PARALLELIZATION OF ENVELOPE DYNAMICS OF HIGH INTENSIVE BEAMS[*]

N. Kulabukhova[†], Faculty of Applied Mathematics and Control Processes,
Saint-Petersburg State University, St. Petersburg, Russia

Abstract

In this work the survey of methods for high intensive beam dynamics is given. As an alternative to them the approach based on envelope dynamics was used. This method is focusing on the use of the matrix form for Lie algebraic methods for calculating the beam dynamics in the presence of self-field of the beam. In particular, the corresponding calculations are based on the predictor-corrector method. Pros and cons of using described approach on hybrid systems are discussed.

INTRODUCTION

The number of methods and software for modeling beam dynamics is out of counting. The most popular software packages are MAD, COSY Infinity, TRANSPORT, BEAMBEAM3D, IMPACT-Z, IMPACT-T and some others [1]. But the problem of gathering different packages under one software product (figure 1) is not yet solved [2, 3]. Software for modeling beam dynamics with the space charge forces in the concept of matrix forms is the part of project of making a unified user interface ”Virtual Accelerator Laboratory”(VAL) [4]. The main use of the VAL is simulation of beam dynamics by different packages with the opportunity to match the results (in case of using different solution methods for the same problem) and the possibility to create pipelines of tasks when the results of one processing step based on a particular software package can be sent to the input of another processing step.

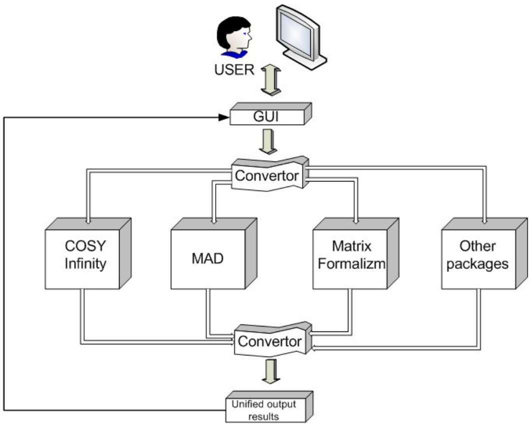

Figure 1: Unified user access.

However, in all these packages methods of tracking particle by particle through the whole system. The most commonly used is Particle-in-Cell method (PIC)[5, 6]. The Fortran-based environment COSY INFINITY is based on computations of perturbation expansions of Poincare maps to high orders. MARYLIE [7] is a FORTRAN program for beam transport and tracking based on a Lie algebraic formulation of charged particle trajectory calculations.

And in case of intensive beams, which can lead to the so called the filamentation effect or to the Halo, the number of particles is bigger then 1 billion. Though, the computer resources allow us to calculate large amount of data, the practice shows that it is better to have a parallel algorithm than a good and powerful machine. That was the goal: to make the algorithm that can be parallelized easily.

ENVELOPE DYNAMICS IN MATRIX FORM

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.

Matrix formalism is a high-performance mapping approach for Ordinary Differential Equation 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. As it was said above, there are different ways of modeling beam dynamics:

- 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 [8]. We will speak about them later.
- 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.

In this paper, we describe an approach to construct analytical expressions for the electric field produced by the

[*] Work supported by SPbSU 0.37.155.2014 and RFBR 16-07-01113A
[†] n.kulabukhova@spbu.ru

ISBN 978-3-95450-181-6

beam particles. These expressions may be derived using the matrix formalism for a trajectory analysis [8], and in terms of the envelope of the beam and/or the distribution function (in accordance with the Vlasov-Maxwell equations).

Particle Tracking

For beam without bunches the equations of motion can be written in the following form [8]

$$\begin{aligned} x'' &= \frac{q}{p}T\left(y'B_s - (1+x'2)B_y + x'y'B_x + T\frac{E_x}{c\beta\gamma}\right) \\ y'' &= -\frac{q}{p}T\left(x'B_s - (1+y'2)B_x + x'y'B_y + T\frac{E_y}{c\beta\gamma}\right) \end{aligned} \tag{1}$$

Using the concept of matrix formalism equations (1) can be rewritten

$$\frac{d\mathbf{X}}{ds} = \sum_{k=1}^{\infty}\left(\mathbb{P}^{1k}_{ext}(s) + \mathbb{P}^{1k}_{self}(s)\right)\mathbf{X}^{[k]} \tag{2}$$

In linear case the equation of motion with the space charge effect (1) will have the following view:

$$\begin{aligned} x'' + k_x x - \frac{q}{\epsilon_0 m_0 c^2\beta^2\gamma^3}E^L_x = 0 \\ y'' + k_y y - \frac{q}{\epsilon_0 m_0 c^2\beta^2\gamma^3}E^L_y = 0 \end{aligned} \tag{3}$$

Matrix $\mathbb{P}^{11}$ in this case can be

$$\mathbb{P}^{11}(s) = \mathbb{P}^{11}_{ext}(s) + \mathbb{P}^{11}_{self}(s)$$

where

$$\mathbb{P}^{11}_{ext}(s) = \begin{pmatrix} 0 & 1 & 0 & 0 \\ -k_x & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & -k_y & 0 \end{pmatrix},$$

$$\mathbb{P}^{11}_{self}(s) = \begin{pmatrix} 0 & 0 & 0 & 0 \\ -\eta_x & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & \eta_y & 0 \end{pmatrix},$$

If we speak about nonlinear case the equations (1) will be

$$\begin{aligned} x'' + (k_x - \alpha k_{xx})x = a_x x^3 + B_x x x'^2 + c_x x y^2 + \\ + k'_x x y y' + d_x x y'^2 + k_x x' y y', \end{aligned} \tag{4}$$

$$\begin{aligned} y'' + (k_y - \alpha k_{yy})y = a_y y^3 + B_y y y'^2 + c_y y x^2 + \\ + k'_y y x x' + d_y y x'^2 + k_y y' x x' \end{aligned} \tag{5}$$

Using matrix form this equations will be

$$\mathbf{X} = \begin{pmatrix} x \\ x' \end{pmatrix}, \quad \mathbf{Y} = \begin{pmatrix} y \\ y' \end{pmatrix}, \tag{6}$$

$$\mathbf{X}^3 = \begin{pmatrix} \mathbf{X} \\ \mathbf{Y} \\ \mathbf{X}^{[3]} \\ \mathbf{X}^{[2]} \bigotimes \mathbf{Y} \\ \mathbf{X} \bigotimes \mathbf{Y}^{[2]} \\ \mathbf{Y}^{[3]} \end{pmatrix}.$$

And for matrices $\mathbb{P}$:

$$\mathbb{P}^{11} = \begin{pmatrix} \mathbb{P}^{11}_x & \mathbb{O} \\ \mathbb{O} & \mathbb{P}^{11}_y \end{pmatrix},$$

$$\mathbb{P}^{13} = \begin{pmatrix} \mathbb{Q}^{11}_x & \mathbb{O} & \mathbb{Q}^{13}_x & \mathbb{O} \\ \mathbb{O} & \mathbb{Q}^{22}_y & \mathbb{O} & \mathbb{Q}^{24}_y \end{pmatrix},$$

where, for example, for {x, x'}

$$\mathbb{P}^{11}_{ext} = \begin{pmatrix} 0 & 1 \\ -k_x & 0 \end{pmatrix},$$

$$\mathbb{P}^{11}_{self} = \begin{pmatrix} 0 & 0 \\ \alpha k_{xx} & 0 \end{pmatrix}.$$

Envelope Dynamics

The approximation of the envelope can be done by different ways (see figure 2):

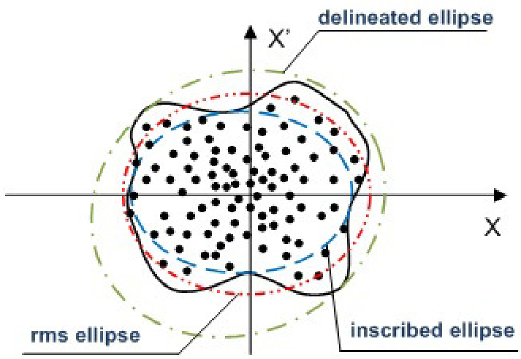

Figure 2: Types of approximated ellipses of the envelope.

- the root mean square approximation;
- approximation by delineate ellipsoid;
- approximation by inscribed ellipsoid.

The general view of envelope matrix is

$$\mathfrak{S}^{ik}_0 = \int_{\mathfrak{M}_0} f_0(\mathbf{X})\mathbf{X}^{[i]}(\mathbf{X}^{[k]})^* d\mathbf{X}$$

In linear case for delineate ellipsoid according to matrix formalizm equations can be

ISBN 978-3-95450-181-6

$$\frac{d\mathfrak{S}^{max}}{ds} = \mathbb{P}^{11}\mathfrak{S}^{max} + \mathfrak{S}^{max}(\mathbb{P}^{11})^*.$$

For root-mean-square envelope the equation with the space charge forces is

$$\frac{d\mathfrak{S}^{rms}}{ds} = \mathbb{P}^{11}\left(\langle\mathfrak{S}^{rms}\rangle_{\Delta s}; s\right)+\mathfrak{S}^{rms}\mathbb{P}^{11}\left(\langle\mathfrak{S}^{rms}\rangle_{\Delta s}; s\right)^*.$$

After that, using algorithm, described in [9] we get the envelope matrices to solve the equation of motion in the matrix form.

BENEFITS OF PARALLELIZATION

Due to the fact that the described approach is based on matrices and the main operations are multiplication and addition of them, the use of GPU programming [10, 11] seems very useful and effective. The present research is shown that there is no great benefit via parallelization of computational code for one particle by using, for example, OpenMP library (see Table 1). In this case overhead on data sending is significant and take the greatest part of time. On the other hand, matrix formalism allows to process a set of initial points as an envelope of the distribution function, where the data sending is lower. But using only GPUs is not justified. However, the results have shown that the described algorithm is well parallelizing.

Table 1: Time (sec.) in parallel code for different number of particles.

Threads/ Particles	10^7	$20 * 10^6$	$30 * 10^6$
Sequential	9,14508	18,3444	28,1367
4	13,1787	26,3788	21,5358
8	7,16977	14,5777	20,8549
16	6,8548	13,6071	20,15
64	6,70448	13,5593	20,0794
128	7,92809	14,8894	22,748

CONCLUSION

Our challenge is to provide computer simulation for developed algorithm for solving the problem of accounting space-charge forces in general and compare this algorithm with other methods. 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 to compare different parallel techniques for Hybrid Systems, in order to effective use of described approach to compute the required number of particles in long-term evolution of the beam.

ACKNOWLEDGMENT

The author would like to express gratitude to Vladimir Korkhov for valuable help. Scientific research were performed using the equipment of the Research Park of St.Petersburg State University. The work was sponsored by the Russian Foundation for Basic Research under the projects: 16-07-01113 ”Virtual supercomputer as a tool for solving complex problems” and by the Saint-Petersburg State University under the project 0.37.155.2014 ”Research in the field of designing and implementing effective computational simulation for hydrophisical and hydrometeorological processes of Baltic Sea (and the open Ocean and offshores of Russia)”.

REFERENCES

[1] Robert D. Ryne, Advanced Computing Tools and Models for Accelerator Physics, Proceedings of EPAC 2008, pp. 2947-2951, 2008.

[2] N. Malitsky and J. Smith and J.Wei and R. Talman, UAL-Based Simulation Environment for Spallation Neutron Sourse Ring, Proceedings of the 1999 Particle Accelerator Conference, 1999.

[3] D. Sagan and at al., Unified Accelerator Modeling Using the Bmad Software Library, Proceedings of IPAC 2011, 2011.

[4] V. Korkhov and A. Ivanov and N. Kulabukhova and A. Bogdanov and S. Andrianov, Virtual Accelerator: Distributed Environment for Modeling Beam Accelerator Control System , Lecture Notes in Computer Science, Vol. 6681116, pp. 166-169,2013.

[5] S. Paret and J. Qiang, Collisional Effects in Particle-In-Cell Beam-Beam Simulation, Proceedings of IPAC 2013, pp. 1700-1702, 2013.

[6] G. Stancari and S. Redaelli and V. Moens, “Beam Dynamics in an Electron Lens with Warp Particle-in-Cell Code“, Proceedings of IPAC 2014, pp. 451-453, 2014.

[7] Alex J. Dragt and Robert D. Ryne at al., “MARYLIE 3.0 Users Manual: A Program for Charged Particle Beam Transport Based on Lie Algebraic Methods“, University of Maryland, 2003.

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

[9] Kulabukhova N. and Andrianov S. and A. Bogdanov and A.Degtyarev, Simulation of Space Charge Dynamics in High Intensive Beams on Hybrid Systems, Lecture Notes in Computer Science, Vol. 9786, pp. 284-295, 2014.

[10] Kulabukhova Nataliia, Space Charge Dominated Envelope Dynamics Using GPUs, Proceedings of IPAC 2013, 2013.

[11] N. Kulabukhova and A. Degtyatev and A. Bogdanov and S. Andrianov, Simulation of Space Charge Dynamics on HPC, Proceedings of IPAC 2014, 2014.