IL NUOVO CIMENTO Vol. 32 C, N. 2 Marzo-Aprile 2009
DOI 10.1393/ncc/i2009-10390-y

Colloquia: CSFI 2008

Development of a Particle-In-Cell code with Structured Adaptive Mesh Refinement for Plasma Focus devices breakdown simulation

G. Grasso(*), F. Rocchi and M. Sumini

Laboratorio LIN di Montecuccolino, Università di Bologna
via dei Colli, 16, 40136 Bologna, Italy

(ricevuto il 22 Giugno 2009; pubblicato online il 4 Settembre 2009)

Summary. — The aim at simulating the breakdown phase of a Plasma Focus (PF) discharge follows the need to fully understand the dynamics of such device, in order to retrieve useful information for the design and optimization of the machine itself. PFs are compact devices able to generate, accelerate, compress and confine a plasma by means of strongly varying electric and magnetic fields. In the final phase of the discharge, the generated plasma collapses in a high density region (the focus) where nuclear reactions occur. The choice of the gases composing the plasma tunes the nuclear reactions in order to characterize the device as a possible neutron-free Short-Life Radioisotopes (SLRs) generator for PET (f.i. ^{18}F and ^{15}O), as well as a neutrons or collimated-electrons-beams source for radio-therapy applications. An electrostatic-collisional Particle-In-Cell (PIC) code for Plasma Focus devices (es-cPIF) has already been developed to investigate the breakdown phenomenon and the formation of the plasma *seed*, the preliminary plasma spot, within the device: the exact knowledge of the phase space distribution function (strongly deviating from the Maxwellian equilibrium one) is a fundamental basis indeed for the whole discharge simulation. In order to extend the present simulations towards the complete evolution of the plasma seed into a running plasma sheath, the code is being re-structured for strong parallelization and inclusion of Structured Adaptive Mesh Refinement (SAMR) capabilities. In this paper the development frame as well as the software design architecture are presented together with the features that will be provided by the new SAMRes-cPIF code.

PACS `52.65.Rr` – Particle-in-cell method.
PACS `52.58.Lq` – Z-pinches, plasma focus, and other pinch devices.

1. – Introduction

The formation of a plasma within a PF device is due to the electric breakdown of the filling gas caused by the closure of a capacitor bank onto the two discharge electrodes: the high-collisional condition leads indeed to the avalanche ionization of the gas and the initial formation of a plasma *seed*. The further evolution of the breakdown and its

---

(*) E-mail: `giacomo.grasso@mail.ing.unibo.it`

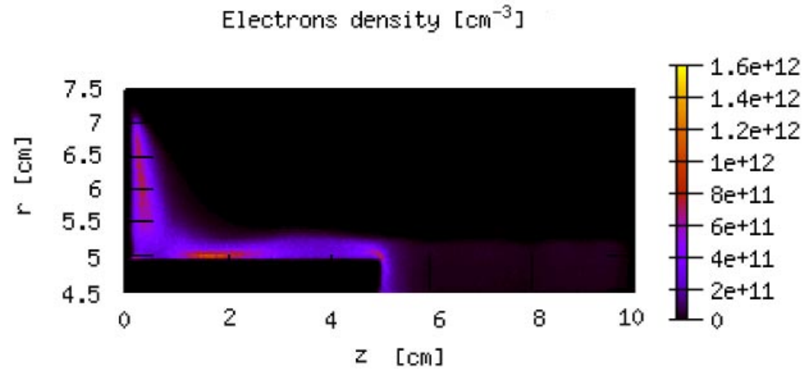

Fig. 1. – Plasma sheath formation by es-cPIF simulation.

sustainment are due to the action of Secondary Electrons Emission (SEE) phenomena from the boundary walls (both the electrodes and the dielectric insulating them), as confirmed by the latest simulations performed with the existing es-cPIF code [1] (fig. 1).

The breakdown phase of a PF discharge is therefore an interesting matter of study both on a physical and a numerical point of view: the non-equilibrium conditions and the complexity introduced by the multitude of involved phenomena require to turn to a microscopical analysis of the breakdown. The simulation of such a phenomenon results therefore in a High-Performance Computing (HPC) task due to the huge number of particles to be studied in order to correctly represent the whole system behavior.

2. – Development framework

The Structured Adaptive Mesh Refinement Application Infrastructure (SAMRAI) [2] is a C++ framework designed to provide a self-consistent environment to develop SAMR codes within. The framework architecture, achieved by means of deep patterns design, structures the tool as a general-purpose host: a framework is indeed a "set of cooperating classes that make up a reusable design for a specific class of software" [3].

A pre-defined infrastructure (providing automatic methods for mesh generation, parallel data communication, time integration and problem solution) dictates the architecture of the application. Within the framework, it is possible to design the layout of the resulting code by choosing among the different multigrid generation techniques and integration schemes provided, and characterize then the most suitable architecture chosen with problem specific code.

2˙1. *Particles management.* – A first important extension of the framework has been achieved by developing the missing classes for particles management on the dynamically evolving multigrid. SAMRAI integrates indeed only grid-located data classes, which are automatically initialized on multilevel patches and communicated every time the grid hierarchy is redefined. Particles, however, cannot be described as grid-located data, since they may move upon the grid, as well as agglomerate or disperse locally. A new kind of variable had therefore to be defined into the framework in order to represent all the dispersed data just slightly connected to the grid.

It is not even possible to introduce particles definition for each grid patch as a simple array of data, since a back trace of particles to the belonging cell is required by the Monte Carlo Collisional (MCC) module (necessary to simulate the ionization of the plasma), by the charge to node deposition function (the shape function proper of PIC codes) and by the optimization techniques already present in the es-cPIF code. Every grid cell must therefore reference the proper (whatever) set of particles belonging to its own domain.

By making use of indexed patch data classes, the new “ParticleData” variable has been defined into the framework: such a kind of data cannot be univocally or regularly defined onto the multigrid hierarchy, nor easily managed for intra-processors communication by means of the existing grid-located data types. All the characterizing classes and communication instances have been therefore completely written from scratch to provide the flexible tool yet missing into the framework.

3. – SAMRes-cPIF architecture

The development of the SAMRes-cPIF code within the SAMRAI framework requires the creation of a specific class implementing the application object, a main program summarizing the overall architecture, and all the physical methods modeling the phenomenon provided as external Fortran subroutines.

3˙1. *Code layout and features*. – The application class must define all data to be allocated on the grid hierarchy (field, charge density, particles), as well as the actual implementations of the virtual methods for grid management and the interfaces to the external physical functions. It therefore characterizes the code by tailoring the whole framework to suite the problem of interest. In particular, the SAMRes-cPIF application object will implement the methods for the dynamic refinement of the grid based upon the field source gradient detection: the exact knowledge of the self-consistent field by means of a fine improvement of the solution is fundamental in order to take into account the local effect of charge separation in the simulation of the plasma dynamics.

The massive parallelization of the code will allow an increase in simulation particles number, required by PIC methods for the statistical representativeness of the simulating system. The multilevel simulation domain will be decomposed in patches and distributed among the available processors together with the simulation particles belonging to each patch. The concurrent solution on every patch and the separate particles management are the key features of the overall scheme: an accurate, efficient load balancing will at least grant the effective speed-up of the code, which will be monitored by means of integrated calls to the Tau profiler.

The numerical features of the present es-cPIF code, such as the MCC module, the improved cache management and the control of the number of simulation particles via the innovative Hierarchical Agglomerative Sub-Clustering (HASC) merging technique [1], will be kept and extended to the new SAMR 3D scheme. Figure 2 presents the general scheme of the SAMRes-cPIF code, whose aimed features are summarized in table I.

4. – Conclusions

The SAMRes-cPIF code under development will represent the most advanced PIC code for electrical breakdown study: by exploiting the SAMR technique for the dynamical refinement of the field solution, a full 3D geometrical description will be possible.

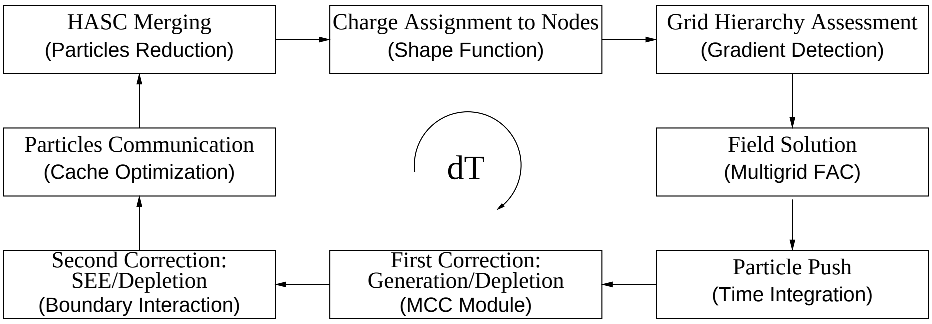

Fig. 2. – General scheme of the SAMRes-cPIF code.

Table I. – *Compared es-cPIF and SAMRes-cPIF properties and main features.*

	es-cPIF	SAMRes-cPIF (expected)
model	$2\frac{1}{2}$D	3D
grid	static	SAMR
solver	SOR	multigrid FAC
simulation particles	10 M	(several G)
breakdown time simulated	32 ns	(200 ns)
simulation time	12 h	(24 h)
MCC	yes	yes
SEE models	1-2D	3D
HASC merging	$2\frac{1}{2}$D	3D
cache optimization	yes	yes
integrated profiler	no	yes

The strong parallelization will furthermore grant the capability of simulating billions of particles when run on suitable HPC clusters.

Such a powerful tool, applied to the simulation of the PF breakdown, will allow to push the simulation towards the full plasma sheath formation and the closure of the electrical circuit. The 3D model will also allow the authors to investigate the angular plasma distribution representing the basis of the regular (modal) filamentation of the sheath, experimentally observed but not yet well understood.

* * *

This work was supported by CINECA. In particular, the authors would like to thank dr. C. Gheller for the valuable collaboration.

REFERENCES

[1] Frignani M., Grasso G., Rocchi F. and Sumini M., *PIC-MCC Simulation of the Electrical Breakdown in a Plasma Focus Device* in *Joint International Topical Meeting on Mathematics & Computation and Supercomputing in Nuclear Applications (M&C + SNA 2007)* Monterey, California, April 15-19, 2007, on CD-ROM (American Nuclear Society, LaGrange Park, IL) 2007.

[2] Hornung R. D. and Kohn S. R., *The use of object-oriented design patterns in the SAMRAI structured AMR framework*, in *SIAM Workshop on Object Oriented Methods for Inter-operable Scientific and Engineering Computing (00'98), Yorktown Heights, New York, October 21-23, 1998.*

[3] Gamma E., Helm R., Johnson R. and Vlissides J., *Design Patterns: Elements of reusable Object-Oriented Software* (Addison-Wesley Publishing Co., Menlo Park, CA) 1995.