Introducing the domestic multi-dimensional particle-in-cell code AZERAP

Yazdanpanah J.*

Plasma and Nuclear Fusion Research School, Nuclear Science and Technology Research Institute, Tehran, Iran

* *Email:* jyazdanpanah@aeoi.org.ir

Abstract

The domestic plasma modeling framework AZERAP is introduced and its capabilities in simulating the plasma based accelerators and intense beam-plasma interaction are discussed. AZERAP is developed based on the fully kinetic, electromagnetic relativistic PIC algorithm. It is implemented in the object oriented language C++ and utilizes the Message Passing Interface (MPI) for parallelization. The main idea behind the development of AZERAP has been establishing a software platform for virtual plasma laboratory. Achieving this goal has implied (1) attaining high functionality in introducing the input problem, (2) supporting abstraction of the field and plasma structures/modules, and (3) supporting high flexibility for future developments. The present first beta-release of AZERAP paws the way toward these objectives. Moreover, it offers a very comfortable user experience with code compile, debugging execution, data accusation and data animation, simulating plasma based accelerators.

Keywords: Advanced plasma simulation, High performance computing, Object oriented programming, Particle-in-cell method, Plasma based accelerators.

Introduction

Plasma state is in the heart of many important concepts in nuclear science, like as thermonuclear fusion, and bright, energetic particle and photon beam sources. In addition to many applications of bright beam sources, the plasma heating in thermonuclear fusion relies on using beam source derives. Due to very high complexity of plasma systems, simulations play a vital role in researches and developments in this area[1]. Especially, the fully kinetic methods which capture self-consistent evolutions of distribution function are of great interest, as they can simulate the plasma in a very close coincidence with the experimental conditions. Among, different kinetic methods the particle-in-cell (PIC) scheme is very promising in comprising the huge computational complexity (scales) inherited with these methods, with the present limitations in hardware resources. In PIC a trade-off occurs between unnecessary details and computational complexity reduction. Many, successful applications of this method to diverse plasma scenarios, especially plasma based beam sources, have demonstrated its promising potential as a candidate for plasma virtual laboratory. However, despite these successes, still there exist challenges utilizing this method to full scale plasma devices and experiments at 3D. In this regard, this simulation method is still a hot topic in computational plasma researches. Problems like improving the computational accuracy, accelerating execution by applying high performance computing (HPC) methods and improving data accusation and visualization through the very big output data are of current interest, in this respect. Nowadays, many modern PIC code are released each with its specific superior features and objectives (see e.g. [2-4]). The author has begun develping PIC codes more than a decade ago(see e.g. [5,6]). The domestic code AZERAP has been developed in quest for establishing a software platform for virtual plasma laboratory characterized mainly by properties like as (1) attaining high functionality in introducing the input problem, (2) supporting abstraction of the field and plasma structures/modules, and (3) supporting high flexibility for future developments. The descriptions and results presented below are due to the first beta-release of AZERAP.

Code discriptions

We have exploited a multi-layered (multitier) architecture in designing AZERAP. An outlook of this architecture in terms of developed components and the top-down hierarchy among them is shown in Fig. 1. The code is implemented in object oriented C++ language and utilizes Message Passing Interface (MPI) for parallelization. Using MPI allows for utilizing the distributed memory resources. An important feature of AZERAP is that, while being quite portable, it is completely self-supporting; it does not link to any external library other than the common standard C/C++ libraries. This feature leads to a very easy installation and makes it possible to be compiled in a single stage and produces a single executable. The scripting language (the standard of input deck) is developed to be in the most natural way and supports functional dialog. To perform a computer experiment, user defines the problem through the input deck by providing plasma and electromagnetic parameters and functions. Then the desired diagnoses are determined in the same way and the compile code is executed. In AZERAP, data accusation and analyses is more facilitated by two ways: 1) It has the capability to directly output the results in the '.bmp' format and online animation. 2) It is possible to use test sample particles to easily understand the physics behind the particle dynamics while avoiding search in extermely large data of plasma particles.

Results and discussion

In Fig. 2, outputs of a typical computer experiment on laser plasma accelerator are presented among results from the simple user graphical interface of the code. Here, the quantitative discussion of results is not intended. A strong laser pulse enters into a column of plasma from the left (the top most panel), excites an accelerating micro-cavity structure very similar to the conventional accelerators, but rather more compact in size (second panel)[7]. The produced structure accelerates an intense bunch of electrons which eventually exit from the right as trail to the laser pulse (third panel).

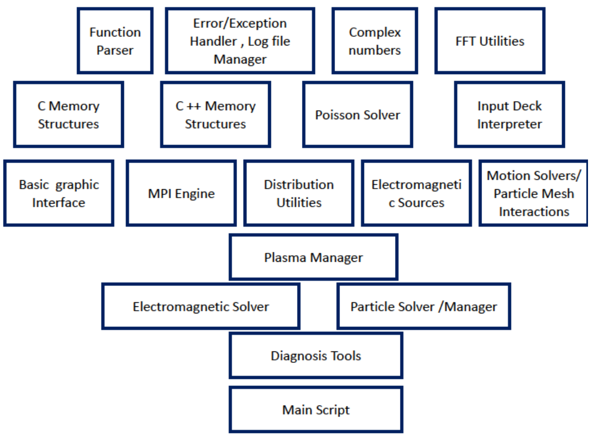
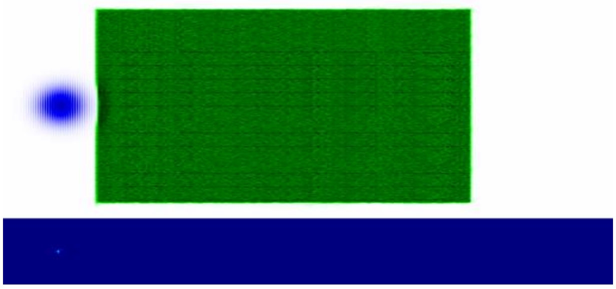


Figure 1. AZERAP architecture in terms of its components and their linkage.



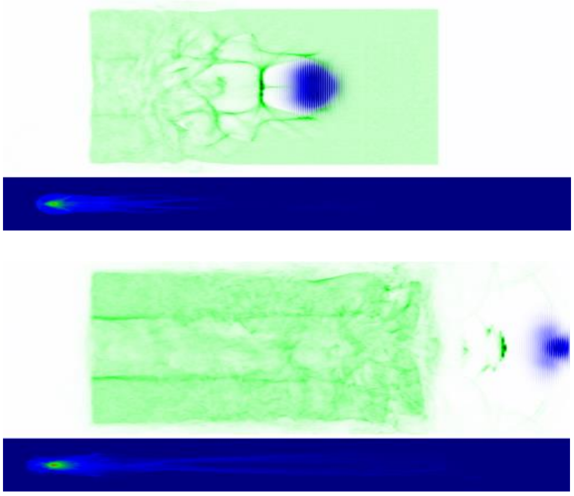


Figure 2. AZERAP outputs for laser and plasma evolutions and beam production at a typical computer experiment of laser plasma accelerator.

Conclusions

In conclusion, we have outlined the properties and advance features of the new domestic PIC code AZERAP. We also presented example results from simulation of laser plasma accelerator using this code.

References

[1] T. Tajima, *Computational plasma physics with applications to fusion and astrophysics*, (Taylor& Francis LLC, 2004)

[2] C. Nieter, J. R. Cary, Vorpal: A Versatile plasma simulation code, J. Comput. Phys. 2 (2004), p. 448-473.

[3] R. A. Fonseca etal, OSIRIS: A Three-Dimensional, Fully Relativistic Particle in Cell Code for Modeling Plasma Based Accelerators, International conference on computational science ICCS 2002, p. 342-351.

[4] J. Derouillat etal, Smilei : *A collaborative, open source, multi-purpose particle-in-cell code for plasma simulation*, *Computer Physics Communications* 222 (2018), p. 351- 373.

[5] J. Yazdanpanah and A. Anvary, *Time and space extended-particle in cell model for electromagnetic particle algorithms* , Phys. Plasmas 19 (2012), p. 033110 ,17 pages.

[6] J. Yazdanpanah, *Self modulation and scattering instability of a relativistic short laser pulse in an underdense plasma*, Plasma Phys. Control.Fusion 61 (2019), 85021, 15pages.

[7] A. Pukhov, etal , *The bubble regime of laser–plasma acceleration: monoenergetic electrons and the scalability*, Plasma Phys. Control. Fusion 46 (2004), B179, 8 pages.