ISBN: 978-3-95450-132-8 doi:10.18429/JACoW-IPAC2014-TUPRI024

SIMULATION OF SPACE CHARGE DYNAMICS ON HPC *

Nataliia Kulabukhova †, Alexander Degtyarev, Alexander Bogdanov, Sergey Andrianov
Faculty of Applied Mathematics and Control Processes, Saint-Petersburg State University,
St. Petersburg, Russia

Abstract

To represent the space charge forces of beam a software based on analytical models for space charge distributions was developed. Special algorithm for predictor-corrector method for beam map evaluation scheme including the space charge forces were used. This method allows us to evaluate the map along the reference trajectory and to analyze beam envelope dynamics. In three dimensional models the number of computing resources we use is significant. For this purpose graphical processors are used. This software is a part of Virtual Accelerator concept which is considered as a set of services and tools of modeling beam dynamics in accelerators on distributed computing resources.

INTRODUCTION

It is well known that the envelope equations for continuous beam with uniform charge density and elliptical cross-section were first derived by Kapchinsky and Vladimirsky (KV). This very useful result has been put into different approaches to charged beams description with any charge distribution with elliptical symmetry. More over this is also true in practice for three dimensional bunched beams with ellipsoidal symmetry. The utility of this rms approach was first demonstrated by Lapostolle for stationary distributions. Subsequently, Gluckstern [1] proved that rms version of KV-equations remain valid for all continuous beams with ellipsoidal form. Here we describe the approach based on these ideas for description of nonlinear space charge forces using ellipsoidal presentation of a space charge distribution. The purpose of analytical models is connected with necessity to improve the efficiency of numerical calculations (especially with the use of parallel and distributed computing systems), and on the other with providing a detailed analysis of the impact on the beam dynamics of various parameters (both the control system itself and the beam parameters). In this paper, we describe an approach to construct analytical expressions for the electric field produced by the beam particles. These expressions may be derived using the matrix formalism for a trajectory analysis [2], and in terms of the envelope of the beam and/or the distribution function (in accordance with the Vlasov-Maxwell equations).

CHARGE DISTRIBUTION DENSITY

In this paper, we develop the main conditions for the field, generated by the beam, but taking into account the three-dimensional distribution of space charge in a bunched beam. It should be noted that a similar approach in the two-dimensional case allowed us to not only build a general analytical expression for a wide class of distributions of the beam, but also to integrate the expressions in the appropriate implementation of the perturbation theory, commonly used in beam physics. In this paper, we focus on how the use of the matrix formalism [2] for Lie algebraic methods [3] in the event of calculating the self-field of the beam. This approach allows not only to carry out numerical experiments, but also to provide accurate analysis of the impact of different effects with the use of ready-made modules. As in [2] we use the method of calculation in symbolic form the components of the tension in two dimensions, for various models of the distribution of the transverse charge density $\rho(x, y)$, where x, y are transverse coordinates in according to the Ferrers'a integrals technology [4]. As a result, the expressions for the electric field of the beam we form the total field as the sum of external and the self-field of the beam, which can be written as

$$\vec{E}(x, y, s) = \sum_{k=1}^{\infty} \left(\vec{E}^k_{\text{out}}(x, y, s) + \vec{E}^k_{\text{self}}(x, y, s) \right), \quad (1)$$

where $\vec{E}^k$ contains the members of k-th order according the variables x, y, correspondingly. Similar presentation allows us to embed the total field in the general equation describing the dynamics of the particles in accordance with the matrix formalism

$$\frac{d\vec{X}}{ds} = \sum_{k=1}^{\infty} \left(P^{[1k]}_{\text{out}}(s) + P^{[1k]}_{\text{self}}(s) \right) \vec{X}^{[k]}, \quad (2)$$

where $\vec{X}^{[k]}$ the vector of k-th phase moments, $P^{1k}(t)$ - matrices size of $d[n, k] = \binom{n+k-1}{k}$ [2].

- Trajectory analysis. In this case the beam is presented as a particles assemble and can be written using the following matrix $X^N = \{\vec{X}^1, \ldots, \vec{X}^N\}$, where $\vec{X}^k$ is a phase vector of k-th particle and N is a number of particles.
- Beam envelope dynamics. In this case the beam is described in the terms of envelope matrices [2]. See figure 1 for example.
- Distribution function dynamics. In this case one present the beam in the terms of a distribution function, which satisfies to the Maxwell-Vlasov equations system.

* Work supported by Saint Petersburg State University, project 9.38.674.2013: "Methods for decomposition and aggregation of complex systems and their implementation into distributed computing environment"
† kulabukhova.nv@gmail.com, nkulabukhova@spbu.ru

ISBN: 978-3-95450-132-8 doi:10.18429/JACoW-IPAC2014-TUPRI024

According to the matrix formalism, we compare the evolution operator and the endless-dimensional matrix
$M = (M^{11} \dots M^{1k} \dots)$,
with the help of the block-matrices and using

$$\sigma^{11}(t) = \sum_{l=1}^{\infty}\sum_{k=1}^{\infty} M^{1l}(t|t)\sigma_0^{lk}(M^{1k}(t|t)),$$

calculate the root-mean-square matrix of envelopes and recognize the current distribution function to calculate the own fields of space charge.

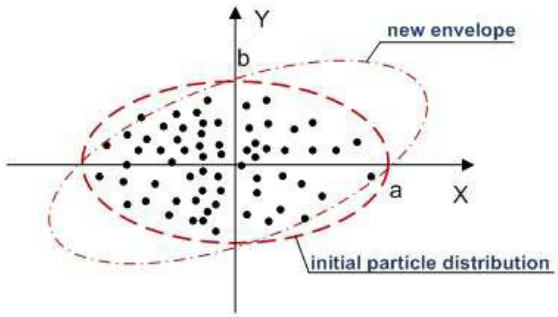

Figure 1: Envelope description

Survey of Distributions

Before speaking about predictor- corrector method we should say some words about the distribution functions. First of all we consider elliptical beam that's why $\varkappa^2 = X^*AX$ is used ($X = (x, y, x', y')^*$ - phase coordinates vector, A - positively definite symmetric matrix). Currently the distribution function can be presented as:

- Uniform distibution :

$$\varphi(\varkappa^2) = \frac{2\sqrt{detA}}{\pi^2}\Theta(1-\varkappa^2), \Theta(x) = \begin{cases} 1, x \geq 0, \\ 0, x < 0. \end{cases}$$

- VK-distribution:

$$\varphi(\varkappa^2) = \frac{\sqrt{detA}}{\pi^2}\delta(1-\varkappa^2).$$

- Normal distribution:

$$\varphi(\varkappa^2) = \frac{\sqrt{detA}}{4\pi^2} exp\left(-\frac{\varkappa^2}{2}\right)$$

Other disributions also exist, they will be described later, but these three are interesting in terms of charge distribution density in real space

$$\rho(x, y) = \int_{R^2} f(x, x', y, y')dx'dy'. \quad (3)$$

Using 3 for three types of distribution function of phase frequency, we get density of charge in real space:

- Uniform distibution :

$$\rho(x, y) = \frac{2qN_0}{\pi}\sqrt{\frac{detA}{A^{22}}}(1-\varkappa_r^2)\Theta(1-\varkappa_r^2) \quad (4)$$

- VK-distribution:

$$\rho(x, y) = \frac{qN_0}{\pi}\sqrt{\frac{detA}{A^{22}}}\Theta(1-\varkappa_r^2) \quad (5)$$

- Normal distribution:

$$\rho(x, y) = \frac{2qN_0}{2\pi}\sqrt{\frac{detA}{A^{22}}} exp\left(-\frac{\varkappa^2}{2}\right) \quad (6)$$

Taking away indetermination in 4 - 6 and assuming current intensity as a constant

$$I(s,t) = \upsilon_0 \int_{R^2} \rho(x, y, s, t)dxdy = const,$$

we get five types of distribution (three discussed above and two more):

- linear: $\rho_1(x, y) = \rho_0(1 - 4\varkappa^2/9)\Theta(1 - 4\varkappa^2/9))$;
- uniform: $\rho_2(x, y) = \rho_0\Theta(1 - \varkappa^2)$;
- normal: $\rho_3(x, y) = \rho_0 exp(-\alpha_3^2\varkappa^2)$, $\alpha_3 = -\frac{\pi}{2ierf(i)}$. Where $erfx = \int_0^x exp(-t^2/2)dt$ - probability integral.
- quadratic: $\rho_4(x, y) = \rho_0(1-(4/5)^4\varkappa^4)\Theta(1-(4/5)^4\varkappa^4)$;
- cosinusoidal: $\rho_5(x, y) = \rho_0\cos^2(\pi\alpha_5^2\varkappa^2)\Theta(1 - \alpha_5^2\varkappa^2)$. Where α_5 calculates with Frenel integral.

The whole solution algorithm - Predictor-Corrector method was described in [8]

PROFIT OF PARALLELIZATION

The natural parallel and distributed structures of beam physics problems allow using parallel and distributed computer systems (see works [5–7]). But usual approaches based on traditional numerical methods demand using the resources of supercomputers. This leads to impossibility of using such multiprocessing systems as computational clusters. There are two classes of problems in beam physics which demand very extensive computer resources. The first class includes long-time evolution problems; the second is concerned with the computer realization of optimization procedures for beam lines. Examples of the first type of problem include multi-turn injection and extraction of the beam in circular accelerators. Usually, these problems do not consider space charge effects. For advanced applications it is essential to study beam dynamics in high-intensity accelerators. Such machines are characterized by large beam currents and by very stringent uncontrolled beam loss requirements. An additional difficulty of numerical simulation

ISBN: 978-3-95450-132-8 doi:10.18429/JACoW-IPAC2014-TUPRI024

is connected with long-time beam evolution that requires the computation of hundreds of thousands or millions of turns. It requires the use of high-performance computers for beam evolution study. The problems of similar multi-turn evolution such as transverse stability with nonlinear space charge, uncontrolled beam losses due to space-charge-induced halo generation, etc. can also be mentioned. These problems are peculiar to modern high-intensity machines and require careful investigations of long-time evolution effects. From the computational point of view there are some problems related to the choice of models for beams with space charge [8], the presentation form of the beam propagator, and so on. As it was said above matrix formalism is a high-performance mapping approach for ODE solving. It allows to present solution of the system in the following form

$$X = \sum_{i=0}^{k} R^{1i}(t) X_0^{[i]}$$

where R^{1i} are numerical matrices. So this approach can be easy implemented in parallel code. Due to the fact that only matrix multiplication and addition are used, GPU programming is especially suitable for this purpose [9]. The research has shown that there is no great benefit via parallelization of computational code for one particle by using GPU. In this case overhead on data sending is significant. On the other hand, matrix formalism allows to process a set of initial points, where parallelization is more preferable. Let us introduce a set of initial particle

$$M = (X_0^1 X_0^2 \dots X^p) \tag{7}$$

According to equation 7 the resulting points can be calculated

$$M = \sum_{i=0}^{k} R^{1i}(t)((X_0^1)^{[i]}(X_0^2)^{[i]} \dots (X_0^p)^{[i]}).$$

Note that the sizes of matrices in this equation are much greater than in 7 when a set of initial particles is quite large. The use of matrix formalism allows to build some important criteria in terms of matrix elements. This significantly reduces computational time.

We evaluate the effectiveness of using data parallelism to program GPUs by providing results for a set of compute-intensive benchmarks. All calculations will be tested on a hybrid cluster of SPbSU computing center. Its nodes contain a NVIDIA Tesla S2050 system that was developed specifically as a GPGPU unit [9]. For our goal we are going to choose OpenCL and CUDA technology [10].

The research have shown that there is no great benefits via parallelization of computational code for one particle by using only GPU. In this case overhead on data sending is significant. On the other hand matrix formalism allows to process a set of the initial points, where parallelization is more preferably.

CONCLUSION

Our challenge is to develop an algorithm for solving the problem of accounting space-charge forces in general and compare this algorithm with other methods. After that, it is necessary to provide computer simulation. It allows simulate both long-term evolution of a set of particles, and evaluating based on envelope description. As it was said above the method can be implemented in parallel codes on GPU+CPU hybrid Cluster. That is why the future development of the research also can be based on writing software using different parallel techniques and complete implementation of the described approaches.

ACKNOWLEDGMENTS

The authors would like to express gratitude to Vladimir Korkhov for valuable help. All computations were partly carried out on cluster HPC-0011654-001 of Saint-Petersburg State University, Faculty of Applied Mathematics and Control Processes.

REFERENCES

[1] Gluckstern R.L., "Analytic Model for Halo Formation in High Current Ion Linacs', Phys. Rev. Lett., August 1994.

[2] Andrianov S.N., "Dynamical Modeling of Control Systems for Particle Beams', SPbSU, Saint Petersburg, 2004 (in Russian).

[3] Dragt A.J., "Lie Methods for Nonlinear Dynamics with Applications to Accelerator Physics. University of Maryland', College Park, www.physics.umd.edu/dsat/

[4] Ferrers N.M. "On the Potentials of Ellipsoids, Ellipsoidal Shells, Elliptic Laminae, and Elliptic Rings of Variable Densities', Quart. J. Pure and Appl. Math. 1877. Vol. 14. N 53. P. 1–22.

[5] M. Giovannozzi, "Space-Charge Simulation Using Parallel Algorithms," http://accelconf.web.cern.ch/AccelConf/e98/PAPERS/THP04B.PDF,pp.1189-1191.

[6] K.J. Bowers, "Accelerating a Particle-in-Cell Simulation Using a Hybrid Counting Sort, Journal of Computational Physics 173, pp.393-411, 2001.

[7] Ji Qiang, R. D. Ryne, S. Habib, V. Decyk, "An Object-Oriented Parallel Particle-in-Cell Code for Beam Dynamics Simulation in Linear Accelerators, Journal of Computational Physics 163, 434-451, 2000.

[8] N. Kulabukhova, "SPACE CHARGE DOMINATED ENVELOPE DYNAMICS USING GPUs, Proceedings of 4th International Particle Accelerator Conference, Shanghai, China, pp.924-926, 2012 http://www.JACoW.org

[9] N Kulabukhova,"GPGPU Implementation of Matrix Formalism for Beam Dynamics Simulation," Proceedings of 11th International Computational Accelerator Physics Conference, pp.143-145, 2012. http://www.JACoW.org

[10] A. Degtyarev, I. Gankevich, "Efficiency Comparison of Wave Surface Generation Using OpenCL, OpenMP and MPI," Proceedings of 8th International Conf. Computer Science and Information Tech., Yerevan, Armenia, 2011, p. 248–251.