DEVELOPMENT OF THE OBJECT-ORIENTED PROGRAM IN C ++ FOR SIMULATION OF BEAM DYNAMICS IN ACCELERATOR INJECTION SYSTEMS

S.A. Kozynchenko *, Saint-Petersburg State University, Saint-Petersburg, Russia

Abstract

In this paper the program for simulation and optimization of beam dynamics in injection systems is considered, which at the same time allows the choice of parameters of the accelerating-focusing system. This permits designing the injection system during optimization process, taking into account the required output characteristics of the beam. The given program is based on Win 32 API dialog boxes and is developed in standard C++, using parallel programming tools based on the MPI-1.

INTRODUCTION

At present both in Russia and abroad, more attention is paid to design and creation of the accelerator complexes which provide generating the high-precision ion beams. When developing an accelerator complex, the injection system design is of importance, because it largely determines the output characteristics of the beam. For the design of such systems it is necessary to carry out numerical simulation and optimization of beam dynamics in the electro-magnetic fields. To reduce the calculation time for beam dynamics, the external fields in each step of optimization are usually approximated by analytical expressions, obtained for the simplified model of the real system under consideration. In order to design the accelerating structures providing high- precision ion beams, it is necessary to optimize the beam dynamics in the fields to be closed to real ones. In this paper, the program for simulation and optimization of beam dynamics in injection systems is considered, which at the same time allows the choice of parameters of the accelerating-focusing system. This permits designing the injection system during optimization process, taking into account the required output characteristics of the beam. Examples of such problems will be considered in the next section. The given program is based on Win 32 API dialog boxes and is developed in standard C++, using parallel programming tools based on the MPI-1.

SOME PROBLEMS OF MODELING AND OPTIMIZATION OF BEAM DYNAMICS IN THE INJECTION SYSTEMS OF ACCELERATORS.

Following D.A. Ovsyannikov [1], the dynamics of the beam in the external field, taking into account the beam space charge, is described by integro-differential equaions:

$$\begin{cases} \frac{dX}{dt} = V, \\ \frac{dV}{dt} = \frac{1}{m_p} f_1\left(t, X, \varphi\left(X, u\right)\right) + \\ \qquad \frac{1}{m_p} \int\limits_{M_{t,u}} f_2\left(t, X, V, \xi\right) \rho\left(t, \xi\right) d\xi = f_3\left(t, X, V, u\right), \\ X\left(t_0\right) = X_0, \quad V\left(t_0\right) = V_0, \quad \left(X_0, V_0\right) \in M_0, \end{cases} \tag{1}$$

$$\frac{\partial \rho\left(t,\eta\right)}{\partial t} + \frac{\partial \rho(t,\eta)}{\partial \eta} f\left(t,\eta,u\right) + \rho\left(t,\eta\right) \, div_{\eta} f\left(t,\eta,u\right) = 0 \tag{2}$$

$$\rho\left(t_0, \eta\right) = \rho_0\left(\eta\right). \tag{3}$$

Here $t \in [t_0, T]$ — the independent variable (time); parameters t_0, T are fixed; m_p, — the mass, $X(t) \in R^3$ — the position, $V(t) \in R^3$ — the velocity of a charged particle, respectively; $u = (u_1, u_2, ..., u_p) \in D$ — vector of control parameters, where $D \subset R^p$ — limited and a closed set; $\eta = (X, V) \in R^6$ — the position of the charged particle in the phase space; $\varphi = \varphi(X, u) \in C^2(G)$ — potential of the external field, where $G \subset R^3$ — limited and open set; function $f_1\left(t, X, \varphi\left(X, u\right)\right)$ describes the force, defined by external field; the choice of the function $f_2\left(t, X, V, \xi\right)$ defines the vay of modeling of the Coulomb interaction of charged particles; vector-function $f\left(t,\eta,u\right) = \left(V\left(t\right), f_3\left(t,\eta,u\right)\right)$; $\rho(t,\eta)$ — density distribution of particles due to the system (1); $\rho_0\left(\eta\right)$ — given charge density in the space M_0 at the moment t_0, where $M_0 \subset R^6$ — bounded closed set of measure zero; $M_{t,u} = \left\{X = X\left(t, X_0, u\right), \, V = V\left(t, V_0, u\right) : \left(X_0, V_0\right) \in M_0\right\}$ — image of the set M_0, due to system (1) under the vector u a the moment t.

For a given vector u the potential of the external electrostatic field φ, defined and continuous in $\bar{G}$, is a solution of the Dirichlet problem for the Laplace equation:

$$\begin{cases} \Delta \varphi\left(x, u\right) = 0, \quad x \in \mathrm{G}, \\ \varphi\left(x, u\right)|_{\Gamma_G(u)} = \varphi_0\left(x\right), \end{cases} \tag{4}$$

where $\Gamma_G\left(u\right)$ — piecewise smooth boundary of G, $\varphi_0\left(x\right)$ – known function.

On the cross-sections of the beam trajectories, the functional characterizing the beam dynamics is introduced. We consider the problem of finding the vector of controls $u^0 \in D$, delivering an extremum to the functional under the restrictions on the beam output energy, the particle losses, the max- imum radius of the beam, the potentials of the electrodes, the value of the functional, and some others.

* Sergey $_K ozyntchenko@hotmail.com$

ISBN 978-3-95450-125-0

Consider the problem of minimizing the growth of the beam emittance. The quality functional that characterizes the growth of beam emittance at the output of the injection system, is given in the form:

$$\begin{cases} I\left(u\right) = \dfrac{0.5\sqrt{\tilde{D_x^T}\tilde{D_{x'}^T} - \tilde{K_{xx'}^T}^2} + 0.5\sqrt{\tilde{D_y^T}\tilde{D_{y'}^T} - \tilde{K_{yy'}^T}^2}}{0.5\sqrt{\tilde{D_x^0}\tilde{D_{x'}^0} - \tilde{K_{xx'}^0}^2} + 0.5\sqrt{\tilde{D_y^0}\tilde{D_{y'}^0} - \tilde{K_{yy'}^0}^2}}, \\ x' = v_x/v_z, \qquad y' = v_y/v_z, \end{cases} \tag{5}$$

where

$$\begin{cases} \tilde{D_x} = \frac{1}{mes(M_{T,u})}\int_{M_{T,u}} x^2\rho\left(t,\eta_T\right)d\eta_T, \\ \tilde{D_{x'}} = \frac{1}{mes(M_{T,u})}\int_{M_{T,u}} x'^2\rho\left(t,\eta_T\right)d\eta_T, \\ \tilde{K_{xx'}} = \frac{1}{mes(M_{T,u})}\int_{M_{T,u}} xx'^2\rho\left(t,\eta_T\right)d\eta_T. \end{cases} \tag{6}$$

The coefficients $\tilde{D_y}$, $\tilde{D_{y'}}$, $\tilde{K_{yy'}}$ are calculated using formulas similar to (6). $mes\left(M_{T,u}\right)$ measure of $M_{T,u}$. Consider the problem of finding of the $u^0 \in D$, which minimizes the function (5) under given constraints.

Consider the problem of matching the beam at the output of the injection system. The quality functional is given as:

$$I\left(u\right) = \int_{M_{T,u}} \Phi_1\left(x_T, x'_T\right)\Phi_2\left(y_T, y'_T\right), \tag{7}$$

where Φ_1, Φ_2 — given functions with compact support, defined on sets $G_1 = \{(x_T, x'_T) : S_1\left(x_T, x'_T\right) < 1\}$, $G_2 = \{(y_T, y'_T) : S_2\left(y_T, y'_T\right) < 1\}$. Where $x'_T = v_{xT}/v_{zT}$, $y'_T = v_{yT}/v_{zT}$; $S_1\left(x_T, x'_T\right) = 1$ and $S_2\left(y_T, y'_T\right) = 1$ — ellipses describing the accepance of the linear accelerator in planes xx' yy' respectively; x, y, z – cartesian coordinates of the point X.

THE PROGRAM FOR MODELING AND OPTIMIZATION OF BEAM DYNAMICS IN INJECTION SYSTEM OF AN ACCELERATOR

Let us consider the computer program designed for solving the nonlinear problems of simulation and optimization of the beam injection systems for non-relativistic beams in the form (1)–(7). The program is written in the standard C++ at Visual C++ 2010 integrated development environment (IDE), using Win32 API and object-oriented programming, and it is intended for the use on personal computers. In this program, a scheme with two threads is applied, with functions of Win32 API being used for working with these threads. The primary thread is mainly used for the output (incl. graphical) of simulation and optimization results, whereas the secondary one is intended for performing calculations. The thread interaction is accomplished using global variables. Mostly, we consider the injection systems consisting of the electrodes in the form of thick disks of arbitrary cross-section with given potentials. As the components of the vector u, such physical (design) parameters of the accelerating system are chosen, whose changes affect the design and, consequently, the field that determines the beam dynamics. So, the program consists of the following interacting parts:

- The graphical user interface (GUI), developed on the basis of modal dialog windows and intended for displaying and editing the parameters, selecting the type of the problem to be solved, methods of modelling the fields, beam dynamics, and optimization, as well as numerical and graphical output. Displaying the results of modelling of beam dynamics is carried out online at the main application window using the methods of the class template **GClass**. The graph output is performed to several dialog windows.
- The unit for calculating the three-dimensional field of the set of electrodes. The main classes developed for the unit are as follows:
  - **Electrode** — classes intended to describe the set of electrodes;
  - **ElStaticField3D** — class developed without usage of parallel computing and intended for simulation of electrostatic fields by the finite-difference method in domains with complex geometry (solving the boundary value problems of type 4));
  - The base classes for ones considered above.
- The unit for calculating the self field of the beam. The main classes included in this unit are as follows:
  - **CylRingBeamModel** — class in which the analytical algorithms of computing the beam Coulomb field are used. It is used a model of the beam in the form of a cylinder with constant radius, as well as a set of charged coaxial hollow circular cylinders;
  - **EFPoisson3D** — class designed to solving the boundary value problems for the Poisson equation by the finite-difference method in domains with complex geometry;
  - **EFPoisson3Dpbc** — class designed for solving the boundary value problems for the Poisson equation by the finite-difference method in a round metal tube with periodic boundary conditions at the ends;
- The unit of modelling of three-dimensional beam dynamics in injection systems. The main classes developed for this unit are as follows:
  - **ChargeDistr** — class intended for modelling of the initial distributions of the beam;
  - **BeamDynamNoMPI** — class intended for modelling of the beam dynamics in external field with taking into account the own beam charge, not using the parallel computing;

ISBN 978-3-95450-125-0

- **BeamDynamMPI** — class designed for modelling of the beam dynamics, which uses the parallel computing based on MPI-1. The number of parallel processes is 5. The zero process is used for the calculation of the external field, forming the initial data for the beam, computing the beam self field and passing the results to corresponding processes, as well as for receiving the results of modelling of dynamics from other processes;
- The classes which are basic to the above. To model the external fields, the unit 2 is used. Consideration of the beam self field is accomplished by the method of "large" particles, as well as using the unit 3;

• Optimization unit. To optimize the beam dynamics in injection systems by the Box-Wilson method under the different number N of control parameters, the class template BoxWilsonN has been developed. It uses the linear model for the description of the response surface in the factor space. Under calculating the components of the gradient vector at each step of the Box-Wilson method with a full factorial design, the number of computational experiments is equal to $2N$. For simulation of beam dynamics, the unit 4 is used. If the calculations at the points where the experiments are performed are carried out in parallel then the total number of processes may consist $5 \cdot 2^N + 1$ $5 \cdot 2^N + 6$.

CONCLUSION

The developed program has shown its efficiency for modelling and optimization of beam dynamics in injection systems, as well as under calculations and analysis of the external fields and the self field of the beam.

REFERENCES

[1] D.A. Ovsyannikov *Modelling and optimization of charged particle beam dynamics*, (Leningrad, 1990), 312.

ISBN 978-3-95450-125-0