Proceedings of the 5th WSEAS International Conference on Applications of Electrical Engineering, Prague, Czech Republic, March 12-14, 2006 (pp131-139)

3D Simulations with Fields and Particles

BRANISLAV RADJENOVIĆ[1], MARIJA RADMILOVIĆ-RADJENOVIĆ[2], PETAR BELIČEV[1]
[1]Laboratory of Physics, Vinča Institute of Nuclear Sciences
P. O. Box 522, 11001 Belgrade
SERBIA AND MONTENEGRO
[2]Institute of Physics
Pregrevica 118, 11080 Belgrade
SERBIA AND MONTENEGRO

Abstract: - In this paper we present a methodology and some of the results for 3D simulations that includes both field and particle abstractions. Electromagnetic field calculations used here are based on the discrete differential form representation of the finite elements method, while the Monte Carlo method makes foundation of the particle part of the simulations. The first example is the simulation of the feature profile evolution during SiO_2 etching in flourocarbon plasma based on the sparse field method for solving level set equations. Etching velocities on the profile surface are determined using Monte Carlo method for the particle flux calculations. The model takes into account pure chemical and ion-enhanced chemical etching mechanisms. Another example is connected with the design of a spiral inflector which is one of the key devices of the axial injection system of the VINCY Cyclotron.

Key-Words: - fields, particles, simulations, finite elements, accelerator, inflector, plasma etching, profile evolution, level set method

1 Introduction

Fields and *particles* are two the most important modeling paradigms in pure and applied science at all. Particles are primarily used in one of the two ways in large scientific applications. The first one is to track sample particles to gather statistics that describe the conditions of a complex physical system. Particles of this kind are often referred to as "tracers". The second one is to perform direct numerical simulation of systems that contain discrete point-like entities such as ions or molecules. In both scenarios, the application contains one or more sets of particles. Each set has some data associated with it that describes its members' characteristics, such as charge, mass or momentum. The state of the physical system is defined by these data.

Particles typically exist in a spatial domain, and they may interact directly with one another or with field quantities defined on that domain. A field, on the other hand, defines a set of values on a region of space. In order to specify a field one must specify the locations at which the field's values are defined, and describe what happens at the boundaries of that region in space. For this purpose we use *meshes* which are discrete representations of the physical domains.

Particle abstractions are usually designed to be used in conjunction with fields. Some types of *interpolators* are used as the glue that bind these together, by specifying how to calculate field values at particle (or other) locations that not happen to lie exactly on mesh points. Interpolators are used to gather values to specific positions in a field's spatial domain from nearby field elements, or to scatter values from such positions into the field. An example of using this kind of interpolation is particle-in-cell (PIC) simulations [1], in which charged particles move through a mesh. The particle interactions are determined by scattering the particle charge density into a field, solving for the self-consistent field, and gathering that field back to the particle positions.

Here we present two examples of the simulations of this type, different in their origin but sharing the same software design concept and implementation components. The first is from the area of the applications of the plasma technologies in microelectronic circuits manufacturing. The second is a part of the ongoing work on the design of the spiral inflector for the VINCY Cyclotron.

2 Software Design Issues

High-performance scientific applications are notoriously difficult and expensive to develop and maintain. Very little progress had been made in scientific programming productivity over decades. On the other hand, in the business programming community giant steps have been made in productivity, through software technology including component-based programming and frameworks. Attempts to transfer this technology to the scientific community have largely been unsuccessful until recently. However, recent advances in software technology signify that the hard to get goal of increased high-performance scientific programming productivity is possible.

Although Object-Oriented Programming itself was not the main reason for the improved productivity the business community (the rise in productivity comes from the use of Object-Oriented Design and Component-Based Programming), the choice of the programming language is of the utmost importance. Development of complex scientific applications requires GUI toolkits, numeric libraries, graphical facilities, libraries for interfacing with code written in other languages, etc. Many scientific programmers and researchers still dream of a general-purpose language that is so expressive, elegant, and efficient that it can cover all needs without significant inconveniences. However, no current language can satisfy all these requirements natively, and there is no a research language that could possibly do that in the short-to-medium term.

C++ is the most popular high-performance, object-oriented programming language in use today. C++ directly supports many programming styles. In this, C++ deliberately differs from other languages. The support of multiple styles (traditional C-style, concrete classes, abstract classes, traditional class hierarchies, abstract classes and class hierarchies, and generic programming) is one of its major strengths. This quality makes C++ the most appropriate (in the context of scientific and engineering applications) programming language for integrating all the necessary components originating from very diverse sources. Our simulation framework (written in C++) is based on Fltk library [2] for building GUI elements, Vtk toolkit [3] for 3D visualization needs, and Pthreads library [4] for managing multiple execution threads. Additional libraries and toolkits are used depending on the specific requirements of the particular applications.

3 Electromagnetic Field Calculations

After the publication of Maxwell's treatise, electromagnetic laws were commonly written in differential formulation. From that moment, electromagnetic field equations were identified with the "Maxwell equations", i.e. with partial differential equations. Like most equations of physics, Maxwell's equations are extremely rich in symmetries and (hence) conservation laws. In the continuum, many conservation laws follow directly from invariances of the Lagrangian (Noether symmetries) such as energy or momentum conservation, while others have an inherent topological aspect, such as magnetic charge. When Maxwell's equations are discretized on a mesh, a number of symmetries of the continuum theory are modified or broken. However, some conservation laws may be preserved on a discrete setting. This is because they often relate a quantity on certain region of space to an associated quantity on the boundary of the region. Because the boundary is a topological invariant, such conservation laws should not depend on the metric of the space. A natural mathematical language that explore this aspect is the calculus of exterior differential forms and associated algebraic topological structures [5, 6].

In this approach the scalar electrostatic potential is a 0-form, the electric and magnetic fields are 1-forms, the electric and magnetic fluxes are 2-forms, and the scalar charge density is a 3-form. The basic operators are the exterior (or wedge) product, the exterior derivative, and the Hodge star. Precise rules (i.e. a calculus) prescribe how these forms and operators can be combined. In this modern geometrical approach to electromagnetics the fundamental conservation laws are not obscured by the details of coordinate system dependent notation. By working within the discrete differential forms framework, we are gauranteed that resulting spatial discretization schemes are fully mimetic.

The calculations of the electric fields in this paper are performed by integrating a general finite element solver GetDP [7, 8] in our simulation framework. GetDP is a thorough implementation of discrete differential forms calculus, and uses mixed finite elements to discretize de Rham-type complexes in one, two and three dimensions. Meshing of the computational domain is carried out by TetGen tetrahedral mesh generator [9]. TetGen generates the boundary constrained high quality (Delaunay) meshes, suitable for numerical simulation using finite element and finite volume methods. As a part of the post-processing procedure the electric fields, obtained on the unstructured meshes, are recalculated on the Cartesian rectangular domains containing the regions of the particles movement. In this manner, the electric field on the particles could be calculated by simple trilinear interpolation.

4 Etching Profile Evolution Simulation

Refined control of etched profile in microelectronic devices during plasma etching process is one of the most important tasks of front-end and back-end microelectronic devices manufacturing technologies. The profile surface evolution in plasma etching, deposition and lithography development is a significant challenge for numerical methods for interface tracking itself. Level set methods for evolving interfaces [10, 11] are specially designed for profiles which can develop sharp corners, change topology and undergo orders of magnitude changes in speed. They are based on a Hamilton-Jacobi type equation [12] for a level set function using techniques developed for solving hyperbolic partial differential equations.

A simple model of Ar^+/F etching process that includes only pure chemical and ion-enhanced chemical etching mechanisms is presented in details.

4.1 Sparse field level set method

The basic idea behind the level set method is to represent the surface in question at a certain time t as the zero level set (with respect to the space variables) of a certain function $\varphi(t, \mathbf{x})$, the so called level set function. The initial surface is given by $\{\mathbf{x} \mid \varphi(0, \mathbf{x}) = 0\}$. The evolution of the surface in time is caused by "forces" or fluxes of particles reaching the surface in the case of the etching process. The velocity of the point on the surface normal to the surface will be denoted by $V(t, \mathbf{x})$, and is called velocity function. For the points on the surface this function is determined by physical models of the ongoing processes; in the case of etching by the fluxes of incident particles and subsequent surface reactions. The velocity function generally depends on the time and space variables and we assume that it is defined on the whole simulation domain. At a later time $t > 0$, the surface is as well the zero level set of the function $\varphi(t, \mathbf{x})$, namely it can be defined as a set of points $\{\mathbf{x}\in\Re^n \mid \varphi(t, \mathbf{x}) = 0\}$. This leads to the level set equation

$$\frac{\partial\varphi}{\partial t} + V(t,\mathbf{x})\left|\nabla\varphi\right| = 0, \qquad (1)$$

in the unknown function $\varphi(t, \mathbf{x})$, where $\varphi(0, \mathbf{x}) = 0$ determines the initial surface. Having solved this equation the zero level set of the solution is the sought surface at all later times. Actually, this equation relates the time change to the gradient via the velocity function. In the numerical implementation the level set function is represented by its values on grid nodes, and the current surface must be extracted from this grid. In order to apply the level set method a suitable initial function $\varphi(0, \mathbf{x})$ has to be defined first. The natural choice for the initialization is the signed distance function of a point from the given surface. This function is the common distance function multiplied by -1 or +1 depending on which side of the surface the point lies on. As already stated, the values of the velocity function are determined by the physical models. In the actual numerical implementation equation (1) is represented by the upwind finite difference schemes (see ref. [10] for the details) that requires the values of this function at the all grid points considered. In reality the physical models determine the velocity function only at the zero level set, so it must be extrapolated suitably at grid points not adjacent to the zero level set. Several approaches for solving level set equations exist which increase accuracy while decreasing computational effort. They are all based on using some sort of adaptive schemes. The most important are narrow band level set method [10, 11], widely used in etching process modeling tools, and recently developed sparse-filed method [13], implemented in medical image processing ITK library [14]. Adaptive methods use the fact that actual calculations should not be performed for points far away from the zero level set, since these points do not have any influence. This is the starting assumption in narrow band methods; the width of the narrow band is predefined and should be as small as possible. In actual implementations it is necessary to choose a new narrow band whenever the front hits the boundary of the current narrow band. Another problem is to find the balance between the width of the narrow band and the frequency of reinitializations. This technique provides a substantial speed up; in three dimensions the computational effort is reduced from $O(n^3)$ to $O(n^2)$ compared to fixed grids on fixed simulation domains. The sparse-field method use an approximation to the distance function that makes it feasible to recompute the neighborhood of the zero level set at each time step. In that way, it takes the narrow band strategy to the extreme. It computes updates on a band of grid points that is only one point wide. The width of the neighborhood is such that derivatives for the next time step can be calculated. This approach has several advantages. The algorithm does precisely the number of calculation needed to compute the next position of the zero level set surface. The number of points being computed is so small that it is feasible to use a linked-list to keep a track of them, so at each iteration only those points are visited whose values control the position of the zero level set surface. As a result, the number of computations increases with the size of the surface, rather than with the resolution of the grid. In fact, the algorithm is analogous to a locomotive engine that lays down tracks before it and picks up them up behind.

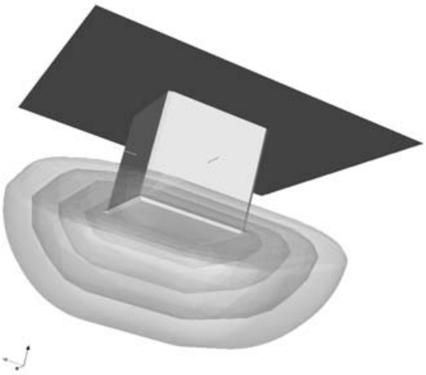

Fig. 1: Isotropic etching – feature profiles at t =0, t = 5 s, t = 10 s, t = 15 s and t = 20 s.

Here we will present some calculations illustrating our approach for etching profile evolution simulation. All the calculations are performed on $128\times128\times384$ rectangular grid. The initial profile surface is a rectangle deep with dimensions $0.1\times0.1\times0.1$ μm. Above the profile surface is the trench region. From its top the particles involved in etching process come from, while bellow is the non-etched material. The actual shape of the initial surface can be described using simple geometrical abstractions. In the beginning of the calculations this description is transformed to the initial level set function using fast marching method [10]. The evolution of the etching profile surface with time is shown in the following figures. In Fig. 1 the results obtained for a test calculation performed with constant velocity function V = V_0 = 5 *nm/s* (purely isotropic etching case) are shown. It is supposed that only the bottom surface could be etched; i.e. that the top and the vertical surfaces belong to photo-resist layer. Behavior of the etching profile is as expected.

The equation (1) can be rewritten in Hamilton–Jacobi form

$$\frac{\partial\varphi}{\partial t}+H(\nabla\varphi(t,x))=0, \qquad (1)$$

where Hamiltonian is given by $H = V(t, \mathbf{x})|\nabla\varphi(t, \mathbf{x})|$ (in this context the term “Hamiltonian” denotes a Hamiltonian function, not an operator). A detailed exposition about the Hamilton–Jacobi equation, the existence and uniqueness of its solution (especially about its viscosity solutions), can be found in [12]. We say that such a Hamiltonian is convex (in $\Re^n$) if the following condition is fulfilled

$$\frac{\partial^2 H}{\partial\varphi_{x_i}\partial\varphi_{x_j}}\geq 0, \qquad (2)$$

where φ_{x_i} is a partial derivative of $\varphi(t,\mathbf{x})$ with respect of x_i . If the surface velocity $V(t,\mathbf{x})$ does not depend on the level set function $\varphi(t,\mathbf{x})$ itself, this condition is usually satisfied. In that case, we can say that the problem is of free boundary type.

The non-convex Hamiltonians are characteristic for plasma etching and deposition simulations. During these processes the etching (deposition) rate, that defines the surface velocity function $V(t,\mathbf{x})$, depends on the geometric characteristics of the profile surface itself, or more precisely, on the angle of the incidence of the incoming particles. In the cases under study here we shall consider an etching beam coming down in the vertical direction. These conditions are characteristic for ion milling technology, but angular dependence of the etching rates appears, more or less, in all etching processes.

The upwind difference scheme cannot be used in the case of non-convex Hamiltonians. The simplest scheme that can be applied in these cases is the Lax–Friedrichs, one which relies on the central difference approximation to the numerical flux function, and preserves monotonicity through a second-order linear smoothing term [11]. In [15] we have shown show that it is possible to use the Lax–Friedrichs scheme in conjunction with the sparse field method, and to preserve sharp interfaces and corners by optimizing the amount of smoothing in it. This is of special importance in the simulations of the etching processes in which spatially localized effects appear, like notching and microtrenching.

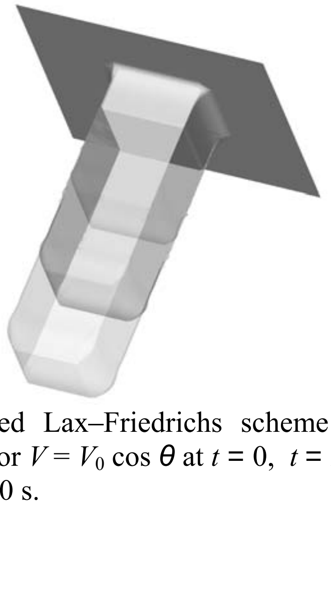

Fig. 2: Optimized Lax–Friedrichs scheme - The etching profiles for $V = V_0 \cos\theta$ at t = 0, t = 20 s, t = 40 s and t = 60 s.

In the case of the ion enhanced chemical etching the dependence of the surface velocity on the incident angle is simple [16]: $V = V_0 \cos \theta$. The pure chemical etching velocity, or more precisely the etching yield, does not depend on the incident angles. This case can be safely treated by the upwind scheme and using the Lax–Friedrichs scheme would lead to unnecessary rounding of the profile surface. The high aspect ratio (depth/width) etching is a common situation in semiconductor technologies. In the Fig. 2 the evolution of the etching profile, when etching rate is proportional to cos*(θ)*, is presented. This is the simplest form of angular dependence, but it describes the ion enhanced chemical etching process correctly [16]. In this case we expect that the horizontal surfaces move downward, while the vertical ones stay still. This figure shows that it with optimal amount of smoothing gives minimal rounding of sharp corners, while preserving the numerical stability of the calculations. Actually, this is one of the most delicate problems in the etching profile simulations.

4.2 Etching rates calculations

A comprehensive simulation of etching requires knowledge of the etching rates at all the points of the profile surface during the etching process. These rates are directly related to fluxes of the etching species on the profile surface, which are themselves determined by the plasma parameters in the etching device. Electrons do not contribute directly to the material removal, but they are the source, together with positive ions, of the profile charging that has many negative consequences on the final outcome of the process especially in the case of insulating material etching, SiO_2 for example. The energy and angular distribution functions for Ar^+ ions (IEDF, IADF) and electrons (EEDF, EADF) are shown in Fig. 3. They are obtained by particle-in-cell (PIC) calculations using XPDC1 code [17, 18, 19]. These data are used as the boundary conditions for the calculations of ion fluxes incident on the profile surface.

Our simulation concept [20] is similar in spirit to the 2D simulations presented in [21], [22] and especially [23], where charging effects in 3D rectangular trench were analyzed. Monte Carlo technique is the only feasible method for calculating particle fluxes in 3D geometries. Trench wall charging strongly influences the charged particles motion and, consequently, particle fluxes which themselves determine the local etching rates. Since the trench boundaries have no regular (rectangular) shape in our simulation, finite element calculations was used for the calculation of the electric field. As the etching profile is not known in advance (it is a result of the calculations itself), the problem of meshing is extremely difficult.

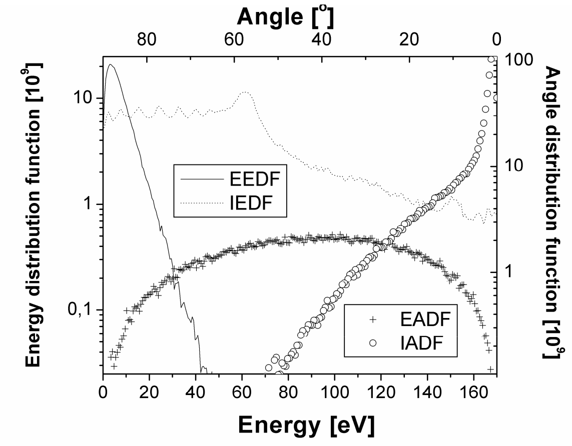

Fig. 3: Electrons (EEDF, EADF) and ions (IEDF, IADF) energy and angular distribution functions.

In Fig. 4 the electrostatic potential map is shown for a test case calculation, for illustrating purposes only. Electric field obtained in that way is used in standard leap-frog particle moving scheme.

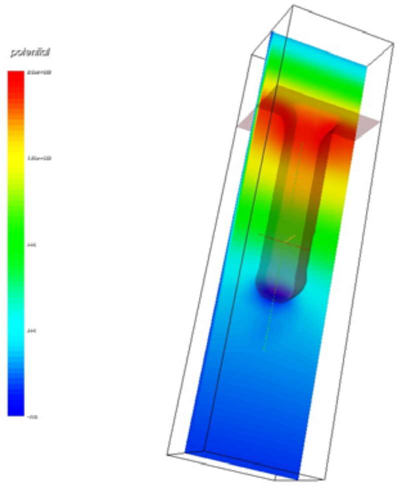

Fig. 4: An example of the electrostatic potential map of the charged feature profile

Although a complete self-consistent cycle that includes profile charging and its influence on the charged particles motion is already implemented, we do not have representative results yet. The reason is the lack of appropriate computational resources, since the Monte Carlo step requires enormous number of particles in every level set step, in order to get satisfying statistics. So, in the next section we present some results that do not include profile charging.

The etching process in medium/high density fluorocarbon plasmas is believed to consist of concurrent etching (of the SiO_2 substrate in our case) and deposition (of a fluorocarbon polymer layer) phenomena [16, 24]. Here the deposition process is neglected for the sake of simplicity. So, we consider only a simple case of chemical etching of SiO_2 under Ar^+ ions bombardment. It is also assumed that the electrons are absorbed at the hitting points, while the neutrals can be absorbed or diffusively reflected, once or many times depending on its sticking coefficient. The positive ions can be absorbed, or specularly reflected, depending on their energy and incident angle. It is assumed that charged particles pass on their charge when they hit the surface, and that this charge does not move after that, what is reasonable for insulating materials. At the boundaries above the profile surface, periodic conditions are assumed.

The surface neutrals coverage (i.e. the fraction of surface covered by free radicals) θ_n statisfy the following balance equation:

$$\frac{d\theta_n}{dt} = J_n S_n (1-\theta_n) - k_{ni} J_{ion} Y_{ni}^{eff} \theta_n - k_{ev} J_{ev} \theta_n , \quad (3)$$

where J_n and J_{ion} are neutral and ion fluxes, S_n is neutral sticking coeficient, k_{ni} and k_{ev} are etchant stoichiometry factors, and Y_{ni}^{eff} is effective etching yield for ion-enhanced chemical etching. Here by the term 'effective' we denote a quantity related to the integral flux, not to an individual particle. J_{ev} is evaporation flux that corresponds to pure chemical etching. It is related to the neutrals flux by Arrhenius law:

$$J_{ev} = K_{SiO_2} e^{-\frac{E_{SiO_2}}{k_b T}} J_n , \quad (4)$$

Balance condition $d\theta_n / dt = 0$ gives the equilibrium surface coverage:

$$\theta_n = \frac{J_n S_n}{J_n S_n \theta_n + k_{ni} J_{ion} Y_{ni}^{eff} + k_{ev} J_{ev}} . \quad (5)$$

So, now we can write equation defining the etching rate ER in the form

$$ER = \frac{1}{\rho_{SiO_2}} [J_{ion} Y_{ni}^{eff} \theta_n + J_{ion} Y_{sp}^{eff} (1-\theta_n) + J_{ev} \theta_n] , \quad (6)$$

where ρ_{SiO_2} is SiO_2 density and Y_{sp}^{eff} is the effective physical spattering etching yield. The etching rate ER defines the velocity function $V(t,\mathbf{x})$ at the profile surface. In actual calculation the feature profile surface is represented by a set of connected triangles, and the above formula should be applied to the every single particular triangle. So, instead of effective etching yields we should define etching yields for every particular ion:

$$Y_{ni}(E_i, \alpha_i) = A_{ni} (\sqrt{E_i} - \sqrt{E_{in}^{th}}) \cos \alpha_i , \quad (7)$$

and

$$Y_{sp}(E_i, \alpha_i) = A_{sp} (\sqrt{E_i} - \sqrt{E_{sp}^{th}}) \cos \alpha_i (1 + B_{sp} \sin^2 \alpha_i) , \quad (8)$$

where E_i is the ion energy and α_i is the angle between the surface normal and the ion incident direction at the point of incidence. Numerical values of the constants appearing in relations (3), (4), (6), (7) and (8) are taken from the reference [13]. The triangular representation of the profile surface requires that instead of integral particle fluxes J_n and J_{ion}, corresponding summations over every particle incident on the particular triangle

$$J_n = \frac{R_n}{A \Delta t_{etch}} N_n , \quad (9)$$

$$J_{ion} Y_{ni}^{eff} = \frac{R_{ion}}{A \Delta t_{etch}} \sum_i Y_{ni}(E_i, \alpha_i) , \quad (10)$$

and

$$J_{ion} Y_{sp}^{eff} = \frac{R_{ion}}{A \Delta t_{etch}} \sum_i Y_{sp}(E_i, \alpha_i) , \quad (11)$$

should be used. Here Nn denotes the number of neutrals absorbed on the particular triangle, R_n (R_{ion}) is the ratio of actual number of neutrals (ions) passing the upper computational domain boundary during the etching time interval and number of neutrals (ions) used in Monte Carlo calculations, A is the particular triangle area, and Δt_{etch} is etching time interval. The ions can be absorbed, or specularly reflected, depending on their energy and incident angle. The probability of specular reflection P_d is given by [25]:

$$P_d = 1 - C_1 \sqrt{E_i} (\frac{\pi}{2} - \alpha_i) . \quad (12)$$

In Fig. 5 the results of the simulation of a highly anisotropic case, that includes both pure and ion-enhanced chemical etching mechanisms, are shown. Neutrals (F radicals) density is supposed to be 10^{19} m^3. The simulation time is $100s$, and it is divided in 100 equal etching intervals (Monte Carlo steps).

5 Inflector Design

Construction of the VINCY Cyclotron, the main part of the TESLA Accelerator Installation, in the

Laboratory of Physics of the Vinča Institute of Nuclear Sciences, has been going on since 1992. It comprises a compact isochronous cyclotron – the VINCY Cyclotron, a volume positive or negative light ion source – the pVINIS Ion Source, two similar electron cyclotron resonance heavy ion sources – the nVINIS Ion Source and the mVINIS Ion Source, and a number of low energy and high energy experimental channels [26, 27]. Part of this work is related to the activities concerning the design of the spiral inflectors for this facility, which is the second example of coupled field and particle simulations treated in this paper.

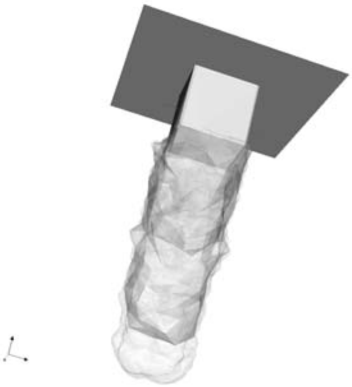

Fig. 5: Anisotropic etching - feature profiles at $t = 0$, $t = 40$ s, $t = 80$ s and $t = 100$ s.

The main roles of the inflector are to bend the injeced ion beam from its initial direction onto the cyclotron’s madian plane and, together with the optical elements in the transport line, to match the beam emittance to the cyclotron’s acceptance. The electrostatic inflector consists of two biased and two grounded electrodes. The electric field produced by these electrodes exerts a force on the ions, simultaneously bending and focusing the beam. The design of an electrostatic inflector is complicated by the fact that in addition to the electrostatic force produced by its electrodes, the ions are also subjected to a magnetic force produced by the magnetic field near the center of the cyclotron. This effect must be taken into account in designing the inflector.

Different types of inflectors have been devised [28-32] for inflecting the axially injected ion beam into the cyclotron median plane. In modern, variable energy, multi-particle, compact cyclotrons, the minimal gap between the magnetic poles tends to be very small (few centimeters - to provide high flutter and high magnetic circuit efficiency). This fact imposes severe restrictions on inflector dimensions, as well as specific demands concerning its optical properties. Owing to its flexibility and relatively low voltage needed for its operation, the electrostatic spiral inflector has become widely used in the multi-particle compact cyclotrons. Due to the complexity of the shape of its electrodes, this particular type of inflector has been chosen as illustrative (see Fig. 6).

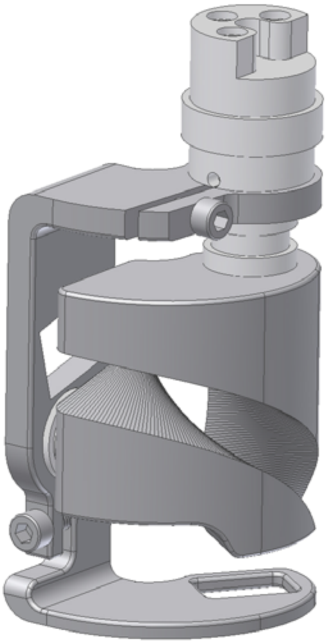

Fig. 6: Mechanical drawing of the spiral inflector.

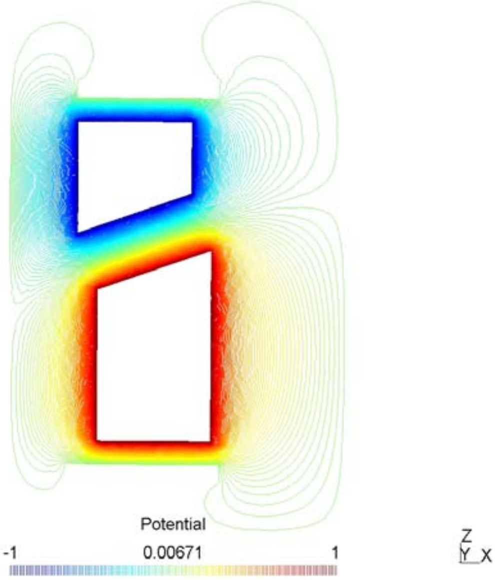

Fig. 7: Electrostatic potential map at the exit plane of the spiral inflector.

Analysis of the optical properties of particular spiral inflector calls for extensive ion beam transport simulations for wide range of ion species and beam emittances. Necessary prerequisite for this is a detailed knowledge of the electrostatic potential map between the biased electrodes as well as in the limited region before/after the inflector entrance/exit (fringe field region). Although several analytic approximation of the spiral inflector electrostatic field exist [30, 31, 32], their precision is limited and can be used only in the initial phase of inflector design. In the final stage of the design, finite elements solution of the 3D Laplace equation is compulsory.

The ion trajectories are calculated by solving Newton-Lorenz equations of motion using Boris integrator scheme [1], since magnetic field must be taken into account also. In Fig. 7 an electrostatic potential map at the exit plane of the inflector is shown. The voltages applied to the upper and the lower electrodes are taken to be -1V and 1V in the potential calculations.

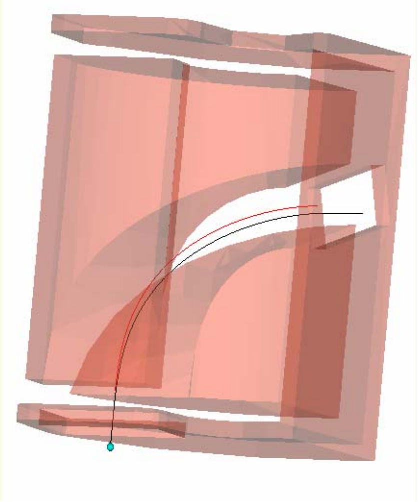

Fig. 8: Analytical (red) and numerical central trajectories of the ion beam in the spiral inflector.

So proper multiplication factors should be included when forces on the ions are calculated. In this way, we do not have to recalculate potential (electric field) for different voltages on the electrodes.

In Fig. 8 an analytical approximation of the central ion trajectory (in red) [31], with no fringe field included, and the trajectory obtained by the described numerical procedure are shown (in black). This result is preliminary, and more detailed calculations that include particle bunches and interparticles interaction are in due course.

6 Conclusion

In this paper we presented some results of two different types of combined particle-field simulations (based on the unified software framework), in which the authors are involved. The first is connected with the applications of the plasma etching technologies in microelectronic circuits manufacturing. Obtained results show that more realistic calculations should include the effects of polymer deposition, profile charging, more complex surface reactions set, as well as better statistic (greater number of particles) in the Monte Carlo step of the calculations, which is now limited by the available computational resources. Another is a part of the ongoing work on the design of the spiral inflector for the VINCY Cyclotron. In present phase this simulations are preliminary, and more comprehensive analysis that includes the detailed maps of central region magnetic field, particle bunches and interparticle interaction is necessary.

Acknowledgments:

The present work has been supported by the the TESLA Project: Science with Accelerators and accelerator technologies (1247) and project 1478 financed by the Ministry of Science and Environmental Protection.

References:

[1] C. Birdsall and A. Langdon, *Plasma Physics via Computer Simulation*, McGraw-Hill, New York, 1985.

[2] Fltk, http://www.fltk.org

[3] Vtk, http://www.vtk.org

[4] Pthreads, http://sourceware.org/pthreads-win32

[5] A. Bossavit, *Computational Electromagnetism*, Academic Press, Boston, 1998.

[6] K. Warnick, , R. Selfridge, , and D. Arnold, 1997. Teaching Electromagnetic Field Theory Using Differential Forms. *IEEE Trans. on Education*, Vol. 40, No. 1, pp. 53–68

[7] C. Geuzaine, *High order hybrid finite element schemes for Maxwell's equations taking thin structures and global quantities into account*, Ph.D. Thesis, Universite de Liege, 2001.

[8] GetDP, http://www.geuz.org/getdp

[9] TetGen, http://tetgen.berlios.de

[10] J. Sethian, *Level Set Methods and Fast arching Methods: Evolving Interfaces in Computational Geometry, Fluid Mechanics, Computer Vision and Materials Sciences*, Cambridge University Press, Cambridge, UK, 1998.

[11] S. Osher and R. Fedkiw, *Level Set Method and Dynamic Implicit Surfaces*, Springer, 2002.

[12] L. Evans, *Partial Differential Equations*, American Mathematical Society, Providence, RI, 1998.

[13] R. Whitaker, “A Level-Set Approach to 3D Reconstruction From Range Data”,*The International Journal of Computer Vision* Vol. 29, No. 3,1998, pp 203-231

[14] NLM Insight Segmentation and Registration Toolkit, http://www.itk.org

[15] B. Radjenović, J. K. Lee and M. Radmilović-Radjenović,“ Sparse field level set method for non-convex Hamiltonians in 3D plasma etching profile simulations“, *Computer Physics Communications* Vol. 174, 2006, pp. 127–132

[16] M. Lieberman and A. Lichtenberg, *Principles of Plasma Discharges and Materials Processing*, John Wiley & Sons, Inc. 1994.

[17] C.K. Birdsall, ``Particle-in-cell charged-particle simulations, plus Monte Carlo collisions with neutral atoms, PIC-MCC", *IEEE Transactions on Plasma Science*., vol. 19, 1991, pp. 65-85

[18] J.P. Verboncoeur, M.V. Alves, V. Vahedi and C. Birdsall, ``Simultaneous Potential and Circuit Solution for 1D Bounded Plasma Particle Simulation Codes", *Journal of Computational Physics*., vol. 104, 1993, pp. 321-328

[19] H.C. Kim, F. Iza, S.S. Yang, M. Radmilović-Radjenović and J.K. Lee, "Particle and fluid simulations of low-temperature plasma discharge: benchmarks and kinetic effects", *Journal of Physics D: Applied Physics,* vol..38, 2005, pp. R283-R301

[20] B. Radjenović and J. K. Lee, “3D feature profile evolution simulation for SiO_2 etching in fluorocarbon plasma“, XXVIIth ICPIG, Eindhoven, the Netherlands, 18-22 July, 2005

[21] G. Hwang and K. Giapis, “On the origin of the notching effect during etching in uniform high density plasmas”, *Journal of Vacuum Science and Technology*, Vol. B15(1), 1997, pp. 70-87

[22] A. Mahorowala and H. Sawin, “Etching of polysilicon in inductively coupled Cl2 and HBr discharges. IV. Calculation of feature charging in profile evolution”, *Journal of Vacuum Science and Technology*, Vol. B20(3), 2002, pp. 1084-1095

[23] H. S. Park, S. J. Kim, Y. Q. Wu and J. K. Lee, “Effects of Plasma Chamber Pressure on the etching of Micro Structures in SiO_2 with the Charging Effects”*IEEE Transactions on Plasma Science*, Vol. 31, 2003, pp. 703-710

[24] A. La Magna and G. Garozzo, “ Factors Affecting Profile Evolution in Plasma Etching of SiO_2 - Modeling and Experimental verification“ *Journal of Electrochemical Society*, Vol. 150, No. 10, 2003, pp. F178-F185

[25] G. S. Hwang, C. Anderson, M. Gordon, T. Moore, T. Minton and K. Giapis, “ Gas-Surface Dynamics and Profile Evolution during Etching of Silicon“ *PhysicalReview Letters* Vol. 77, No. 14, 1996, pp. 3049-3052

[26] N. Nešković, et al., "Status report on the VINCY Cyclotron", N*UKLEONIKA* 2003; 48 (Supplement), pp. S135-S139.

[27] N. Nešković et al., "TESLA Accelerator Installation", Proceedings of the Third European Particle Accelerator Conference (Frontières, Gif-sur-Yvette), 1992, Vol. 1, pp. 415-417

[28] W.B. Powell, B.L. Reece, "Injection of ions into a cyclotron from an external source", *Nulcear Instruments and Methods,* Vol. 32, 1965, pp. 325-332.

[29] J.L. Belmont, J.L. Pabot, "Study of Axial Injection for the Grenoble Cyclotron", *IEEE Transactions on Nucear Sciences* Vol. NS-13, 1966, pp. 191- 193

[30] R.W. Müller, "Novel Inflectors for Cyclic Accelerators", *Nulcear Instruments and Methods* Vol. 54 1967, pp. 29-41

[31] L.W. Root, *Experimental and Theoretical Studies of the Behaviour of an H^- Ion Beam During Injection and Acceleration in the TRIUMF Central Region Model Cyclotron*, Ph.D. Thesis, Univ. of British Columbia, Vancouver, 1974.

[32] P. Beličev, D.V. Altiparmakov, "An Efficient Electric Field pproximation for Spiral Inflector Calculations", *Nulcear Instruments and Methods*, 2001, Vol. 456, pp. 177- 189