

## Using of Hybrid Cluster Systems for Modeling of a Satellite and Plasma Interaction by the Molecular Dynamics Method

Leonid Zinin<sup>[0000-0002-9098-9515]</sup> (✉), Alexander Sharamet<sup>[0000-0002-5306-0141]</sup> (✉) and Sergey Ishanov<sup>[0000-0002-8908-8955]</sup>

Immanuel Kant Baltic Federal University, Kaliningrad, Russia  
leonid.zinin@gmail.com, alexsharamet@gmail.com, sishanov@kantiana.ru

**Abstract.** This article deals with a model of interaction between a positively charged microsatellite and thermal space plasma. The model is based on the method of molecular dynamics (MMD). The minimum possible number of particles necessary for modeling in the simplest geometric problem formulation for a microsatellite in the form of a sphere 10 cm in diameter is ten million. This value is determined by the plasma parameters, including the value of the Debye radius, which is the basis for estimating the spatial dimensions of the modeling domain.

For the solution, MPI and CUDA technologies are used in the version of one MPI process per node. An intermediate layer in the form of multithreading, implemented on the basis of the C++ library of threads, is also used, this provides more flexible control over the management of all kinds of node memory (video memory, RAM), which provides a performance boost. The GPU optimizes the use of shared memory, records the allocation of registers between threads and the features associated with calculating trigonometric functions.

The results of numerical simulation for a single-ion thermal plasma showed significant changes in the spatial distribution of the concentration around the satellite, which depends on three main parameters - the temperature of the plasma components, the velocity of the satellite relative to the plasma and the potential of the spacecraft. The presence of a region of reduced ion concentration behind the satellite and the region of condensation in front of it is shown.

**Keywords:** Parallel computing · Thermal space plasma · Charged satellite · Molecular dynamics method

## 1 Introduction.

The presence of an electric charge of the satellite is an important factor affecting the experimental changes in thermal near-Earth space plasma. The influence of this interaction on the results and processing of measurements has been widely discussed and continues to be discussed in literature (see, for example, the analysis of the measurement results from the Auroral probe [1, 2]). Modern measurement models take into account the effect of temperature anisotropy [3-5]. The analysis of experimental measurements of thermal plasma and their simulation unambiguously indicate that the positive potential of the satellite significantly affects the spatial distribution of the ion flux values. In addition, the temperature anisotropy affects the angles of ion arrival [5].

These features of measurements can significantly complicate the processing of experimental data in determining the direction of the ion flux, and consequently, the magnitude of the ion velocity components.

At present, there are a number of successful interaction models of a charged satellite and plasma. Separately note the NASA model (NASCAP-2K [6]) and ESA (SPIS [7,8]), see also the article [9], where the results of calculations and model codes are compared.

There are enough examples of successful simulation of the electric field distribution around the satellites real form. Thus, for example, [10] present calculations based on the variants of NASCAP and SPIS models have already mentioned for the SCATHA satellite and the modern telecommunication satellite in the geostationary orbit.

Most modern models are based on the Particle-in-Cell method, which has proven its effectiveness. At the same time, this method does not allow us to obtain the trajectories of individual plasma particles, which it is necessary in some cases. The method of molecular dynamics (MMD) can be applied for any parameters of thermal plasma, regardless of the magnitude of the Debye radius, in spite of its computational complexity.

In this article, we consider an interaction model of thermal plasma with a charged microsatellite, which has the simplest spherical shape. The model is based on the molecular dynamics method.

## 2 Description of the Model

Now the method of classical molecular dynamics is being widely used for modeling physical, chemical and biological systems.

The research of Alder and Wainwright [11, 12] can be considered as the first attempt to apply MMD. (A review of the application of MMD in physics and chemistry is given in the article [13]). In classical MMD, quantum and relativistic effects are neglected, and the particles move according to the laws of classical mechanics.

Let us consider the 3D simulation of a charged microsatellite and thermal Maxwellian plasma in detail [14-16]. Let the plasma consist of protons and electrons. The force of Lorentz acts on every particle in the process of its motion:

$$m\mathbf{a} = q\mathbf{E} + q[\mathbf{v} \times \mathbf{B}], \quad (1)$$

here  $m$  is the mass of the ion or electron,  $q$ ,  $\mathbf{a}$ ,  $\mathbf{v}$  is its charge, acceleration and velocity, respectively,  $\mathbf{E}$  is the electric field strength,  $\mathbf{B}$  is the magnetic field.

The electric field strength at any point in the simulation area is calculated according to the Coulomb law. For its calculation, all the electrons and ions which are in the computational domain are used. The electric field from the satellite is also taken into account.

$$\mathbf{E} = \sum_{i=1}^n \frac{q_i}{4\pi\epsilon_0 r_i^2} \frac{\mathbf{r}_i}{r_i} + \mathbf{E}_{sat} \quad (2)$$

Thus, for each particle one can find the value and direction of force in the right side of the equation (1) to obtain the positions and velocities of the particles in the new time level.

The particles at the initial time of simulation are evenly distributed in space and have a Maxwell velocity distribution.

$$F(v) = \left(\frac{m}{2\pi kT}\right)^{3/2} \exp\left(-\frac{mv^2}{2kT}\right) \quad (3)$$

where  $v$  - velocity of the particle,  $m$  - its mass,  $k$  - Boltzmann constant,  $T$  - temperature.

The number of particles  $N$  in the speed range  $[v, v + dv]$  for the density  $n$  is calculated as

$$N = \int_v^{v+dv} 4\pi n v^2 F(v) dv \quad (4)$$

The model domain is a cube with edge of 1 m. In its center there is a microsatellite which is a ball of 5 cm radius.

The relative velocity of the plasma and the satellite is directed along the OX axis. The magnetic field in these calculations was not taken into account. The temperature of ions and electrons is the same and it is equal to 5000 K, which corresponds to the conditions in the Earth's magnetosphere. The total number of particles in the modeling area was  $2 \times 10^7$ . Thus, the unperturbed concentration of ions and electrons was  $n = 10 \text{ cm}^{-3}$ . The plasma velocities relative to the satellite were assumed to be equal to 10 km/s and 20 km/s. The potential of the satellite  $U_{sat}$  was positive and its values were taken equal to +5 V and +10 V. The time step was  $10^{-8}$  seconds. Such a small amount is due to the need to accurately calculate the trajectories of electrons which thermal velocities are about 500 km/sec. The spatial step for the selected time step value does not exceed 5 mm. At each time step, some of the particles leave the simulation area, while others are injected into this region in accordance with the specified unperturbed Maxwell distribution function. The charge of electrons that hit the satellite surface is compensated by photoelectron emission.

### 3 Method of Calculations.

#### Algorithm Analysis and Parallel Algorithm Scheme

The model has a good margin of scaling on large computing systems, but, nevertheless, it requires huge computing resources. To calculate the particle's coordinates at the next time step, it is necessary to calculate the electric field strength for this particle, which includes the sum of the stress vectors from all other plasma particles and the electric field strength from the satellite.

The convergence of this method requires from 3000 to 7000 time iterations. With a large number of equations, influenced by the size of the region and the concentration of particles in  $\text{cm}^{-3}$ , amount of calculations is significantly increased. To solve this problem a huge computing power, concentrated on each of the nodes, is required. So, for example, the computational complexity of the algorithm for the calculated region of  $1 \text{ m}^3$  and the concentration of  $10 \text{ cm}^{-3}$  is  $10^{17}$ .

To solve the problem we use cluster, with MPI technology to communicate between nodes. On each node we have two accelerators and use CUDA to calculate on them. To optimize GPU communication inside node and reduce communication between nodes we don't use a classical scheme "MPI process per GPU". Our scheme assumes that one MPI thread contains three CPU threads (one for communication, other to GPU). To create these threads we use "C++11 thread" library.

For MPI communication we use a classical master-slave scheme. Master reads information about the satellite, a field size, a magnetic field and other. On the second step it generates particles or reads their state from the file. It is very important to have an option to load data from some iteration, because the calculation may take a long time. Later master smashes data into pieces and sends them to slaves. To send data we don't use MPI\_Bcast/Mpi\_Ibcast, instead we use a cycle with MPI\_Isend and synchronization after. This is faster on Intel mpi compilers, and we can control the size of any portion of data. The same way we collect data after iteration. MPI\_Isend has another advantage, it doesn't use buffer. Master process gathers data with MPI\_Recv and we add the correct address for any pies, so data isn't copied at all in an MPI subsystem.

MPI process consists of two or more threads. These threads have master-slave architecture too. At the beginning the master thread asks CUDA driver and creates as many threads as devices in node. Another thread function is the synchronization between controlling GPU threads and MPI communications. For synchronization we use barriers. The barrier is created on the basis of the "Singleton" pattern and uses mutable exclusion with conditional variables.

So the master thread receives (mpi-slave) or sends (mpi-master) tasks. This task is divided into portions and distributed to slave threads, followed by waiting for the result with barrier synchronization. After calculation some particles leave field or pierce satellite, and disappear. On their places the master thread generates new particles. We don't do it in parallel because rand depends on started seed, and if we try to use time as this seed we should de-synchronize time of starting command. Of course it can be found another way to solve this problem, but the task is small now

and the current method is working well. The last master step is saving results to the file (only in some iteration) and sending input data to MPI-slaves.

### **CUDA calculation model and optimization.**

The calculation is based on starting some consistent kernels. Kernel is started in CPU function, but calculated at GPU. The main kernel is an electric tension vector calculating for the electric field, because this part of the program contains the main complexity of the algorithm. After it, the new positions of the particles are calculated, later consistent on the CPU calculates an intersection with the satellite and leaving of the space. The only feature of the recalculation particle position is to save results to random access memory (non GPU) with the zero-copy technology, because we need further global exchange and current step data is done once.

Let's review the electric tension vector calculation. The first idea is to give one particle and its electric tension vector one GPU thread, taking into account that we have all particles, but recalculate only some parts of them which are defined at planning iteration in an MPI layer and later in cpu-thread layer.

Besides this condition, amount of threads can be large and greatly exceed the limit of GPU thread, which can be calculated on GPU at one time, so scheduler should be started and exchange calculation blocks. To avoid scheduler loading we give each thread some of the particles to calculate them. Obviously, all discussed above will be effective only with a multiple excess of the number of particles over the number of started threads.

Common model of calculation for one thread is look like:

- 1). Load data from DRAM (particles data for which the vector is calculated).
- 2). Cooperative load by block threads portion of particles in shared memory, synchronization
- 3). Cycle of calculation of the partial vector of electric tension, synchronization
- 4). Return to step 2, while all data will not be read.
- 5). Choose another particle and Return to step 1

The first level of optimization is coalesced requests to DRAM, and shared memory hasn't bank conflicts. Further, three parameters must be considered to achieve 100% theoretical and practical performance.

- 1). Loading of streaming multiprocessors (SM)
- 2). Size of shared memory
- 3). Count of registers per SM

Due to the fact that in common situations we have the number of particles significantly exceeding the capabilities of computing devices, it will be better to start from the original data about optimal parameters from the driver and execute an appropriate number of blocks and threads in block.

On the current device per SM 1536 and a max block size is 1024, so for full loading we use size of block 512 or less.

Amount of shared memory per SM consists of 48Kb, and if we have a block size equal to 512 it will split into three blocks. This size is enough to load all necessary data.

The last optimization parameter is registers. We have no direct influence on it, but we can get it count in use by different debugger tools, and take into account that it is affected by the size and the type of variables in kernel and compiler optimization.

For this parameter optimization we move some parameters from an electric tension formula, to multiply them later, thereby reducing the amount of used registers and allowing more threads to be started

Also one more GPU optimization is the optimization of the algorithm with hard to device trigonometric functions(sin, cos, tg ,...), we change these parts of algorithm to their math analogs with the similar coefficient, which reduce the number of operations to the whole calculation.

In order to evaluate the scaling of the algorithm, we estimated the speed of the algorithm for one and two GPUs as shown in Table 1.

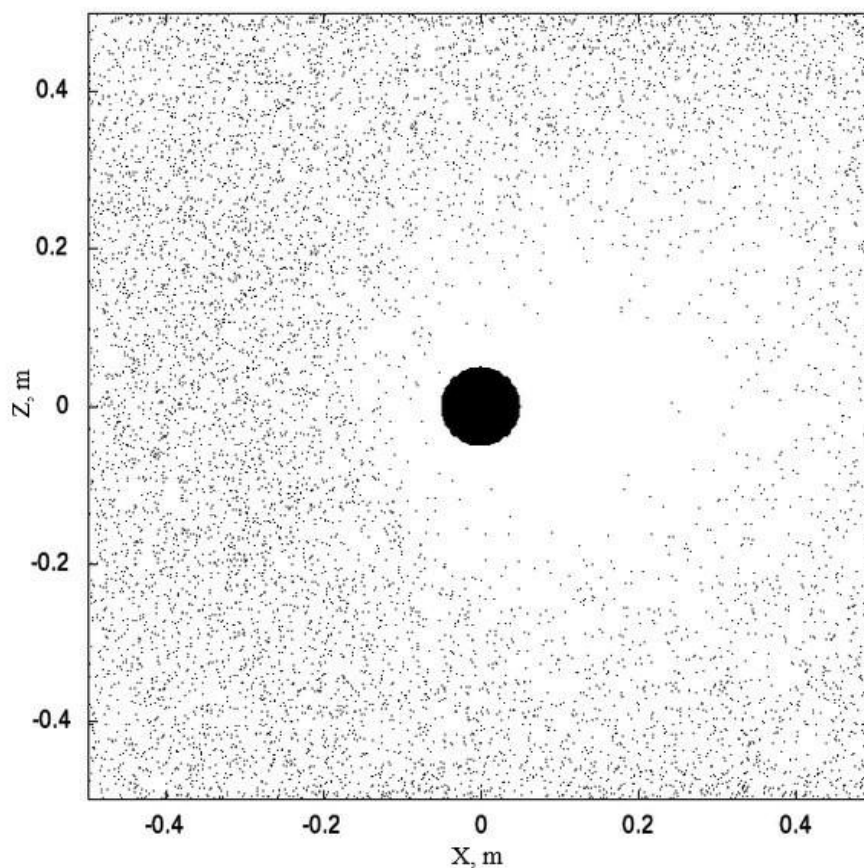
**Table 1.** Algorithm scalability

Numbers of particles	Time in one GPU(sec)	Time in two GPU(sec) (in one node)	Performance increase
$10^4$	0.621	0.567	9%
$10^5$	4.549	2.923	55%
$10^6$	44.40	28.69	54%
$10^7$	398.69	272.00	46%

Based on the sibling table, it is obvious that the task scales well. With a particle count of  $10^4$ , the performance gain is not large, because the amount of data is not sufficient to load two GPU at once. As the number of particles increases, the performance begins to decrease, because the effects associated with data transfer start to play the more important role.

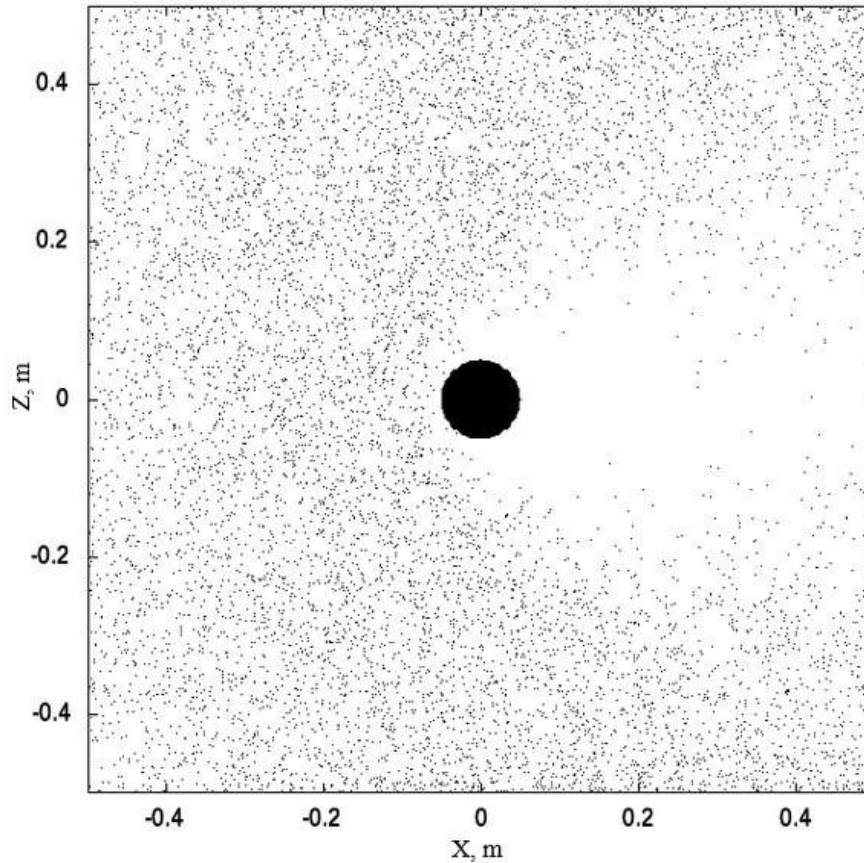
#### 4 Results of Numerical Modeling

Figures 1-4 show the results of numerical modeling of the interaction of a charged satellite and thermal Maxwellian plasma. The spatial distribution of the concentration of hydrogen ions in the plane XOZ ( $y = 0$ ) after 3500 time steps beginning from the numerical experiment, which represents  $3.5 \cdot 10^{-5}$  sec from the time the initial modeling is presented. Each point represents a proton, crossed the XOZ plane of the two time steps. Initial conditions were only a spatially uniform distribution of protons and electrons with isotropic velocities Maxwellian distribution corresponding to the temperature 5000 K. The plasma velocity relative to the satellite is  $V$  and consisted of 10 km/sec (Fig. 1 and 3) and 20 km/sec (Fig. 2 and 4) along the horizontal axis OX. For all calculations, the plasma velocity relative to the satellite is directed along the horizontal axis OX (in all figures from left to right).  $U_{sat}$  satellite potential of +5 Volts (Fig. 1 and 2) and +10 Volts (Figures 3 and 4), was adopted.



**Fig.1.** The spatial distribution of the concentration of hydrogen ions. The axes are the distance from the center of the microsatellite in meters. The plasma velocity relative to the satellite is directed along the horizontal axis OX.  $V = 10 \text{ km / s}$ .  $U_{\text{sat}}$  is electric potential of satellite equal to +5 Volts

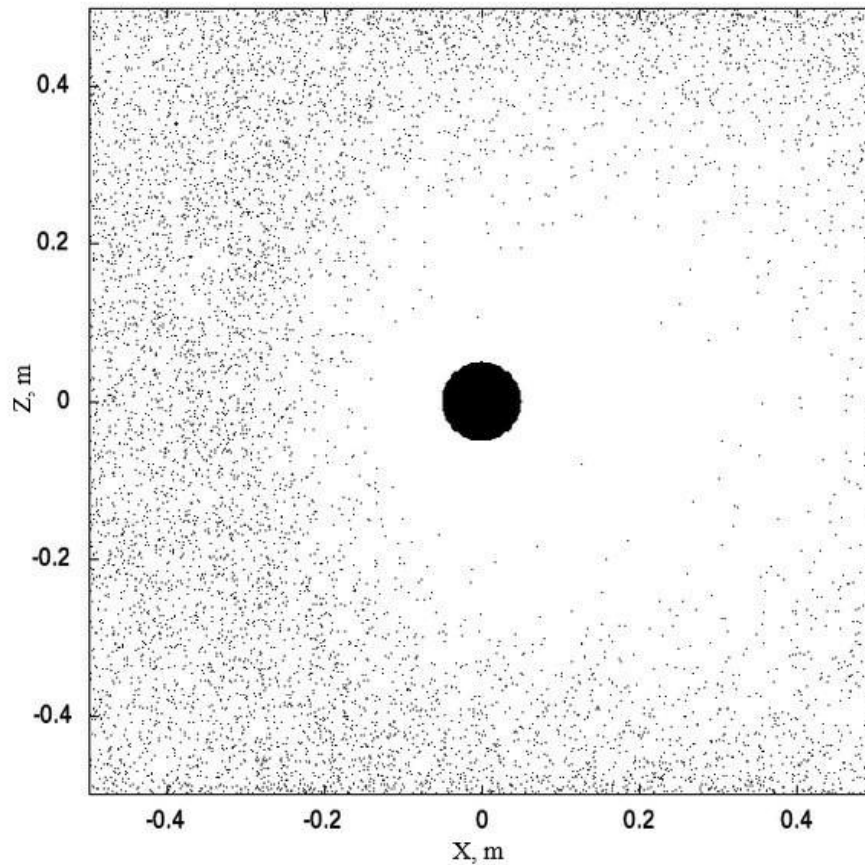
In all the figures there is a strongly marked ion shadow of microsatellite. Spatial characteristics of the shadows become more clear with increasing relative velocity of the plasma and the satellite and the satellite's positive potential. Thus, for a plasma speed of 10 km/s, and the electric potential of the satellite 5 Volts (Figure 1), the characteristic dimensions of the shadow of the satellite ion about 20 cm. Moreover, before a satellite there is also a region of a few cm greatly reduced ion concentration. By increasing the relative speed of 20 km/s at the same value of the satellite electric potential (Figure 2) ion shadow increases up to half of meter (maybe further, but we are limited in spatial scales of modeling the field).



**Fig. 2.** The same as on Fig.1 and 2, but for plasma velocity relative to the satellite  $V = 20$  km/s

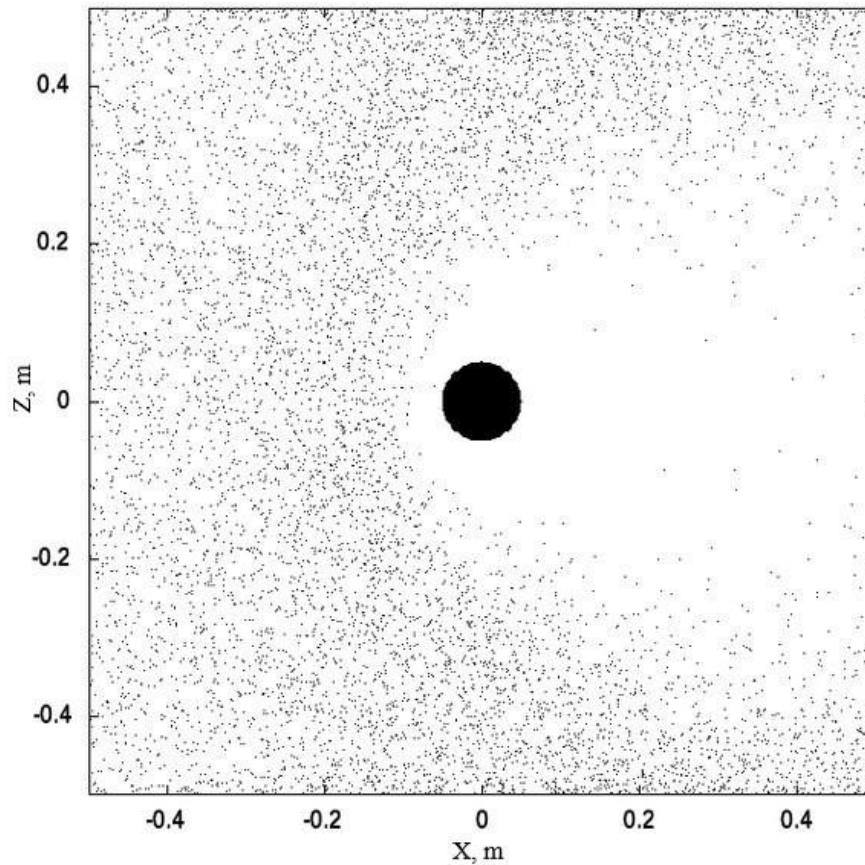
Consider the effect of increasing the positive potential of the satellite. Figures 3 and 4 show the results of modeling the spatial distribution of protons near a satellite with an electrical potential of +10 Volts. When plasma speed of 10 km/sec region of reduced concentration increases significantly as a companion to and behind it. Before the satellite, at a distance of about 10 cm, concentration drops almost to zero. Shadow is greatly increased in the transverse dimension and the satellite extends almost to the boundaries of modeling region. A different picture is observed for twice the relative velocity of the satellite and plasma. Directly in front of the satellite it is observed a thin layer (3 - 5 cm) lower concentration of protons. But directly in front of this layer the concentration exceeds the background undisturbed conditions. The shadow of the satellite is clear and has about half a meter in diameter. The length of the shade clearly exceeds the spatial domain simulations and probably reaches the meter and more. As expected, pattern is symmetrical along axes OY (not shown), and OZ.





**Fig. 3.** The same as on Fig.1, but for satellite electric potential  $U_{\text{sat}} = +10$  Volts

The features of the spatial distribution of the proton concentration near the charged satellite qualitatively correspond to calculations on other models and theoretical concepts. Thus, calculations for the hydrodynamic model for a dense plasma [17] indicate the presence of an ionic shadow behind a satellite with spatial dimensions on the order of the Debye radius. Note that the presence of this shadow was predicted in [18] on the basis of analytical calculations.



**Fig. 4.** The same as on Fig.2, but for satellite electric potential  $U_{\text{sat}} = +10$  Volts

## 5 Conclusion

The numerical simulation results show a significant distortion of the spatial distribution of thermal ions in the presence of the positive potential of the satellite, which are expressed in the presence of an area of low concentration to the satellite and ion shadows behind him. These distortions are the more pronounced, the greater the relative velocity of the satellite and the plasma and the magnitude of the positive potential of the satellite.

Thus, the algorithms considered in this paper are easily scaled and can be modified for the case of real microsatellites, such as 1U-3U CubeSats. Modeling the interaction

between thermal plasma and larger satellites is currently difficult due to the considerable increase in computational complexity with increasing modeling domain.

**Acknowledgements.** This work was made by support of Russian Foundation for Basic Research, grant 18-01-00394

## References

1. Bouhram, M., Dubouloz, N., Hamelin, M., Grigoriev, S.A., Malingre, M., Torkar, K., Veselov, M.V., Galperin, Y., Hanasz, J., Perraut, S., Schreiber, R., Zinin, L.V.: Electrostatic interaction between Interball-2 and the ambient plasma. 1. Determination of the spacecraft potential from current calculations. *Ann. Geophys.* **20**, 365-376 (2002)
2. Hamelin, M., Bouhram, M., Dubouloz, N., Malingre, M., Grigoriev, S.A., Zinin, L.V.: Electrostatic interaction between Interball-2 and the ambient plasma. 2. Influence on the low energy ion measurements with Hyperboloid. *Ann. Geophys.* **20**, 377-390 (2002)
3. Zinin, L.V., Galperin, Y.I., Gladyshev, V.A., Grigoriev, S.A., Girard, L., Muliarchik, T.M.: Modelling of the anisotropic thermal plasma measurements of the energy-mass-angle ion spectrometers onboard a charged satellite. *Cosmic Research.* **33**, 511-518 (1995)
4. Zinin, L.V., Galperin, Yu.I., Grigoriev, S.A., Muliarchik, T.M.: On measurements of polarization jet effects in the outer plasmasphere. *Cosmic Res.* **36**, 39-48 (1998)
5. Zinin, L.V.: Modelling of thermal H<sup>+</sup> ions on charged satellite taking into account temperature anisotropy. *Vestnik IKSUR*, 56-63 (2009) in russian
6. Mandell, M. J., Katz, I., Hilton, M. et al.: Nascap-2K spacecraft charging models: algorithms and applications. In: 2001: A spacecraft charging odyssey. Proceeding of the 7th Spacecraft Charging Technology Conference, pp. 499—507. ESTEC. Noordwijk. The Netherlands. (2001)
7. Hilgers, A., Clucas, S., Thiébaud, B., Roussel, J.-F., Matéo-Vélez, J.-C., Forest, J., Rodgers, D.: Modeling of Plasma Probe Interactions With a PIC Code Using an Unstructured Mesh. *IEEE Trans. Plasma Sci.* **36**, 2319-2323 (2008). doi: 10.1109/TPS.2008.2003360
8. Thiébaud, B., Jeanty-Ruard, B., Souquet, P., Forest, J., Matéo-Vélez, J.-C., Sarrailh, P., Rodgers, D., Hilgers, A., Cipriani, F., Payan, D., Balcon, N.: SPIS 5.1: An Innovative Approach for Spacecraft Plasma Modeling. *IEEE Transactions on Plasma Sci.* **43**, (2015). doi:10.1109/TPS.2015.2425300
9. Novikov, L.S., Makletsov, A.A., Sinolits, V.V.: Comparison of Coulomb-2, NASCAP-2K, MUSCAT and SPIS codes for geosynchronous spacecraft charging. *Advances in Space Research.* (2015). doi:10.1016/j.asr.2015.11.003
10. Matéo-Vélez, J.-C., Theillaumas, B., Sévoz, M., Andersson, B., Nilsson, T., Sarrailh, P., Thiébaud, B., Jeanty-Ruard, B., Rodgers, N., Balcon, N.,

- Payan, D.: Simulation and analysis of spacecraft charging using SPIS and NASCAP/GEO. IEEE Trans. Plasma Sci.,43, 2808-2816. (2015). doi:10.1109/TPS.2015.2447523
11. Alder, B. J., Wainwright, T. E.: Phase Transition for a Hard Sphere System. J. Chem. Phys. **27**, 1208–1209 (1957)
  12. Alder, B. J., Wainwright, T. E.: Transport processes in statistical mechanics. Ed. I. Prigogine, New York (1958)
  13. Kholmurodov, H.T., Altaisky, M.V., Pusynin, M.V. et al: Methods of molecular dynamics to simulate the physical and biological processes. Physics of Elementary Particles and Atomic Nuclei. **34**, 474-515 (2003)
  14. Zinin, L.V., Ishanov, S.A., Sharamet, A.A., Matsievsky, S.V.: Modeling the distribution of charged ions near the satellite method of molecular dynamics. 2-D approximation. Vestnik IKBFU. 53-60 (2012). In Russian
  15. Sharamet, A.A., Zinin, L.V., Ishanov, S.A., Matsievsky, S.V.: 2D modelling of a ion shadow behind charged satellite by molecular dynamics method. Vestnik IKBFU. 26-30 (2013). In Russian
  16. Zinin, L.V., Sharamet, A.A., Vasileva, A.Yu.: Modeling the formation of the ion shadow behind a positively charged microsatellite in an oxygen plasma by the molecular dynamics method. Vestnik IKBFU. 48-52 (2017). In Russian
  17. Rylyna, I.V., Zinin, L.V., Grigoriev, S.A., Veselov M.V.: Hydrodynamic approach to modeling the thermal plasma distribution around a moving charged satellite. Cosmic Res. **40**, 367-377 (2002)
  18. Alpert, Ya. L., Gurevich, A. V., Pitaevskiy, L. P. Artificial satellites in rarefied plasma: Nauka, Moscow. (1964). In Russian.