges contact single neighboring part. Triangles contacting three parts are not refined.

In Fig. 6 we present partitioning result with all refinements are done. From the Fig. 6 it is seen that ‘KD’ produces more stable parts in terms of size  $<n_i>^p$  deviations. The shared boundary size  $l_{\Sigma}$  is nearly the same for both methods. We can conclude that ‘greedy’ is not so bad compared to KD, having certain important advantages over KD:

- ‘greedy’ not sensitive to non-trivial geometries
- KD produces  $2n$  parts, ‘greedy’ can produce any number of parts.

We hope that partitioning will allow using all the above methodology with any distributed system from classical clusters to GPU or Intel Phi clusters.

**Conclusion**

Using unstructured meshes seems to be quite interesting for modern physics applications dealing with sophisticated geometries like nano-structured surfaces and particle arrays. Numerical methods for unstructured meshes appear to be a natural generalization of methods used for

structured meshes, preserving their important features. Simulations with unstructured meshes seems to be easy adoptable for modern computational systems. In this article we studied 2d case for easiest linear system of Maxwell equations. Methods used in this work can be generalized to 3d tetrahedral meshes, while components of electromagnetic fields, currents and plasma density can be still bind to edges, faces and nodes correspondingly.

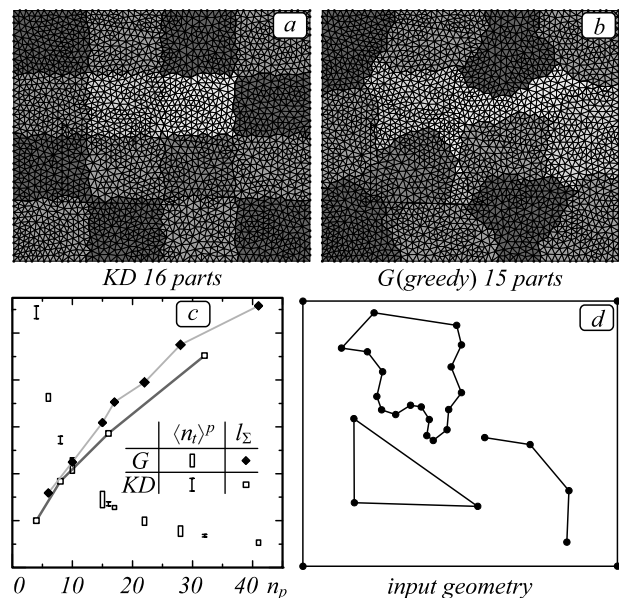


Fig. 6. Comparison of partitioning algorithms. Partitioning examples with nearly the same number of parts with KD (a) and ‘greedy’ (b); input geometry (d) and overall shared boundary length  $l_s$  i.e. number of shared edges and average number of triangles per part  $\langle n_i \rangle^p$  for KD and ‘greedy’ methods (error bars show root mean square deviation) with respect to number of partitions  $n_p$  (c)

In many practical applications on the boundary of mesh PML layers are adopted to simulate open boundary. The proposed method can work with mixed type of mesh e.g. triangular plus quad. So the meshes having rectangular boundaries can be tied with structured rectangular frame shaped mesh with PML layers.

All numerical codes used in this article are open source and located at GitHub [20].

### References

[1] Yee KS. Numerical solution of initial boundary value problems involving Maxwell's equations in isotropic media. IEEE Trans Antennas Propagat 1966; 14: 302.  
 [2] Kolesik M, Moloney JV. Nonlinear optical pulse propagation simulation: From Maxwell's to unidirectional equations. Phys Rev E 2004; 70: 036604.

[3] Skobelev SA, Kartashov DV, Kim AV. Few-optical-cycle solitons and pulse self-compression in a Kerr medium. Phys Rev Lett 2007; 99(20): 203902.  
 [4] Kosareva OG, Liu W, Panov NA, Bernhardt J, Ji Z, Sharifi M, Li R, Xu Z, Liu J, Wang Z, Ju J, Lu X, Jiang Y, Leng Y, Liang X, Kandidov VP, Chin SL. Can we reach very high intensity in air with femtosecond PW laser pulses? Laser Phys 2009, 19: 1776-1792.  
 [5] Dey I, Jana K, Fedorov VY, Koulouklidis AD, Mondal A, Shaikh M, Sarkar D, Lad AD, Tzortzakis S, Couairon A, and Kumar GR. Highly efficient broadband terahertz generation from ultrashort laser filamentation in liquids. Nature Communications 2017; 8: 1184.  
 [6] Déchard J, Debayle A, Davoine X, Gremillet L, Bergé L. Terahertz pulse generation in underdense relativistic plasmas: From photoionization-induced radiation to coherent transition radiation. Phys Rev Lett 2018; 120: 144801.  
 [7] Fadeev DA, Oladyshkin IV, Mironov VA. Terahertz emission from metal nanoparticle array. Opt Lett 2018; 43: 1939-1942.  
 [8] Nesterenko DV, Kotlyar VV. Hybrid finite element method and boundary element method for analysis of light diffraction on diffraction gratings [In Russian]. Computer Optics 2008; 32(3): 238-245.  
 [9] Sprangle P, Peñano JR, Hafizi B, Kapetanakis CA. Ultrashort laser pulses and electromagnetic pulse generation in air and on dielectric surfaces. Phys Rev E 2004; 69: 066415.  
 [10] Leuchtman P, Fumeaux C, Baumann D. Comparison of errors and stability in FDTD and FVTD. Advances in Radio Science 2003; 1: 87-92.  
 [11] Fumeaux C, Baumann D, Sankaran K, Krohne K, Vahldieck R, Li E. The finite-volume time-domain method for 3D solutions of Maxwell's equations in complex geometries: a review. Proceedings of the European Microwave Association 2007; 3: 136-146.  
 [12] Gansen A, Hachemi ME, Belouettar S, Hassan O, Morgan K. EM modelling of arbitrary shaped anisotropic dielectric objects using an efficient 3D leapfrog scheme on unstructured meshes. Comput Mech 2016; 58: 441-455.  
 [13] Keranen J, Kanagas J, Ahola A, Kettunen L. Implicit Yee-like scheme on tetrahedral mesh. IEEE Transactions on magnetics 2002; 38(2): 717-720.  
 [14] Monk P, Suli E. Error estimates for Yee's method on non-uniform grids. IEEE Transactions on magnetics. – 1994. – Vol. 30, Issue 5. – P. 3200-3203.  
 [15] Taflove A, Hagness S. Computational electrodynamics: The finite-difference time-domain method. 3rd ed. Boston, London: Artech House Publishers; 2005.  
 [16] Aurenhammer F, Klein R, Lee D-T. Voronoi diagrams and Delaunay triangulations. World Scientific; 2013. ISBN: 978-981-4447-63-8.  
 [17] Birdsall CK, Langdon AB. Plasma physics via computer simulations. McGraw-Hill Book Company; 1985: 56.  
 [18] Dynamics of EMP in the presence of metallic particle. Source: (<https://www.youtube.com/watch?v=SryTESknKDE>).  
 [19] Delaunay B. Sur la sphère vide. A la mémoire de Georges Voronoi, Bulletin de l'Académie des Sciences de l'URSS, Classe des sciences mathématiques et naturelles 1934; 6: 793-800.  
 [20] Why GitHub? Source code. Source: (<https://github.com/dafadey/hed.git>).

### Author's information

**Daniil Aleksandrovich Fadeev** (b. 1984) graduated from Lobachevsky State University of Nizhny Novgorod in 2008 (High School of General and Applied Physics sub-department). Currently works as junior scientist in Institute of Applied Physics of Russian Academy of Sciences (short IAP RAS). Scientific interests are numerical simulation, optimization of numerical codes for graphical processing units and systems with non-uniform memory access, simulation of plasma electrodynamics, algorithms generating meshes for calculations, imaging methods for electrodynamic characteristics of flaky media using reflected electromagnetic pulse waveform.

Received November 22, 2018. The final version – May 13, 2019.