PAPER

2D axial-azimuthal particle-in-cell benchmark for low-temperature partially magnetized plasmas

To cite this article: T Charoy *et al* 2019 *Plasma Sources Sci. Technol.* **28** 105010

View the article online for updates and enhancements.

2D axial-azimuthal particle-in-cell benchmark for low-temperature partially magnetized plasmas

T Charoy[1] ⓘ, J P Boeuf[2] ⓘ, A Bourdon[1] ⓘ, J A Carlsson[3] ⓘ, P Chabert[1],
B Cuenot[4] ⓘ, D Eremin[5], L Garrigues[2] ⓘ, K Hara[6,12] ⓘ, I D Kaganovich[7] ⓘ,
A T Powis[8], A Smolyakov[9] ⓘ, D Sydorenko[9,10], A Tavant[1,11] ⓘ,
O Vermorel[4] ⓘ and W Villafana[4,11]

[1] Laboratoire de Physique des Plasmas, CNRS, Ecole polytechnique, Sorbonne Université, Université Paris-Sud, Observatoire de Paris, Université Paris-Saclay, PSL Research University, F-91128 Palaiseau, France
[2] LAPLACE, Université de Toulouse, CNRS, F-31062 Toulouse, France
[3] RadiaSoft LLC, Boulder, CO 80301, United States of America
[4] CERFACS—42, avenue Gaspard Coriolis, F-31057 Toulouse, France
[5] Ruhr University Bochum, Universitaetsstrasse 150, D-44801 Bochum, Germany
[6] Texas A&M University, College Station, TX 77843, United States of America
[7] Princeton Plasma Physics Laboratory, Princeton, NJ 08540, United States of America
[8] Princeton University, Princeton, NJ 08544, United States of America
[9] Department of Physics and Engineering Physics, University of Saskatchewan, 116 Science Place, Saskatoon SK S7N 5E2, Canada
[10] University of Alberta, Edmonton, Alberta T6G 2E1, Canada
[11] Safran Aircraft Engines, F-27208 Vernon, France
[12] Stanford University, Stanford, CA, 94305, United States of America

E-mail: thomas.charoy@lpp.polytechnique.fr

CrossMark

Abstract

The increasing need to demonstrate the correctness of computer simulations has highlighted the importance of benchmarks. We define in this paper a representative simulation case to study low-temperature partially-magnetized plasmas. Seven independently developed particle-in-cell codes have simulated this benchmark case, with the same specified conditions. The characteristics of the codes used, such as implementation details or computing times and resources, are given. First, we compare at steady-state the time-averaged axial profiles of three main discharge parameters (axial electric field, ion density and electron temperature). We show that the results obtained exhibit a very good agreement within 5% between all the codes. As $E \times B$ discharges are known to cause instabilities propagating in the direction of electron drift, an analysis of these instabilities is then performed and a similar behaviour is retrieved between all the codes. A particular attention has been paid to the numerical convergence by varying the number of macroparticles per cell and we show that the chosen benchmark case displays a good convergence. Detailed outputs are given in the supplementary data, to be used by other similar codes in the perspective of code verification.

Supplementary material for this article is available online

Keywords: $E \times B$ discharges, benchmark, particle-in-cell, electron drift instability

1. Introduction

In different applications of low-temperature plasmas, such as Hall Thrusters for electric propulsion [1, 2] or magnetron discharges for plasma processing [3, 4], the gas pressure is relatively low (typically between 0.1 and 10 mTorr) and the plasma is confined by a magnetic field to enhance ionization. This external static magnetic field is imposed in the direction perpendicular to the electric field from the cathode to the anode, and hence a cross-field drift is induced in the $E \times B$ direction (E is the electric field and B the imposed magnetic field). For an efficient plasma confinement, the $E \times B$ direction is closed, corresponding to the azimuthal direction in a cylindrical geometry. The main difference with fusion plasmas is that the electrons are strongly magnetized, while the ions are not (the ion Larmor radius is much bigger than the plasma dimensions), which is the reason why these plasmas are often called 'partially magnetized plasmas'.

The presence of the magnetic field can trigger many fluctuations in $E \times B$ discharges thus increasing significantly the physics complexity, and, in particular, resulting in electron cross-field mobility several orders of magnitude higher than the expected classical collisional mobility. Combinations of gradients of plasma density, temperature and magnetic field, electron $E \times B$ drift, ionization and collisions can all be sources of fluctuations in various regions of the discharges [5, 6]. Recently, the kinetic instability due to strong electron drift, often called Electron Cyclotron Drift Instability (ECDI) [7] has attracted much attention as a possible source of the anomalous electron transport in Hall Thrusters [8–10]. This instability does not require any gradients nor collisions and may be active in the region of large electric field. It has been further studied for conditions of Hall Thrusters [11–13] and magnetron discharges [14, 15]. This instability is kinetic in nature but the Boltzmann equation is so complex in these systems that no good analytical solution can be derived. Hence, particle-in-cell (PIC) simulations are required to better understand its origins and effects on the electron transport.

However, one of the challenges in these devices is that the collisionless instabilities and collisional phenomena (e.g. ionization) occur simultaneously [16]. Due to the relatively high plasma density (typically $n \approx 10^{18}$ m^{-3}), small cells (typically $\Delta x \approx 20 \, \mu$m) and time steps (typically $\Delta t \approx 1$ ps) are required to simulate device scale phenomena on the order of a few cm and 10 kHz. In addition, the multi-dimensional nature (axial convection, azimuthal $E \times B$ drift, radial wall effects) of the plasma flow makes PIC simulations of $E \times B$ discharges computationally expensive.

In the last decade, the growing performances of computer facilities have stimulated the development of simulation codes, that have become indispensable tools in plasma studies. However, as the numerical models have become more and more complex, the validity of the results must be investigated. They could be affected by various numerical errors and uncertainties (such as numerical noise), algorithms and models used, or even by the configuration of input parameters. Therefore, there is an increasing need for verification and validation (V&V) of simulation codes. While validation implies comparison with real experiments, verification could be done in many ways such as unit and mezzanine tests for specific parts of a code [17], or benchmarking, i.e. code-to-code verification. The early work of Surendra [18], in which the results of twelve different codes (kinetic, hybrid and fluid) on a 1D low-pressure (30, 100 and 300 mTorr) radio-frequency discharge in helium were compared, is considered as a pioneer for the benchmarking of simulation codes in the low-temperature plasma community. Later, a similar 1D case in helium was benchmarked by Turner *et al* [19] with five independently developed PIC codes, and they demonstrated that the results obtained for 4 pressures (30, 100, 300 and 1000 mTorr) were statistically indistinguishable. It paved the way to an increased benchmarking activity for different types of plasma discharges. In particular, in [20], two 1D PIC codes have been compared on a parallel plate glow discharge in helium at 3.5 Torr. In [21], six 2D fluid codes have been compared on the simulation of axisymmetric positive streamers in dry air at atmospheric pressure on three test cases of increasing complexity, and the authors stated that 'the results agree reasonably well'.

Even though a 1D helium benchmark is an efficient tool to verify the main algorithms of a PIC code (such as the Poisson solver and equations of motion) along with the Monte Carlo collision (MCC) module, this case is only one dimensional with no magnetic field and hence, it would be beneficial for the low-temperature plasma community to benchmark simulation codes using a more complex model, such as $E \times B$ discharges. Moreover, it has been observed recently by Janhunen *et al* [22, 23] that numerical noise may influence the results of PIC simulations by imitating the effect of collisions and hence, it is important to better understand the influence of the numerical parameters. The chosen simulation model should exhibit the relevant physics of an $E \times B$ discharge (high peak value of axial electric field, azimuthal instabilities, etc.) and in the meantime, it should be simple enough to be simulated in a reasonable computational time. In this paper, a 2D simulation model close to the one proposed by Boeuf and Garrigues [24] is adopted, with a longer azimuthal length and a higher number of macroparticles per cell to assess numerical convergence. The advantage of this test case is that a steady state is reached quickly, which facilitates comparison of the results. Moreover, the computational cost of PIC simulations has increased the need for quicker algorithms and made code parallelization compulsory. Hence, the seven independent codes considered here exhibit different features to decrease computational times and it makes this benchmarking activity even more relevant. An agreement on insightful parameters of the discharge will strengthen the confidence in our codes and legitimize them for further analysis of $E \times B$ discharges. This test case is not only intended to study the physics of a Hall Thruster, but could also be used in a general way to benchmark all $E \times B$ discharge codes.

In this paper, we first describe in section 2 the simulation model chosen, along with the detailed algorithms used. Then, the specificities of each independent PIC code are given in section 3 along with the computational times and resources. Section 4 is dedicated to the comparison of the results.
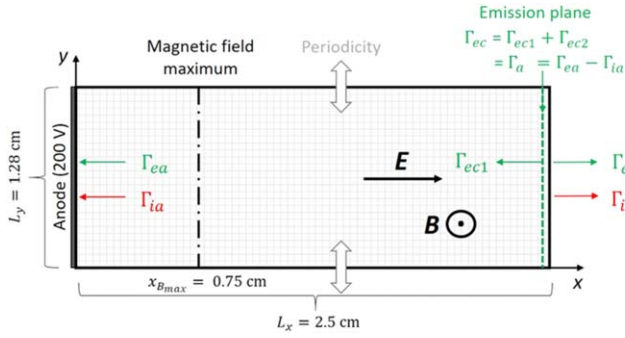
Figure 1. Simulation domain. *x* is the axial direction, *y* the (periodic) azimuthal direction. Black pointed dashed line ($x_{B_{\max}} = 0.75$ cm): position of maximum radial magnetic field. Green dashed line ($x_e = 2.4$ cm): plane from which electrons are emitted uniformly along the azimuthal direction. The azimuthally averaged fluxes are represented. Γ_{ea} and Γ_{ia}: respectively electron and ion fluxes through the left boundary. Γ_{ec1}: electron flux from the emission plane going into the discharge. Γ_{ec2} and Γ_{ic}: respectively electron and ion fluxes through the right boundary.

Azimuthally and time averaged (at steady state) axial profiles of main discharge parameters (axial electric field, ion density and electron temperature) are first compared and then, we look at the characteristics of the azimuthal instabilities. The sensitivity of the benchmark and the numerical convergence according to the number of macroparticles per cell is then discussed in section 5, prior to conclude on the agreement obtained between all the codes.

2. Description of the model

To study the azimuthal $\boldsymbol{E} \times \boldsymbol{B}$ electron drift instability and the associated axial electron transport, a 2D axial-azimuthal particle-in-cell benchmark case is considered with conditions close to those of a typical $\boldsymbol{E} \times \boldsymbol{B}$ discharge. Some simplifying assumptions have been made to make the case reproducible in a reasonable computational time. Indeed, the intermolecular collisions and neutral transport are neglected while a given ionization source term is imposed [24] and hence, we are able to obtain a steady state result in a short time (i.e. 10 μs).

2.1. Simulation domain

As illustrated in figure 1, the computational domain corresponds to a 2D structured Cartesian mesh, which models the axial (*x*) and azimuthal (*y*) directions of an $\boldsymbol{E} \times \boldsymbol{B}$ discharge. Hence, the curvature of the (*x, y*) plane is neglected. The left-hand side boundary of the domain represents the anode plane, with a fixed potential of 200 V, while the right-hand side corresponds to the cathode plane, where electrons are emitted. The distance between the anode and the cathode corresponds to the axial length of $L_x = 2.5$ cm, with the position of radial magnetic field maximum at $x = 0.75$ cm. To reduce computational times, a small region ($L_y = 1.28$ cm) in the azimuthal direction is taken into account and periodic boundary conditions are imposed.

Table 1. Input parameters.

| Parameter	Symbol	Value	Unit
Computational parameters			
Time step	Δt	5×10^{-12}	s
Cell size	$\Delta x = \Delta y$	5×10^{-5}	m
Final time	T_{final}	20×10^{-6}	s
Cells in axial direction	N_x	500	
Cells in azimuthal direction	N_y	256	
Axial length	L_x	2.5	cm
Azimuthal length	L_y	1.28	cm
Initial state			
Macroparticles per cell	$N_{\mathrm{ppc,ini}}$	75/**150**/300	
Plasma density	$n_{p,\mathrm{ini}}$	5×10^{16}	m^{-3}
Physical parameters			
Discharge voltage	U_0	200	V
Electron initial temperature	$T_{e,\mathrm{ini}}$	10	eV
Ion initial temperature	$T_{i,\mathrm{ini}}$	0.5	eV

As most of the codes used are explicit, the cell size Δx and time step Δt needed to satisfy the PIC stability conditions:

$$\begin{cases} \Delta t \leqslant \frac{0.2}{\omega_p} \\ \Delta x \leqslant \lambda_d \end{cases} \tag{1}$$

with $w_p = \sqrt{\frac{n_e e^2}{m \epsilon_0}}$ and $\lambda_d = \sqrt{\frac{\epsilon_0 k_B T_e}{n_e e^2}}$ being respectively the angular plasma frequency and the electron Debye length, with n_e the electron density, e the electron charge, m the electron mass, T_e the electron temperature and ϵ_0 the vacuum permittivity. In our case, the current density is fixed at 400 A.m^{-2}, which gives a maximum plasma density of around 5×10^{17} m^{-3} and electron temperatures of about 50 eV. Hence, the minimum values for Δt and Δx will respectively be 6×10^{-12} s and 70 μm. For the benchmark case, a time step of $\Delta t = 5 \times 10^{-12}$ s and a grid spacing of $\Delta x = 50$ μm with a grid of 500×256 cells are used. 4×10^6 time steps are simulated, i.e. 20 μs of the discharge, and the diagnostics are averaged every 5000 time steps.

Electrons and ions are initially loaded with a density of 5×10^{16} m^{-3} uniformly throughout the simulation domain, with velocities chosen from a Maxwellian distribution with a temperature $T_e = 10$ eV and $T_i = 0.5$ eV, respectively. To reduce numerical heating due to statistical noise, the number of macroparticles per cell at initialisation for the nominal case is fixed at $N_{\mathrm{ppc,ini}} = 150$ (case 1), i.e. 150 electrons and 150 ions per cell. Then, approximately $N_{\mathrm{ppc,fin}} \approx 550$ macroparticles per cell are obtained at stationary state. As mentioned before, this parameter could have an influence on the numerical results and hence, an extensive study of code convergence has also been conducted by simulating two other cases with $N_{\mathrm{ppc,ini}} = 75$ (case 2) and $N_{\mathrm{ppc,ini}} = 300$ (case 3).

All the simulation parameters are summarized in table 1.
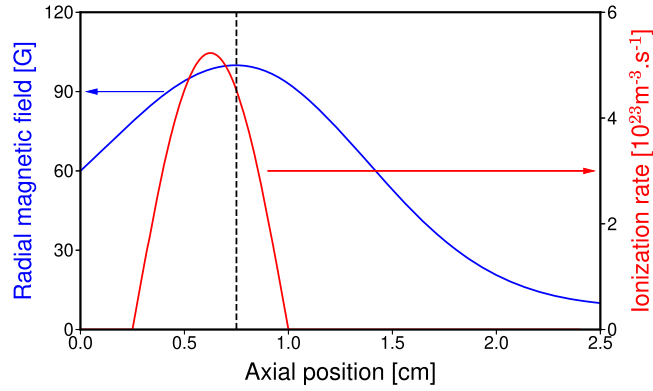
Figure 2. Axial profiles of the imposed radial magnetic field and ionization rate. Dashed line corresponds to the position of maximum magnetic field.

2.2. Imposed axial profiles

2.2.1. Radial magnetic field. The axial profile of the radial magnetic field is imposed with a Gaussian shape, as shown in figure 2:

$$B(x) = a_k \exp\left(-\frac{(x - x_{B_{max}})^2}{2\sigma_k^2}\right) + b_k \qquad (2)$$

with $k = 1$ for $x < x_{B_{max}}$ and $k = 2$ for $x > x_{B_{max}}$. The values of the a_k and b_k coefficients can be easily calculated from the given parameters: $B_0 = B(x = 0) = 6$ mT, $B_{L_x} = B(x = L_x) = 1$ mT, $B_{max} = 10$ mT, $x_{B_{max}} = 0.3L_x = 0.75$ cm and $\sigma_1 = \sigma_2 = 0.25L_x = 0.625$ cm. Their formula are given in appendix A.

2.2.2. Ionization profile. For this benchmark case, no collisions are considered. However, ionization events are taken into account as a source term for the plasma to sustain the discharge. To do so, electron–ion pairs are injected at each time step according to the profile of a given ionization rate $S(x)$, dependent on x only (uniform in azimuthal direction). $S(x)$ has a cosine shape, as shown on figure 2:

$$\begin{cases} S(x) = S_0 \cos\left(\pi\frac{x - x_m}{x_2 - x_1}\right) \text{ for } x_1 \leqslant x \leqslant x_2 \\ S(x) = 0 \text{ for } x < x_1 \text{ or } x > x_2 \end{cases} \qquad (3)$$

with $x_1 = 0.25$ cm, $x_2 = 1$ cm and $x_m = \frac{x_1 + x_2}{2} = 0.625$ cm. The maximum ion current density J_M can be extracted from the steady-state continuity equation, accounting for the ionization profile in equation (3) by:

$$J_M = e \int_0^{L_x} S(x)dx = \frac{2}{\pi}(x_2 - x_1)eS_0. \qquad (4)$$

Hence, we impose $J_M = 400$ A m^{-2} by fixing the maximum value of the ionization profile to $S_0 = 5.23 \times 10^{23}$ m^{-3} s^{-1}.

The number of electron–ion pairs to inject at each time step Δt is given by $L_y\Delta t \int_0^{L_x} S(x)dx$ and the positions (x_i, y_i) are chosen randomly such as:

$$\begin{cases} x_i = x_m + \sin^{-1}(2r_1 - 1)\frac{(x_2 - x_1)}{\pi} \\ y_i = r_2 L_y \end{cases} \qquad (5)$$

with r_1 and r_2 two random numbers uniformly distributed over the interval [0, 1]. For one pair, the electron and the ion are injected at the exact same position. Their velocities are chosen from a Maxwellian distribution with the same temperature as initialisation ($T_e = 10$ eV and $T_i = 0.5$ eV).

2.3. Boundary conditions

Electrons and ions which cross the left or right boundary plane of the domain are removed from the simulation. However, to ensure current continuity and neutralization of the extracted ion beam, electrons are injected from the cathode plane. The emission line is set on the downstream of the simulation domain, at 1 mm from the right domain boundary. The number of electrons injected at each time step is calculated using the current conservation through the discharge to obtain Γ_{ec}, the absolute value of azimuthally averaged emitted electron flux:

$$\Gamma_{ec} = \Gamma_a = \Gamma_{ea} - \Gamma_{ia} \qquad (6)$$

with Γ_{ea} and Γ_{ia} being respectively the absolute values of azimuthally averaged electron and ion fluxes to the anode side, displayed on figure 1. Hence, by counting the number of electrons and ions that cross the anode boundary at each time step (respectively ΔN_{ea} and ΔN_{ia}), the corresponding number of electrons emitted from the emission plane can be calculated as:

$$\Delta N_{e,emi} = \Delta N_{ea} - \Delta N_{ia}. \qquad (7)$$

These electrons are injected uniformly in the azimuthal direction, with a Maxwellian velocity distribution with a temperature $T_{e,emi} = 10$ eV.

However, this method for calculating the number of emitted electrons does not prevent an artificial cathode sheath to form. To suppress artificially this sheath, the emission plane is shifted by 1 mm from the right boundary plane (i.e. to the position $x_e = 2.4$ cm) and its potential is adjusted at each time step by imposing a zero azimuthally averaged potential at this location. Hence, the azimuthally averaged potential drop between the anode and the emission plane is maintained constant and equal to the applied voltage (200 V). To do so, we solve the Poisson equation for U:

$$\Delta U = -\frac{e}{\epsilon_0}(n_i - n_e) \qquad (8)$$

with boundary conditions $U(0, y) = U_0$ and $U(L_x, y) = 0$. Then, we obtain the electric potential ϕ by subtracting the azimuthally averaged potential at the emission plane $\overline{U_e}$ from the solution $U(x, y)$ of Poisson equation:

$$\phi(x, y) = U(x, y) - \frac{x}{x_e}\overline{U_e} \qquad (9)$$

Table 2. Main codes specificities.

	LPP	LAPLACE	CERFACS	RUB	USask	TAMU	PPPL
Algorithms							
Pusher solver	Explicit	Explicit	Explicit	Implicit	Explicit	Explicit	Explicit
Poisson solver	Hypre	Pardiso	Maphys	FFT Thomas	FFT	Hypre	Hypre
Floating-point precision	Double	Single(pusher) Double (Poisson)	Double	Single(pusher) Double (Poisson)	Double	Double	Double
Code acceleration							
Architecture	CPU	CPU	CPU	GPU	CPU	CPU	CPU
Parallelization	MPI	MPI/OpenMP	MPI	CUDA	MPI	MPI	MPI/OpenMP
Decomposition	Domain	Particle	Domain	Both	Domain	Particle	Particle
Language	Fortran	Fortran	Fortran	C+Cuda C	Fortran	C++	C
Simulation times (days)							
Case 1 ($N_{\text{ppc,ini}} = 150$)	8 (360 CPU)	5 (108 CPU)	7 (360 CPU)	14 (1 GPU)	21 (32 CPU)	15 (300 CPU)	2.5 (224 CPU)
Case 2 ($N_{\text{ppc,ini}} = 75$)	5 (360 CPU)	3 (108 CPU)	4 (360 CPU)	9 (1 GPU)	11 (32 CPU)	11 (200 CPU)	2.5 (112 CPU)
Case 3 ($N_{\text{ppc,ini}} = 300$)	14 (360 CPU)	6 (180 CPU)	13 (360 CPU)	14 (2 GPU)	20 (64 CPU)	22 (400 CPU)	2.5 (448 CPU)

with:

$$\overline{U_e} = \frac{1}{L_y} \int_0^{L_y} U(x_e, y)\mathrm{d}y. \tag{10}$$

The right boundary plane will have a varying negative potential but this drop in potential between the emission plane and the right boundary plane does not have any useful physical meaning and does not affect the main discharge physics.

3. Code details

Seven groups participated in this study, each group using its own independently developed simulation code. While the codes are all PIC codes, they mainly differ in the way the equation of motion and the Poisson equation are solved. All the codes (except RUB, see subsection 3.4 for details) are using a bilinear interpolation scheme (cloud-in-cell) and ions are considered unmagnetized, due to their large Larmor radius compared to the domain dimensions. The exact physical charge-to-mass ratio for ions (here Xenon ions are considered) is used by all the codes. As described in the previous section, periodic boundary conditions are imposed in the azimuthal direction, whereas the plasma potential is fixed at 200 V at the left boundary (anode) and 0 V at the right boundary (cathode).

As the benchmark cases are quite computationally expensive, the code performances are obtained through parallelization. This could be done via message passing interface (MPI), combined or not with OpenMP (Open Multi-Processing), or using GPU instead of CPU. Each processor can consider one portion of the computational grid (domain

decomposition) or one portion of the particles in the domain (particle decomposition) in order to speed-up the computation. Another way of decreasing significantly the computational time is to move the ions every N_{sub} electron time steps, as they are way slower than the electrons and barely move during one time step [25].

A summary of the codes specificities is provided in table 2, along with the simulation times for the 3 benchmark cases.

3.1. Group **LPP**: T Charoy, A Tavant, A Bourdon, P Chabert

The 2D-3V PIC-MCC simulation code *LPPic* was used. The code features a structured Cartesian mesh fixed in time, the Poisson equation is solved using an iterative parallel multigrid solver (PFMG from the open source HYPRE library [26]) and the particles are advanced via a classic leapfrog scheme, along with a Boris scheme [27]. The code is parallelized via MPI through a domain decomposition. It has been verified with the 1D helium benchmark of Turner *et al* [19] (further details in [28, 29]) and extensively used to simulate the radial-azimuthal plane of a Hall Thruster [28, 30, 31]. For this benchmark, the code was adapted to the axial-azimuthal plane and accelerated via a load-balancing algorithm (adjusting periodically the size of each processor domain to have approximately the same number of particles inside each processor). Ions are moved every 11 electron time steps to decrease computational time [25]. It was checked that it has a negligible influence on the obtained results. The random number generator (RNG) used is the *Fortran 2003* RNG, seeded by the internal clock of every CPU.

3.2. Group **LAPLACE**: L Garrigues, J P Boeuf

Explicit electrostatic PIC-MCC models developed at Laplace resolve the space in two-dimensions [32–38] and three-dimensions [39, 40] (Cartesian coordinates with structured meshes) and three dimensions in velocity phase. Trajectories of charged particles are integrated according to a standard leap-frog scheme with a Boris algorithm [27]. Poisson's equation is solved using the direct PARDISO solver included in the MKL library of INTEL. A particle decomposition method is used and an hybrid approach coupling MPI and OpenMP techniques is used to accelerate parallelization. Typically, a MPI thread per socket is attached and a number of OpenMP threads is taken identical to the number of cores per socket. No subcycling technique is used (ions are moved every time step). The RANDU function is used to generate pseudo-random numbers between 0 and 1 [41].

3.3. Group **CERFACS**: W Villafana, B Cuenot, O Vermorel

The PIC version of the AVIP code was used. AVIP is a massively parallel code able to model the plasma dynamics of Hall thrusters in complex 2D/3D geometries using unstructured grids. AVIP has been built upon the AVBP combustion code structure [42], which has been extensively validated and specifically designed for efficient calculations with a high number of processors [43]. Both PIC and fluid modelings are available [44, 45]. For the present PIC simulations, ions and electrons velocities are respectively updated with standard Leap-Frog and Boris schemes [27]. Poisson equation is solved thanks to the open-source solver MAPHYS currently developed by INRIA. It combines both direct and iterative methods for fast and accurate results [46, 47]. Domain decomposition is performed using the external library PAR-METIS [48]. Domain partitioning is regularly updated to ensure a correct load balancing between processors. In order to speed up the simulation, subcycling is used for the ion motion and their position and speed are updated every 5 electron time steps. The standard RNG of Fortran 95 is used with the same seed for each run in order to ensure reproducibility of the results.

3.4. Group **RUB**: D Eremin

The 2D-3V PIC code used in the present work is adopted from the implicit energy- and charge-conserving scheme suggested in [49]. The approach iteratively utilizes the Crank–Nicolson method to calculate the electrostatic field on the new time level, simultaneously with the particle orbit integration. Note that this approach employs the same shape function for the electric field and the current density (as opposed to the same shape function for the electric field and the charge density in the conventional explicit leapfrog scheme-based PIC variant implemented in all the other codes used in this benchmark), which makes the linear momentum conservation inexact. However, as the results demonstrate, this does not seem to hamper the model.

In order to reduce the computational cost, the Poisson equation rather than Ampere's formulation was adopted for the electrostatic field. In case of a charge-conserving scheme, both approaches are equivalent, but the Poisson equation requires calculating the charge density only at the end of a time step, instead of having to add contribution to the current density each time a particle crosses a grid cell if the divergence of Ampere's law is used. The charge density was calculated using the quadratic spline shape function, whereas the linear shape function was used for the electric field. Since the original work [49] did not contain treatment of boundaries at the electrode surfaces, the modifications necessary to include such effects were introduced. In order to account for the periodicity in the azimuthal direction, the field solver used the discrete Fourier transformation in this direction, after which the axial profile for each of the azimuthal field harmonics was obtained by solving the corresponding one-dimensional Poisson equation with the Thomas algorithm.

It should be emphasized that the implicit iterative scheme employed for the RUB code is quite different from and is much more computationally expensive compared to the commonly used explicit algorithm. In contrast to the latter, the implicit scheme is predominantly aimed at self-consistent modelling of plasmas with high densities, where the need to resolve the Debye length to avoid the finite-grid instability makes the explicit approach prohibitively expensive. However, the energy-conserving implicit algorithm being relatively new, it is important to establish equivalence of its' results to those provided by the explicit scheme whenever possible. This dictated our choice of the algorithm for this particular benchmark. Because of the high computational cost of the scheme, everything except the field solver (which was implemented on CPU) was parallelized on GPU (NVIDIA V100 32GB) using CUDA C. Due to the limited amount of GPU memory, the case 3 was parallelized on two GPUs using an additional domain decomposition in the azimuthal direction to ensure even load balance. For the RNG, the xorshift128 method suggested in [50] has been used.

3.5. Group **USask**: D Sydorenko, A Smolyakov

The code is an explicit electrostatic particle-in-cell 2D-3V resolving 2 coordinates (*x* and *y*) and 3 velocity components for each particle. It was used in [23] and the earlier 1D version [51] was used in [22]. A leap-frog numerical scheme is used and the velocity is advanced using the Boris scheme [27]. The Poisson equation solver involves discrete Fourier transformation along the periodic direction to reduce dimensionality of the problem. The code is parallelized with MPI and domain decomposition is used. Subcycling of electrons relative to ions is used (ions moved every 11 electron time steps) to reduce numerical cost [25]. The RNG is the maximally equidistributed version of Mersenne Twister 19937 [52, 53].

3.6. Group **TAMU**: K Hara

The explicit PIC code is written in C++ using Message Passing Interface (MPI). Particle decomposition is used to make the number of particles per processors to be
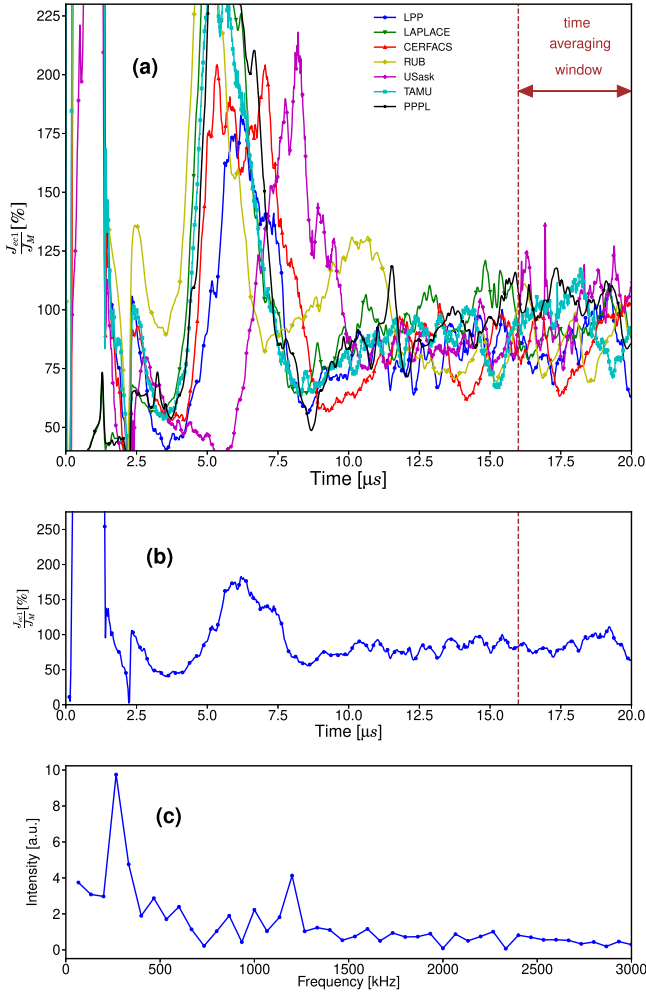
Figure 3. (a) Time evolution of $\frac{J_{ec1}}{J_M}$ for Case 1 for all the codes. Brown dashed line: beginning of time averaging interval (until 20 μs). (b) Time evolution of $\frac{J_{ec1}}{J_M}$ for Case 1 with code **LPP**. (c) Corresponding fast-Fourier-transform taken from data between 12 and 20 μs.

approximately equal. Domain decomposition is used to calculate the potential via the Poisson equation using HYPRE [26], in which a multigrid method is used as a preconditioner to the GMRES solver. Double precision is used for all numerical variables and no electron subcycling is used, i.e. the ions and electrons are updated at the same time step. For the results shown, ion and electron densities as well as the potential are calculated on cell centers while similar results are obtained in calculations based on storing information on nodes. Random numbers are generated using the C Standard General Utilities Library by initializing different seed values in the individual processors.

*3.7. Group **PPPL**: A T Powis, J A Carlsson, I D Kaganovich*

This new electrostatic particle-in-cell code was developed at the Princeton Plasma Physics Laboratory and Princeton University. It was designed from the ground up for scalability and performance on modern super-computing facilities. The code features parallelism via MPI, OpenMP, and algorithms are designed to take advantage of modern vector registers. Poisson's equation is solved over the grid using domain decomposition and via the HYPRE [26] package, which has demonstrated excellent scalability on up to 100 000 cores [54]. Particles are distributed and shared as a list, rather than via domain decomposition. The software is capable of modeling a two-dimensional box, with arbitrary boundary conditions (conducting, periodic, mirror) and allows particle sources and losses through the walls. Particles are evolved explicitly with double precision in 2D-3V via the Boris algorithm [27]. Random numbers are generated using the double precision SIMD oriented Fast Mersenne Twister (dSFMT) package [55].

Before converging to the final benchmark results presented in the next section, many small implementation mistakes were found. To guide the next users of this benchmark, some general guidelines are given in appendix A.

4. Results

Prior to performing any detailed benchmarking, it was important to make sure that all codes converge to a steady state. To do so, the time evolution of the electron axial current is compared. The electron current density injected at the emission plane J_{ec} is split in two parts: $J_{ec} = J_{ec1} + J_{ec2}$, where J_{ec1} corresponds to the electron current density entering the channel and J_{ec2} is used to neutralize the extracted ion beam $J_{ec2} = J_{ic}$, fixed by the imposed ion current density. Hence, J_{ec1} could be used to characterize the anomalous cross-field transport in the discharge. For the comparison of results, J_{ec1} is normalized by the imposed total ion current $J_M = J_{ia} + J_{ic}$, which is set to 400 A m^{-2} at steady state.

The time evolution of $\frac{J_{ec1}}{J_M}$ is shown in figure 3(a) for all simulation codes. It can be seen that all simulation codes reach a steady state after around 10 μs. However, it can be noticed that a small oscillation appears at steady state, with a frequency of the order of hundreds of kHz, as shown on figures 3(b) and (c) for code **LPP**. As this phenomena is retrieved for all the codes, we decided to average our results in time to smooth out these oscillations, which could be physical or numerical, focusing on benchmarking of time-averaged phenomena. The period has been chosen as a compromise between the need to capture enough oscillation periods at steady state and keeping a reasonable computational time, i.e. 4 μs between 16 and 20 μs.

Moreover, as mentioned before, some high-frequency instabilities propagate in the azimuthal $\boldsymbol{E} \times \boldsymbol{B}$ direction, as seen on figure 4 for the azimuthal electric field and the ion density, obtained with code **LPP** at $t = 20$ μs. To make the benchmarking of large-scale phenomena, it was decided to average in this direction. It can also be noticed that two distinct zones for the oscillations of the azimuthal electric field exist: a short wavelength zone between the anode and the location where the radial magnetic field is at maximum, called zone (I), and a long wavelength zone downstream, called
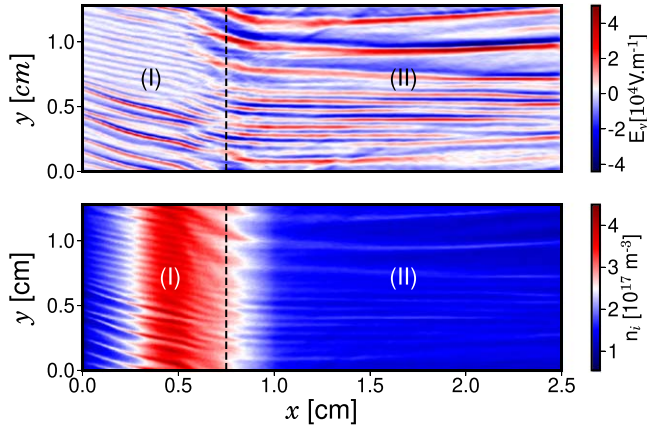
Figure 4. 2D axial-azimuthal maps of the azimuthal electric field (top) and ion density (bottom) obtained with code **LPP** at $t = 20\,\mu s$. Dashed line corresponds to the position of maximum magnetic field that separates zone (I) and zone (II).

Table 3. Three benchmark cases. N_{ppc} is the number of macroparticles per cell. The nominal case is Case 1.

Case	$N_{ppc,ini}$ at initialisation	$N_{ppc,fin}$ at steady state
1	150	550
2	75	275
3	300	1100

zone (II). Such transition of the plasma waves is discussed more in detail in section 4.2.

One can notice on table 2 that the computational times and resources needed to reach this steady state are quite high, with around 10 days in average for the nominal case (which corresponds to around $60\,000$ CPU hours). Compared with the computational time of the 1D helium benchmark of Turner *et al* [19] that was around a couple of hours, parallelization of computational codes is needed to increase drastically the code performances. Moreover, it can be seen that the 7 codes simulate the cases with a broad range of computational times (between 2.5 and 21 days for the nominal case) and it shows that this benchmark is also a powerful tool to characterize the performance of a simulation code.

Below in section 4.1, the azimuthally and time averaged axial profiles of the main discharge parameters (axial electric field, ion density and electron temperature) are first analysed. Due to the interest towards understanding the effects of azimuthal plasma waves on the electron anomalous transport, the azimuthal instabilities are compared in section 4.2, investigating their dominant mode characteristics (wavelength and frequency). These comparisons are done for the 3 cases considered in this benchmark, that differ only by their number of macroparticles per cell, given in table 3.

4.1. Main plasma parameters

For benchmarking purposes, three parameters are chosen to be shown in this paper: the axial electric field, the ion density and the electron temperature. Since the low-frequency

oscillations on the order of hundreds of kHz are not the focus of the benchmarking, the results are averaged in the azimuthal direction and in time (between 16 and $20\,\mu s$) to obtain a steady state result.

The axial profiles for the nominal case are shown on figure 5. We can see that the 7 codes display a good agreement for all the parameters. The differences are mainly on the peak value and the profile in zone (II) but the overall mean relative error is less than 5%. This difference is measured by using the mean value between all the curves and calculating the mean relative error for every curve, that has a maximum below 5%. It is also important to notice that all results from different codes capture the characteristics of an $\boldsymbol{E} \times \boldsymbol{B}$ discharge: a high axial electric field peaks near the maximum of radial magnetic field while the ion density is high on the anode side, just before the magnetic field peak. In particular, the results display an important feature of Hall thrusters, namely the overlapping between the ion density peak (ionization zone) and the axial electric field peak (acceleration zone). One can notice the sharp increase of the axial electric field near the right boundary that is due to the artificial sheath created outside of the region of interest (between the emission plane ($x_e = 2.4$ cm) and the right boundary). It has been shown that this region does not affect the main discharge results [24].

The same comparison is done for the two other cases and the results are displayed on figures 6 and 7 respectively. They both exhibit a similar behaviour than the nominal case shown in figure 5, with an overall mean relative error between the codes less than 5%. The main reason for the slight discrepancies comes from the low-frequency oscillation behaviour, as can be seen from figure 3.

4.2. Azimuthal instabilities

In addition to the time-averaged plasma properties, the instabilities propagating in the azimuthal direction, shown on figure 4, also serve as a useful phenomenon for benchmarking of different simulation codes. Usually in 2D, when the direction parallel to the magnetic field is neglected, the ECDI exhibits a discrete behaviour around the cyclotron resonances $k_y V_E = n\omega_{ce}$, $n = 1, 2,..$ [10]. When the wave propagation along the magnetic field is included, one can show from the linear theory that the ECDI can transition to an ion-acoustic instability [7, 9, 56, 57]. Nonlinear effects can also potentially result in transition to the ion-acoustic instability [22, 58]. The quasilinear theory of the anomalous transport based on the modified ion-acoustic instability in the conditions of Hall thruster has been proposed [56, 57]. The dispersion relation for the ion-acoustic instability in plasmas with moving ions has the form:

$$\omega_R \approx \boldsymbol{k}.\boldsymbol{v_i} \pm \frac{kc_s}{\sqrt{1 + k^2\lambda_d^2}}. \tag{11}$$

A 2D fast-Fourier-transform (FFT) is applied to the azimuthal electric field at fixed axial positions (between 16 and $20\,\mu s$) to get the corresponding spectrum. The results for two different axial positions are displayed on figure 8 for code **LPP**. It is
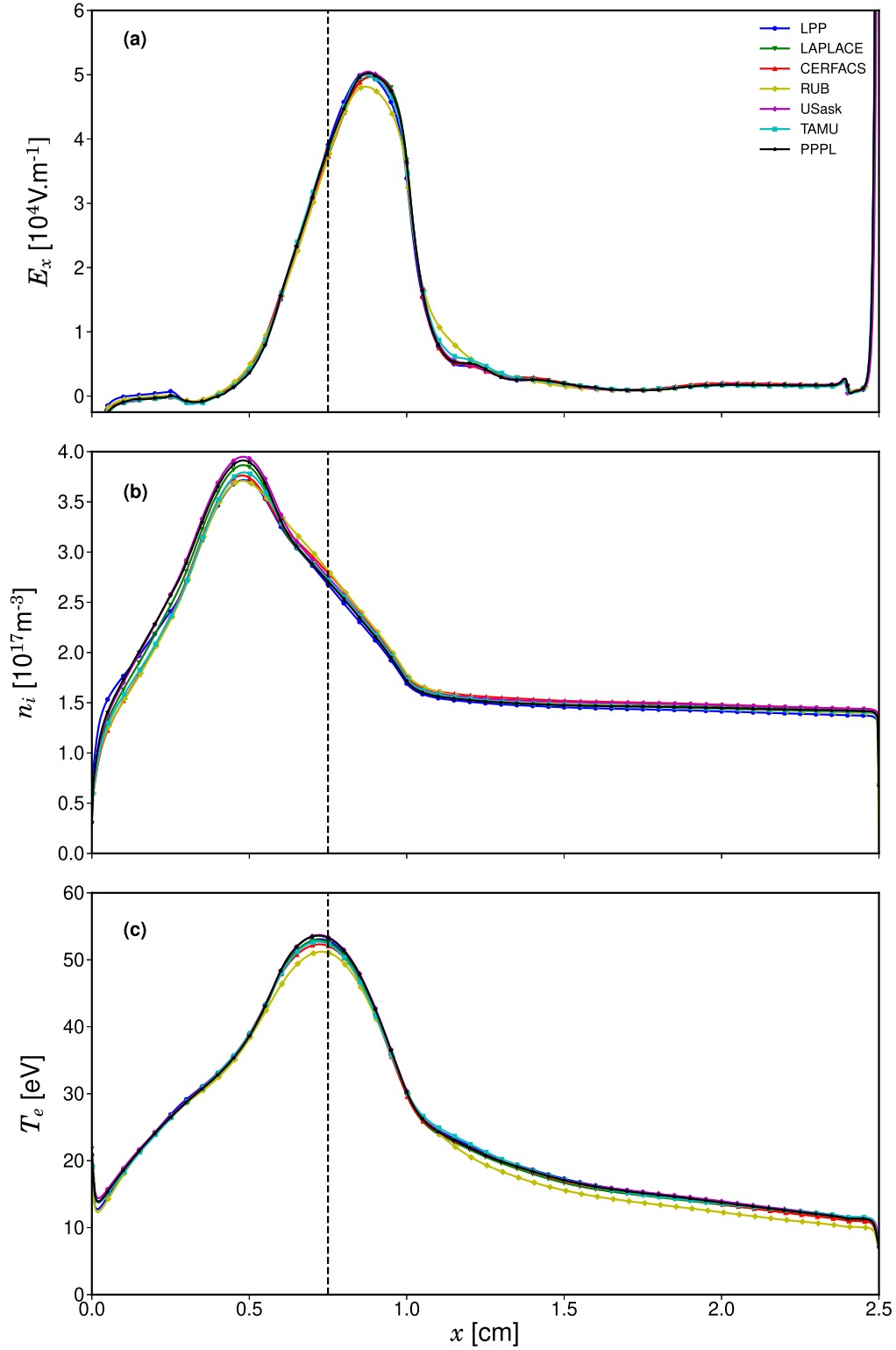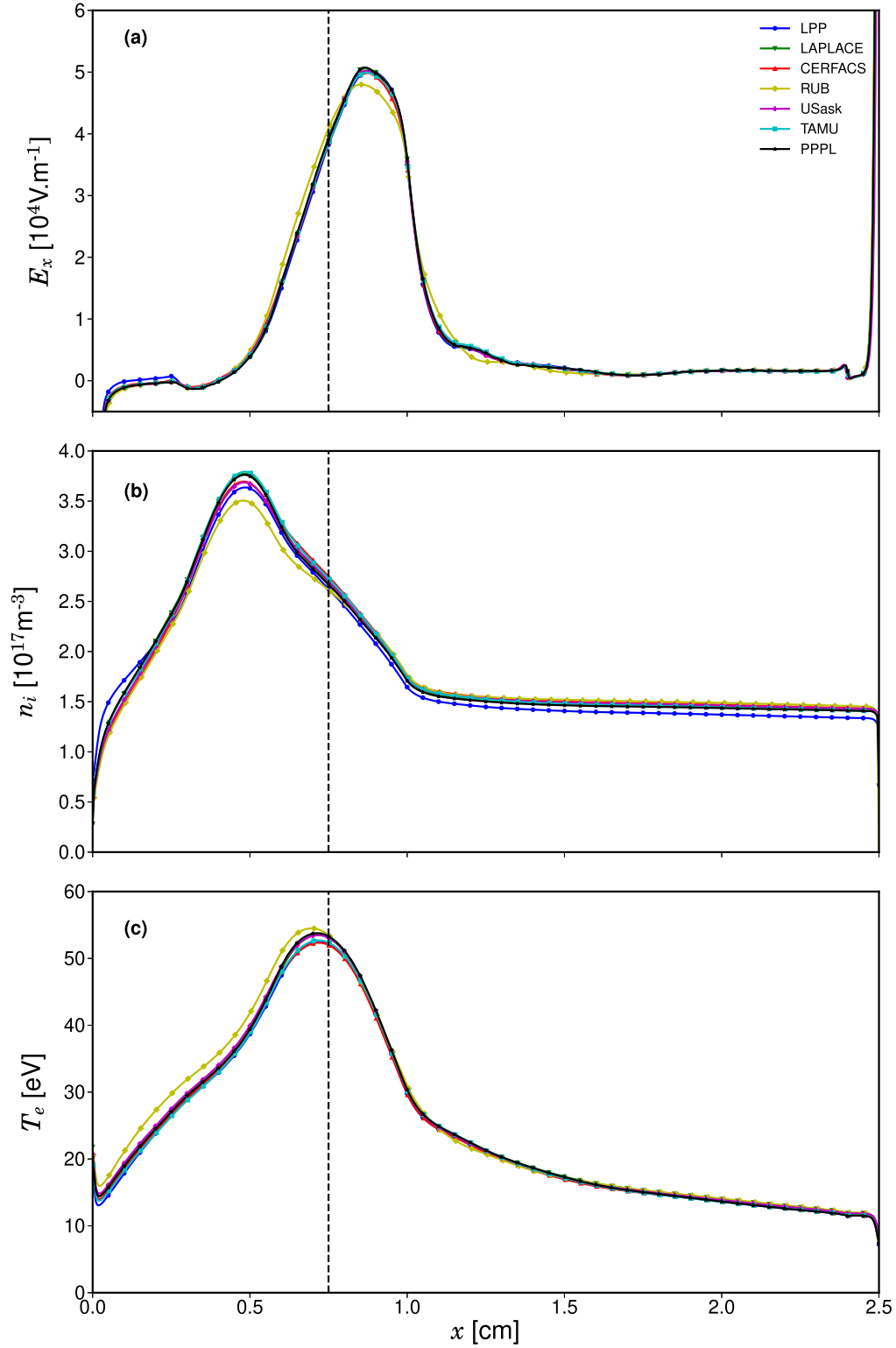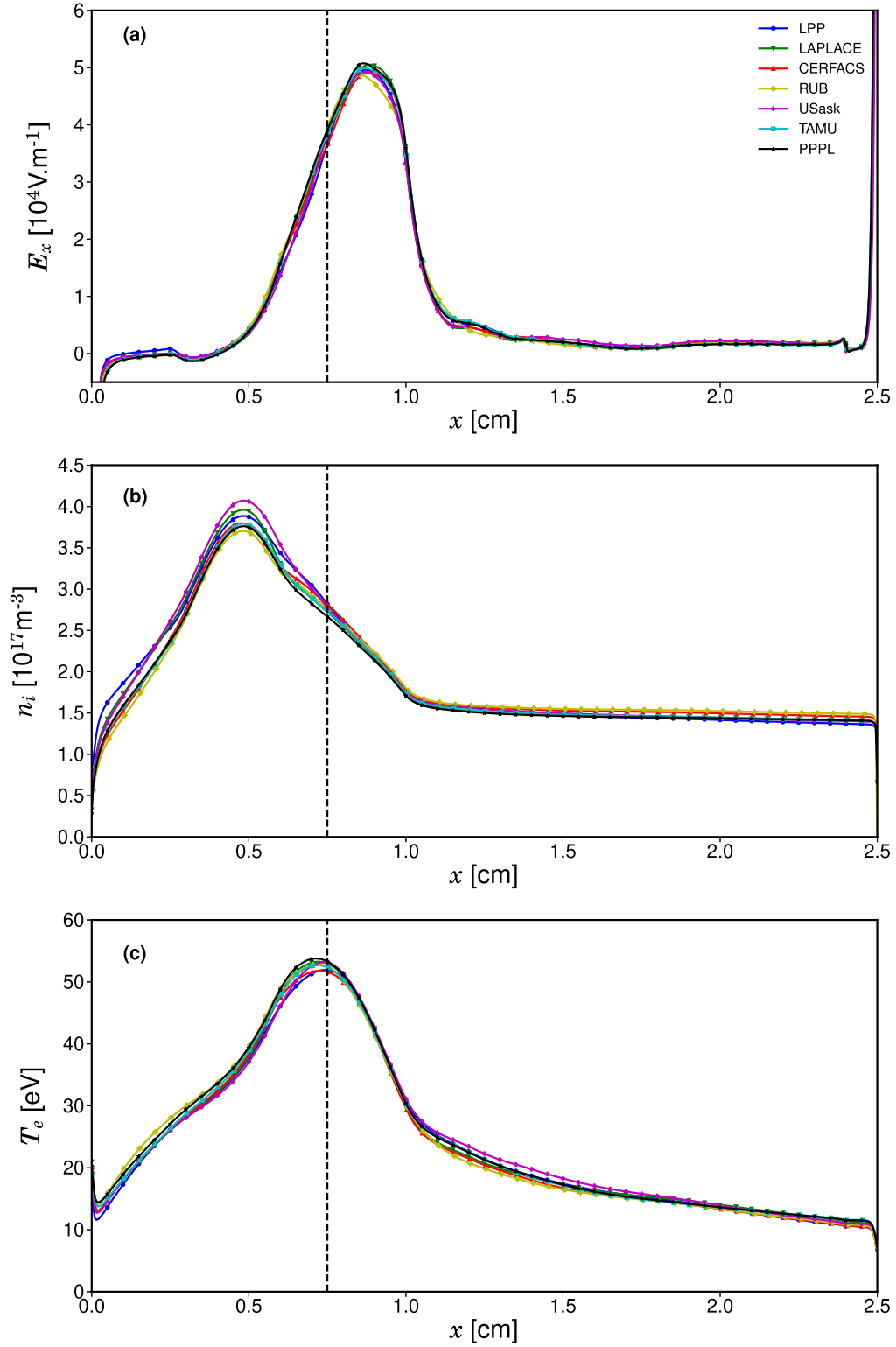
Figure 5. *Case* 1: Azimuthally and time (between 16 and 20 μs) averaged axial profiles of axial electric field (a), ion density (b) and electron temperature (c). Dashed line corresponds to the position of maximum magnetic field.

shown that in zone (I) (at $x = 0.3$ cm) a continuous dispersion relation is well fitted to the analytical expression of equation (11). This continuous behaviour was already observed in experiments [59] and in other 2D PIC codes that are self-consistent with the plasma generation [13]. However, closer to the cathode in zone (II) (at $x = 1.5$ cm), the

dispersion relation exhibits a different behaviour, which seems more discrete.

As this paper is focused on the benchmarking of different simulation codes, detailed study of the dispersion relations of the plasma waves is out of scope. Instead, to be able to compare the results of the different codes, it was decided to

Figure 6. *Case* 2: Azimuthally and time (between 16 and 20 μs) averaged axial profiles of axial electric field (a), ion density (b) and electron temperature (c). Dashed line corresponds to the position of maximum magnetic field.

extract the dominant mode at each axial position. Hence, the wave characteristics (wavelength and frequency) are compared as function of the axial position.

This axial dependence of the dominant mode characteristics for all the codes is shown on figure 9 for the nominal case. It can be seen that, in all simulation results, the wavelength and the frequency change abruptly at the position of maximum radial magnetic field. In zone (I) near the anode, the oscillations have a small wavelength ($\lambda \approx 0.5$ mm) and a high frequency ($f \approx 5$ MHz) while near the cathode in zone (II), the frequency drops to $f \approx 3$ MHz with a wavelength almost 4 times bigger ($\lambda \approx 2$ mm). The Debye length and ion

Figure 7. *Case* 3: Azimuthally and time (between 16 and 20 μs) averaged axial profiles of axial electric field (a), ion density (b) and electron temperature (c). Dashed line corresponds to the position of maximum magnetic field.

plasma frequency are displayed on the bottom row for comparison. The same behaviour is retrieved for the two other cases, as seen on figures B1 and B2, that have been placed in appendix for clarity purpose. Considering the complexity of the phenomena involved, the agreement between the codes is satisfactory.

The maximum wavelength for the dominant mode is around 2 mm which is well-resolved by the length in the azimuthal direction of 1.28 cm. However, further development of long-wavelength structures may be limited either by this limited azimuthal length or by the axial convection of the modes (due to the ion flow) from the most unstable region. As
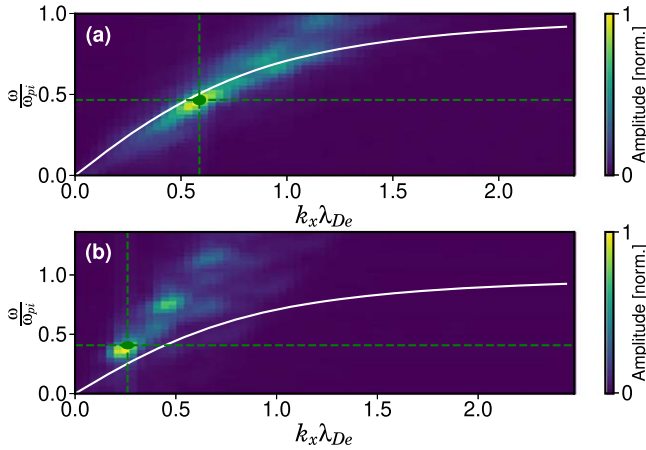
Figure 8. 2D FFT of the azimuthal electric field at $x = 0.3$ cm (a) and $x = 1.5$ cm (b), obtained with code **LPP**. Solid white line: ion acoustic dispersion relation. Green dashed lines: wavelength and frequency of the dominant mode.

this work is focused on a benchmark comparison with a simplified domain, this question will require further studies with a larger domain.

5. Discussion

5.1. Numerical convergence

In a PIC simulation, we consider finite-sized particles and hence, numerical collisions play a role. They can lead to fluctuations induced by thermal noise and this noise could have an impact on the study of the azimuthal instabilities and the related anomalous electron transport.

Okuda and Birdsall [60] defined a frequency for these numerical collisions in 2D simulations:

$$\nu_{\text{num}} \approx \frac{\pi \omega_{pe}}{16 N_D} \tag{12}$$

with N_D the number of macroparticles in a Debye sphere. For our nominal case (Case 1, $N_{\text{ppc,ini}} = 150$), we will have around $N_{\text{ppc,fin}} \approx 550$ macroparticles per cell at steady state, in average. Looking at figure 9, we can see that the minimum Debye length is around $70 \, \mu$m and hence, we have around 2π cells in a Debye sphere (worst case). It corresponds to a numerical collision frequency of $\nu_{\text{num}} \approx 5.6 \times 10^{-5} \omega_{pe}$. Turner [61] has shown that the numerical collisions can be neglected if $\frac{\nu_{\text{num}}}{\omega_{pe}} \leqslant 10^{-4}$ and hence our case complies with this criterion.

It is also important to assess the numerical convergence of this benchmark case more rigorously, by varying the number of macroparticles per cell. The mean value at steady state of the ratio $\frac{J_{ec1}}{J_M}$ of electron current entering the channel to the total ion current was used as a convergence criterion. Indeed, as mentioned earlier, this ratio is related to the electron axial transport in the discharge. This transport can be enhanced by numerical collisions and hence, decreasing N_{ppc} will increase the transport and $\frac{J_{ec1}}{J_M}$ will be higher. We can retrieve this behaviour by looking at how the averaged profiles of ion density and axial electric field evolve when the number of macroparticles per cell is

decreased. We observe on figure 10 for code **LPP** that if N_{ppc} is decreased, the axial electric field is increased and the ion density is decreased which is characteristic of a higher axial transport.

Figure 11 shows the mean value of $\frac{J_{ec1}}{J_M}$ at steady state for all seven codes, as function of the number of macroparticles per cell at steady state. We can see that we obtain a good convergence: when N_{ppc} is increased, $\frac{J_{ec1}}{J_M}$ is decreased and reaches a plateau. This plateau corresponds to the three benchmark cases that we have chosen (Case 1 with $N_{\text{ppc,final}} = 550$, Case 2 with $N_{\text{ppc,final}} = 275$ and Case 3 with $N_{\text{ppc,final}} = 1100$). It is interesting to notice that the curve has a knee at around 250 macroparticles per cell which gives a numerical collision frequency close to the criterion of [61]. Furthermore, this benchmark case shows that the number of macroparticles per cell commonly used in 2D PIC simulations (i.e. $N_{\text{ppc,final}} = 100$) is not enough to reach convergence. This need to increase the number of macroparticles per cell to prevent numerical collisions was also reported in [8] and more recently in [13], in which 800 macroparticles per cell were used on average at the end of the simulation.

One can also notice that the differences between the codes are this time bigger than 5%. The origin of this discrepancy still remains unclear and its analysis is let as future work.

5.2. Case sensitivity

The agreement obtained between the seven codes in section 4 is good, but it is worth noting that the results obtained are not 'statistically indistinguishable' (corresponding to less than 1% difference) as in the 1D helium benchmark [19]. As the present benchmarking test case is more challenging and more complex (two dimensions, magnetic field, emitting cathode, etc.) with the presence of turbulent phenomena, it is expected to obtain bigger differences.

To better characterize the sensitivity of this benchmark case, one code (code **LPP**) is used and the same simulation (same input parameters, corresponding to Case 2) is repeated 3 times. Figure 12(a) shows the different time evolutions of the $\frac{J_{ec1}}{J_M}$ ratio for these 3 simulations. It can be seen that while the beginning of the transient state (first 4 μs) is quite similar, some differences appear quickly and the oscillations at steady state become quite different. In fact, when time averaging is done between 11 and 15 μs, different axial profiles are retrieved for the ion density and axial electric field (the electron temperature is not shown here but displays a similar behaviour), as seen on figure 12(b). These differences are of the order of 5% and could be considered as the closest agreement we would get between the seven codes.

There is a reason why identical results are not obtained with the same code. In fact, as described in section 2, a RNG is frequently used for routines that are crucial for the discharge behaviour (ionization and electron emission). Hence, as this RNG is seeded randomly depending on the processors used, differences are expected on the random numbers that will propagate due to the inherent chaotic behaviour of the discharge.

Indeed, it was confirmed that when the seed number of this RNG is fixed in the **LPP** code (and the same procedure as
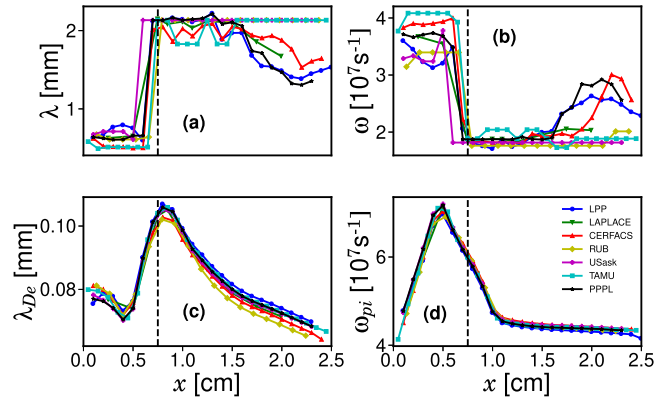
Figure 9. *Case* 1: Axial evolution of dominant mode characteristics for azimuthal electric field (azimuthal instabilities). (a) Wavelength. (b) Frequency. (c) Debye length. (d) Ion plasma frequency. Dashed line corresponds to the position of maximum magnetic field.
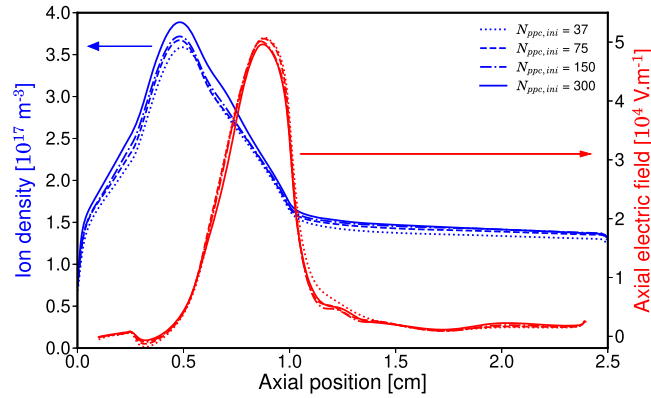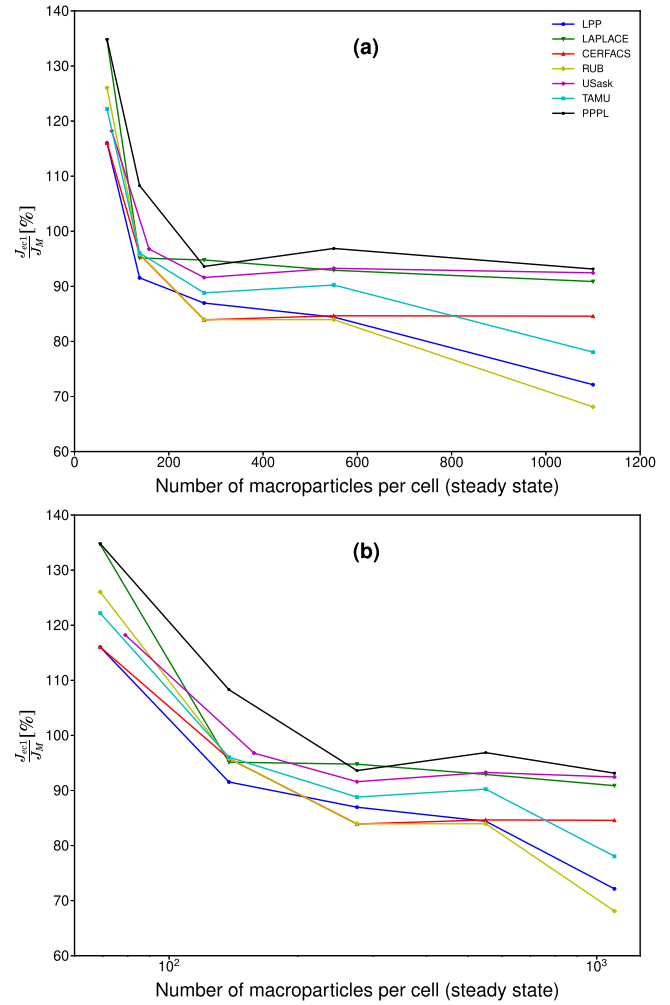

Figure 10. Azimuthally and time (between 16 and 20 μs) averaged axial profiles of axial electric field and ion density for different number of macroparticles per cell at initialisation $N_{\text{ppc,ini}}$. Results obtained with code **LPP**.

Figure 11. Mean value (between 12 and 20 μs) of $\frac{J_{ec1}}{J_M}$ depending on $N_{\text{ppc,final}}$ in normal scale (a) and semi-log scale (b). The three benchmark cases correspond to $N_{\text{ppc,final}} = 550$ (Case 1), $N_{\text{ppc,final}} = 275$ (Case 2) and $N_{\text{ppc,final}} = 1100$ (Case 3).

before is repeated), a perfect overlap is obtained for the time evolution of the current (and hence for the averaged parameters). The result is not shown here for clarity purpose but this study has shown clearly that the intrinsic turbulent nature of the discharge makes this case very sensitive and it made us more confident on the quality of the agreement obtained for this benchmark.

6. Conclusion

A 2D axial-azimuthal benchmark model for low-temperature partially magnetized plasmas has been presented here. Seven independently developed PIC codes have been used to simulate this case and their results are compared.

Despite the relative complexity of this benchmark, a good agreement was obtained on the averaged axial profiles of the main discharge parameters (axial electric field, ion density and electron temperature). All codes show the presence of a very strong kinetic instability propagating in the $E \times B$ azimuthal direction that plays a significant role on the cross-field electron transport. The characteristics of the

dominant mode of these instabilities have been compared and exhibit a good agreement between all the codes. The remaining differences of around 5% are explained by the inherent unstable nature of the discharge in this case, correlated with the fact that different RNG were used between the codes. The issue of numerical noise due to a too-low number of macroparticles was also assessed. It appears that around 250 macroparticles per cell were needed to get convergence in this 2D benchmark, which is however much less than the 10 000 macroparticles per cell used in the 1D case studied in [22].

The seven participants converged on the main purpose of this 2D benchmark that was to increase confidence in our codes by verifying that the results produced were consistent with other implementations. Moreover, as mentioned earlier, these codes are often used to simulate cases that are very computationally expensive. With this in mind, this work also gave insights on the computational efficiency of different solvers, with computational times that could vary from 2.5 to 21 days for the nominal case. It is important to highlight that for simulations of $E \times B$ discharges, the required computing
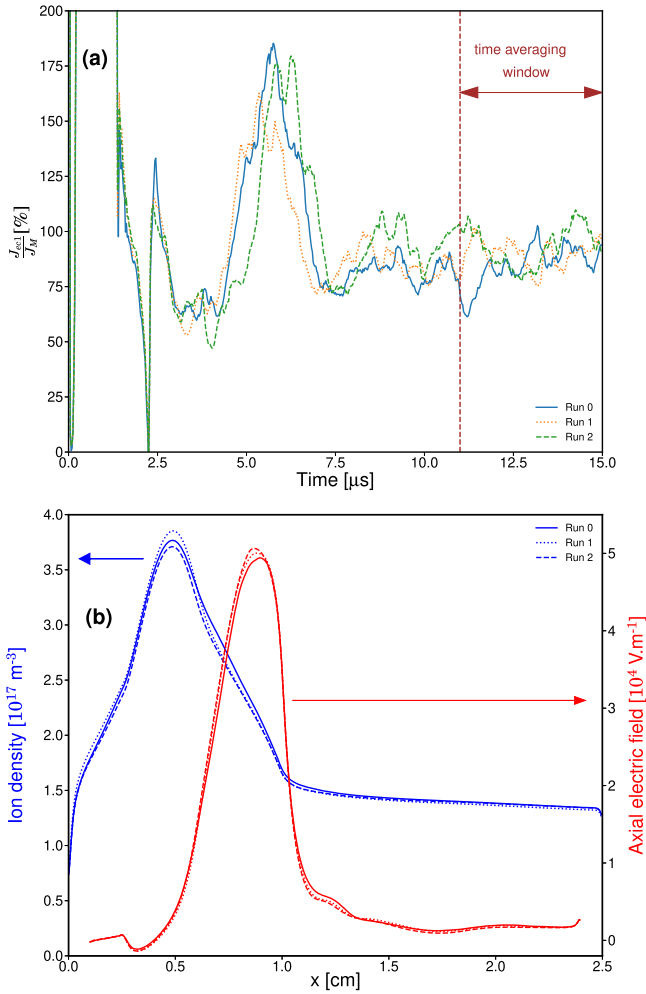
Figure 12. Comparison of 3 simulations with the same code (**LPP**) for the same input parameters (Case 2). (a) Time evolution $\frac{J_{ec1}}{J_M}$. (b) Azimuthally and time (between 11 and 15 μs as shown on (a)) averaged axial profiles of ion density (blue) and axial electric field (red).

resources are quite large (around 60 000 CPU hours in average for the nominal case of this benchmark) and it makes the need for benchmarking even more important.

Even though the case chosen here enabled to test different aspects of a 2D axial-azimuthal electrostatic PIC code, some simplifying assumptions have still been made, particularly concerning the absence of collisions. The earlier work of Turner *et al* [19] could be used to verify the Monte Carlo collision (MCC) module of PIC codes, or another benchmark case could be defined with a self-consistent treatment of ionization with the addition of neutrals and collisions. Hence, the work presented here should be considered as a step towards the benchmarking of PIC codes of low-temperature partially magnetized plasmas.

Acknowledgments

Appendix A. Supplementary implementation details

A.1. Radial magnetic field

The parameters for the radial magnetic field have the following formulas:

$$
\begin{cases}
a_1 = \dfrac{B_m - B_0}{1 - \exp\left(-\dfrac{1}{2}\left(\dfrac{x_{B_{\max}}}{\sigma_1}\right)^2\right)} \\[4ex]
a_2 = \dfrac{B_m - B_{L_x}}{1 - \exp\left(-\dfrac{1}{2}\left(\dfrac{L_x - x_{B_{\max}}}{\sigma_2}\right)^2\right)} \\[4ex]
b_1 = \left(B_0 - \dfrac{B_m.\exp\left(-\dfrac{1}{2}\left(\dfrac{x_{B_{\max}}}{\sigma_1}\right)^2\right)}{1 - \exp\left(-\dfrac{1}{2}\left(\dfrac{x_{B_{\max}}}{\sigma_1}\right)^2\right)}\right) \\[4ex]
b_2 = \left(B_{L_x} - \dfrac{B_m.\exp\left(-\dfrac{1}{2}\left(\dfrac{L_x - x_{B_{\max}}}{\sigma_2}\right)^2\right)}{1 - \exp\left(-\dfrac{1}{2}\left(\dfrac{L_x - x_{B_{\max}}}{\sigma_2}\right)^2\right)}\right)
\end{cases}
\tag{A.1}
$$

A.2. Benchmark guidelines

We listed here some general advices to perform this benchmark:

- *Temperature of emitted electrons*: it needs to be full 3D-Maxwellian electrons with a temperature of 10 eV. It was

found that the discharge behaviour was very sensitive to the temperature of these electrons.

- *Velocity and temperature diagnostics*: if the leapfrog scheme is used to solve the equations of motion, the velocity needs to be adjusted by half a time step before using it for the diagnostics. It can lead to important differences on the electron velocities and temperatures.

- *Total axial current*: at steady state, as a current equality is imposed in the system, the total axial current (ion +electron) should be constant axially. One can also

check that the divergence of the total current is null, with the divergence of the ion current being the imposed ionization source term.

Appendix B. Supplementary comparison figures

For the azimuthal instabilities (section 4.2), we also obtained a satisfactory agreement for the cases 2 and 3, as seen on figures B1 and B2 respectively.
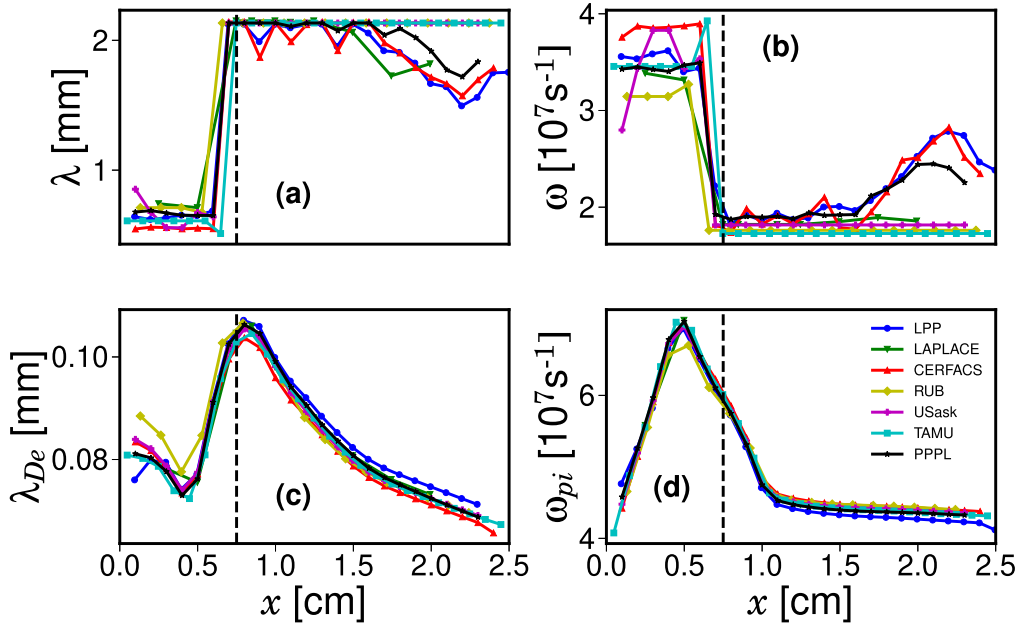
Figure B1. *Case* 2: Axial evolution of dominant mode characteristics for azimuthal electric field (azimuthal instabilities). (a) Wavelength. (b) Frequency. (c) Debye length. (d) Ion plasma frequency. Dashed line corresponds to the position of maximum magnetic field.
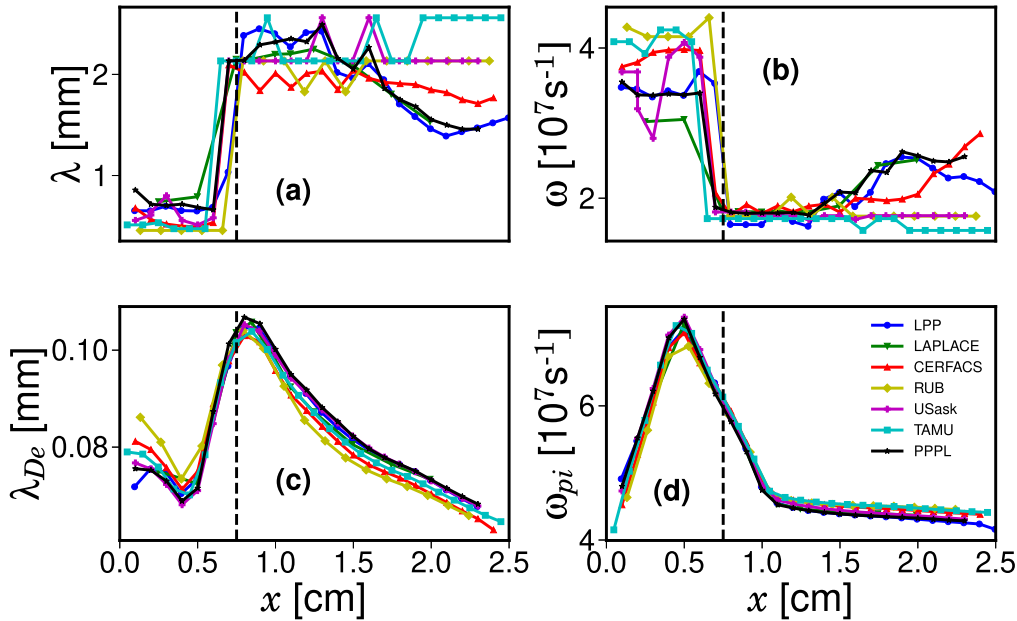

**Figure B2.** *Case* 3: Axial evolution of dominant mode characteristics for azimuthal electric field (azimuthal instabilities). (a) Wavelength. (b) Frequency. (c) Debye length. (d) Ion plasma frequency. Dashed line corresponds to the position of maximum magnetic field.

Appendix C. Supplementary data files

The averaged axial profiles of axial electric field, ion density and electron temperatures for the 3 benchmark cases displayed in section 4.1 can be found on the journal website., along with the wavelength and frequency of the instabilities dominant mode displayed in section 4.2 and appendix B, for the 3 benchmark cases. More benchmark cases on ExB discharges are currently discussed in the frame of the LAND-MARK project and the main author email can be used for more information.

ORCID iDs

T Charoy https://orcid.org/0000-0003-1191-2657

J P Boeuf https://orcid.org/0000-0003-1023-222X

A Bourdon https://orcid.org/0000-0003-3662-4651

J A Carlsson https://orcid.org/0000-0003-4614-8150

B Cuenot https://orcid.org/0000-0002-7061-3668

L Garrigues https://orcid.org/0000-0001-6855-098X

K Hara https://orcid.org/0000-0002-1816-165X

I D Kaganovich https://orcid.org/0000-0003-0653-5682

A Smolyakov https://orcid.org/0000-0002-4975-2743

A Tavant https://orcid.org/0000-0003-0010-8073

O Vermorel https://orcid.org/0000-0003-2294-8110

References

[1] Adam J *et al* 2008 *Plasma Phys. Control. Fusion* **50** 124041
[2] Boeuf J 2017 *J. Appl. Phys.* **121** 011101
[3] Anders A 2017 *J. Appl. Phys.* **121** 171101
[4] Gudmundsson J T, Brenning N, Lundin D and Helmersson U 2012 *J. Vac. Sci. Technol.* A **30** 030801
[5] Smolyakov A, Chapurin O, Frias Pombo W, Koshkarov O, Romadanov I, Tang T, Umansky M, Raitses Y, Kaganovich I and Lakhin V 2017 *Plasma Phys. Control. Fusion* **59** 014041
[6] Choueiri E 2001 *Phys. Plasmas* **8** 1411
[7] Gary S P and Sanderson J J 1970 *J. Plasma Phys.* **4** 739–51
[8] Adam J C, Heron A and Laval G 2004 *Phys. Plasmas* **11** 295–305
[9] Cavalier J, Lemoine N, Bonhomme G, Tsikata S, Honore C and Gresillon D 2013 *Phys. Plasmas* **20** 082107
[10] Ducrocq A, Adam J C, Heron A and Laval G 2006 *Phys. Plasmas* **13** 102111
[11] Lafleur T, Baalrud S D and Chabert P 2016 *Phys. Plasmas* **23** 053502
[12] Mikellides I G, Jorns B, Katz I and Lopez Ortega A 2016 *52nd AIAA/SAE/ASEE Joint Propulsion Conf. (American Institute of Aeronautics and Astronautics)*
[13] Lafleur T and Chabert P 2018 *Plasma Sources Sci. Technol.* **27** 015003
[14] Boeuf J P and Chaudhury B 2013 *Phys. Rev. Lett.* **111** 155005
[15] Tsikata S and Minea T 2015 *Phys. Rev. Lett.* **114** 185001
[16] Hara K 2019 *Plasma Sources Sci. Technol.* **28** 044001
[17] Turner M 2016 *Plasma Sources Sci. Technol.* **25** 054007
[18] Surendra M 1995 *Plasma Sources Sci. Technol.* **4** 56
[19] Turner M, Derzsi A, Donkó Z, Eremin D, Kelly S, Lafleur T and Mussenbrock T 2013 *Phys. Plasmas* **20** 013507
[20] Carlsson J, Khrabrov A, Kaganovich I, Sommerer T and Keating D 2017 *Plasma Sources Sci. Technol.* **26** 014003
[21] Bagheri B *et al* 2018 *Plasma Sources Sci. Technol.* **27** 095002
[22] Janhunen S, Smolyakov A, Chapurin O, Sydorenko D, Kaganovich I and Raitses Y 2018 *Phys. Plasmas* **25** 011608
[23] Janhunen S, Smolyakov A, Sydorenko D, Jimenez M, Kaganovich I and Raitses Y 2018 *Phys. Plasmas* **25** 082308
[24] Boeuf J P and Garrigues L 2018 *Phys. Plasmas* **25** 061204
[25] Adam J C, Serveniere A G and Langdon A B 1982 *J. Comput. Phys.* **47** 229–44
[26] Falgout R D and Yang U M 2002 *Computational Science– ICCS 2002* ed P M A Sloot *et al* (Berlin: Springer) pp 632–41
[27] Boris J P and Shanny R A 1972 *Proc. 4th Conf. on Numerical Simulation of Plasmas* (Washington DC: Naval Research Laboratory) pp 3–67
[28] Croes V, Lafleur T, Bonaventura Z, Bourdon A and Chabert P 2017 *Plasma Sources Sci. Technol.* **26** 034001
[29] Croes V 2018 *PhD Thesis* Université Paris-Saclay, France
[30] Croes V, Tavant A, Lucken R, Martorelli R, Lafleur T, Bourdon A and Chabert P 2018 *Phys. Plasmas* **25** 063522
[31] Tavant A, Croes V, Lucken R, Lafleur T, Bourdon A and Chabert P 2018 *Plasma Sources Sci. Technol.* **27** 12
[32] Boeuf J P, Chaudhury B and Garrigues L 2012 *Phys. Plasmas* **19** 113509
[33] Boeuf J P, Claustre J, Chaudhury B and Fubiani G 2012 *Phys. Plasmas* **19** 113510
[34] Boeuf J P and Chaudhury B 2013 *Phys. Rev. Lett.* **111** 155005
[35] Boeuf J P, Fubiani G and Garrigues L 2016 *Plasma Sources Sci. Technol.* **25** 045010
[36] Garrigues L, Fubiani G and Boeuf J P 2016 *J. Appl. Phys.* **120** 213303
[37] Garrigues L, Fubiani G and Boeuf J P 2017 *Nucl. Fusion* **57** 014003
[38] Coche P and Garrigues L 2014 *Phys. Plasmas* **21** 023503
[39] Fubiani G and Boeuf J P 2015 *Plasma Sources Sci. Technol.* **24** 055001
[40] Fubiani G, Garrigues L, Hagelaar G, Kohen N and Boeuf J P 2017 *New J. Phys.* **19** 015002
[41] Knuth D E 1997 *The Art of Computer Programming, Volume 2: Seminumerical Algorithms* (Boston, MA: Addison-Wesley Longman Publishing)
[42] Schonfeld T and Rudgyard M 1999 *AIAA J.* **37** 1378–85
[43] Gourdain N, Gicquel L, Montagnac M, Vermorel O, Gazaix M, Staffelbach G, Garcia M, Boussuge J F and Poinsot T 2009 *Comput. Sci. Discovery* **2** 015003
[44] Joncquieres V 2019 *PhD Thesis* Institut National Polytechnique de Toulouse, France
[45] Joncquieres V, Pechereau F, Laguna A A, Bourdon A, Vermorel O and Cuenot B 2018 *Joint Propulsion Conference, AIAA 2018-4905 (Cincinnati, Ohio, 9-11 July)*
[46] Agullo E, Giraud L and Poirel L 2017 *Int. Conf. on Domain Decomposition Methods, DD24*
[47] Poirel L 2018 *PhD Thesis* Université de Bordeaux, France
[48] Karypis G and Kumar V 1995 *MeTis: Unstructured Graph Partitioning and Sparse Matrix Ordering System, Version 4.0*
[49] Chen G, Chacón L and Barnes D C 2011 *J. Comput. Phys.* **230** 7018–36
[50] Marsaglia G 2003 *J. Stat. Softw.* **8** 1–6
[51] Sydorenko D Y 2006 *PhD Thesis* University of Saskatchewan, Canada
[52] Harase S 2014 *Math. Comput. Simul.* **100** 103–13
[53] Panneton F, L'Ecuyer P and Matsumoto M 2006 *ACM Trans. Math. Softw.* **32** 1–16
[54] Baker A H, Falgout R D, Kolev T V and Yang U M 2011 *High-Performance Scientific Computing* (Berlin: Springer)

[55] Saito M and Matsumoto M 2009 *Monte Carlo and Quasi-Monte Carlo Methods 2008* ed P L' Ecuyer and A B Owen (Berlin: Springer) pp 589–602

[56] Lafleur T, Baalrud S D and Chabert P 2017 *Plasma Sources Sci. Technol.* **26** 024008

[57] Lafleur T, Martorelli R, Chabert P and Bourdon A 2018 *Phys. Plasmas* **25** 061202

[58] Lampe M, Manheimer W, B McBride J, Orens J H, Shanny R and Sudan R N 1971 *Phys. Rev. Lett.* **26** 1221–5

[59] Tsikata S, Lemoine N, Pisarev V and Grésillon D M 2009 *Phys. Plasmas* **16** 033506

[60] Okuda H and Birdsall C 1970 *Phys. Fluids* **13** 2123

[61] Turner M 2006 *Phys. Plasmas* **13** 033506