The study of the role of collisions within the beam-plasma interaction with a finite difference Vlasov solver*

A.V. Snytnikov

Abstract. A 3D kinetic study of relaxation processes caused by the electron beam propagation in high-temperature plasma was carried out. This problem has two different spatial scales: the plasma Debye length and the beam-plasma interaction wavelength, that is, some 10 or 100 times larger, thus one needs high-performance computing to observe the two lengths at once. The mathematical model is built on the basis of the Particle-in-Cell (PIC) method and, also, the finite-difference kinetic approach is employed. The question to be answered within the model is how the numerical (model) collisions affect the course of interaction. To answer this question, the Bolzmann equation is solved with the Bhatnagar–Gross–Krook (BGK) collision term. The result is the following: the initial two-stream velocity distribution becomes uniform due to the role of collisions in the BGK equation in the same way as it happens in the collisionless PIC model with a coarse grid. This means that a coarse grid imposes collisions that are to be taken into account.

1. Introduction

This research was inspired by the effect of anomalous heat conductivity observed in the GOL-3 facility at the Budker Institute of Nuclear Physics [1]. The GOL-3 facility is a long open trap where dense plasma is heated up in a strong magnetic field when injecting a powerful relativistic electron beam of a microsecond duration. The effect is a decrease of the plasma electron heat conductivity by 100 or 1000 times as compared to a classical value for the plasma with the temperature and density experimentally observed. Anomalous heat conductivity arises because of the turbulence that is caused by the relaxation of a relativistic electron beam in the high-temperature Maxwellian plasma. The physical problem is to define the origin and mechanism of the heat conductivity decrease. This is of importance for the fusion devices because the effect of anomalous heat conductivity contributes to heating plasma and, also, to confine it. The problem of heat transport in fusion devices was widely discussed, e.g., [2, 3] and some recent works [4].

The novelty of the present research has two aspects: physical aspect and numerical aspect. From the physical point of view, heating of a beam of

plasma for a long time by now was being studied, but the details of the process, namely, parameters of the arising plasma instability are still unknown. The current theory of heating a beam uses too many simplifications like a strictly monochromatic beam, Maxwellian plasma, etc. Our objective is to determine the instabilities and to evaluate their parameters.

The numerical aspect of the novelty is that such problems are usually solved by means of the direct Boltzmann equation solution (see, e.g., [5]). The PIC method is expected to give a better picture of the turbulence and the underlying plasma instabilities, although it requires greater computer costs, as mentioned in [2], in order to obtain a physically realistic picture. The fact is, in Russia there is no finite difference Vlasov solver aimed at a high-temperature turbulent plasma simulation, though about ten foreign solvers could be mentioned. The present work is the first step in developing such a solver.

This problem needs a high-performance computation because of the necessity to have a sufficiently large grid to simulate the resonance interaction of a relativistic electron beam with plasma. A beam interacts with plasma through the electric field (similar to the Landau damping), thus it is necessary to simultaneously observe two different scales. The first is the plasma Debye length and the second is the beam-plasma interaction wavelength, which is 10 or 100 times larger than the Debye length. Since one must provide at least 8 grid cells for the Debye length, it is possible to estimate the size of a grid.

It is also necessary to provide a large number of superparticles for each cell of a grid for the simulation of turbulence. The level of non-physical statistical fluctuations is inversely proportional to the number of superparticles per cell. Thus, if there are far too few superparticles, all the physical plasma waves and oscillations will be suppressed by non-physical noise.

2. Model description

The mathematical model employed for the solution of the problem of beam relaxation in plasma consists of the Vlasov equations for the ion and electron components of plasma as well as of the Maxwell equation system. These equations in the usual notation have the following form:

$$\frac{\partial f_{i,e}}{\partial t} + \vec{v}\frac{\partial f_{i,e}}{\partial \vec{r}} + \vec{F}_{i,e}\frac{\partial f_{i,e}}{\partial \vec{p}} = 0, \qquad \vec{F}_{i,e} = q_{i,e}\Big(\vec{E} + \frac{1}{c}[\vec{v}, \vec{B}]\Big)$$

$$\operatorname{rot}\vec{B} = \frac{4\pi}{c}\vec{j} + \frac{1}{c}\frac{\partial \vec{E}}{\partial t}, \qquad \operatorname{div}\vec{B} = 0, \qquad (1)$$

$$\operatorname{rot}\vec{E} = -\frac{1}{c}\frac{\partial \vec{B}}{\partial t}, \qquad \operatorname{div}\vec{E} = 4\pi\rho.$$

In this paper, this equation system is solved by the method described in [8]. All the equations will be further given in the non-dimensional form. The following basic quantities are used for the transition to the non-dimensional form:

- characteristic velocity is the velocity of light $\tilde{v} = c = 3 \times 10^{10}$ cm/s;
- characteristic plasma density $\tilde{n} = 10^{14}$ cm^{-3};
- characteristic time $\tilde{t}$ is the plasma period (a value inverse to the electron plasma frequency) $\tilde{t} = \omega_p^{-1} = \sqrt{4\pi n_0 e^2 / m_e} = 5.3 \times 10^{-12}$ s.

The Vlasov equations are solved by the PIC method. This method implies the solution of the equation of motion for model particles, or superparticles. The quantities with the subscript i are related to ions, while with the subscript e – to electrons:

$$\frac{\partial \vec{p}_e}{\partial t} = -\big(\vec{E} + [\vec{v}_e, \vec{B}]\big), \qquad \frac{\partial \vec{p}_i}{\partial t} = \kappa\big(\vec{E} + [\vec{v}_i, \vec{B}]\big),$$

$$\frac{\partial \vec{r}_{i,e}}{\partial t} = \vec{v}_{i,e}, \quad \kappa = \frac{m_e}{m_i}, \quad \vec{p}_{i,e} = \gamma \vec{v}_{i,e}, \quad \gamma^{-1} = \sqrt{1 - v^2}.$$

The leapfrog scheme is employed to solve these equations:

$$\frac{\vec{p}_{i,e}^{\,m+1/2} - \vec{p}_{i,e}^{\,m-1/2}}{\tau} = q_i\bigg(\vec{E}^m + \Big[\frac{\vec{v}_{i,e}^{\,m+1/2} - \vec{v}_{i,e}^{\,m-1/2}}{2}, \vec{B}^m\Big]\bigg),$$

$$\frac{\vec{r}_{i,e}^{\,m+1} - \vec{r}_{i,e}^{\,m}}{\tau} = \vec{v}_{i,e}^{\,m+1/2}.$$

Here τ is the time step.

The scheme proposed by Langdon and Lasinski is used to obtain the values of electric and magnetic fields. The scheme employs the finite-difference form of the Faradey and the Ampere laws. A detailed decsription of the scheme can be found in [8]. The scheme gives the second order of approximation with respect to space and time.

3. The finite-difference Vlasov solver

In the present paper, the simplest and reliable scheme is employed for the numerical approximation. Let us employ the indices i, l, k for the spatial grid nodes and the indices p, q, r for the velocity space grid nodes. The superscript n or $n+1$ denotes the number of the time step. Although the full notation of a grid node is $f_{i,l,k,p,q,r}^n$, let us write down only the indices that are changed, so f_{q+1}^{n+1} actually represents $f_{i,l,k,p,q+1,r}^{n+1}$:

$$f^{n+1} = f + \Delta_{i,l,k}, \quad 2 \le i \le N_X - 1, \quad 2 \le l \le N_Y - 1, \quad 2 \le k \le N_Z - 1;$$

$$\Delta_{i,l,k} = -\tau\big(v_x\nabla_x f + v_y\nabla_y f + v_z\nabla_z f F_x\nabla_{v_x} f + F_y\nabla_{v_y} f + F_z\nabla_{v_z} f\big).$$

Here

$$\nabla_x f = \begin{cases} \dfrac{f - f_{i-1}}{h_x}, & v_x > 0, \\ \dfrac{f_{i+1} - f}{h_x}, & v_x < 0, \end{cases} \qquad \nabla_{v_x} f = \begin{cases} \dfrac{f - f_{i-1}}{h_x}, & F_x > 0, \\ \dfrac{f_{i+1} - f}{h_x}, & F_x < 0. \end{cases}$$

It is the same for the nabla operator for Y and Z directions.

The collision operator shape. Let us define the collision operator following [11]. Since the transport and collisions are treated separately, the Boltzmann equation is reduced in this section to the collision operator only:

$$\frac{\partial f}{\partial t} = \frac{1}{k_n} Q(f, f). \tag{2}$$

Let d be the velocity space dimension. The solution f of (2) evolves towards a steady state called a Maxwellian:

$$f_\infty = \frac{\rho}{(2\pi T)^{d/2}} \exp\Big(-\frac{|V - v|^2}{2T}\Big).$$

The Maxwellian depends on a few moments of the initial distribution:

- the density

$$\rho = \int_{R^d} f(v)\, dv;$$

- the mean/bulk velocity

$$V = \frac{1}{\rho} \int_{R^d} v f(v)\, dv;$$

- the temperature

$$T = \frac{1}{3\rho} \int_{R^d} |V - v|^2 f(v)\, dv.$$

In further computations with the Boltzmann equation, the Bhatnagar–Gross–Krook (BGK) collision operator will be used. The approximation [12] rests on the fact that the effect of the collision term in (2) is mainly in pushing the distribution function towards the equilibrium state. Therefore, the idea here is to simply replace $Q(f, f)$ by a difference between f and its equilibrium value, divided by the relaxation time controlling the evolution speed:

$$Q^{\text{BGK}}(f) = \frac{1}{\tau}(f_\infty - f).$$

4. Testing the model

The first test of the Boltzmann solver with collisions is the relaxation of an arbitrary distribution function to a Maxwellian (Figure 1). Then the conservation of impulse and energy is tested. Figure 2 shows the deviation of the energy from the initial value.
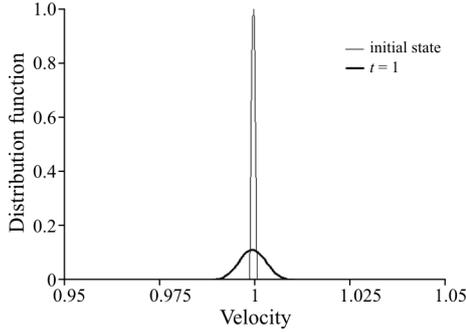
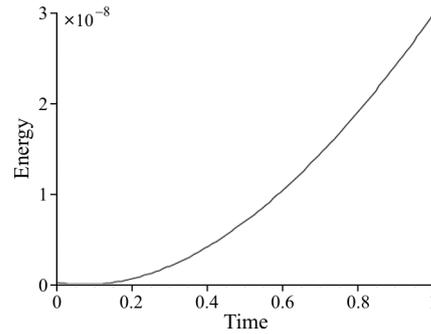**Figure 1.** Relaxation of a point distribution function to a Maxwellian

Figure 2. Deviation of energy from the initial value

5. Studying the role of collisions in beam-plasma interaction simulation

The PIC model of beam-plasma interaction is declared to be collisionless. But in fact in the course of numerical simulation, the model particles do deviate from the straight trajectory and they may also either gain or loose energy. This occurring means that in the PIC plasma model there are collisions. The reason of these "collisions" is purely numerical (self-forces, round-off errors, etc.), they are a non-physical collision, but, nevertheless, they do exist.

Thus, we come to a necessity to study these numerical collisions. In particular, the following questions are to be answered:

- What is the actual level of collisions in the model?
- Could the model be reckoned as collisionless?
- If yes, under what conditions?
- Or, will a certain collisional model with a necessarily low collision frequency give the same results as the PIC model with numerical collisions?

To answer these questions, one must measure the actual deviation of particles in the PIC model, and figure out the numerical "collision frequency". Then if the real physical collision frequency is considerably greater than the virtual collision frequency, then the model is really collisionless.

Another question will be addressed the following way: a collisional finite-difference Boltzman solver will be taken in order to compare the evolution of the initially two-stream distribution function with the same process within the collisionless PIC model.
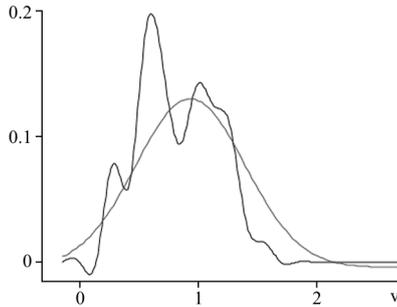

Figure 3. Mixing the initially two-stream distribution

To answer this question, the Bolzmann equation is solved with the Bhatnagar–Gross–Krook collision term. The result is the following: the initially two-stream velocity distribution becomes somewhat like Maxwellian (Figure 3) due to the role of collisions in the BGK equation in the same way as it happens in the collisionless PIC model with a coarse grid. This means that a coarse grid imposes collisions that are to be taken into account.

Problem statement. Let the 3D computational domain have the shape of a cube with the following dimensions:

$$0 \leq x \leq L_X, \qquad 0 \leq y \leq L_Y, \qquad 0 \leq z \leq L_Z.$$

Within this domain, there is model plasma. The model plasma particles (superparticles) are uniformly distributed within the domain. The density of plasma is specified by the user as well as the electron temperature. The temperature of ions is considered to be zero. Beam electrons are also uniformly distributed along the domain. Thus, a beam is considered to be already present in plasma, and the effects that occur while the beam is entering the plasma, are beyond the scope of this study.

The superparticles simulating beam electrons differ from those simulating plasma electrons by their energy value. Beam electrons initially have the energy of about 1 MeV, while plasma electrons have the energy of about 1 keV. Moreover, beam electrons have one direction of movement strictly along the axis X, and plasma electrons have the Maxwellian velocity distribution for all the three dimensions.

There is one more difference between the superparticles simulating beam electrons and plasma electrons. They have different weights when computing the current density and the charge density. Let us consider the ratio of the beam density to the plasma density, α (usually α varies from 10^{-3} to 10^{-6}), then the contribution of a beam electron superparticle is α from the contribution of a plasma electron superparticle. In such a way it is possible to provide a large number of beam superparticles.

The main physical parameters of the problem under study are the following: the density and the temperature of the plasma electrons, the ratio of the beam density to the plasma density and the energy of a beam.

6. Parallel implementation

The program was parallelized by the domain decomposition method. The computational domain is divided into parts along the direction orthogonal to the direction of a beam (along the axis Y, the beam moving along the axis X). The computational grid in the whole domain is divided into equal parts (subdomains) along the axis Y. Each subdomain is assigned to a group of processors (in the case of a multicore system, a single core would be called a processor, since no hybrid parallelization like MPI+OpenMP is employed, just mere MPI). Furthermore, the superparticles of each subdomain are uniformly distributed between processors of a group with no regard to their position, as is shown in Figure 4.
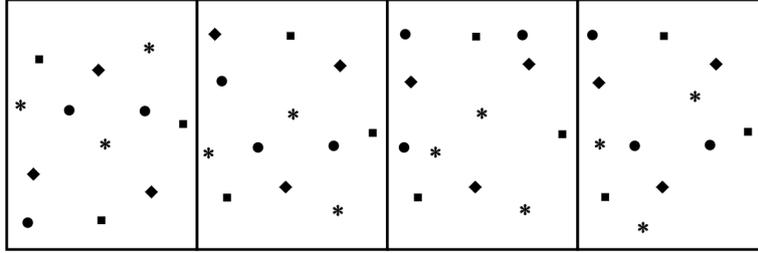

Figure 4. The scheme of domain decomposition. The computational domain is divided into 4 subdomains. The superparticles of each subdomain are uniformly distributed between four processors with no regard to their position. Different symbols (circle, square, diamond, star) denote superparticles belonging to different processors in the same subdomain

Every processor in the group solves the Maxwell equations in the whole subdomain, and exchanges boundary values of the fields with processors assigned to the adjacent subdomains. Then the equations of motion for superparticles are solved, and the 3D matrix of the current density and the charge density are evaluated by each processor. However, since a processor has only a part of superparticles located inside the subdomain, it is necessary to sum the matrices through all the processors of the group to obtain the whole current density matrix in the subdomain. The interprocessor data exchange is performed by the MPI subroutines.

Parallelization efficiency. A parallel program has been primarily developed for the simulation of the beam interaction with plasma on large computational grids and with large numbers of superparticles. That is why the parallelization efficiency was computed in the following way:

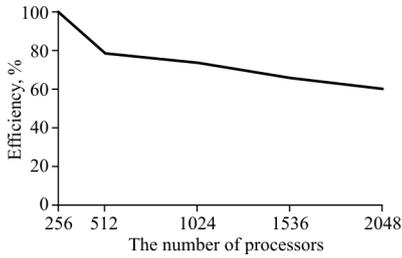$$k = \frac{T_2}{T_1} \times \frac{N_1}{N_2} \times \frac{S_2}{S_1} \times 100\,\%. \tag{3}$$

Figure 5. The parallelization efficiency measured on MVS-100K cluster of the Joint Supercomputer Center of the RAS. The grid size along Y and Z is 64 nodes, the grid size along X is equal to the number of processors, 150 superparticles per cell are used for all the cases

Here T_1 is the computation time with N_1 processors, T_2 is the computation time with N_2 processors, S_i is the characteristic size of the problem in each case, $i = 1, 2$.

Here the characteristic size is the grid size along X axis. In this section, the characteristic size S is proportional to the number of processors N. This means that the workload of a single processor is constant. The purpose of such a definition of efficiency is to find out what the communication overhead is when the number of processors is increased with a constant workload for each processor. In the ideal case, the computation time must remain the same (the ideal $k = 100\,\%$). In the computations dealing with the efficiency evaluation, only the grid size along X axis was increased, all other parameters remaining constant, the results are shown in Figure 5.

7. Electron heat conductivity in computational experiments

In order to simulate the interaction of an electron beam with plasma, the following values of the main physical parameters were set:

- electron temperature of 1 KeV;

- the mass of ion 1836 electron masses (hydrogene ions);

- plasma density of 10^{17} cm^{-3};

- the ratio of beam density to plasma density of 10^{-3};

- beam energy of 1 MeV;

- the size of the domain $L_X = 0.065$ cm and $L_Y = L_Z = 0.008$ cm;

- the grid size of $512 \times 64 \times 64$ nodes, 150 superparticles per cell.

Density modulation was observed in the computational experiments. The amplitude of the modulation is $220\,\%$ of the initial value of density. Modulation in this case means regions with very high or a very low density presence in the previously uniform-density plasma as is shown in Figure 6. It is seen that the density becomes non-uniform not only along the direction of the beam (X axis), but also along Y axis. Thus, the density is modulated not only along X, that seems quite natural, but also along Y and Z. This

Figure 6. Electron density contours in the XY plane, $z = L_Z/2$, the moment of time $t = 91.7$ (in terms of the plasma period). The density is given in terms of the initial values of density

corresponds the physics of the process sufficiently well, because it is known that the waves propagating in plasma due to beam relaxation have all the three components of a wave-vector as non-zeros.

Moreover, it was found out that the movement of beam electrons becomes eddy as a result of the beam interaction with plasma. At the initial moment of time, all the beam electrons have the same velocity strictly along X axis. This results in the eddy structure of the electron heat flux

$$q(x, y) = \left| T_e(x, y, L_Z/2) \vec{v}_e(x, y, L_Z/2) \right|.$$

The electron heat flux also gains modulations along X and Y. Moreover, there are regions with a very low value of the electron heat flux as is shown in Figure 7 (less than 1% of the initial electron heat flux). This means that only in small and isolated regions, the value of the electron heat flux is close to the initial one, but generally, in the computational domain, an electron heat flux is very low. Thus, the domain as a whole has a very low heat conductivity after the beam relaxation.
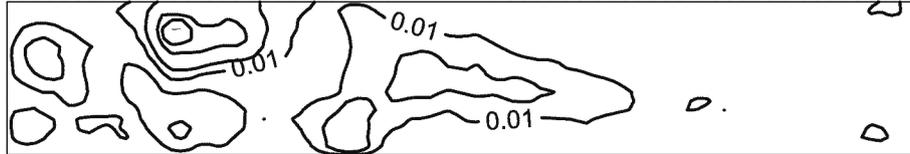

Figure 7. The electron heat flux contours in the XY plane, $z = L_Z/2$, the moment of time $t = 91.7$ (in terms of the plasma period). The flux is given in terms of the initial values of the electron heat flux

8. Conclusion

The parallel implementation of the numerical model of the interaction of an electron beam with plasma is described, and the parallel efficiency is considered. The worktime of the program has been measured on various clusters and then analyzed. It was found out that the memory bandwidth of the cluster is the most important feature for the PIC programs for attaining

a good performance. For the proposed program it has been shown that there are great potentialities for the optimization and decrease of the computation time.

It is necessary to underline that for a grid with a small number of nodes or with a small number of superparticles per cell, there is no resonance interaction between a beam and plasma, and the physical effect of the heat conductivity decrease is not observed. Thus, high-performance computing is essentially necessary to solve the present problem.

References

[1] Astrelin V.T., Burdakov A.V., Postupaev V.V. Generation of ion-acoustic waves and suppression of heat transport during plasma heating by an electron beam // Plasma Physics Reports. — Vol. 24, No. 5. — P. 414–425.

[2] Cohen B.I., Barnes D.C., Dawson J.M., et al. The numerical tokamak project: simulation of turbulent transport // Computer Physics Communications. — 1995. — Vol. 87, Iss. 1–2. — P. 1–15.

[3] Jaun A., Appert K., Vaclavik J., Villard L. Global waves in resistive and hot tokamak plasmas // Computer Physics Communications. — 1995. — Vol. 92, Iss. 2–3. — P. 153–187.

[4] Gardarein J.-L., Reichle R., Rigollet F., Le Niliot C., Pochea C. Calculation of heat flux and evolution of equivalent thermal contact resistance of carbon deposits on Tore Supra neutralizer // Fusion Engineering and Design. — 2008. — Vol. 83, Iss. 5–6. — P. 759–765.

[5] Wright J.C., Bonoli P.T., D'Azevedo E., Brambilla M. Ultrahigh resolution simulations of mode converted ion cyclotron waves and lower hybrid waves // Computer Physics Communications. — 2004. — Vol. 164, Iss. 1–3. — P. 330–335.

[6] TOP500 Supercomputing Sites. — http://www.top500.org.

[7] "SKIF" – Joint Belorussian-Russian Program. — http://www.skif.bas-net.by/index_en.htm.

[8] Vshivkov V.A., Grigoryev Yu.N., Fedoruk M.P. Numerical "Particle-in-Cell" Methods. Theory and Applications. — VSP, 2002.

[9] Krall N., Trivelpiece A. Principles of Plasma Physics. — McGraw-Hill, 1973.

[10] Bidsall Ch., Langdon B. Plasma Physics via Computer Simulation. — McGraw-Hill, 1985.

[11] Narayan A., Klickner A. Deterministic Methods for the Boltzmann Equation. // arXiv:0911.3589. Numerical Analysis Series.

[12] Bhatnager P.L., Gross E.P., Krook M. // Phys. Rev. — Vol. 94. — P. 511.