

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

Jam Yazdanpanah

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

## HIGHLIGHTS

- Development of advanced PIC code AZERAP as the first one in its type in the country is reported.
- Its major goals, potentials, architecture and advanced features are briefly outlined.
- Its computational core is outlined and discussed.
- Examples of its operation in the case of laser plasma accelerator are provided.

## 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. The current first beta-release of AZERAP exploits the fully kinetic, electromagnetic relativistic PIC algorithm as its numerical engine. 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 for plasma based particle beam sources and high power electromagnetic generators. Achieving this goal has implied attaining high functionality in introducing the input problem, supporting abstraction of the field and plasma structures/modules, and 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

## HISTORY

Received: 7 May 2022  
Revised: 2 July 2022  
Accepted: 11 July 2022  
Published: Winter 2023

## 1 Introduction

Particle and radiation beam sources have many scientific and medical applications. Especially, beam sources are used as heating drivers in different concepts of thermonuclear fusion. High power ion and neutral sources are needed to be developed for the giant magnetically-confined fusion machine ITER, and great achievements attained up to now (Toigo et al., 2017). We also should add active ongoing projects on direct-drive heavy ion beam inertial confinement fusion (Kawata, 2021; Horioka, 2018). Plasma based beam sources are unique in achieving high brightness, strong currents and high power (Brown, 2004). This is because the plasma provides a high density source of charged particles which may be controlled and accelerated by applying electric and magnetic field configurations. Moreover, recent technological developments in construction of ultra-intense laser pulses and super-energetic parti-

cle bunches, have demonstrated the plasma potentials as a unique accelerating medium capable of supporting ultra-high acceleration gradients of  $\text{GeV}\cdot\text{cm}^{-1}$  (Tajima et al., 2020; Esarey et al., 2009); at least 1000 times higher than was possible before with the conventional cavity accelerators. Nowadays, plasma is considered not only as a high density charged particle source, but also as a unique candidate for compact accelerator. Projects like FACET (Yakimenko et al., 2019), EuPRAXIA (Assmann et al., 2020), and AWAKE (Gschwendtner et al., 2016) are examples of major active projects planning to utilize these compact plasma based accelerators.

Optimized construction of stable plasma based beam sources demands for extensive studies on plasma dynamics and transport during its formation and interactions. Generally, we need to precisely assess the plasma/gas response when it is subjected to external strong fields under diverse conditions and when subjected to strong electromagnetic

\*Corresponding author: [jjazdanpanah@aeoi.org.ir](mailto:jjazdanpanah@aeoi.org.ir)

signals and particle beams. Moreover, these assessments are critical to understand the fuel heating during the nuclear fusion.

Due to very high complexity of plasma systems, simulations play a vital role in researches and developments in this area (Tajima, 2018). 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 fully-kinetic methods the particle-in-cell (PIC) scheme (Tajima, 2018; Hockney and Eastwood, 1988; Birdsall and Langdon, 1991) 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. Very successful experiences with application of this method in diverse plasma scenarios, especially plasma based beam sources, have demonstrated its promising potential as a candidate for virtual plasma laboratory. However, despite these successes, still there exist challenges utilizing this method to full scale plasma devices and experiments at 3D. In this regard, PIC method is still a hot ongoing topic in computational plasma researches. Problems like improving the computational accuracy, accelerating execution by exploiting new high performance computing (HPC) methods, and improving data accusation and visualization through the very big output data are of current interest.

Nowadays, many modern PIC code are released each with its specific superior features and objectives (see e.g. (Nieter and Cary, 2004; Fonseca et al., 2002; Derouillat et al., 2018; Pukhov, 1999; Verboncoeur et al., 1995)). The author has begun developing PIC codes more than a decade ago (see e.g. (Yazdanpanah and Anvari, 2012, 2014; Yazdanpanah, 2017, 2019; Khalilzadeh et al., 2015; Pishdast et al., 2018)). The domestic code AZERAP (abbreviation of the Persian phrase Azemayeshgahe Rayanaye Plasma, meaning virtual plasma laboratory) has been developed in quest for establishing a software platform for virtual plasma laboratory, especially in the field of plasma based beam sources and their interactions. This platform is 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 which is currently a 2D3V implementation (2-dimensional field solver with arbitrary (3D) polarization, 3-dimensional motion solver), with potential for introduction of (3D) field solvers in near future.

The paper is organized as follows; In Sec. 2 we discuss the code physics base and its general algorithm. In Sec. 3 the computational core of the code and its extensibility potentials are described. Sec. 4 briefly summarizes the code structure and implementation, and Sec. 5 gives examples of code operation and discusses the importance of its results in the context of laser-plasma accelerators.

## 2 General physics basics and algorithm

$$\left. \frac{\partial f_\alpha}{\partial t} + \mathbf{v} \cdot \frac{\partial f_\alpha}{\partial \mathbf{x}} + (\mathbf{E} + \mathbf{v} \times \mathbf{B}) \cdot \frac{\partial f_\alpha}{\partial \mathbf{p}} = \frac{\partial f_\alpha}{\partial t} \right)_c \quad (1a)$$

$$\nabla \cdot \mathbf{E} = \frac{\rho}{\varepsilon_0}, \quad \nabla \cdot \mathbf{B} = 0 \quad (1b)$$

$$\nabla \times \mathbf{E} = -\frac{\partial \mathbf{B}}{\partial t}, \quad \nabla \times \mathbf{B} = \frac{1}{c^2} \frac{\partial \mathbf{E}}{\partial t} + \mu_0 \mathbf{J} \quad (1c)$$

$$\rho(r, t) = \sum_\alpha q_\alpha \int d^3p f_\alpha(r, p, t) \quad (1d)$$

$$\mathbf{J}(r, t) = \sum_\alpha q_\alpha \int d^3p \mathbf{v} f_\alpha(r, p, t)$$

where  $f$ ,  $t$ ,  $\mathbf{x}$ ,  $\mathbf{v}$ , and  $\mathbf{p}$  are distribution function, time, position, velocity and momentum, respectively.  $\alpha$  shows the plasma species index. The right hand in Eq. (1a) represents the collisions.  $\mathbf{E}$ ,  $\mathbf{B}$ ,  $\mathbf{J}$ , and  $\rho$  are electric field, magnetic field, current density and charge density respectively.  $\varepsilon_0$ ,  $\mu_0$  and  $c$  are vacuum permittivity, permeability and light speed respectively.  $q$  is the electric charge. The variants of kinetic numerical methods (e.g. PIC, direct Vlasov (Sonnendrücker et al., 1999) etc.) are mostly based on integration of the Vlasov equation along its Lagrangian characteristics. For plasmas out of thermal-equilibrium (non collision dominated) the solution for Boltzmann equation may be obtained from this Vlasov solution by reconciling for comprising the particle collisions. The Vlasov characteristics are  $\mathbf{x} = \mathbf{x}(t; \mathbf{x}_0; \mathbf{v}_0)$  and  $\mathbf{v} = \mathbf{v}(t; \mathbf{x}_0; \mathbf{v}_0)$  derived by integration of Lorentz-Newton equations of motion:

$$\begin{aligned} \frac{d\mathbf{x}}{dt} &= \mathbf{v} \\ \frac{d\mathbf{p}}{dt} &= q(\mathbf{E} + \mathbf{v} \times \mathbf{B}) \end{aligned} \quad (2)$$

which once inverted give equations for initial phase-space coordinates according to time and new coordinates  $\mathbf{x}_0 = \mathbf{x}_0(t; \mathbf{x}, \mathbf{v})$  and  $\mathbf{v}_0 = \mathbf{v}_0(t; \mathbf{x}, \mathbf{v})$ . In the direct Vlasov integration (DVI) method (Sonnendrücker et al., 1999), the distribution function at time  $t$  is obtained from its initial value by using the conservation of distribution function along the characteristics,

$$f_\alpha(t, \mathbf{x}, \mathbf{v}) = f_\alpha(t = 0, \mathbf{x}_0(t; \mathbf{x}, \mathbf{v}), \mathbf{v}_0(t; \mathbf{x}, \mathbf{v})) \quad (3)$$

Note that this equation is equivalent to the more familiar statement of conservation of particle-number inside the commoving phase-space element, i.e.  $f_\alpha(t, \mathbf{x}, \mathbf{v}) = f_\alpha(t = 0, \mathbf{x}_0(t; \mathbf{x}, \mathbf{v}), \mathbf{v}_0(t; \mathbf{x}, \mathbf{v})) [\partial(\mathbf{x}_0, \mathbf{v}_0) / \partial(\mathbf{x}, \mathbf{v})]$  where the partial derivative represents the Jacobian determinant of the transformation which equals 1 as the phase space volumes between adjacent trajectories remains conserved. To determine the distribution function over a discretized region of phase-space (PS) (defined as the phase-space grid) at a given time  $t$ , for each grid point we should trace back the characteristic passed through that point, find the original grid-point at an initial time  $t_0$ , and then assign the original distribution function according to Eq. (3). As the original PS grid-points do not necessarily coincide with the current (time  $t$ ) grid-points (the back traced characteristics give a different region of phase), evaluation of

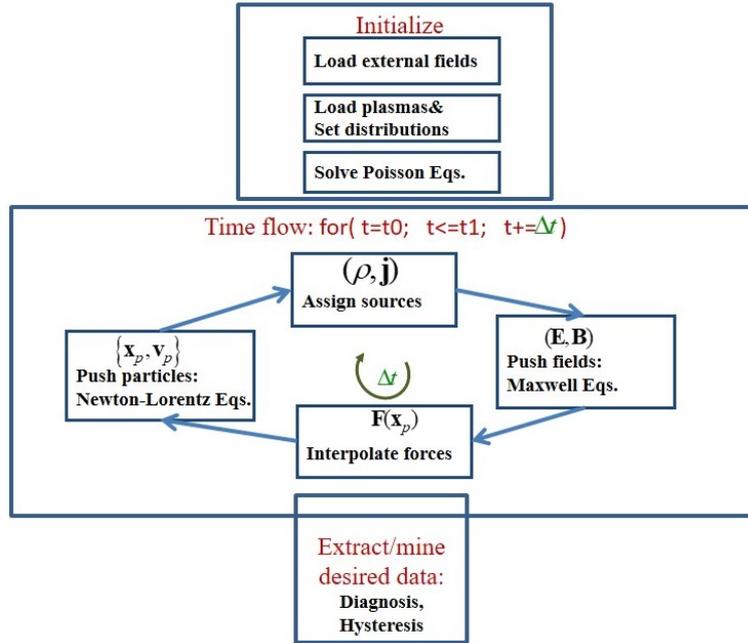


Figure 1: The typical flowchart of a user flexible PIC code.

distribution function usually inherited with interpolations and/or extrapolations from this grid, which are very time consuming. Moreover, as it is obvious, in the DVI method, integration is done over the full range discretized 6 dimensional phase space, rather than usual 3 dimensional spatial space. Therefore, DVI implies tremendous computational complexity, such that it becomes practically inapplicable for real scale or 3 (space) dimensional problems, even with parallel processing.

In the PIC method, in the other hand, the computational complexity is greatly reduces by using the method of monte-carlo sampling of phase space (see e.g. (Aydemir, 1994)), meaning the larger the distribution function the larger the samples' population. The phase space samples are Lagrangian markers called superparticles (SP) which move according to equations of motion (Eq. (2)), identically as characteristics. SPs have the proper shape factors. Despite the method of characteristics back-tracing which is used in DVI (Eq. (3)), here the distribution function is reconstructed explicitly at each time by proper counting of shaped SPs:

$$f_{\alpha}(t, \mathbf{x}, \mathbf{v}) = \langle F_{\alpha}(t, \mathbf{x}, \mathbf{v}) \rangle$$

$$F_{\alpha}(t, \mathbf{x}, \mathbf{v}) = \sum_{P_{\alpha}} W(\mathbf{x} - \mathbf{x}_{p\alpha}(t)) \delta(\mathbf{v} - \mathbf{v}_{p\alpha}(t)) \quad (4)$$

where  $\langle \dots \rangle$  represents averaging over a finite spatial element, which is equivalent to the usual ensemble averaging i.e.  $f_{\alpha}(t, \mathbf{x}, \mathbf{v})$  is the macroscopic while  $F_{\alpha}(t, \mathbf{x}, \mathbf{v})$  is microscopic distribution function. Summation goes over all the SPs of type  $\alpha$ . In this way, the microscopic fluctuations occurring inside a relevant finescale (typical grid spacing) are assumed (as are in the most cases) unimportant. In fact, the key feature of PIC method versus the direct Vlasov, is the trade-off between these unimportant fluctuations and computational performance.

The charge and current densities carried by SPs are assigned to a computational grid which is used to integrate the inhomogeneous Maxwell equations in Eq. (1b) and Eq. (1c). The obtained fields are then interpolated on markers to move them according to the Lorentz-Newton equations (Eq. (8)). The typical, numerical algorithm for a flexible PIC code is shown in Fig. 1. The Initialization is done by loading external fields, sources and plasmas. It includes distributing particles according the distribution functions. Also, field initialization is done, which itself needs solving Poisson equations for electromagnetic potentials  $\mathbf{A}$  and  $\phi$  for initial electromagnetic sources. Afterward, the program enters into the time flow which contains four major steps per each time-step, for collisionless simulations: 1) Assigning sources into the computational grid, 2) advancing fields through solution of Maxwell equations using assigned sources, 3) interpolating new fields into SPs positions, and 4) moving SPs to new phase coordinates according to equations of motion. The diagnosis and desired hysteresis are taken during or at the end of the time flow, depending on their type.

### 3 Computational core and its extensibility

The main numerical solvers used in a general purpose PIC code are distribution generators, Poisson solvers, electromagnetic solvers equipped with absorbing boundary conditions, and particle pushers. In addition, sophisticated numerical schemes should be applied to source assignment and field interpolations in order to retain charge conservation and obtain a proper noise control. The full description of numerical methods applied to a general purpose PIC code like AZERAP is very extensive and out the scope of the present paper. Here, we briefly comment on some

selective topics to elucidate the numerical context of the code (See the issues mentioned on Fig. 1).

### 3.1 Poisson solver

The Poisson solver is usually a combination of Fourier/Fast Fourier analyses and sparse matrix methods applied to the inhomogeneous Helmholtz equation. For example, in 2D geometry the second order finite-difference approximation of Poisson equation is:

$$\frac{\phi_{l,j+1} - 2\phi_{l,j} + \phi_{l,j-1}}{DY^2} + \frac{\phi_{l+1,j} - 2\phi_{l,j} + \phi_{l-1,j}}{DX^2} = -\rho_{lj} \quad (5a)$$

which, applying the complex Fourier transform in  $y$  direction, reduces to the following 1D Helmholtz equation:

$$\frac{1}{DX^2} \left( \psi_{l+1,k} - 2 \left[ 1 + 2 \left( \frac{DX}{DY} \right)^2 \sin^2 \left( \frac{\pi k}{N} \right) \right] \psi_{l,k} + \psi_{l-1,k} \right) = -R_{lk} \quad (5b)$$

Therefore, we may develop a 1D Helmholtz solver accepting general boundary conditions, using the sparse matrix methods like as the three-diagonal methods and/or cyclic reduction, and then construct a general 2D Poisson solver by adding proper suits for Fourier transform in  $y$  direction.

### 3.2 Distribution generator

For each plasma case, reconstruction of position and velocity distributions is done by inverting the distribution function using the method of random sampling. The inversion is performed numerically by establishing a one-to-one correspondence between the produced uniform random numbers and the discrete evaluation of the indefinite integral of the given distribution function.

### 3.3 Maxwell solver

In the current version of AZERAP, the electromagnetic (EM) fields are advanced through the time using the well-known finite difference time domain (FDTD) or Yee method (Tajima, 2018). In this method electric and magnetic fields are staggered in time and space with respect each other. An example of Yee element in time-space and corresponding locations of different fields is shown in Fig. (2) (See e.g. (Eastwood, 1991; Yazdanpanah and Anvari, 2012)). Accordingly, the second-order numerical Yee scheme for Maxwell equations, in 2D, read as

$$\frac{\mathbf{E}_{x,l+1/2,m}^{n+1} - \mathbf{E}_{x,l+1/2,m}^n}{DT} = c^2 \frac{\mathbf{B}_{z,l+1/2,m+1/2}^{n+1/2} - \mathbf{B}_{z,l+1/2,m-1/2}^{n+1/2}}{DY} - \frac{\mathbf{J}_{x,l+1/2,m}^{n+1/2}}{\epsilon_0} \quad (6a)$$

$$\frac{\mathbf{E}_{y,l,m+1/2}^{n+1} - \mathbf{E}_{y,l,m+1/2}^n}{DT} = -c^2 \frac{\mathbf{B}_{z,l+1/2,m+1/2}^{n+1/2} - \mathbf{B}_{z,l-1/2,m+1/2}^{n+1/2}}{DX} - \frac{\mathbf{J}_{y,l,m+1/2}^{n+1/2}}{\epsilon_0} \quad (6b)$$

$$\frac{\mathbf{E}_{z,l,m}^{n+1} - \mathbf{E}_{y,l,m}^n}{DT} = c^2 \left[ \frac{\mathbf{B}_{y,l+1/2,m}^{n+1/2} - \mathbf{B}_{y,l-1/2,m}^{n+1/2}}{DX} - \frac{\mathbf{B}_{x,l,m+1/2}^{n+1/2} - \mathbf{B}_{x,l,m-1/2}^{n+1/2}}{DY} \right] - \frac{\mathbf{J}_{z,l,m}^{n+1/2}}{\epsilon_0} \quad (6c)$$

$$\frac{\mathbf{B}_{x,l,m+1/2}^{n+1/2} - \mathbf{B}_{x,l,m+1/2}^{n-1/2}}{DT} = - \frac{\mathbf{E}_{z,l,m+1}^n - \mathbf{E}_{z,l,m}^n}{DY} \quad (6d)$$

$$\frac{\mathbf{B}_{y,l+1/2,m}^{n+1/2} - \mathbf{B}_{y,l+1/2,m}^{n-1/2}}{DT} = \frac{\mathbf{E}_{z,l+1,m}^n - \mathbf{E}_{z,l,m}^n}{DX} \quad (6e)$$

$$\frac{\mathbf{B}_{z,l+1/2,m+1/2}^{n+1/2} - \mathbf{B}_{z,l+1/2,m+1/2}^{n-1/2}}{DT} = - \left[ \frac{\mathbf{E}_{y,l+1,m+1/2}^n - \mathbf{E}_{y,l,m+1/2}^n}{DX} - \frac{\mathbf{E}_{x,l+1/2,m+1}^n - \mathbf{E}_{x,l+1/2,m}^n}{DY} \right] \quad (6f)$$

where  $DX$ ,  $DY$ , and  $DT$  are finite differences along  $x$ ,  $y$  and  $t$  coordinate, respectively.  $l$ ,  $m$ , and  $n$  are  $x$ ,  $y$ , and  $t$  indices, respectively. As  $E$  and  $B$  fields are staggered by a half-step in this discretization, the  $E$  field may be first advanced using old values of  $B$ , then the new  $E$  values may be used to obtain new  $B$  fields. Using explicit boundary conditions in above formulation is trivial, but employing a free-space or open boundary conditions (when the simulation box boundaries are transparent with respect to outgoing waves) is much more complicated. In the current version of AZERAP, we use the second order Mur conditions (Mur, 1981) which is not further expanded here.

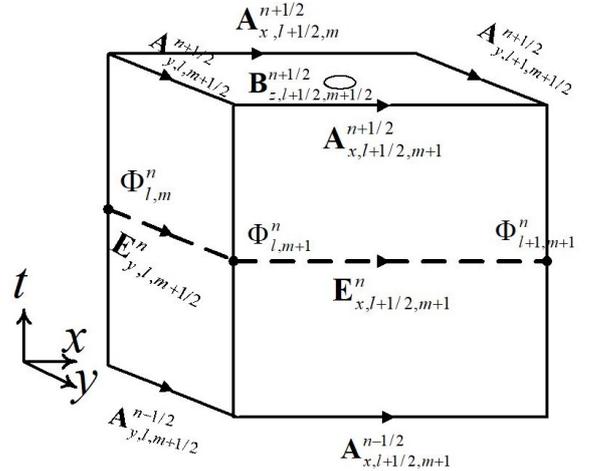


Figure 2: The space-time presentation of a 2D Yee elements and locations of nodal electromagnetic fields and potentials.

### 3.4 Particle mover

In the presence of computational grid for EM fields (self-consistent fields) Particles push is done using variants of Boris algorithm (Tajima, 2018; Hockney and Eastwood, 1988; Birdsall and Langdon, 1991). In the continuous space, in the other hand, we may use in addition the well known 4<sup>th</sup> order Runge-Kutta algorithm. The Boris algorithm is based on the second-order leapfrog

(midpoint) scheme, which produces the semi-explicit discretized Lorentz-Newton equations:

$$\frac{\mathbf{X}^{n+1} - \mathbf{X}^n}{DT} = \mathbf{V}^{n+1/2} \quad (7a)$$

$$\frac{\mathbf{u}^{n+1/2} - \mathbf{u}^{n-1/2}}{DT} = \frac{q}{m} \left( \varepsilon^n + \frac{\mathbf{u}^n}{\gamma^n} \times \mathbf{b}^n \right) \quad (7b)$$

where  $\mathbf{u} \equiv \mathbf{p}/m$  is the mass normalized momentum and  $\gamma = (1 + \mathbf{u} \cdot \mathbf{u}/c^2)^{1/2}$  is the relativistic factor.  $\varepsilon^n$  and  $\mathbf{b}^n$  are electric and magnetic field values at  $n^{\text{th}}$  time level on the particle position. The different variants of Boris algorithm rely on different definition of mid-point values  $\mathbf{u}^n$  and  $\gamma^n$  in terms of  $\mathbf{u}^{n-1/2}$  and  $\mathbf{u}^{n+1/2}$ . In the traditional Boris form,  $\mathbf{u}^n/\gamma^n$  is written as  $\mathbf{u}^n/\gamma^n \equiv (\mathbf{u}^{n-1/2} + \mathbf{u}^{n+1/2})/2\gamma^{n-1/2}$ . In the modern Vay (Vay, 2008) method, it is written as  $\mathbf{u}^n/\gamma^n \equiv (\mathbf{u}^{n-1/2} + \mathbf{u}^{n+1/2})/(\gamma^{n-1/2} + \gamma^{n+1/2})$ , and in the Higuera-Cary (Higuera and Cary, 2017) as  $\mathbf{u}^n/\gamma^n \equiv (\mathbf{u}^{n-1/2}/\gamma^{n-1/2} + \mathbf{u}^{n+1/2}/\gamma^{n+1/2})/2$ . As the mid-point momentum is better approximated, the latter two are demonstrated to be more accurate specially at relativistic velocities. In the simplest Boris case, the implicit Eq. (7b) may be converted to an explicit one by separating its translational and rotation parts. To do this, the auxiliary momentum quantities  $\mathbf{u}_1^*$  and  $\mathbf{u}_2^*$  are defined such that

$$\begin{aligned} \mathbf{u}_1^* &= \mathbf{u}^{n-1/2} + \varepsilon^n \frac{DT}{2} \\ \mathbf{u}_1^{n+1/2} &= \mathbf{u}_2^* + \varepsilon^n \frac{DT}{2} \end{aligned} \quad (8)$$

By substituting these definitions into Eq. (7b) we get the fully rotational equation  $\mathbf{u}_2^* = \mathbf{u}_1^* + (\mathbf{u}_1^* + \mathbf{u}_2^*) \times \theta^n$  where  $\theta^n \equiv (qDT/2m\gamma^{n-1/2})\mathbf{b}^n$ . Now if we take the cross product of this equation with  $\theta^n$  and eliminate the term  $\mathbf{u}_2^* \times \theta^n$  in its right hand side via obtained equation, after some mathematical manipulations, we end up with

$$\mathbf{u}_2^* = \mathbf{u}_1^* + \frac{2(\mathbf{u}_1^* + \mathbf{u}_1^* \times \theta^n) \times \theta^n}{1 + \theta^n \cdot \theta^n} \quad (9)$$

which upon using Eq. (8) gives an explicit equations for  $\mathbf{u}^{n+1/2}$  in terms of  $\mathbf{u}^{n-1/2}$ . Another important feature of Boris algorithm, is possibility of enhancing its accuracy in the presence of strong magnetic fields and finite time steps, i.e. when  $\theta^n \equiv (qDT/2m\gamma^{n-1/2})\mathbf{b}^n$  is computationally large. This is done by noting that the purely rotational equation  $\mathbf{u}_2^* = \mathbf{u}_1^* + (\mathbf{u}_1^* + \mathbf{u}_2^*) \times \theta^n$  admits an analytical solution. The details may be found in the literature (Hockney and Eastwood, 1988). Here we give the result; the present  $\theta^n$  is replaced by  $\theta^n \tan(\theta^n) = \mathbf{b}^n \tan(\theta^n)$  in Eq. (9).

### 3.5 Force interpolation

Now let discuss briefly, the concepts of field interpolation on SP locations and assignment of particle sources to computational grid. Different schemes have been suggested for field interpolations. The two well known cases are the so-called energy and momentum conservation schemes. Especially, the energy conservation scheme is very popular and used extensively, due to very good overall energy conservation. It has been argued by Eastwood (Eastwood,

1991) that in this case, the interpolation functions for different fields are produced from a single generating function  $H_{l,m}^{n+1/2}(t, \mathbf{x}) = \Lambda^{n+1/2}(t)\Lambda_l(x)\Lambda_m(y)$  where the triangle-shape function is defined as

$$\begin{aligned} \Lambda_l(x) &\equiv \Lambda(x - x_l) \\ &\equiv \begin{cases} 1 - \frac{|x - x_l|}{DX} & \text{if } |x - x_l| \leq DX \\ 0 & \text{if } |x - x_l| > DX \end{cases} \end{aligned} \quad (10)$$

Interpolations for different fields are obtained from this generating function by taking proper time/space derivatives. For example for the TM fields we have:

$$e_x(t, x, y) = \sum_{n,l,m} E_{x,l+1/2,m}^n \Pi^n(t) \Pi_{l+1/2}(x) \Lambda_m(y) \quad (11a)$$

$$e_y(t, x, y) = \sum_{n,l,m} E_{y,l,m+1/2}^n \Pi^n(t) \Lambda_l(x) \Pi_{m+1/2}(y) \quad (11b)$$

$$\begin{aligned} b_z(t, x, y) &= \sum_{n,l,m} B_{z,l+1/2,m+1/2}^{n+1/2} \\ &\quad \times \Lambda^{n+1/2}(t) \Pi_{l+1/2}(x) \Pi_{m+1/2}(y) \end{aligned} \quad (11c)$$

where the flattop function is defined as,

$$\Pi_l(x) \equiv \Pi(x - x_l) \equiv \begin{cases} 1 & \text{if } |x - x_l| \leq DX \\ 0 & \text{if } |x - x_l| > DX \end{cases} \quad (12)$$

However, recent studies on modern PIC codes (Pukhov, 1999) have demonstrated enhanced accuracy by replacing the flattop function with the smoother triangle shape function.

### 3.6 Charge assignment

The current and charge assignments should ensure the satisfaction of the charge conservation (continuity equation) (Villasenor and Buneman, 1992). Otherwise, the Poisson equation should be solved at each time step to eliminate the error electric field accumulated due to the conservation violating current in Eqs. (6a) to (6c). As solving Poisson equation is usually non-local, it is hard to parallelize and is whence time consuming. Therefore, the charge conservation current assignment methods are always preferred. The charge conservation assignment schemes are best understood in terms of SP time-space shape factor (see (Yazdanpanah and Anvari, 2012)). The current and charge densities produced by SPs with shape factor  $S(t' - t, \mathbf{x}' - \mathbf{x}_p(t))$  at each space-time point  $(t', \mathbf{x}')$  are given by

$$\rho(t', \mathbf{x}') = \sum_p q_p \int dt S(t' - t, \mathbf{x}' - \mathbf{x}_p(t)) \quad (13a)$$

$$\mathbf{j}(t', \mathbf{x}') = \sum_p q_p \int dt [\mathbf{v}_p(t) S(t' - t, \mathbf{x}' - \mathbf{x}_p(t))] \quad (13b)$$

where summation is over all SPs. The nodal values of current and charge used in solving Maxwell equations (Eqs.

(6a) to (6c)) are assigned after integration of these equations over the grid element. Therefore, we would have,

$$\mathbf{Q}_{l,m}^n = \int_{l-1/2}^{l+1/2} dx' \int_{m-1/2}^{m+1/2} dy' \rho(t^n, \mathbf{x}') \quad (14a)$$

$$\mathbf{J}_{x,l+1/2,m}^{n+1/2} = \int_n^{n+1} dt' \int_{m-1/2}^{m+1/2} dy' j_x(t', x'_{l+1/2}, y') \quad (14b)$$

$$\mathbf{J}_{y,l,m+1/2}^{n+1/2} = \int_n^{n+1} dt' \int_{l-1/2}^{l+1/2} dx' j_y(t', x', y'_{m+1/2}) \quad (14c)$$

Using the popular shape function  $S(t' - t, \mathbf{x}' - \mathbf{x}_p, \mathbf{y}' - \mathbf{y}_p) = \delta(t' - t)\Pi(\mathbf{x}' - \mathbf{x}_p)\Pi(\mathbf{y}' - \mathbf{y}_p)$  one may obtain (Yazdanpanah and Anvari, 2012)

$$\mathbf{Q}_{l,m}^n = \sum_p q_p \int dt F_{t,l,m}^n(t, \mathbf{x}_p) \quad (15a)$$

$$\mathbf{J}_{x,l+1/2,m}^n = \sum_p q_p \int dt \left[ F_{x,l+1/2,m}^n(t, \mathbf{x}_p(t)) v_{p,x}(t) \right] \quad (15b)$$

$$\mathbf{J}_{y,l,m+1/2}^n = \sum_p q_p \int dt \left[ F_{y,l,m+1/2}^n(t, \mathbf{x}_p(t)) v_{p,y}(t) \right] \quad (15c)$$

where  $F_{t,l,m}^n(t, x, y) = \delta^n(t)\Lambda_l(x)\Pi_m(y)$ ,  $F_{x,l+1/2,m}^{n+1/2}(t, x, y) = \Pi^{n+1/2}(t)\Pi_{l+1/2}(x)\Lambda_m(y)$ , and  $F_{y,l,m+1/2}^{n+1/2}(t, x, y) = \Pi^{n+1/2}(t)\Lambda_l(x)\Pi_{m+1/2}(y)$ . With this choice of shape-function, the obtained assignment is identical to that of Villasenor and Buneman (Villasenor and Buneman, 1992).

### 3.7 Accuracy and Stability

Accuracy and stability of the computational core used in the first code-release, summarized in above sections, have been fully discussed in my earlier works (Yazdanpanah and Anvari, 2012, 2014; Yazdanpanah, 2017, 2019). Especially, its high accuracy has been demonstrated through two standard methods: 1) Successful examination of global conservation laws (e.g. total charge and energy) during the long term simulations (Yazdanpanah and Anvari, 2012), 2) Successful confirmation of simulation results by comparison with the exact analytical solutions for problems admitting such solutions (e.g. plane wave interaction with plasma, single particle motion, electromagnetic pulse propagation in vacuum, etc.) (Yazdanpanah and Anvari, 2012, 2014; Yazdanpanah, 2017, 2019). Here, we briefly discuss the main remarks on computational stability and accuracy of the solvers discussed above.

In the fully relativistic regime the stability analyses become cumbersome and are out of the scope of the present paper. Nonetheless, very good insights are obtained from non-relativistic analyses which may be generalized to the relativistic regime. The stability analysis is usually done

via the so-called method of error amplification matrix (Hockney and Eastwood, 1988). In the case of linear equations, this matrix is identical to the transformer matrix which relates the old and new quantities during a time-step. In the case of motion equations (7a) and (7b) (or Eq. (9)), when electric field is assumed zero, magnetic field in  $z$  direction, and motion non-relativistic, it may be shown that these equations could be represented by the following matrix form:

$$\begin{pmatrix} u_x^{n+1/2} \\ u_y^{n+1/2} \\ x^n \\ y^n \end{pmatrix} = \mathbf{M}_T \begin{pmatrix} u_x^{n-1/2} \\ u_y^{n-1/2} \\ x^{n-1} \\ y^{n-1} \end{pmatrix} \quad (16a)$$

where  $\mathbf{M}_T$  is the transformer matrix given by,

$$\mathbf{M}_T = \begin{pmatrix} (1 - \theta^2)/(1 + \theta^2) & 2\theta/(1 + \theta^2) & 0 & 0 \\ -2\theta/(1 + \theta^2) & (1 - \theta^2)/(1 + \theta^2) & 0 & 0 \\ DT & 0 & 1 & 0 \\ 0 & DT & 0 & 1 \end{pmatrix} \quad (16b)$$

Also, it easily follows by

$$\begin{pmatrix} u_x^{n+1/2} \\ u_y^{n+1/2} \\ x^n \\ y^n \end{pmatrix} = \mathbf{M}_T^n \begin{pmatrix} u_x^{1/2} \\ u_y^{1/2} \\ x^0 \\ y^0 \end{pmatrix} \quad (16c)$$

The stability of the scheme is determined by eigenvalues of the transformer matrix  $\mathbf{M}_T$  which are easily determined to be

$$\lambda_{1,2} = \frac{1 - \theta^2}{1 + \theta^2} \pm 2i \frac{|\theta|}{1 + \theta^2}, \quad \lambda_3 = 1 \quad (17)$$

It may be easily shown that  $|\lambda_{1,2}| = 1$ , therefore the rotational part of Boris scheme is unconditionally stable. Furthermore, unity norm of eigenvalues indicates that norm of velocity ( $|v^2|$ ) remains constant, i.e. energy conservation during the time iteration.

Moreover, based on Eqs. (16a), (16b), (16c), and (17), we farther may perform an accuracy analysis on the Boris solver. To do this, we note that Eq. (16c) admits spectral solutions in the form  $\mathbf{u}^{n+1/2} = e^{\pm i\Omega_n DT} \mathbf{u}^{1/2}$ , where  $\Omega_c$  is the numerical cyclotron frequency which is determined from  $e^{\pm i\Omega_c DT} = \lambda_{1,2}$ , giving the following equation:

$$\tan(\Omega DT) = \frac{2|\theta|}{1 - \theta^2} = \frac{\omega_c DT}{1 - (\omega_c DT/2)^2} \quad (18)$$

where  $\omega_c = qB/m$  is the exact cyclotron frequency. This equation indicates that computational frequency matches the physical frequency at small  $DT$ . Also, by Taylor expansion of  $\tan(\Omega_c DT)$ , the second order accuracy of the scheme is recovered.

When the set of equations could be ordered into a single equation, it is possible to perform the instability and accuracy analyses at once within the spectral analyses, i.e. the transformer matrix is replaced by a single scalar. This is the case, for example in analysis of a charged particle motion in a parabolic potential or in the case of Maxwell

**Table 1:** Summary of solvers/methods implemented in the current version of AZERAP and their extensibility offered by the code design.

Issue	Current Implemented Methods/Solvers	Supported Extensibility
Maxwell Solver	2D Standard Finite Difference Time Domain	High order FDTD/Generalization to 3D
Motion Solver	Relativistic Boris, Cary-Higuera, Vay in presence of mesh, +RK4 in mesh free	Generalization to allow RK4 in presence of mesh
Poisson Solver	FFT, FACR and SOR Chebyshev Acc eleration	Generalization to 3D
Fields Boundary Conditions	Second order Absorbing Mur	Implementing with PML
Source assignment	Villasenor Buneman charge conserving	Generalization to arbitrary particle shape
Force Inter polation	Selectable composition of Linear and Constant	Generalization to arbitrary interpolation schemes

equations collapsed into the wave equation. The latter case is trivially related to examination of Eqs. (6a) to (6f) and the former encountered in the case of translational motion of a charged particle in the electric field (the translational part of the Boris algorithm, see e.g. (Hockney and Eastwood, 1988)). We do not repeat the calculations here, as the procedure is more or less same as one performed above and may be easily found in literatures (e.g. (Hockney and Eastwood, 1988; Birdsall and Langdon, 1991)); we only give the final results:

In the case of translational motion of charged particle in electric fields as a part of Boris motion, the general stability analysis is difficult/impossible. However, the most error pileup may be expected when particle performs harmonic motion in a static parabolic potential (a static dipole electric field). In this case we get (Hockney and Eastwood, 1988),

$$\sin\left(\frac{\Omega_p DT}{2}\right) = \frac{\omega_p DT}{2} \quad (19a)$$

where  $\omega_p$  and  $\Omega_p$  are physical and numerical frequency of the charge particle inside the electric potential. This indicates that time step must obey the following relation to guarantee the stability

$$\omega_p DT \leq 2 \quad (19b)$$

Also, Taylor expansion of Eq. (19a) reveals the second order accuracy of the motion solver in terms of frequency. Nonetheless, Eq. (19b) is a less restrictive condition when compared to the stability condition for Maxwell solver commented bellow. In the case of Maxwell equations (Eqs. (6a) to (6f)) we get the following numerical dispersion relation (relating the numerical frequency  $\Omega$  to wave number  $k$ ) through spectral analyses (see e.g. (Birdsall and Langdon, 1991)):

$$\sin^2\left(\frac{\Omega DT}{2}\right) = c^2 DT^2 \left[ \frac{1}{DX^2} \sin^2\left(\frac{k_x DX}{2}\right) + \frac{1}{DY^2} \sin^2\left(\frac{k_y DY}{2}\right) \right] \quad (20a)$$

It follows that scheme is stable under the so-called Courant-Friedrichs-Lewy (CFL) condition

$$cDT \sqrt{\frac{1}{DX^2} + \frac{1}{DY^2}} < 1 \quad (20b)$$

In the above discussions, we have treated the radiation and particles separately, i.e. collective behaviors of plasma

model not included. When particles are subjected to longitudinal plasma modes, comprehensive analyses indicate (see e.g. (Birdsall and Langdon, 1991)) that wave-particle interactions impose limitation over spatial differences:

$$\frac{\{DX, DY\}}{\lambda_D} \leq 1 \quad (21)$$

Violation of Eq. (21) leads to the so-called finite-grid instability. This may be the most sever restriction on spatial differences when the initial plasma temperature is not so high (of order of electron volts or less). Otherwise, when high initial temperatures are considered, this instability will disappear. In practice, when Eq. (21) is violated, the plasma temperature quickly rises to a level that  $\lambda_D = \sqrt{k_B T_e / ne^2}$  equals the spatial differences. Using high order super-particle shapes, as included in the future extensions of the code, are very effective in mitigating this noise amplifying effect at low temperatures.

### 3.8 Extensibility

In Table 1, a summary of methods/solvers implemented in the present version of AZERAP is given, according to issues described above. In addition, the possible extensions allowed by the code design and libraries are listed for given issues. For example, in near future we aim to implement the full 3D PIC algorithm and predicted the necessary flexibilities in writing the base libraries of the code. In addition, we aim to implement the high stencil order FDTD schemes for solution of EM fields to improve the dispersion properties of wave propagation necessary for simulating ultra-relativistic velocities. Further, we planned for future using of Runge-Kutta solvers for particles in presence of self-consistent fields (computational mesh) along with high time order Maxwell solvers.

## 4 Coding structure and implementation

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. 3. Especially, the top layers like memory structures, interpreter, MPI manager, etc are prepared general purpose and admit the maximum flexibility for future developments. This makes it possible to include these components in other code applications

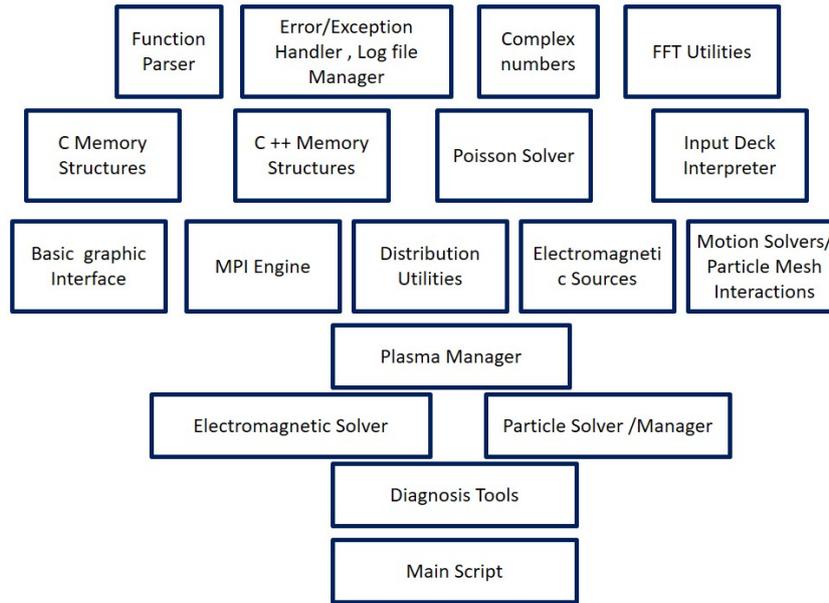


Figure 3: AZERAP architecture in terms of its components and their hierarchy.

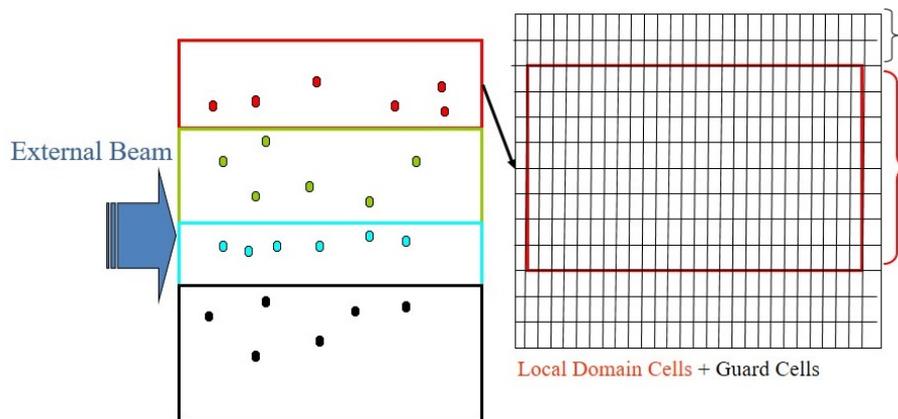


Figure 4: An illustration of domain decomposition (patch-based) parallelization; points represent super-particles.

with different purposes. The code is implemented in object oriented C++ language. Working objects like fields and plasmas are created by instantiation of the abstract objects by using encapsulated primary objects like grid object, mover objects, etc.

AZERAP utilizes Message Passing Interface (MPI) for parallelization, which is suitable for the distributed memory resources. The main parallelization (during the time loop) is based on domain decomposition, i.e. patch-based parallelization. The spatial domain of the simulation is decomposed into sub-domains via divisions in both x and y directions (2D) or divisions in one of these directions (1D). The local field data at different domains are patched together via guard cell messages. Particles may pass through different domains via appropriate message. An illustration of domain decomposition in 1D is given in Fig. 4. Functions are available to change the domain partition during the execution to keep the load balance. All MPI routines are developed based on the blocking send and receive func-

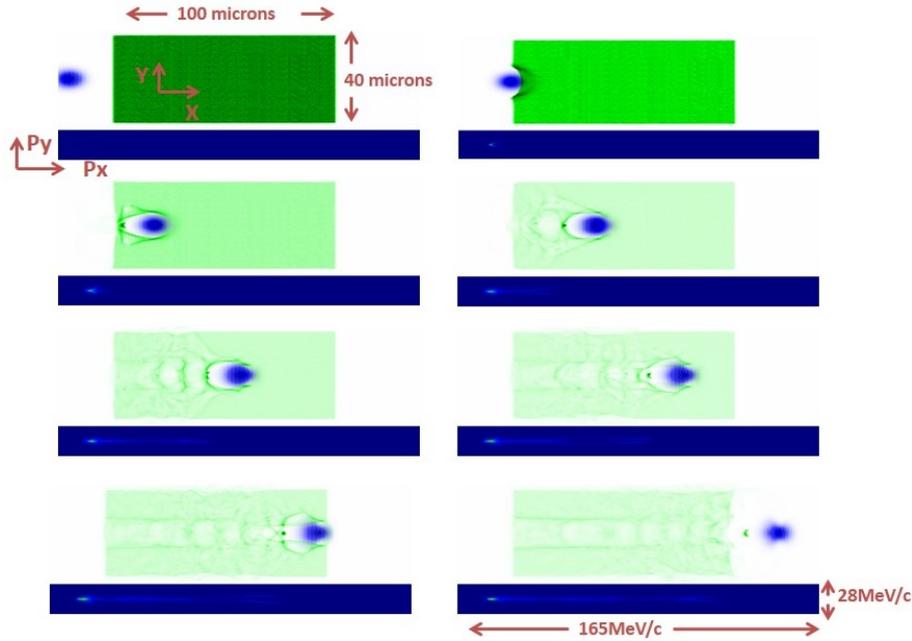
tions and are designed to keep the maximum concurrency among different messages passing between different peer-processes in the communicator.

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. The exception is using a brief open source C++ function parser library *fparser* (<http://warp.povusers.org/FunctionParser/> and <https://github.com/thliebig/fparser>) which is included in the code package in a distinct directory. Having no external linkage leads to a very easy installation of the code and makes it possible to be compiled in a single stage and produces a single executable.

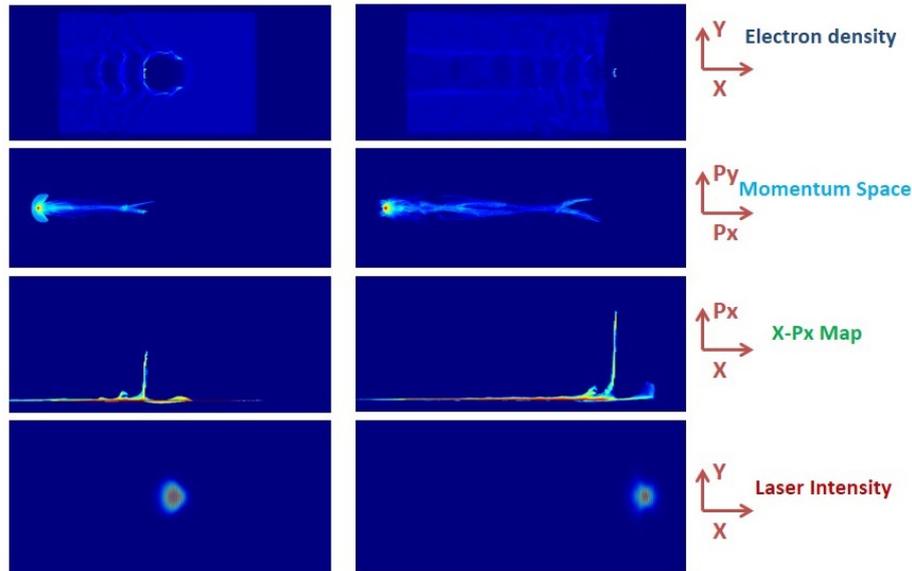
The scripting language (the standard of input deck programming) is developed to be in the most natural way and supports functional dialog. It uses *fparser* to parse the input function strings and convert them into the built in functions as a part of token list. To perform a com-

**Table 2:** Summary of important parameters used in different computer experiments.

Experiment No.	Plasma density ( $\text{cm}^{-3}$ )	Laser Spot Size ( $\mu\text{m}$ )	Laser Pulse Length (fs)	Laser Wavelength ( $\mu\text{m}$ )	Laser Intensity ( $\text{W}\cdot\text{cm}^{-2}$ )
1	$2.28 \times 10^{19}$	4	28.3	1	$2.7 \times 10^{20}$
2	$2.28 \times 10^{19}$	4	28.3	1	$5.5 \times 10^{18}$
3	$2.28 \times 10^{20}$	4	28.3	1	$2.7 \times 10^{20}$



**Figure 5:** A time sequence of frames chosen from the simple graphical interface of AZERAP (animated overview of experiment) , showing snapshots of laser (blue) electron- plasma (green) plus the phase space record of electrons (dark blue screen), all for experiment no. 1.



**Figure 6:** Detailed views of electron density (first row), momentum space (second row), x-px phase slice (third row) and laser intensity (fourth row) at two different simulation times, for experiment no. 1.

puter 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 exe-

cuted.

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 ani-

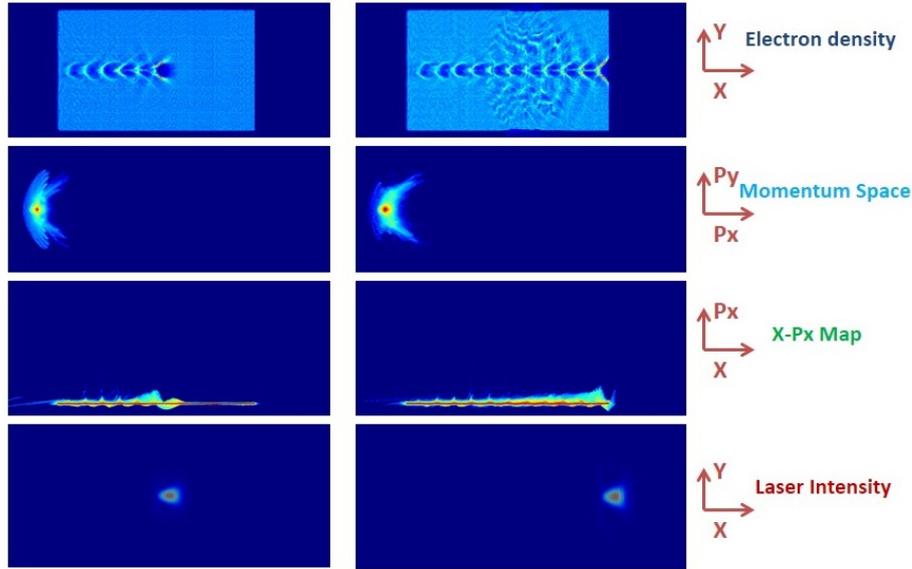


Figure 7: Same as Fig. 6 but for experiment no. 2.

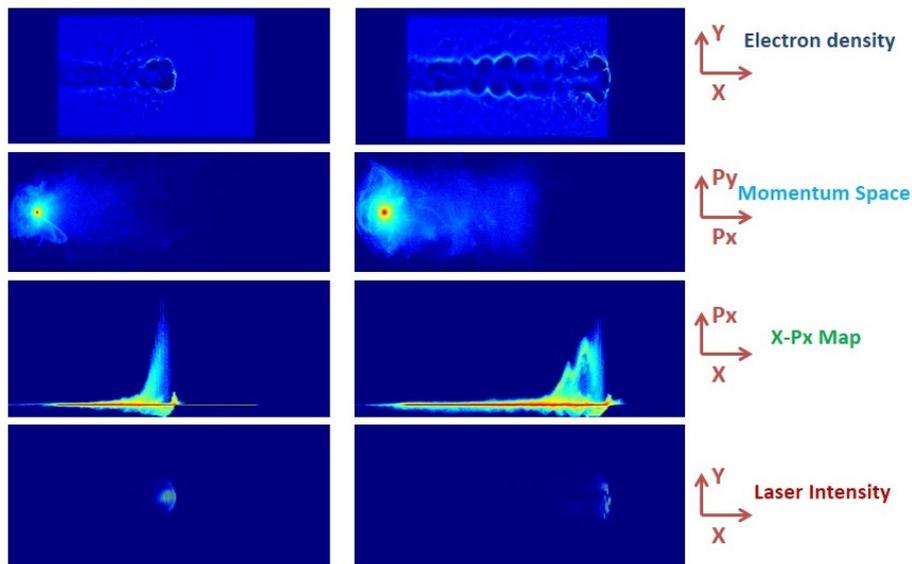


Figure 8: Same as Fig. 6 but for experiment no. 3.

mation. 2) It is possible to use test sample particles to easily understand the physics behind the particle dynamics while avoiding search in extremely large data of plasma particles.

### 5 Code operation examples

Here, three examples of computer experiments performed by AZERAP, on the important scenario of laser plasma accelerator (LPA) (see e.g. (Lu et al., 2007; Pukhov et al., 2004)), are presented. The quantitative discussion of results is not intended, but rather we aim to demonstrate the code operation and importance of such simulations in discovering the proper parameter space for operation of the LPA. The important parameters of these experiments are summarized in Table 2. In Fig. 5, a time-sequence of

snapshots of computer experiment no. 1 is presented from the simple user graphical interface of the code. Propagating rightward, a strong laser pulse (blue color) enters into a column of plasma (green color) from the left, excites an accelerating micro-cavity structure very similar to the conventional accelerators, but rather more compact in size (less than  $1/1000$ ). The produced structure accelerates an intense bunch of electrons initially trapped from the plasma background. This electron bunch gets an energy-gain of order of 100MeV and eventually exits from the right as a trail following the laser pulse.

As, the simple graphical interface, is usually intended to present the overview/main features of the experiment, more detailed outputs from other desired properties may be requested by user through the input deck. In Fig. 6 we show examples of these outputs, including plasma map

(first row), electron momentum space (second row), electron longitudinal phase space (third row) and laser intensity map (fourth row) at two different times, for experiment no. 1. These maps and phase-space slices are very important in understanding the performance of LPA, and are usually used to infer different LPA schemes and regimes. The data outlined in Figs. 5 and 6 manifest the characters of the so called bubble regime (Pukhov et al., 2004).

To emphasize the importance of simulations in assessment of different LPA schemes, we repeat the same data as Fig. 6 for different laser intensity (experiment no. 2, Fig. 7) and plasma density (experiment no. 3, Fig. 8). These results show that when the laser field amplitude is reduced almost by an order of magnitude (intensity reduced by two orders of magnitude) (Fig. 7) the accelerated beam does not form and we only observe a mild plasma heating. On the other hand, when the plasma density increases by an order of magnitude (but still quite transparent to the laser pulse) (Fig. 8), no isolated accelerated beam is formed but instead, we observe strong volumetric plasma heating.

## 6 Conclusions and final remarks

In conclusion, we have outlined the properties and advanced features of the new domestic PIC code AZERAP. We also presented example simulations of laser plasma accelerator using this code and discussed the importance of such simulations in discovering the proper LWFA parameter space and assessment of different schemes of LPA. As the final remark, it should be mentioned that the current version of the code is available to researchers through scientific collaboration. A version of the code will be documented and uploaded to a public repository (like as GitHub) under the GNU public license in the near future.

## Conflict of Interest

The authors declare no potential conflict of interest regarding the publication of this work.

## References

- Assmann, R., Weikum, M., Akhter, T., et al. (2020). Eu-praxia conceptual design report. *The European Physical Journal Special Topics*, 229(24):3675–4284.
- Aydemir, A. Y. (1994). A unified Monte Carlo interpretation of particle simulations and applications to non-neutral plasmas. *Physics of Plasmas*, 1(4):822–831.
- Birdsall, C. and Langdon, A. (1991). Plasma Physics via Computer simulation (Bristol, UK). *IOP Publishing*, 11610:0750301171.
- Brown, I. G. (2004). *The physics and technology of ion sources*. John Wiley & Sons.
- Derouillat, J., Beck, A., Pérez, F., et al. (2018). Smilei: A collaborative, open-source, multi-purpose particle-in-cell code for plasma simulation. *Computer Physics Communications*, 222:351–373.
- Eastwood, J. W. (1991). The virtual particle electromagnetic particle-mesh method. *Computer Physics Communications*, 64(2):252–266.
- Esarey, E., Schroeder, C., and Leemans, W. (2009). Physics of laser-driven plasma-based electron accelerators. *Reviews of Modern Physics*, 81(3):1229.
- Fonseca, R. A., Silva, L. O., Tsung, F. S., et al. (2002). OSIRIS: A three-dimensional, fully relativistic particle in cell code for modeling plasma based accelerators. In *International Conference on Computational Science*, pages 342–351. Springer.
- Gschwendtner, E., Adli, E., Amorim, L., et al. (2016). AWAKE, the advanced proton driven plasma wakefield acceleration experiment at CERN. *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment*, 829:76–82.
- Higuera, A. V. and Cary, J. R. (2017). Structure-preserving second-order integration of relativistic charged particle trajectories in electromagnetic fields. *Physics of Plasmas*, 24(5):052104.
- Hockney, R. and Eastwood, J. (1988). *Computer Simulation Using Particles* Taylor & Francis. Inc., USA.
- Horioka, K. (2018). Progress in particle-beam-driven inertial fusion research: Activities in Japan. *Matter and Radiation at Extremes*, 3(1):12–27.
- Kawata, S. (2021). Direct-drive heavy ion beam inertial confinement fusion: a review, toward our future energy source. *Advances in Physics: X*, 6(1):1873860.
- Khalilzadeh, E., Yazdanpanah, J., Jahanpanah, J., et al. (2015). Electron residual energy due to stochastic heating in field-ionized plasma. *Physics of Plasmas*, 22(11):113115.
- Lu, W., Tzoufras, M., Joshi, C., et al. (2007). Generating multi-GeV electron bunches using single stage laser wakefield acceleration in a 3D nonlinear regime. *Physical Review Special Topics-Accelerators and Beams*, 10(6):061301.
- Mur, G. (1981). Absorbing boundary conditions for the finite-difference approximation of the time-domain electromagnetic-field equations. *IEEE Transactions on Electromagnetic Compatibility*, (4):377–382.
- Nieter, C. and Cary, J. R. (2004). VORPAL: a versatile plasma simulation code. *Journal of Computational Physics*, 196(2):448–473.
- Pishdast, M., Yazdanpanah, J., and Ghasemi, S. (2018). Electron acceleration by an intense laser pulse inside a density profile induced by non-linear pulse evolution. *Laser and Particle Beams*, 36(1):41–48.
- Pukhov, A. (1999). Three-dimensional electromagnetic relativistic particle-in-cell code VLPL (Virtual Laser Plasma Lab). *Journal of Plasma Physics*, 61(3):425–433.
- Pukhov, A., Gordienko, S., Kiselev, S., et al. (2004). The bubble regime of laser-plasma acceleration: monoenergetic electrons and the scalability. *Plasma Physics and Controlled Fusion*, 46(12B):B179.

Sonnendrücker, E., Roche, J., Bertrand, P., et al. (1999). The semi-Lagrangian method for the numerical resolution of the Vlasov equation. *Journal of Computational Physics*, 149(2):201–220.

Tajima, T. (2018). *Computational plasma physics: with applications to fusion and astrophysics*. CRC press.

Tajima, T., Yan, X., and Ebisuzaki, T. (2020). Wakefield acceleration. *Reviews of Modern Plasma Physics*, 4(1):1–72.

Toigo, V., Piovan, R., Dal Bello, S., et al. (2017). The PRIMA Test Facility: SPIDER and MITICA test-beds for ITER neutral beam injectors. *New Journal of Physics*, 19(8):085004.

Vay, J.-L. (2008). Simulation of beams or plasmas crossing at relativistic velocity. *Physics of Plasmas*, 15(5):056701.

Verboncoeur, J. P., Langdon, A. B., and Gladd, N. (1995). An object-oriented electromagnetic PIC code. *Computer Physics Communications*, 87(1-2):199–211.

Villasenor, J. and Buneman, O. (1992). Rigorous charge conservation for local electromagnetic field solvers. *Computer Physics Communications*, 69(2-3):306–316.

Yakimenko, V., Alsberg, L., Bong, E., et al. (2019). FACET-II facility for advanced accelerator experimental tests. *Physical Review Accelerators and Beams*, 22(10):101301.

Yazdanpanah, J. (2017). Nonlinear evolutions of an ultra-intense ultra-short laser pulse in a rarefied plasma through a new quasi-static theory. *Plasma Physics and Controlled Fusion*, 60(2):025014.

Yazdanpanah, J. (2019). Self modulation and scattering instability of a relativistic short laser pulse in an underdense plasma. *Plasma Physics and Controlled Fusion*, 61(8):085021.

Yazdanpanah, J. and Anvari, A. (2012). Time and space extended-particle in cell model for electromagnetic particle algorithms. *Physics of Plasmas*, 19(3):033110.

Yazdanpanah, J. and Anvari, A. (2014). Effects of initially energetic electrons on relativistic laser-driven electron plasma waves. *Physics of Plasmas*, 21(2):023101.

©2023 by the journal.

RPE is licensed under a [Creative Commons Attribution-NonCommercial 4.0 International License](https://creativecommons.org/licenses/by-nc/4.0/) (CC BY-NC 4.0).



#### To cite this article:

Yazdanpanah, J. (2023). Introducing the domestic multi-dimensional particle-in-cell code AZERAP. *Radiation Physics and Engineering*, 4(1), 1-12.

DOI: [10.22034/rpe.2022.341201.1086](https://doi.org/10.22034/rpe.2022.341201.1086)

To link to this article: <https://doi.org/10.22034/rpe.2022.341201.1086>