PARALLEL AND DISTRIBUTED COMPUTING IN CIRCULAR ACCELERATORS

S. Andrianov[*], N. Edamenko[†], A. Dvoeglazov[‡], SPbSU, St.-Petersburg, Russia

Abstract

There are some problems of accelerator simulation (for example, computer design of new accelerator projects, optimization of working machines and quasi-real time accelerator control) which demand high performance computing using parallel and distributed computer systems. In particular, usage of computer clusters set is perspective for this purpose. In this reports we develop the approach suggested in our previous works. This approach is based on several levels of process of parallel and distributed computing. The structured physical and mathematical description of a beam evolution behavior (including space charge forces) allows to distribute calculations on several computing clusters with parallel computing inside each structural block. Computer algebra tools play the great role in such approach. Indeed, on the one hand, the used matrix formalism for Lie algebraic methods admits the symbolic computations and storage of the results in corresponding databases. On the other hand, the possibilities of visual 2D- and 3D-presentation allows to analyze the studied effects more carefully and intuitive. Multivariate description of beam evolution, space charge forces, and optimization procedures permits to compute various representative models simultaneously on different clusters.

1 INTRODUCTION

High-intensity accelerator technology has made much progress over the last years. It has led to growth of the number of application of these machines. In particular, they are proposed for nuclear waste management, tritium production, material sciences and other fields. However, particles losses are still a main problem in the design of high-current accelerators (for example, such as the Spallation Neutron Source – SNS). At high energies particles losses can lead to large induced activation and radiation damage in the structures. The main cause of particle losses is a halo, surrounding the beam core. The halo radius can reach several core radii — a central compact part of a beam phase portrait. This problem produced to great number of works devoted to studying halo formation processes and to possibilities to control them. As a main cause of the halo formation is considered mismatching between the transport system and beam characteristics. Besides the halo problem there some another problems of long time evolution of particle beams.

In this report we discuss some problems of computer modeling to study the halo developed by an intense proton beam. Some different mathematical backgrounds for

parallel and distributed computing providing high optical qualities of a beam along its transport through a long periodic focusing channel.

Computer modeling of long time beam evolution in circular accelerators makes exacting requirements to mathematical and algorithmic tools. The most of present programs are based on numerical integration methods for motion equations. Usually these motion equations have the form of ordinary differential equation. The external (control) and space charge forces are included into the model in a different way. The external forces are usually presented as a set of control functions depending on independent variables s measured along a reference trajectory. This corresponds to widely used approach in beam dynamics — the high-order aberrations description. The space charge forces can presented both in the perturbation picture and in the form of the Coulomb interaction forces. In the last case the problem of computing these forces clash with some difficulties in realization (see, e.g. [1]). In this report we consider the first approach and use some different methods for beam description. These methods have advantages from computer point of view both in numerical and symbolic modes.

For SNS space charge forces play an important role in the beam dynamics and tracking many particles for the enough long period of the circular accelerator including the effects of space charge becomes impractical for serial calculations. Naturally all forms of beam descriptions need in sufficient computational burden. That is why usage of parallel and distributed computers is preferable.

2 PARALLEL AND DISTRIBUTED COMPUTING

In this report we use two different concepts: parallel and distributed computing. We mean by *parallel computing* realization of homogeneous calculations on identical computers (e.g. using a cluster system). The *distributed computing* represents realization of rather independent computational streams. The nature of these computational streams are different and can have an original aim. In particular, for the beam physics problems we can introduce the following computational streams: *the dynamics stream*, *the visualization stream*, *the analytical stream* and *the space charge stream*. The interaction between these streams are shown on the Figures 1–2.

Evidently that similar interprocessing communications can be realized by different way depending on the particular problem. The great role plays the developed proved data in the symbolic presentation. Corresponding contents can be used in the computing process and thereby it can

---

[*] serge@apmath.spbu.ru

[†] nick@apmath.spbu.ru
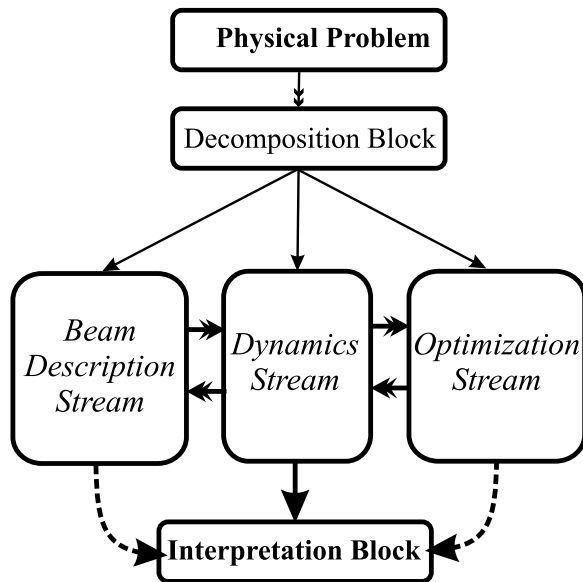
[‡] andrew@apmath.spbu.ru
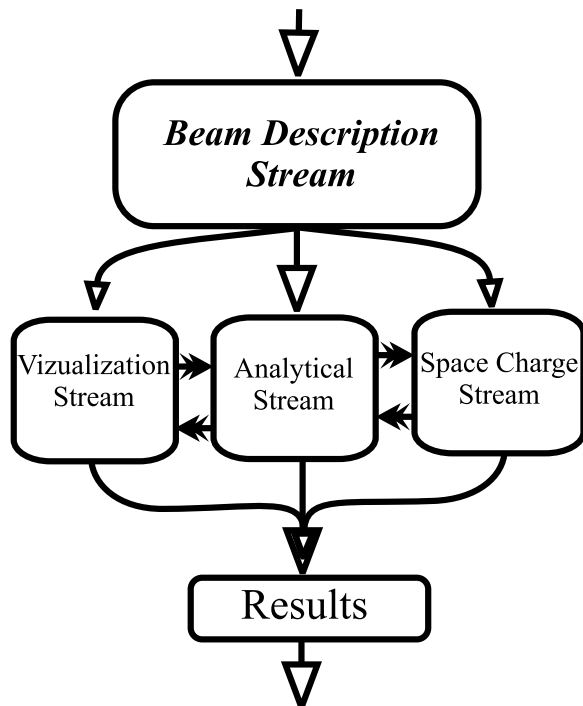
Figure 1: The main computing streams.

Figure 2: The beam description computing streams.

enhance the computational effectiveness. Moreover similar symbolic information can give support for design and optimization problems (see, e.g. [2]).

2.1 Mathematical Backgrounds

The effectiveness of the computer experiments strongly depends on used mathematical methods and their algorithmic realization. This is very actual for parallel computing. Here we suggest the matrix formalism for Lie algebraic tools

as appropriate tools for computing process on parallel and distributed computers. Usual used methods need in rather powerful computers with very big numbers of parallel processors. Ideally a researcher needs a computer with N processors, where N is equal to number of particles presenting a beam. The matrix formalism has natural parallel structure on the one hand and its inner structure admits distribution of computing processes. Let demonstrate the basic features of this formalism.

The Beam Propagator Construction According to the matrix formalism the map generated by a beamline can be presented in the following form

$$\mathcal{M}(s|s_0) \circ \mathbf{X}_0 = \sum_{k=0}^{\infty} \mathbb{M}^{1k}(s|s_0) \mathbf{X}_0^{[k]},$$

where $\mathbf{X}^{[k]}$ is the Kronecker power of the k-th order for the phase vector $\mathbf{X}$, $\mathbb{M}^{1k}(s|s_0)$ are two-dimensional matrices – aberrations matrices for the beam line under study. In the case space charge dominated beam for these matrices can be presented as a sum

$$\mathbb{M}^{1k}(s|s_0) = \mathbb{M}_{\text{ext}}^{1k}(s|s_0) + \mathbb{M}_{\text{self}}^{1k}(s|s_0),$$

where the first term is the aberration matrices describing the external (control) fields an the second term describes a contribution of the space charge forces. These matrices can be calculated both in numerical and symbolic forms. The second type of these matrices we use a set of model functions for particle beam distribution [3]. For cyclic accelerators one of the main aim is the creation of the one turn map. In this case we use the matrix presentation for normal form approach. Besides one can evaluate exact solutions for some special classes of the Hamiltonian in the matrix form too [4].

The Beam Phase Portrait Modeling Effectiveness of our computing process depends on the presentation form of the beam phase portrait. For our purpose it is convenient to use the two following forms:

- The phase space set occupied by particles is presented as a set of embedded subsets describing by vector functions $\mathbf{G}_j$. 2. The phase space distribution functions are modeled is presented as a set of embedded subsets describing by vector functions $\mathbf{G}_j$.

- The phase space set is described with help of a distribution function $f(\mathbf{X}, s)$.

The knowledge of the dynamics of theses functions allows to investigate the beam dynamics accurately and carefully. In our approach we build the analytical tools for the evolution description of these functions (see e.g. [3]). These functions can be used in different ways: at first for a visualization procedure, secondly for analytical investigation of particular features of the the phase portraits and at last — for computing of the space charge forces. The space charge

computational aspects require more attention in comparison with the external forces computing. The main distinguishing feature of our approach is that we do not use macro-particles presentation for beam description on the first step of our computing. The matrix presentation of all computational levels (see [3]) allows to realize all necessary calculations using sufficiently restricted computational resources.

2.2 The Computing Implementation

All calculations for every computing stream shown on the Figures 1–2 are performed on some dedicated set of computers. Similar groups of computers carry out homogeneous operations which reside to a corresponding stream. For either of these computing streams there is a special database of ready for operation objects. These objects are kept in symbolic forms and prepared in advance using computer algebra codes (e.g. *Maple V*) [5]–[6].

The Kronecker Operations Database According to the matrix formalism [7] all calculations are based on the Kronecker extension of the usual matrix algebra. These operations on two-dimensional matrices can be fulfilled up in a abstract form for arbitrary matrices and kept in the special database. Similar calculations are made only once.

The Dynamics Database This database keeps the block-matrices for control elements of the designed beam-line in the general form. We should note that for every element there are a set of similar matrices from which we collect a matrix presentation for the whole map for this element. As an example we can mention the fringe fields maps (they have virtual characters) and the central (for constant field) map. Their combination should generated in a special manner $k \geq 1$):

$$\mathbb{M}_{\text{whole}}^{1k}(s_3|s_0) =$$
$$\sum_{j \geq 1, n \geq 1, n \geq 1} \mathbb{M}_{\text{right}}^{1j}(s_3|s_2) \cdot \mathbb{M}_{\text{central}}^{jn}(s_2|s_1) \cdot \mathbb{M}_{\text{left}}^{nk}(s_1|s_0),$$

where the matrices $\mathbb{M}_{\text{left}}^{nk}(s_1|s_0)$, $\mathbb{M}_{\text{right}}^{1j}(s_3|s_2)$ describe input and output fringe fields respectively, the matrix $\mathbb{M}_{\text{central}}^{jn}(s_2|s_1)$ — the central part of the element. For every element one can use a set of fringe fields and study their influence on the beam motion. Similar technique is used for other combinations of virtual and reference parts of control elements. These matrices are kept in special sub-databases.

The Distribution Functions Database Described above two pictures for beam description have own presentations [3] and corresponding information in a symbolic form is kept in this database.

The Computational Module ("Calculator") All matrix objects have special attributes describing rules of concatenation of particular matrices in a full matrix. This operation is maintained by a special interface. For circular accelerators the full turn matrix is calculated using the matrix presentation for normal forms. The so called kick approximation for control elements plays a the particular role. In this case all calculations made in a symbolic form are very simple and need sufficiently small computational resources.

2.3 A Computer Experiments Realization

The suggested approach was realized using the following two ways of parallel and distributed computing. The first is based on the client-server technology and uses the TCP/IP protocol for the information interchange. The second was realize on a cluster computer class. This cluster is a set of MPI-based parallel machines and consists of eight Pentium II (with 400 MHz processors and 128 Mb of memory) machines. Test computational experiments demonstrated wholly satisfactory results on computational speedup.

An implementation of the moving window to run in MPI-based parallel machines was also developed.

3 ACKNOWLEDGEMENTS

4 REFERENCES

[1] Giovannozzi M. *Space-Charge Simulation using Parallel Algorithms*. Proc. of the Sixth European Particle Accelerator Conference — EPAC'98, Stockholm. P. 1189–1191 (1998).

[2] Andrianov S. N., ,Dvoeglazov A. I. *Beam-Lines Design Codes: Dynamical Modeling Approach*. Proc. of the Sixth European Particle Accelerator Conference — EPAC'98, Stockholm (1998). http://www.cern.ch/accelconf/e98/PAPERS/TPH01J.PDF

[3] Andrianov S.N., *High-Order Optics with Space Charge: Analytical Approach*. Proc. of the Sixth European Particle Accelerator Conference — EPAC'98, Stockholm. P. 1091–1093 (1998). http://www.cern.ch/accelconf/e98/PAPERS/TPH03J.PDF

[4] Andrianov S.N. *The Explicit Form for Lie Transformations*. Proc. of the Fifth European Particle Accelerator Conference EPAC 96, SITGES (Barcelona, Spain), Barcelona. P. 998–1000 (1996).

[5] Andrianov S.N. *Algebraic Approach to Pertubation Theory and Computer Algebra*. Proc. of the Int. Workshop New Comp. Technology in Control Systems. 1994. Pereslavl-Zalesky. 1994. Pereslavl-Zalesky: P. 10–12 (1994).

[6] Andrianov S.N. *Computer Algebra Application for Dynamical Systems Modeling*. Extended Abstrs. of the Intern. Workshop on Computer Algebra in Scientific Computing, 1998, SPb, Russia. SPb: Euler Mathem. Inst. P. 40–42 (1998).

[7] Andrianov S.N. *A Matrix Representation of the Lie Algebraic Methods for Design of Nonlinear Beam Lines*. Proc. of the 1996 Computational Accelerator Physics Conference – CAP'96, Williamsburg, Virginia, USA /Eds. J.J.Bisognano, A.A.Mondelli/. AIP Conf. Proc. No 391. NY: AIP, P. 355–360 (1997).