Current status of the development of the problem-oriented software package GYREOSS

M Damyanova[1,3], S Sabchevski[1], I Zhelyazkov[2], E Vasileva[1], E. Balabanova[1], P Dankov[2] and P Malinov[2]

[1]Acad. E. Djakov Institute of Electronics, Bulgarian Academy of Sciences,
Association EURATOM-INRNE, 72 Tsarigradsko Chaussee, 1784 Sofia, Bulgaria
[2]Faculty of Physics, St. Kl. Ohridski University of Sofia,
Association EURATOM-INRNE, 5 James Bourchier Blvd., 1164 Sofia, Bulgaria

E-mail: milena@ie.bas.bg

Abstract. A great number of problem-oriented software packages (e.g., DAPHNE, ESRAY, GYROSIM, just to name a few) are used for computer-aided design (CAD) and optimization of high-power gyrotrons for fusion research (most notably for ECRH and ECCD). In them, adequate self-consistent physical models formulated in a two-dimensional space (2D) are implemented. Most of the problems encountered in recent years in the development of megawatt-class gyrotrons, however, can be attributed to physical factors and phenomena that are both non-stationary (for instance various electron beam instabilities, dynamics of the trapped particles) and inherently three-dimensional by nature (e.g., non-uniformity of the emission, misalignment of the electrodes and magnetic coils, etc.). In order to address these problems and to take into account most of the physical factors that have so far been neglected, a work on the development of a novel problem-oriented software package called GYREOSS (which stands for Gyrotron Electron-Optical System Simulation) was initiated recently. Since then, GYREOSS has evolved into a test bed for experimenting with different numerical methods, algorithms, and programming techniques for ray tracing (trajectory analysis) and particle-in-cell (PIC) simulations. In this paper, we outline the current status of GYREOSS and present both its physical model formulated in 2D and 3D and its program realization. We illustrate the capabilities of the latest version of the code, which includes a novel 3D field solver and an efficient relativistic particle pusher. Finally, we present an outlook for a further development of GYREOSS.

1. Introduction

The gyrotrons are fast-wave vacuum tubes operating on a physical principle known as electron-cyclotron maser instability, which is caused by the dependence of the relativistic cyclotron frequency of accelerated electrons gyrating in a strong magnetic field on their energy. The interaction of a helical beam in which most of the kinetic energy of its electrons is associated with their rotational motion (gyro-motion) with the transverse components of the electric field of an electromagnetic wave having a phase velocity greater than the speed of light in vacuum takes place in a resonant structure (cavity resonator) with characteristic dimensions that are much larger than the wavelength of the radiation.

[3] To whom any correspondence should be addressed.

Such operational principle, which lacks the disadvantages and the limitations of the conventional (slow-wave) vacuum tubes makes the gyrotrons the most powerful sources of coherent radiation at millimetre and sub-millimetre wavelengths (sub-THz and THz frequencies) in a CW mode. Definitely, among numerous applications of gyrotrons in the physical research and the technologies, the most prominent one is their use as sources of high-power radiation for electron cyclotron resonant heating (ECRH) and electron cyclotron current drive (ECCD) of magnetically confined plasmas in fusion devices (e.g., tokamaks and stellarators). The current state-of-the-art of these devices is well represented by a series of megawatt-class tubes in the frequency range 140 – 170 GHz [1]. For their computer-aided design (CAD), optimization and analysis, many problem-oriented software packages are available that are specialized to one or another subsystem of the tube, e.g., EOS (DAPHNE, ESRAR, GUN-MIG/CUSP), electrodynamical system (CAVITY, SELFT) or the whole device (GYROSIM). In all of them, adequate physical models formulated in a two–dimensional (2D) space are realized. Many physical factors and phenomena (for instance non-uniformity of the emission, misalignment of the electrodes and magnetic coils, various instabilities, etc.), however, are inherently three-dimensional (3D) and non-stationary by nature and are therefore neglected in the 2D simulations. In order to take them into account, the development of GYREOSS package (which stands for Gyrotron Electron-Optical System Simulation) was initiated recently. It follows a concept presented elsewhere [2, 3] and is based on a physical model formulated in both 2D and 3D and is being developed as a problem-oriented platform for particle in cell (PIC) simulations and trajectory analysis (ray tracing). In this paper we present both the current status of GYREOSS and an outlook for its further development.

2. Novel field solver and relativistic particle pusher of the GYREOSS package

Starting from an initial version of the package [4], which was based on the *gmsh* (a 3D mesh generator with a build-in CAD system with pre-, post-processing, and visualization capabilities [5]) and GetDP (a General Environment for Treatment of Discrete Problems [6]) in recent years GYREOSS has evolved into a test bed for experimenting with different numerical methods, algorithms, and programming techniques for ray tracing (trajectory analysis) and particle-in-cell (PIC) simulations. Its current version is implemented as a library of scripts using FreeFEM++ [7] and its integrated development environment (IDE) FreeFEM++-cs. The most characteristic for the current status of GYREOSS are two of its recently developed computational modules, namely a novel 3D (2D) field solver and a relativistic particle pusher. They will be presented briefly below.

The distribution of the electrostatic potential u in the electron-optical system (EOS) is governed by the Poisson's equation and the electric field $\mathbf{E} = -\nabla u$ is a gradient of that potential. More specifically, we consider a bounded computational domain $\Omega \subset \mathrm{R}^d$ of dimension $d = 3$ (3D) or $d = 2$ (2D) with a boundary $\partial\Omega = \bigcup_{i=1}^{N} \partial\Omega_i$, which is composed of N surfaces $\partial\Omega_i$ in the case of 3D or N contours in 2D. The electrostatic potential is a solution of the following boundary value problem for the Poisson equation

$$\Delta u = -f \text{ in } \Omega, \tag{1}$$

with mixed boundary conditions, i.e., Dirichlet boundary conditions (DBCs) on the electrodes (designated by the indices l =1,…, L)

$$u = U_l \text{ on } \partial\Omega_{Dl}, \tag{2}$$

and Neumann boundary conditions (NBCs)

$$\frac{\partial u}{\partial n} = 0 \text{ on } \partial\Omega_{Nm}, \tag{3}$$

on the appropriately specified closing boundaries (e.g., symmetry planes), designated by the indices m = 1,…, M. The regions with Dirichlet $\partial\Omega_D$ and Neumann $\partial\Omega_N$ boundary conditions do not overlap, i.e., $\partial\Omega = \partial\Omega_D \bigcup \partial\Omega_N$ and $\partial\Omega_D \bigcap \partial\Omega_N = \varnothing$, where $\partial\Omega_D = \bigcup_{l=1}^{L} \partial\Omega_{Dl}$, $\partial\Omega_N = \bigcup_{m=1}^{M} \partial\Omega_{Dm}$ and L+M = N. It is well known that with such mixed boundary conditions the Poisson equation has a unique solution. Here the right-hand side of the Poisson equation $f = \rho / \varepsilon_0$ is defined by the space charge distribution ρ and the permittivity of vacuum ε_0.

For the solution of the above boundary value problem the finite element method (FEM) and the tools of the FreeFEM++ software package are used. The corresponding weak variation form of the Poisson equation reads: Find $u \in H^1$ such that

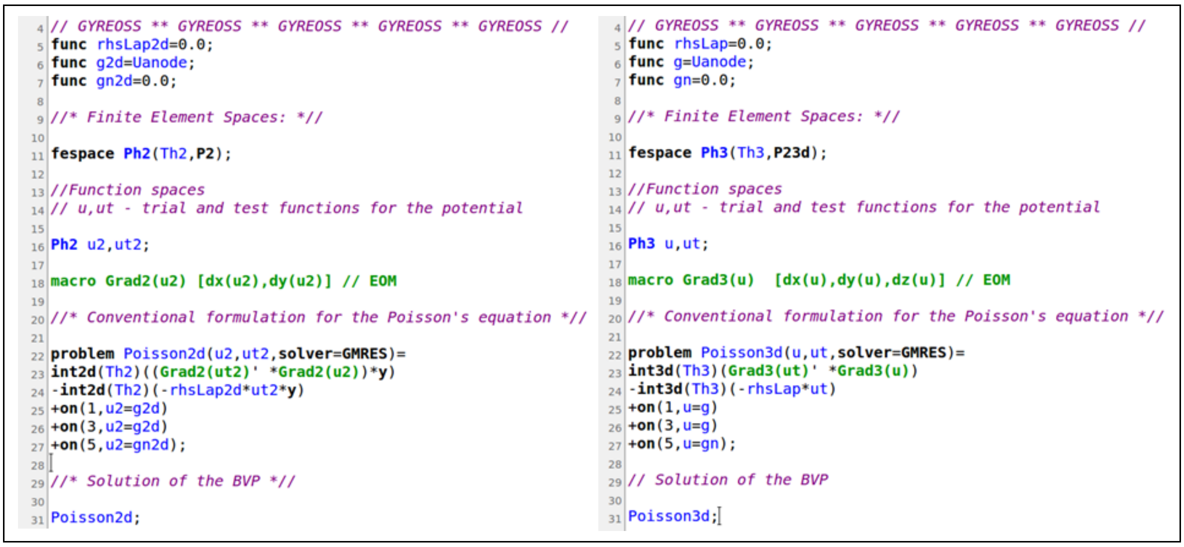
$$\int_{\Omega} \nabla u . \nabla u_\tau = \int_{\Omega} f u_\tau \, , \tag{4}$$

for all $u_\tau \in H_0^1$, where H^1 is the Hilbert space and H_0^1 consists of continuously differentiable functions u_τ in Ω and vanishing on the boundary. Here and below we use the traditional notations for the Sobolev function spaces $H^m(\Omega)$ that consist of all functions on Ω that are square integrable together with their derivatives up to the order m, i.e.

$$H^m(\Omega) = \left\{ u \in L_2(\Omega) : \frac{\partial^j u}{dx^j} \in L_2(\Omega), 1 \le j \le m \right\}, \tag{5}$$

where $L_2(\Omega)$ denotes the space of square integrable functions on Ω. In this paper, for convenience we denote both the unknown (trail) functions and the test functions by one and the same letter but mark the latter functions by a subscript τ .

The above formulation of the boundary value problem of the field solver is implemented using the finite elements and the tools provided by the FreeFEM++ [7] package for solving partial differential equations by the FEM. Two pieces of code with its program realization for the 2D and 3D cases respectively are shown in figure 1.



```
// GYREOSS ** GYREOSS ** GYREOSS ** GYREOSS ** GYREOSS //
func rhsLap2d=0.0;
func g2d=Uanode;
func gn2d=0.0;

//* Finite Element Spaces: *//

fespace Ph2(Th2,P2);

//Function spaces
// u,ut - trial and test functions for the potential

Ph2 u2,ut2;

macro Grad2(u2) [dx(u2),dy(u2)] // EOM

//* Conventional formulation for the Poisson's equation *//

problem Poisson2d(u2,ut2,solver=GMRES)=
int2d(Th2)((Grad2(ut2)' *Grad2(u2))*y)
-int2d(Th2)(-rhsLap2d*ut2*y)
+on(1,u2=g2d)
+on(3,u2=g2d)
+on(5,u2=gn2d);

//* Solution of the BVP *//

Poisson2d;
```

```
// GYREOSS ** GYREOSS ** GYREOSS ** GYREOSS ** GYREOSS //
func rhsLap=0.0;
func g=Uanode;
func gn=0.0;

//* Finite Element Spaces: *//

fespace Ph3(Th3,P23d);

//Function spaces
// u,ut - trial and test functions for the potential

Ph3 u,ut;

macro Grad3(u)  [dx(u),dy(u),dz(u)] // EOM

//* Conventional formulation for the Poisson's equation *//

problem Poisson3d(u,ut,solver=GMRES)=
int3d(Th3)(Grad3(ut)' *Grad3(u))
-int3d(Th3)(-rhsLap*ut)
+on(1,u=g)
+on(3,u=g)
+on(5,u=gn);

// Solution of the BVP

Poisson3d;
```

Figure 1. Excerpts from the codes of the field solvers in 2D (left) and 3D (right).

The computational domain Ω is discretized by a mesh T_h consisting of cells K, which are triangles in 2D and tetrahedrons in 3D. Both externally generated meshes (using gmsh and TetGen) as in the previous version of GYREOSS and grids generated by the embedded mesh tools of FreeFEM++.

An example which shows the surface meshes that represent the electrodes of the gun and the closing boundaries of the computational domain as well as the 3D tetrahedral mesh used in the simulations of a coaxial EOS are presented in figure 2.

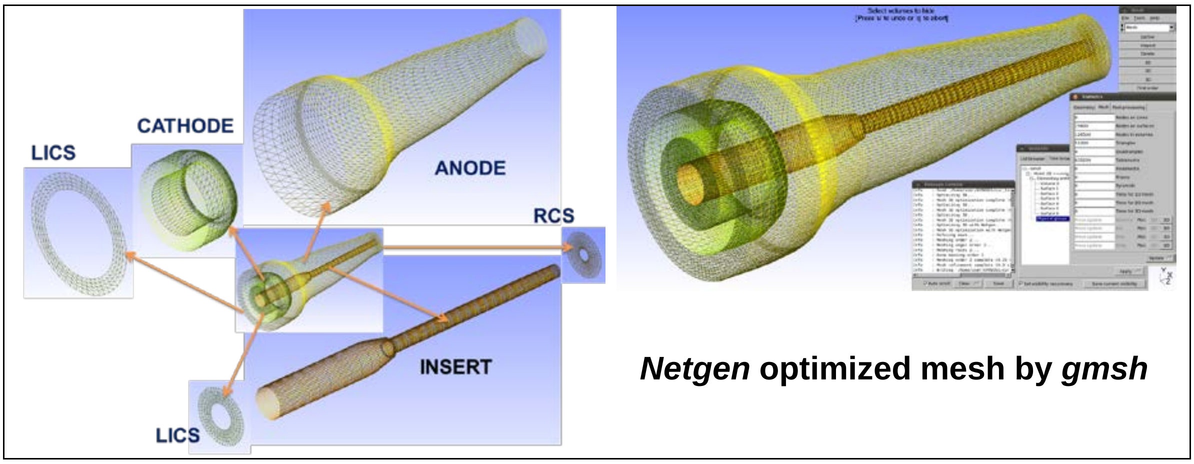


Figure 2. Surface meshes forming the boundary of the computational domain (left) and the 3D tetrahedral mesh (right) of a coaxial EOS for a 2 MW/170 GHz gyrotron.

From a physical point of view, the DBC correspond to the potentials applied to the electrodes that form the EOS and are also known as geometric boundary conditions. The electrodes are considered as perfectly conducting equipotential surfaces, where the tangential component of the electric field vanishes. The NBC are usually specified on the closing contours (surfaces) between the electrodes, and the condition $\frac{\partial u}{\partial n}=0$ implies simply that the equipotential lines cross these boundaries at a right angle. It should be noted that the physical boundary conditions (DBCs or NBCs) are treaded by the finite element method as either *natural* or *essential*. For example, the DBCs are essential boundary conditions (EBCs) while the NBCs are natural boundary conditions. The main difference between them is the following. The natural BCs are already included in the integration in the finite element equations and there is no need to impose them additionally. On the other hand, although the EBCs are satisfied automatically in the finite-element formulation, they still need to be implemented (using one or another method, e.g., direct application; Lagrangian multiplier or Penalty Method). In another words, the essential boundary conditions are built into the solution space, while the natural boundary conditions are built into the weak form.

The particle in cell (PIC) method implies representing the electron beam by an ensemble of N macro particles and tracing their motion in a self-consistent electromagnetic field whose electric $\mathbf{E}$ and magnetic $\mathbf{B}_i$ components include not only the applied external fields but the self-fields of the space charge as well [8, 9]. According to the PIC algorithm, this is being performed integrating numerically the following relativistic Newton–Lorentz equations of motion

$$\frac{\mathrm{d}\mathbf{w}_i}{\mathrm{d}t}=\frac{q}{m_0}\left(\mathbf{E}_i+\frac{\mathbf{w}_i}{\gamma_i}\times\mathbf{B}_i\right),\ \frac{\mathrm{d}\mathbf{r}_i}{\mathrm{d}t}=\mathbf{v}_i,\quad \mathbf{w}_i=\gamma_i\mathbf{v}_i \tag{6}$$

for all electrons (i = 1, 2,…, N), where q/m_0 is the charge to mass ratio of an electron at rest, $\mathbf{E}_i$, $\mathbf{B}_i$ being the electric and the magnetic field at the current particle position $\mathbf{r}_i$, and the relativistic Lorentz factor of the i-th particle is defined by its velocity $\mathbf{v}_i$ and the speed of light in vacuum c as $\gamma=[1-\frac{|\mathbf{v}|^2}{c^2}]^{-1/2}=\sqrt{1+\frac{\mathbf{w}^2}{c^2}}$. Among the numerous explicit and implicit numerical methods for integration of the equations of motion (e.g., schemes of Boris, Boris–Bunneman, Tajima, Runge–Kutta method, Verlet's method, and predictor–corrector method) evaluating all important factors (accuracy, stability, CPU time etc.) finally we have selected the leapfrog method of the relativistic Boris–Bunneman scheme. The leap-frog method [8, 9] reduces the relativistic equations of motion (6) to the following time–centered implicit relations

$$\frac{\mathbf{w}_i^{n+1/2}-\mathbf{w}_i^{n-1/2}}{\Delta t}=\frac{q}{m_0}\left(\mathbf{E}^n+\frac{\mathbf{w}_i^{n+1/2}-\mathbf{w}_i^{n-1/2}}{2\gamma_i^n}\right);\quad \frac{\mathbf{r}_i^{n+1}-\mathbf{r}_i^{n}}{\Delta t}=\mathbf{v}_i^{n+1/2}=\frac{\mathbf{w}_i^{n+1/2}}{\gamma_i^{n+1/2}},\qquad (7)$$

where the advancement in time $t^n = n\Delta t$ is performed with a step Δt. Integration of (7) is performed following the classical 3-step method by Boris [10], which involves a complete separation of the fields $\mathbf{E}_i$, $\mathbf{B}_i$ (introducing two intermediate velocities and using the scalings $\mathbf{E}\leftarrow\frac{q\Delta t}{m}\mathbf{E}$, $\mathbf{B}\leftarrow\frac{q\Delta t}{m}\mathbf{B}$), followed by a half-step electrical acceleration ($\mathbf{w}\leftarrow\mathbf{w}+\mathbf{E}$), then a Larmor rotation (gyromotion) of the velocity vector ($\mathbf{w}\leftarrow\frac{2}{1+\mathbf{B}^2}(\mathbf{w}+\mathbf{w}\times\mathbf{B})\times\mathbf{B}$) and, finally, another half-step acceleration ($\mathbf{w}\leftarrow\mathbf{w}+\mathbf{E}$). This stepping scheme introduces an error which is second order with respect to the ratio of the time step to the cyclotron period. By satisfying the condition $\omega_c\Delta t<0.35$ (ω_c being the electron cyclotron frequency) the relative error of the rotation angle of the Boris scheme is kept to values less than 1%. This stepping technique has two important physical, namely it is time-reversible and energy-conserving. Since the leap-frog method is not self-starting initially a half step backwards from the initial conditions for the particles is performed in order to kickstart the process. The method described above is coded in the novel relativistic particle pusher (aka mover), which is used to update (advance) the positions and the velocities of the traced particles on each time step of the PIC algorithm as well as for calculation of the electron trajectories if the code is used as a ray-tracing tool for trajectory (electron–optical) analysis. As an example, figure 3 shows the electron trajectories in the coaxial gyrotron, which was presented in figure 2.

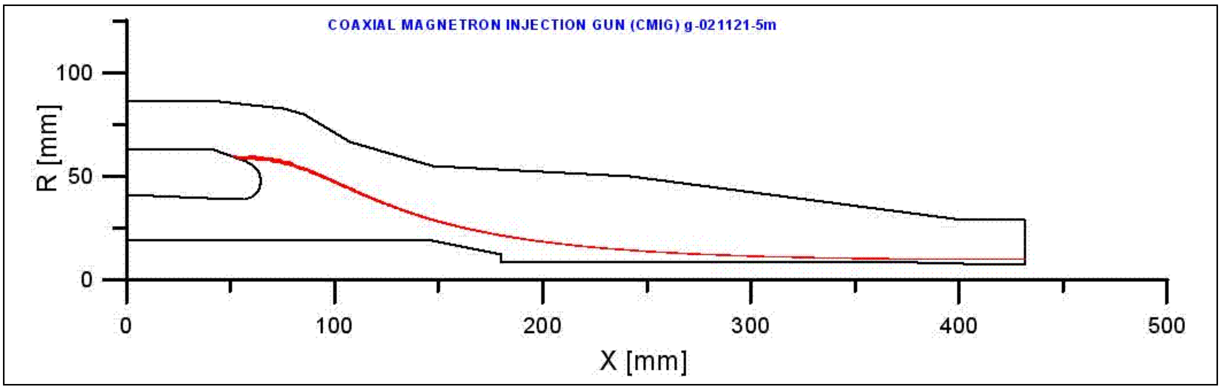


Figure 3. Electron trajectories in the coaxial gyrotron shown in figure 2.

3. Conclusions and outlook

The problem-oriented software package GYREOSS is being developed as a versatile tool for numerical studies and CAD of EOS of gyrotrons. Its development follows a well-defined concept formulated earlier, which sets the following guiding requirements to the code: (i) based on adequate (self-consistent) physical models; (ii) utilizing state-of-the-art computational platforms, numerical libraries and programing techniques; (iii) extensibility; (iv) portability; (v) efficiency. This paper presents a novel electrostatic field solver of the GYREOSS software package. It is parameterized in such a way as to provide convenient data structures of the electromagnetic fields at the current particle positions in both 2D and 3D which are used by the newly developed second-order relativistic particle pusher in which the leap-frog method and the Boris–Buneman scheme are implemented. Although the current version is sequential, both the data structures and the algorithms are coded in a way that would allow an easy parallelization of the code in the future. Currently we are studying and comparing various schemes for allocation of the currents and charges to the computational grid. The aim is to develop efficient modules that not only realize the above mentioned step ("scattering") of the PIC algorithm but also take full advantage of both the advanced interpolator and the mesh adaptation (adaptive mesh refinement) provided by FreeFEM++.

Acknowledgements

This work was carried out in the framework of Task 2.1.2 of the scientific program of the EURATOM-INRNE Association.

References

[1] Thumm M 2013 State-of-the-art of high power gyro-devices and free electron masers 2012 *KIT Sci. Reports* 7641 (KIT Scientific Publishing, 2013) 1–144

[2] Sabchevski S, Zhelyazkov I, Thumm M, Piosczyk B, Tran T M, Hogge J-Ph and Pagonakis J Gr 2007 Recent evolution of the simulation tools for computer aided design of electron-optical systems for powerful gyrotrons *Computer Modelling in Engin. & Sci.* **20** 203–20

[3] Sabchevski S, Illy S, Piosczyk B, Borie E and Zhelyazkov I 2008 Towards the formulation of a realistic 3D model for simulation of magnetron injection guns for gyrotrons (A Preliminary study) *Wissenschaftliche Berichte* FZKA 7409 1–34

[4] Damyanova M, Sabchevski S and Zhelyazkov I 2010 Pre- and post-processing of data for simulation of gyrotrons by the GYREOSS software package, *J. Phys.: Conf. Series* **207** 012032

[5] Geuzaine Ch and Remacle J-F 2013 *Gmsh: a three-dimensional finite element mesh generator with built-in pre- and post-processing facilities* (version 2.8.2 July 16 2013) http://geuz.org/gmsh/

[6] Dular P and Geuzaine Ch 2013 *GetDP: a general environment for the treatment of discrete problems* (version 2.4.1 July 16 2013) http://geuz.org/getdp/

[7] Hecht F 2013 *FreeFem++* 3rd edition version 3.23 (Laboratorie Jacques-Louis Lions, Université Pierre et Marie Curie, Paris) 1–392 http://www.freefem.org/ff++/ftp/freefem++doc.pdf

[8] Hockney R W and Eastwood J W 1981 *Computer simulation using particles* (McGraw-Hill New York)

[9] Birdsall C and Langdon A 2004 *Plasma physics via computer simulation* (Series in Plasma Phys.) (Taylor & Francis London)

[10] Buneman O 1976 The advance from 2D electrostatic to 3D electromagnetic particle simulation *Computer Phys. Communic.* **12** 21–31