

Interim report

Software development for electron cooling calculation

Test version of the program realizing multi-particle tracking

I.Meshkov, A.Smirnov, A.Sidorin, G.Trubnikov,
JINR, Dubna, Russia

Abstract

The report describes a test version of the BETACOOOL program developed at the first stage of the work in the frame of Attachment to the Accord on a Software Development for Electron cooling Calculation. In the test version two algorithms for simulation of the ion distribution function evolution are implemented:

algorithm based on calculation of the particle momentum variation (SimCool);

algorithm based on calculation of the particle motion invariant variation (Monte-Carlo).

Two effects are realized in both algorithms: intrabeam scattering and electron cooling.

Intrabeam scattering action can be calculated on the basis of analytical models (mean or detailed) and in the frame of Monte-Carlo algorithm – using Molecular Dynamics technique.

In this report the description of the algorithm schemes and basic numerical models are described. Changes in the interface part of the program are included into new version of the user manual.

Contents

Introduction	3
1. General description of the program structure	5
1.1. Goals of the software development. Description of the algorithms	5
1.2. Program structure. Concept of the data input and output	7
2. SimCool algorithm	10
2.1. Block scheme of the algorithm	10
2.2. Generation of the ion initial distribution in the given ring position	12
2.3. Intrabeam scattering simulation	13
2.3.1. Beam emittance calculation	13
2.3.1.1. <i>Statistical approach</i>	13
2.3.1.2. <i>Emittance calculation at given lattice functions</i>	14
2.3.2. Mean growth rates	15
2.3.3. Detailed calculation of the growth rates	15
2.4. Electron cooling simulation	16
2.5. The beam rotation with the ring matrix	17
2.6. Calculation of output parameters	17
3. Monte-Carlo algorithm	19
3.1. Algorithm scheme	19
3.2. Program structure on the basis of module algorithm	21
3.3. The module library	23
3.3.1. The ring optic structure	23
3.3.2. Description of the optic library	23
3.3.2.1. <i>Variables, structure of the optic element</i>	23
3.3.2.2. <i>Drift space</i>	25
3.3.2.3. <i>Bending magnet</i>	25
3.3.2.4. <i>Quadrupole lens</i>	27
3.3.2.5. <i>RF Cavity</i>	28
3.3.2.6. <i>Solenoid of the longitudinal field</i>	29
3.4. Initialization of the ring structure	29
3.4.1. Initialization using output MAD file	30
3.4.2. Initialization using input MAD file. Tracking of Twiss parameters	31
3.4.3. Format of input file describing the accelerator structure	32
3.5. Intrabeam scattering simulation	33
3.5.1. Calculation of the growth rates on the basis of analytical models	33
3.5.2. Molecular dynamics	34
Annex. IBS growth rates calculation in accordance with Jie Wei model	36

Introduction

The version of the BETACOOOL program described in this report is a further modification of the program developed in cooperation with BNL, GSI and RIKEN in 2002-2003 years [1-3]. The initial version of the program calculates evolution in time of the r.m.s. parameters of the ion beam distribution function and particle number. Characteristic times of the beam parameter evolution for a few general effects are calculated under assumption of Gaussian shape of the distribution function. The algorithm is based on a simplest model of the ion beam (the beam is represented with four parameters only - three r.m.s. emittances and the particle number) that determines its general advantages: high calculation speed, a possibility to include into calculations many effects which can be described in the terms of characteristic times - electron and stochastic cooling, intrabeam scattering (IBS), scattering with residual gas, interaction with internal target, particle loss in collision point and so on.

The program is object oriented and its structure

- permits to use different formulae for friction force calculation, a few models of the electron beam,
- allows to take into account errors in the cooler geometry and so on;
- includes different analytical models for IBS growth rates calculation (for this aim an output MAD file of a ring is used).

The program is divided in two independent parts: physical code and interface part. Physical code is made using only standard C++ syntax and, in principle, is platform independent. The interface part is an executable program working under Windows environment and provides a service for input files preparation and for data post processing.

The base physical model permits to estimate the ion beam parameters in equilibrium or to calculate the beam dynamics during thermodynamically equilibrium process, when the ion distribution function is close to the Gaussian one. For RHIC electron cooling it is important to simulate an evolution of the ion distribution function during cooling. One can expect a formation of a dense core of the beam, which will determine the luminosity value. It follows to strong dependence of the friction force on the ion velocity, therefore the cooling time for central part of the ion beam can be sufficiently shorter than for the distribution tails. The situation is more serious in the case of cooling with a bunched electron beam, when the maximum of the electron density is in a central part of the electron bunch.

The structure of the base program version, and general objects were developed to be compatible for calculation of the ion distribution function evolution using Monte-Carlo method or other methods of multi particle simulation. Monte-Carlo method was realized for simulation of the electron cooling and interaction with internal target [2].

The test version of the multi-particle program includes two tracking procedures, which provide simulation of intrabeam scattering process at arbitrary shape of the ion

distribution function. First procedure repeats the algorithm of SimCool program [4] on the basis of the IBS and electron cooling models existing in the BETACOOOL. Second one realizes Monte-Carlo (MC) simulations and it can use as the analytical IBS models as the numerical growth rates calculations using Molecular Dynamics (MD) technique. For realization of the MD algorithm a translator of input MAD file was developed. To speed up the calculation the additional input file describing the accelerator structure (ASD file) is used. The program structure is based on module algorithm, which permits reducing the ring optic structure in accordance with the ASD file. The module library, which includes all general optic elements of accelerator, was developed as well as the algorithm of Twiss parameters tracking. Simplest procedures for data post-processing are implemented into the program.

Description of two tracking procedures, example of ASD file format, structure of the module library, base numerical models and first results of the test version benchmarking are the topics of this report. Modifications of the interface part of the program are described in new version of User Manual made on the basis of user manual for BETACOOOL program [1]. The user manual is a part of this report and included as an Addendum.

Some basic ideas of multi-particle tracking were described in the physical description of the BETACOOOL program [2]. For instance, the MD algorithm was realized in the base version of BETACOOOL program and in the frame of this work it was modified in accordance to the module structure of the program. To simplify the understanding of the document the corresponding parts (with necessary corrections) are briefly repeated here.

References

1. I.Meshkov, A.Smirnov, A.Sidorin, G.Trubnikov, E.Syresin, Description of software for BETACOOOL program based on BOLIDE interface, Joint Institute for Nuclear Research, Dubna, 2003
2. Yu.Korotaev, I.Meshkov, A.Sidorin, A.Smirnov, E.Syresin, G.Trubnikov, Software for Beam Parameter Simulations with Electron Cooling, Joint Institute for Nuclear Research, Dubna, 2003.
3. A.Smirnov, A.Sidorin, G.Trubnikov BOLIDE Interface, Joint Institute for Nuclear Research, Dubna, 2003, <http://lepta.jinr.ru/bolide.htm>
4. V.Parkhomchuk, I. Ben-Zvi, Electron cooling at RHIC, BNL, 2000

1. General description of the program structure

1.1. Goals of the software development. Description of the algorithms

To provide simulation of the ion distribution function evolution in time one needs to develop algorithm resolving the following problems: to simulate dynamics of a large number of particles on the one hand and to simulate a large number of the revolution in the ring on the other hand.

Required period of the beam dynamics calculation is related to characteristic times of investigated processes. Revolution period of a beam inside a storage ring usually is of the order of $10^{-5} - 10^{-6}$ sec. Characteristic times of the general processes, which lead to variation of the beam distribution function, depend on the beam energy, number of particles, target density, luminosity and so on and can be of the order of $10^2 - 10^3$ sec. In more important cases the time of experiment is about one order of magnitude longer. Correspondingly one needs to calculate the particle dynamics during $10^8 - 10^9$ revolutions in the ring.

Presently two base algorithms are used for simulation of a long-term evolution of the particle distribution function:

the algorithm realized in the SimCool program by V.Parkhomchuk's group from BINP (Novosibirsk),

and Monte-Carlo simulation performed in the MOCAC program developed by P.Zenkevich and A.Bolshakov from ITEP (Moscow).

The goal of the BETACOOOL development is to combine both the algorithms (with their modifications) in the same program, to use a standard input file format for all algorithms, standard output data format and uniform post processing. Such an approach permits to perform benchmarking of the algorithms step by step comparing results of the simulation with the results of r.m.s. dynamics simulation inside the BETACOOOL program and with the results of independent programs. Besides that each algorithm can use all the models of general effects acting on distribution function.

Both the algorithm of the multi-particle simulation are based on amplification of the heating and cooling effect action in accordance to a large step over time during dynamics simulation.

In the frame SimCool algorithm the step in dynamics simulation corresponds to large number of the turns in the ring ($\sim 10^5 - 10^6$). Each effect acting on the ion beam distribution function is represented by the variation of the particle momentum components after revolution in the ring. These **momentum variations are amplified** by the turn number, which is used as a step for dynamics simulation. The momentum variation due to IBS is calculated in accordance with analytical models and, therefore, the structure of the ion ring can be described in the same way as in the r.m.s. dynamics simulations (by import of MAD output file). The required particle number is determined by the accuracy of the luminosity calculation and can be of the order of a few thousands.

Detailed description of the SimCool algorithm realization in the test version is presented in the chapter 2.

In the frame of Monte-Carlo algorithm the beam dynamics, in principle, can be simulated accurately during revolution in the ring. The idea to speed up the calculation with this algorithm is the following. For each particle deviations of its motion invariants (two Courant-Snyder invariants for the transverse degrees of freedom and square of the maximum relative momentum deviation for the longitudinal one) are calculated. At given step of the dynamics simulation for each particle new values of **the invariants are calculated in accordance with the corresponding invariant deviations**. New phases of the particle betatron and synchrotron oscillations are generated randomly. Description of the MC algorithm is a topic of chapter 3.

Electron cooling, interaction with an internal target and accuracy of the luminosity calculation are not depending on the ion beam intensity and the number of particles required for MC algorithm is determined by the required accuracy of IBS simulation. We plan to use two possibilities to reduce number of particles involved into calculations.

For a coasting ion beam one can use Molecular Dynamics technique. In this case the motion equations are solved for small number of particles located inside a short cell. Influence of all other particles is taking into account through periodic boundary conditions for the particle distribution function and usage of Ewald's sum for calculation of Coulomb forces. For bunched beam the MD cannot be applied directly, but the bunch evolution can be investigated by analysis of the dynamics of a few slices along the bunch length. MD looks very attractive for simulations because it permits to perform realistic IBS calculation at the particle number of about 1000 and it does not require an increase of the particle mass and charge.

For bunched beam we plan to simulate the beam dynamics using also so called "macro" particles, each of them has charge to mass ratio the same as a real particle, but the mass of the macro particle is larger than the mass of real particle by the factor of the particle increase. The number of macro particles required for calculations is determined by the accuracy of the beam space charge calculation. Usually to perform particle distribution function accurately one needs about 10-30 thousand of particles for each degree of freedom or for three dimensional problem of about 100 thousands of particles.

The IBS process in the frame of MC algorithm can be calculated using analytical models (in this case only an output MAD file is necessary for the calculations), or by numerical integration of the particle motion equation in the ring taking into account Coulomb interactions between particles. In the last case one needs to input the ring optic structure from input MAD file.

Other problem of the Monte-Carlo method related to the calculation speed is simulation of the particle dynamics in a long optic channel. Necessary step of integration of the particle motion equations along the longitudinal co-ordinate depends on minimum impact parameter of interparticle interaction and relative particle velocity. This value can be of

the order of a few mm. Stable integration of the motion equation at the length of a few km (as it takes place in the RHIC case) is a complicated task. One can solve the problem by reduction of the optic element number in the ring structure. In this case the particle dynamics is simulated by numerical solution of the motion equation only in a few characteristic points on the ring circumference. Between these points the particle coordinates are transformed linearly with transformation matrices of the ring optic elements. This procedure is based on the module structure of the ring optics representation. The module of the optic structure can correspond to a real optic element or, in the case of a reduced optic structure, to a few optic elements placed in a line one after another.

The module corresponding to a real optic element contains all the parameters of the element: geometry of the electromagnetic fields, transformation matrix and lattice functions of the ring at the module entrance. Inside such a module one can perform all types of the particle coordinate transformation – linear transformation using the element matrix and solution of the particle motion equation. Each module can be associated with an effect acting on the particle motion invariant such as electron cooling, internal target, collision point and so on. The module parameters are input from the input MAD file, the parameters of the effects – from the input BETACool file.

If the module corresponds to a few real optic elements it can provide only linear transformation of the particle coordinates. Reduction of the ring optic structure is provided by the program in accordance with the Accelerator Structure Description (ASD) file. This file contains numbers of the optic elements in the input MAD file, where the IBS process is numerically calculated. All the elements located between these points are presented by a module, which contains only a transformation matrix and the ring lattice functions at the entrance. Modules corresponding to real optic elements with a total set of parameters are located in positions of the IBS calculation. The effects acting on the beam emittance are located also in these points.

In the test version of the program the simplest format of the ASD file was developed. The file contains only the element number, distance from the start point, the element length and type of the element or effect. Description of the ASD file format is presented in chapter 3.3.

1.2. Program structure. Concept of the data input and output

In the frame of the initial version of the program a few models of the following effects were developed and tested using r.m.s. dynamics simulation:

- electron cooling,
- intrabeam scattering,
- external heating,
- particle losses in collision point,
- particle losses due to finite beam life-time.

The test version of the program includes presently the following kit of the algorithms:

- r.m.s. dynamics simulation,
- SimCool algorithm, which is based on the analytical model of IBS process simulation (calculating mean rates or detailed rates for each particle),
- MC algorithm using analytical models of IBS process,
- MC algorithm based on MD technique.

Storage ring optic structure, depending on the effect model and numerical algorithm, can be represented in the following forms:

- mean lattice parameters,
- lattice parameter variations along the ring,
- set of optic elements from the optic library,
- module structure.

The ion beam can be represented by

- set of Twiss parameters (for the ring optic structure initialization),
- set of the r.m.s. emittances and particle number,
- real particle array at the distribution corresponded to the distribution function of the beam (for SimCool algorithm, MC with analytical models of IBS),
- real particle array at the realistic distribution in the transverse degrees of freedom and periodic boundary condition in the longitudinal one (for MD simulations),
- array of macro particles.

Further development of the program is related to extension of its entire general components – library of the effects, kit of the algorithms, kit of the ring models, optic element library, algorithms of the Twiss parameters tracking, kit of the ion beam models.

Nearest steps, which have to be done in the frame of this contract, are to develop the model of the scattering with residual gas, to realize MC algorithm simulating the macro particles dynamics, improvement of the ASD file format, improvement of the models of some effects (electron cooling – in the order to include numerical results of the friction force calculation, different variants of the detailed calculation of IBS rates). Depending on the results of benchmarking of the presented version and practice of the beam dynamics simulation it is possible to develop and include new numerical algorithms into the program or to provide a modification of existing ones.

To simplify the benchmarking of the program and to provide fast development of the program the software components have to satisfy the following requirements.

The model of each effect has to be adapted for uniform usage in the frame of each algorithm of dynamics simulation.

General input parameters has to be independent on the using ring and beam models and on the algorithm of the dynamics simulation.

Each numerical algorithm has to provide output not only unique parameters necessary for it, but the same set of general beam parameters. Output of the standard beam parameters has to be uniformed and independent on the beam model and algorithm of calculations.

The concept of uniform data input is realized by development of input BETACOOOL files, further development of ASD file format. Uniform output of the data is based now on the following principles:

- each effect acting on the beam emittance calculates heating (cooling) rates and loss rate at given r.m.s. beam parameters,
- each effect calculates 3-D diagram of the rates in given range of the r.m.s. beam parameters,
- each numerical algorithm calculates the time dependence of the rates, time dependence of the r.m.s beam parameters, particle number and luminosity,
- each numerical algorithm using multi particle tracking outputs the particle distribution function in a few standard files.

2. SimCool algorithm

Initial plans for this work did not presume the development of the SimCool algorithm in the frame of the BETACOOOL program and main attention was attracted to development of Monte-Carlo algorithm. However, the SimCool algorithm has a few obvious advantages in comparison with the MC one. Main of them is simplicity and as a result - high calculation speed. Therefore, the results of simulations using SimCool algorithm can be effectively used for benchmarking of MC.

From the other hand, the SimCool algorithm is under independent development in the frame of the SimCool program. The possibility to compare results of the same numerical model realized in independent programs is very fruitful for benchmarking each of them. For this aim in the frame of BETACOOOL program one can realize data output in the SimCool program format. In the frame of BETACOOOL the SimCool algorithm can be used with all developed models of general effects. In principle one can repeat accurately the models of the processes using in SimCool program (it is mainly related to IBS and electron cooling processes) and after benchmarking the code using results of SimCool program to compare different models in the BETACOOOL program.

2.1. Block scheme of the algorithm

The SimCool algorithm can be illustrated by the scheme presented in the Fig. 2.1 and includes the following steps:

- initially the storage ring optic structure is loaded from external file (MAD output file), mean ring parameters and RF system parameters are loaded from input BETACOOOL file,
- in the initial position of the ring an array of particles is generated using random number generator and matched with the ring lattice functions and synchrotron function,
- thereafter regular algorithm of the particle dynamics simulation is started (“cycle of dynamics simulation” in the Fig. 2.1).

Each step of the dynamics simulation consists of the following procedures:

- to calculate IBS rates in accordance with analytical model one needs to know r.m.s emittance values of the particle array and these values are calculated as described in the chapter 2.2
- in the case of IBS kick calculation using mean diffusion coefficients the IBS growth rates are calculated using lattice parameters loaded from MAD file, thereafter the momentum components of each particle are changed randomly with by the angle, which has the same value for all particles. The scattering angle is calculated in accordance with the calculated rate values, lattice parameters in the initial position and number of the revolution step used for dynamics simulation.
- in the case of detailed IBS kick calculation the emittances of the particle array are used for calculation of the growth rate for each particle individually, the momentum variation of the particle is calculated in the way as in previous point. For detailed IBS kick calculation the IBS process models developed in the BETACOOOL cannot be used and realization of this possibility required development of new models.

- action of the electron cooling on the particle momentum is calculated in the frame of the cooler representation as a thin lens. The friction force components can be calculated using different formulae and different models of the electron beam.

Detailed description of all general stages of the algorithm is presented in the next chapters, the algorithm of the detailed IBS calculation based on Burov's model is described in ch. 2.3.3.

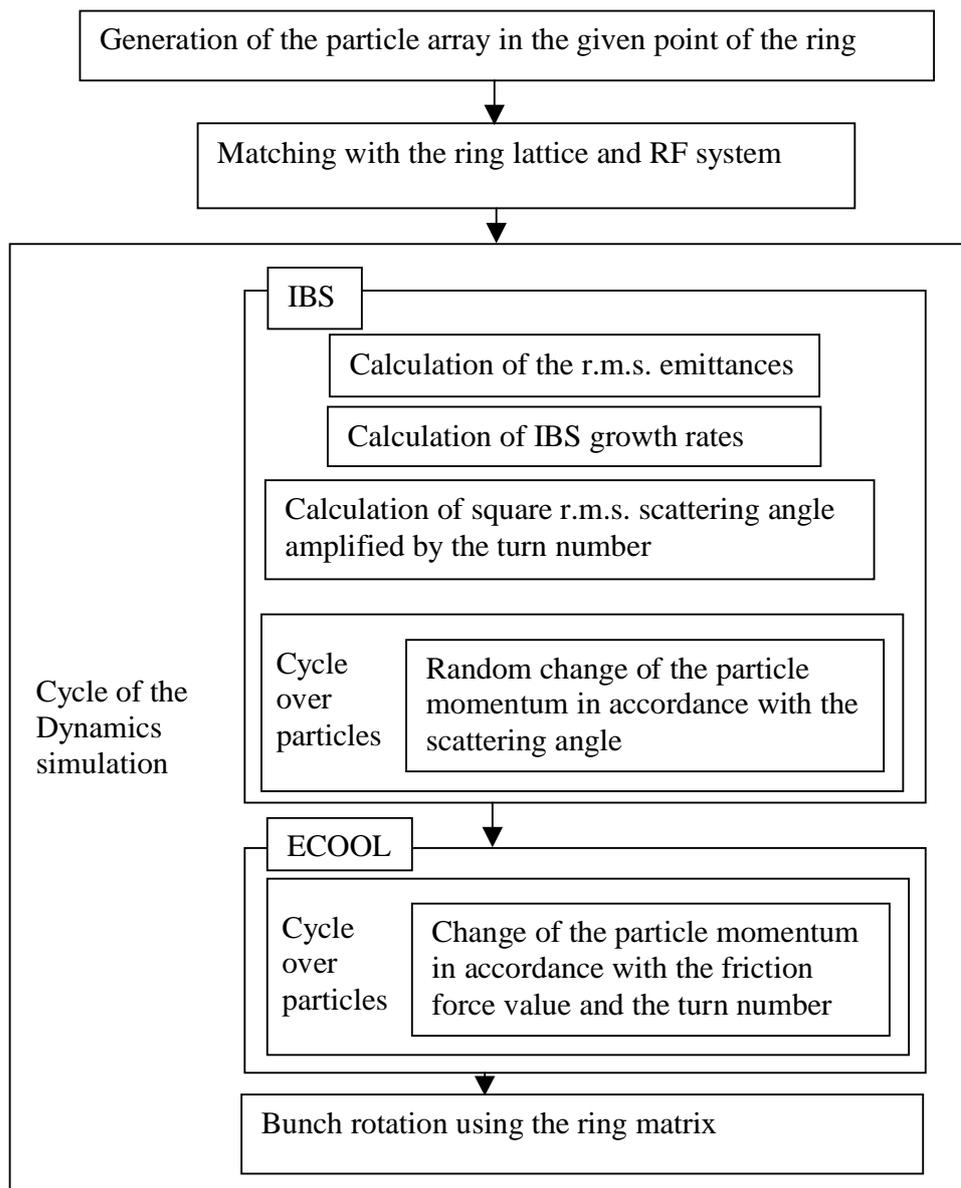


Fig. 2.1. Block scheme of the SimCool algorithm. In the case of **detailed IBS calculation** the calculation of IBS growth rates and square of the scattering angle are performed inside cycle over particles

2.2. Generation of the ion initial distribution in the given ring position

The particle array is generated under assumption of Gaussian shape of the distribution function in all degrees of freedom. Each particle co-ordinate is calculated in accordance with:

$$x_i = \sigma_{x_i} \sqrt{-2 \ln \xi_{1,i}} \sin(2\pi \xi_{2,i}), \quad (2.1)$$

ξ_1 and ξ_2 are random values uniformly distributed in the interval from 0 to 1 generated with the standard random generator.

In the longitudinal plane the standard deviations of co-ordinate and momentum are calculated in accordance with the formulae:

$$\sigma_\delta = \sqrt{\varepsilon_l}, \quad \sigma_s = B_s \sigma_\delta \quad (2.2)$$

where the longitudinal emittance value is determined as $\varepsilon_l = \left(\frac{\Delta p}{p_{rms}} \right)^2$ B_s is the synchrotron function calculated from the RF system parameters:

$$B_s = \frac{C\eta}{2\pi Q_s}, \quad (2.3)$$

where C is the ring circumference, Q_s is the synchrotron tune:

$$Q_s = \frac{1}{\beta} \sqrt{\frac{ZehV\eta}{A2\pi m_p c^2 \gamma}}, \quad (2.4)$$

h is the harmonic number, V is the RF voltage amplitude, β, γ are relativistic parameters, Z and A are the ion charge and atomic numbers, m_p is the proton mass, $\eta = \frac{1}{\gamma^2} - \frac{1}{\gamma_{tr}^2}$.

In the transverse planes one needs to match the particle array with the lattice parameters of the ring. This procedure is realized by the following steps.

1. The particle array of a round shape of cross-section is generated using the standard deviations for transverse co-ordinates and their derivatives:

$$\sigma_x = \sqrt{\varepsilon_x \beta_x}, \quad \sigma_{x'} = \sqrt{\varepsilon_x \gamma_x} \quad (2.5)$$

2. Matching with the lattice parameters the derivatives of the particle transverse co-ordinates are not changed, the co-ordinates are changed:

$$x_{\beta} = \frac{x_{round}}{\sqrt{1 + \alpha_x}} - \frac{\alpha_x x'}{\gamma_x}, \quad (2.6)$$

3. The matching with the dispersion and its derivative:

$$x = x_{\beta} + D_x \delta, \quad x' = x'_{\beta} + D'_x \delta. \quad (2.7)$$

Here and below we use the notation $\delta = \frac{\Delta p}{p}$.

2.3. Intrabeam scattering simulation

2.3.1. Beam emittance calculation

The IBS growth rates in the SimCool algorithm can be calculated using analytical model only. For this aim the beam emittance values are required. The emittance of the particle array can be calculated using statistical approach or as a mean value of the particle motion invariants.

2.3.1.1. Statistical approach

The statistical approach in the case of uncoupled transverse motion gives the following expression for the beam transverse emittance value:

$$\varepsilon_x = \sqrt{\langle (x - \langle x \rangle - D\delta)^2 \rangle \langle (x' - \langle x' \rangle - D'\delta)^2 \rangle - \langle (x - \langle x \rangle - D\delta)(x' - \langle x' \rangle - D'\delta) \rangle^2}. \quad (2.8)$$

where dispersion and its derivative are calculated in accordance with:

$$D_x = \frac{\langle (x - \langle x \rangle)(\delta - \langle \delta \rangle) \rangle}{\langle (\delta - \langle \delta \rangle)^2 \rangle}, \quad (2.9)$$

$$D'_x = \frac{\langle (x' - \langle x' \rangle)(\delta - \langle \delta \rangle) \rangle}{\langle (\delta - \langle \delta \rangle)^2 \rangle}. \quad (2.10)$$

To control the quality of the distribution function matching with the lattice one can numerically evaluate the beam Twiss parameters:

$$\beta_x = \frac{\langle (x - \langle x \rangle)^2 \rangle}{\varepsilon_x}, \quad (2.11)$$

$$\alpha_x = -\frac{\langle (x - \langle x \rangle)(x' - \langle x' \rangle) \rangle}{\epsilon_x}, \quad (2.12)$$

The longitudinal emittance is calculated as a square of relative momentum spread:

$$\epsilon_l = \langle (\delta - \langle \delta \rangle)^2 \rangle. \quad (2.13)$$

In all the formulae above the angular brackets mean averaging over the array, for instance:

$$\langle x \rangle = \frac{\sum_{i=1}^N x_i}{N}, \quad (2.14)$$

where index i is related with the i -th particle and N is the particle number.

2.3.1.2. Emittance calculation at given lattice functions

In the case, when the beam emittance is calculated in the ring position with known lattice functions and dispersion, the emittance of the particle array can be calculated as mean values of the particle motion invariant in corresponding degree of freedom. Invariant of the particle longitudinal motion is given by the formula:

$$I_l = \delta^2 + \frac{(s - s_0)^2}{B_s^2} \quad (2.15)$$

Invariants of the motion in the transverse planes are the Courant-Snaider ones and are calculated as usual. Initially one needs to calculate “betatron” particle co-ordinates:

$$x_\beta = x - D_x \delta, \quad x'_\beta = x' - D'_x \delta, \quad (2.16)$$

after that the invariant value is calculated in accordance with the Courant-Snyder definition:

$$I_x = \frac{1 + \alpha_x^2}{\beta_x} x_\beta^2 + 2\alpha_x x_\beta x'_\beta + \beta_x x'^2_\beta \quad (2.17)$$

R.m.s. emittance is related to the mean invariant value as:

$$\epsilon = \frac{\langle I \rangle}{2} \quad (2.18)$$

Both the methods of the emittance calculation were implemented and tested in the program. Relative difference of the results at the same particle array decreases with the increase of the particle number. For IBS growth rates calculation an accuracy of a few percents is good enough. For instance, the difference between emittance value used for the array generation and calculated from the generated array is of the order of a few percent when the particle number is a few thousand. Thus both the methods are equivalent for the IBS rates calculation, however the situation is dramatically changed in the case of output parameters calculation during dynamics simulation. The second procedure of the emittance calculation does not sensitive to small mismatching of the array and only it gives reasonable results of characteristic time calculation.

2.3.2. Mean growth rates

In the case of IBS calculation using mean growth rates, the mean growth rates are calculated in accordance with one of the analytical model developed in BETACOOOL and the ring structure loaded from output MAD file. When the growth rates are known one can calculate mean square of the scattering angle taking into account multiplication factor. The mean square angle after one revolution in the ring is equal:

$$\langle \theta^2 \rangle = \frac{\varepsilon}{\beta} \frac{T_{rev}}{\tau}, \quad (2.19)$$

where β is the beta function in the point of the particle array generation, τ is characteristic growth time in corresponding degree of freedom. Here the angular deviation of the particle trajectory means relative momentum components: $\theta_{x,y} = \frac{p_{x,y}}{p}$, $\theta_s = \frac{\Delta p}{p}$. After N_{turn} revolutions in the ring the square of the scattering angle is equal to the sum of the square angles at each revolution:

$$\langle \theta^2 \rangle = \frac{\varepsilon}{\beta} \frac{T_{rev}}{\tau} N_{turn}. \quad (2.20)$$

The variation of the particle trajectory angular deviation is calculated in accordance with

$$\Delta\theta = \sqrt{\langle \theta^2 \rangle} \xi \quad (2.21)$$

where ξ is the random value with Gaussian distribution at unit dispersion.

2.3.3. Detailed calculation of the growth rates

In the case of detailed calculation of the growth rates the algorithm is similar to the previous case. However, the characteristic times are calculated for each particle individually. For this aim one needs to know not only r.m.s. parameters of the particle array distribution function, but the amplitudes of the betatron and synchrotron oscillations

of the particle. The inverse growth times for each particle can be calculated by numerical evaluation of the scattering integrals. Under some additional assumption the integration can be performed analytically in one degree of freedom. For instance, in the frame of Burov's model of detailed IBS calculation, the longitudinal IBS rate is presented as:

$$\Lambda_i^{\parallel} \equiv \frac{1}{V_{zm}^2} \frac{d}{dt} V_{zm}^2 = \sqrt{\frac{2}{\pi}} \frac{N_i r_i^2 c L_i}{\gamma^2 V_{zm}^2 \sigma_x^2 \sigma_z u_x} \exp\left(-\frac{z_m^2}{4\sigma_z^2}\right) I_0\left(\frac{z_m^2}{4\sigma_z^2}\right) \Phi\left(\frac{V_{xm}}{u_x}, \frac{V_{ym}}{u_x}\right) \left(1 - \sqrt{\frac{V_{zm}}{u_x \sqrt{2}}}\right) \quad (2.22)$$

Here u_x are rms velocities in PRF, $s_{x,y}$ – rms beam sizes. z_m – amplitude of the ion oscillations over longitudinal co-ordinate, $V_{xm,y}$ are the amplitudes of oscillations over transverse velocities, I_0 – modified Bessel function. The form-factor Φ is given by the expression:

$$\Phi(x, y) \equiv \frac{2}{\pi} \exp\left(-\frac{x^2 + y^2}{2}\right) \iint_0^{\infty} \frac{d\xi d\eta}{\sqrt{\xi^2 + \eta^2}} \exp\left(-\frac{\xi^2 + \eta^2}{2}\right) I_0(\xi x) I_0(\eta y) \quad (2.23)$$

Horizontal IBS rate relates to the longitudinal one as:

$$\Lambda_i^x / \Lambda_i^{\parallel} = \frac{\overline{D^2 + (D' \beta_x + \alpha_x D)^2}}{x_m^2} V_{zm}^2 \quad (2.24)$$

Where the bar stays for the orbit-averaging.

The Burov's model was realized in the program and is under the test now.

2.4. Electron cooling simulation

In the frame of SimCool algorithm the action of the electron cooling on the ion momentum components is calculated in accordance of the cooler representation as a thin lens. In this case the ion angle variation is calculated in accordance with

$$\Delta\theta = \frac{F}{Mc^2 \beta^2 \gamma} l_{cool} N_{turn}, \quad (2.25)$$

where F is the friction force in LRF, M is the ion mass, l_{cool} is the cooling section length, N_{turn} is the revolutions number using as a multiplication factor. The ion co-ordinates are not changed inside cooler. The friction force components can be calculated using all the friction force formulae and all the models of electron beam developed in the BETACOOOL.

2.5. The beam rotation with the ring matrix

To avoid the problems related to the beam mismatching and coupling between longitudinal and transverse coordinates the beam rotation in the ring is provided in the following steps.

Initially the “betatron” transverse particle co-ordinates are calculated in accordance with the formulae presented in chapter 2.3.1.2. Thereafter the betatron coordinates are transformed in accordance with the coefficients of the ring matrix in the point of the array generation:

$$\begin{pmatrix} x \\ x' \end{pmatrix}_{i+1} = \begin{pmatrix} R_{11} & R_{12} \\ R_{21} & R_{22} \end{pmatrix} \begin{pmatrix} x \\ x' \end{pmatrix}_i, \quad \begin{pmatrix} y \\ y' \end{pmatrix}_{i+1} = \begin{pmatrix} R_{33} & R_{34} \\ R_{43} & R_{44} \end{pmatrix} \begin{pmatrix} y \\ y' \end{pmatrix}_i, \quad (2.26)$$

where notation of the matrix coefficients is explained in the following table:

$$M_{RING} = \begin{pmatrix} R_{11} & R_{12} & R_{13} & R_{14} & R_{15} & R_{16} \\ R_{21} & R_{22} & R_{23} & R_{24} & R_{25} & R_{26} \\ R_{31} & R_{32} & R_{33} & R_{34} & R_{35} & R_{36} \\ R_{41} & R_{42} & R_{43} & R_{44} & R_{45} & R_{46} \\ R_{51} & R_{52} & R_{53} & R_{54} & R_{55} & R_{56} \\ R_{61} & R_{62} & R_{63} & R_{64} & R_{65} & R_{66} \end{pmatrix}. \quad (2.27)$$

The beam rotation in the longitudinal plane is provided with the usual rotation matrix:

$$\begin{pmatrix} s - s_0 \\ \frac{\Delta p}{p} \end{pmatrix}_{i+1} = \begin{pmatrix} \cos \mu & B_s \sin \mu \\ -\frac{1}{B_s} \sin \mu & \cos \mu \end{pmatrix} \begin{pmatrix} s - s_0 \\ \frac{\Delta p}{p} \end{pmatrix}_i, \quad (2.28)$$

where μ value can be arbitrary non-zero.

After the beam rotation in the longitudinal plane new values of the particles transverse co-ordinates are calculated from the betatron ones and new value of the longitudinal momentum deviation.

2.6. Calculation of output parameters

To provide benchmarking of the algorithm used results of r.m.s. dynamics simulation, one needs to perform calculations of the output data standard for all algorithms. The emittance and momentum time dependencies are calculated automatically because the emittance calculation is necessary for IBS calculation. Calculation of the emittance variation characteristic times is not necessary for the SimCool algorithm as itself. However these values can be calculated with the standard formulae for the rate values

$$\frac{1}{\tau} = \frac{1}{\varepsilon} \frac{\Delta\varepsilon}{\Delta t}. \quad (2.29)$$

The Δt in the SimCool algorithm is equal to N_{turn} multiplied by revolution period. General problem of these calculations is related to the emittance calculation accuracy. Algorithm of the emittance calculation can be tested by calculation of the characteristic times in the absence of heating and cooling effects.

In the absence of heating and cooling effects in the case of statistical emittance calculation the emittance values are changed after the beam rotation in the ring due to initial mismatching of the particle array with the ring lattice functions. The relative variation of the emittance value is proportional to

$$\frac{\Delta\varepsilon}{\varepsilon} \sim \frac{1}{\sqrt{N_p}}, \quad (2.30)$$

where N_p is the particle number. At small particle number or at large N_{turn} value this variation can be comparable with the emittance variation due to cooling and IBS.

In the case of emittance calculation as mean value of the motion invariant the emittance variation after rotation in the ring takes a place only due to the accuracy of the lattice parameters calculation through the transformation matrix coefficients. The emittance variation value does not correspondingly depend on the particle number and is sufficiently less than emittance variation due to cooling or IBS.

3. Monte-Carlo algorithm

3.1. Algorithm scheme

The problems of the calculation speed and numerical algorithm stability in simulation of a long-term process can be solved using Monte Carlo method for the beam dynamics simulation. As against the SimCool algorithm, one can provide numerical simulation of the IBS process by direct calculation of particle interactions in the frame of MS model.

MC method permits to realize both the possibility

- to calculate IBS growth rates in accordance with analytical model under some assumption about distribution function shape,
- to simulate the particle dynamics in the ring optic structure.

The first possibility permits to compare results of the calculations using SimCool algorithm and results of the MC simulations. It is very fruitful for benchmarking of both algorithms.

The second possibility presumes a solution of the problem related to the integration of the particle motion equation in a long optic structure. This problem is solved in the BETACOOOL program with the module algorithm of the accelerator structure representation used.

The algorithm of the Monte-Carlo simulations is realized in the following steps (see Fig. 3.1):

1. Loading the ring optic structure from input MAD file.
2. Reduction of the ring structure in accordance with input ASD file.
3. Thereafter the program generates the beam as a particle array in accordance with initial distribution. Each particle is presented by six-component vector including three co-ordinates and three components of momentum. The array has to be matched with the ring lattice.
4. The invariants of the motion for each particle in the beam are calculated using particle co-ordinates and ring lattice parameters in the particle position.
5. The beam is propagated through the ring by solution of the motion equation in each optic element of the ring. In the case of module structure of the accelerator the motion equations are solved only in a few points of the ring. Transformation of the particle co-ordinates between these points is provided using transformation matrices.
6. After defined number of revolution the new values of the invariants for each particle are calculated.
7. From initial and new values of invariants the rates of invariant variation for each particle are calculated.
8. One makes a step over long period of time - for each particle new values of invariants expected after given long period of time are calculated.
9. New particle co-ordinates are calculated in accordance with ring lattice parameters by random generation of phases of the betatron and synchrotron oscillations.

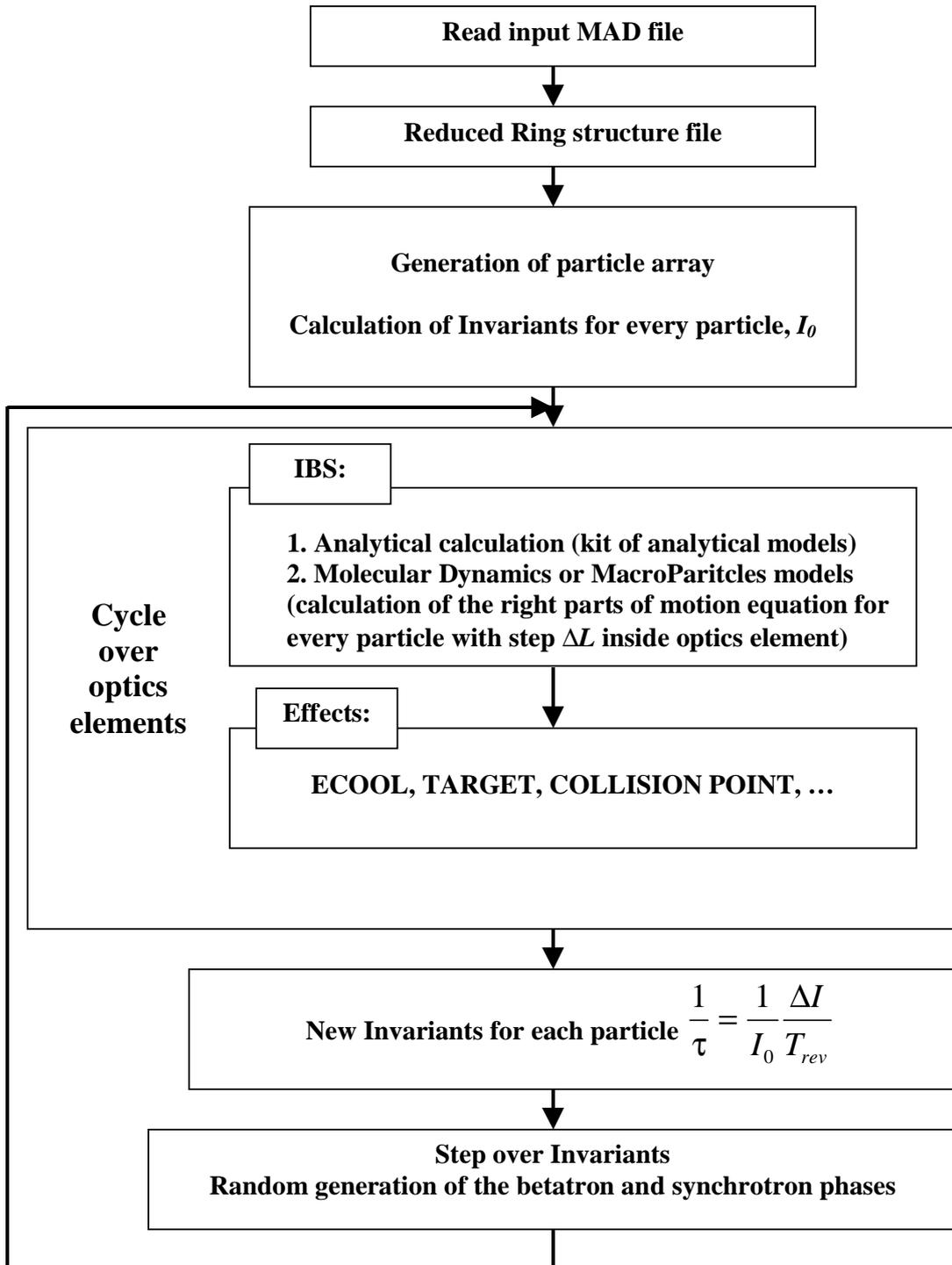


Fig.3.1. Block scheme of the Monte Carlo algorithm

The simulations of the particle interaction can be performed with two methods – macro-particle dynamics simulation or using Molecular Dynamics technique. In the test version presented in this report the MD simulations are realized. However, the module structure

of the program can be used for both the methods and after benchmarking of MD procedure the development of the macro particle simulation can be sufficiently faster.

3.2. Program structure on the basis of module algorithm

Both the algorithms – Molecular Dynamics and Macro Particle simulations are based on the solution of the particle motion equation in the external electric and magnetic fields tacking into account interaction between particles (Fig 3.2). External fields are calculated for different optic elements of the ring in accordance with the model of the element. The models of the optic elements are collected in the Library of elements – **xOptics**. The ion ring structure is described in an input file. The program is oriented to the input file format using in MAD program. The procedures required for reading the MAD input file were developed and tested.

3.3. The module library

3.3.1. The ring optic structure

The storage ring in the program is presented as array of modules. Each module of the accelerator optic structure can contain the following parameters:

- type of the module,
- the module length,
- set of the field harmonics coefficients (multipole components),
- set of the lattice functions at the entrance of the module,
- transformation matrix of the module,
- right parts of the particle motion equation inside the module,
- procedures executing with the particle at the entrance and at exit of the module.

The module parameters are initialized after the program starts during loading of input file. There are two basic format of the optic structure input file: output MAD file or input lattice file (MAD input file or SFX input file).

In the case of the ring initialization using output MAD file the module contains only the module length, set of the lattice functions and transformation matrix. The module transformation matrix is calculated from the lattice functions at the entrance and at the end of the module. In this case the numerical solution of the particle motion equations inside the ring is impossible. Correspondingly the output MAD file is used only in algorithms calculating IBS process in accordance with analytical models.

When the ring is initialized with the input MAD file initially each module contains transformation matrix and can return the right parts of the particle motion equation. At the second step of the ring initialization the program calculates matrix of the ring, lattice parameters in the ring start position (Twiss parameters of a matched beam) and provides tracking of the Twiss parameters. After completion of this procedure each module contains a total set of the parameters and the ring structure permits to realize all the procedures of the dynamic simulations: r.m.s. dynamics, SimCool, MC at analytical model of IBS and MC using solution of the particle motion equations for IBS simulation.

In the case when the reduced ring structure is used the structure initialization is provided in two stages. The first stage is the same as in the case of real structure usage – the program translates input MAD file, provides the tracking of Twiss parameters and initializes all the ring elements. At the second stage in accordance with the ASD file some lines of the elements of the ring are replaced by the module, which contains transformation matrix.

3.3.2. Description of the optic library

3.3.2.1. Variables, structure of the optic element

The expressions for the transformation matrix coefficients and right parts of the particle motion equation depend on the type of canonical conjugated variables used in the

program. In the BETACOOOL program transverse motion is described in the traditional variables: co-ordinate and transverse momentum normalized on the longitudinal momentum. Longitudinal motion is defined by the “scaled arrival time” and the longitudinal momentum spread. The vector of canonical variables is

$$X = \left(x, p_x = \frac{P_x}{P_s}, y, p_y = \frac{P_y}{P_s}, z = -(t - t_0)\beta_0 c, p_z = \frac{E - E_0}{P_s \beta_0 c} \right), \quad (3.1)$$

where x, y are the horizontal and vertical positions, p_x, p_y are the corresponding normalized momenta, z is the “arrival time” of the particle multiplied by $-c\beta_0$, p_z is the normalized momentum deviation $p_z = \Delta P / P_s$.

The particle motion equation in the optic element

$$\frac{d\ddot{X}}{dz} = \ddot{F} \quad (3.2)$$

is solved using one of numerical methods. The right part of the particle motion equation \ddot{F} can be calculated in the program as a sum of force components, in accordance with different components of electric and magnetic fields. For instance, each element at non-zero length includes the components of the right parts corresponding to the drift in a free space. These right part components are described in the chapter dedicated to “drift” element and do not repeated in other elements for economy of space.

A few elements (such as bending magnet, solenoid and so) have to provide modification of the particle co-ordinate at the entrance and exit. Generally this modification is provided by the matrices of the element’s entrance and exit, but in some cases the action can have a specific character. In the last case the element action is described in the corresponding chapter.

The element identifiers and field components variables coincide with the corresponding variables in the MAD input file. Similarly to the MAD program the effect of multipole components of the magnets is included as the parameters K_1 and K_2 , which are defined as

$$K_1 \equiv \frac{1}{B\rho} \frac{\partial B_y}{\partial x}, \quad K_2 \equiv \frac{1}{B\rho} \frac{\partial^2 B_y}{\partial x^2}, \quad (3.3)$$

where $B\rho = p_0 / q$ is the magnetic rigidity.

Matrix of some optic element can be presented as a multiplication of the matrices of the entrance, body and exit of the element. Below we present the force components, on enter and on exit actions, transformation matrices for all types of the optic library elements.

3.3.2.2. Drift space

Identifier **DRIFT**

List of parameters: **L** – the length of the drift space

Right parts of the particle motion equation:

$$F_{drift} = \left(p_x, 0, p_y, 0, \frac{p_z}{\gamma_0^2}, 0 \right) \quad (3.4)$$

Transformation matrix:

$$M_{drift} = \begin{pmatrix} 1 & L & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & L & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & \frac{L}{\gamma^2 \beta^2} \\ 0 & 0 & 0 & 0 & 0 & 1 \end{pmatrix} \quad (3.5)$$

3.3.2.3. Bending magnet

Identifier **BEND**

Parameters : *L, ANGLE, E1, E2, K1, K2, HGAP, FINT* .

L - is a length of the orbit inside the magnet, and length is a parameter of any optic element.

ANGLE - is the bending angle.

E1, E2 - are the magnet pole angles at the exit and at the entrance of the magnet correspondingly.

K1, K2 - are the coefficients of multipole components of the field.

HGAP - is the height of the magnet gap.

FINT - is a fringe field integral.

The trajectory curvature *h* is calculated as:

$$h = \frac{ANGLE}{L} \quad (3.6)$$

Transformation matrix of the magnet body:

$$M_{BEND_BODY} = \begin{pmatrix} cx & sx & 0 & 0 & 0 & \frac{dx \cdot h}{\beta} \\ -kx^2 \cdot sx & cx & 0 & 0 & 0 & \frac{sx \cdot h}{\beta} \\ 0 & 0 & cy & sy & 0 & 0 \\ 0 & 0 & -ky^2 \cdot sy & cy & 0 & 0 \\ -\frac{h \cdot sx}{\beta} & -\frac{dx \cdot h}{\beta} & 0 & 0 & 1 & \frac{L}{\gamma^2 \beta^2} - \frac{J1 \cdot h^2}{\beta} \\ 0 & 0 & 0 & 0 & 0 & 1 \end{pmatrix}, \quad (3.7)$$

where

$$\begin{aligned} kx &= \sqrt{h^2 + K1}, & ky &= \sqrt{-K1}, \\ cx &= \cos(kx \cdot L), & sx &= \frac{\sin(kx \cdot L)}{kx}, & \text{if } ANGLE = 0: & dx = 0, J1 = 0 \\ cy &= \cos(ky \cdot L), & sy &= \frac{\sin(ky \cdot L)}{ky}, & \text{if } K1 = 0: & cy = 1, sy = L, ky = 0. \\ dx &= 1 - \frac{\cos(kx \cdot L)}{kx^2}, & J1 &= \frac{L - sx}{kx^2} \end{aligned} \quad (3.8)$$

The transformation matrices of the magnet entrance and exit are calculated using the following parameters:

$$\begin{aligned} A &= tg(E1), & B &= tg(E2) \\ e1 &= A - (2h \cdot HGAP \cdot FINT(1 + \sin^2(A))), \\ e2 &= A - (2h \cdot HGAP \cdot FINT(1 + \sin^2(B))) \end{aligned} \quad (3.9)$$

and their coefficients are equal to:

$$M_{ENTER} = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ h \cdot e1 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & -h \cdot e1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{pmatrix}, \quad M_{EXIT} = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ h \cdot e2 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & -h \cdot e2 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{pmatrix} \quad (3.10)$$

Matrix of the magnet is calculated by multiplication of three matrices:

$$M_{BEND} = M_{EXIT} \cdot M_{BEND_BODY} \cdot M_{ENTER} \quad (3.11)$$

Right parts of the particle motion equation are calculated as a sum of the drift space right parts and the following components related to the bending field and inertia force:

$$F_{bend} = \left(0, \frac{p_z}{\rho} - \frac{x}{\rho^2}, 0, 0, -\frac{x}{\rho}, 0 \right) \quad (3.12)$$

During integration of the particle motion equation the particle co-ordinates at the entrance and at the exit of the magnet are transformed with the matrices M_{ENTER} and M_{EXIT} .

3.3.2.4. Quadrupole lens

Identifier: **QUADRUPOLE**

Parameters : $L, K1, TILT$.

$K1$ - is the coefficients of the quadrupole component of the field.
 $TILT$ – is a rotation angle of skew quadrupole.

Transformation matrix is calculated as following:

$$M_{QUADRUPOLE} = M_{Q_EXIT} \cdot M_{Q_BODY} \cdot M_{Q_ENTER} \quad (3.13)$$

Where the matrix of the quadrupole body is:

$$M_{Q_BODY} = \begin{pmatrix} cx & sx & 0 & 0 & 0 & 0 \\ -kx^2 \cdot sx & cx & 0 & 0 & 0 & 0 \\ 0 & 0 & cy & sy & 0 & 0 \\ 0 & 0 & -ky^2 \cdot sy & cy & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & \frac{L}{\gamma^2 \beta^2} \\ 0 & 0 & 0 & 0 & 0 & 1 \end{pmatrix}, \quad (3.14)$$

and coefficients are calculated from input parameters as follows:

$$\begin{aligned} kx &= \sqrt{K1}, & ky &= \sqrt{-K1}, \\ cx &= \cos(kx \cdot L), & sx &= \frac{\sin(kx \cdot L)}{kx}, \text{ if } K1 = 0: & cy &= 1, sy = L, ky = 0. \end{aligned} \quad (3.15)$$

$$cy = \cos(ky \cdot L), \quad sx = \frac{\sin(ky \cdot L)}{ky}$$

Matrices of the quadrupole entrance and exit are calculated in accordance with:

$$M_{Q_ENTER} = \begin{pmatrix} C & 0 & S & 0 & 0 & 0 \\ 0 & C & 0 & S & 0 & 0 \\ -S & 0 & C & 0 & 0 & 0 \\ 0 & -S & 0 & C & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{pmatrix}, M_{Q_EXIT} = (M_{Q_ENTER})^{-1}, \quad (3.16)$$

where $S = \sin(TILT)$, $C = \cos(TILT)$.

Right parts of the particle motion equation are:

$$F_{quad} = (0, -K_1x, 0, K_1y, 0, 0) \quad (3.17)$$

During integration of the particle motion equation the particle co-ordinates at the entrance and at the exit of the magnet are transformed with the matrices M_{Q_ENTER} and M_{Q_EXIT} .

3.3.2.5. RF Cavity

Identifier **RFCAVITY**

Parameters f, V

f - is the RF frequency,
 V - is the RF voltage amplitude.

Transformation matrix:

$$M_{RFcavity} = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & -\frac{\omega e V \cos\varphi}{pc} & \frac{1}{c^2} \end{pmatrix} \quad (3.18)$$

where

$$\omega = 2\pi f$$

$$\varphi = \varphi_0 - \omega t$$

The RF cavity is an example of optic element at zero length. Inside such element the particle motion equation is not integrated. The particle co-ordinates at the entrance of this element are transformed in accordance with the transformation matrix.

3.3.2.6. Solenoid of the longitudinal field

Identifier: **SOLENOID**

Parameters: Ks , L .

$Ks = \frac{B_s}{B\rho}$ - is the longitudinal field strength.

Transformation matrix:

$$M_{Solenoid} = \begin{pmatrix} C^2 & \frac{SC}{Ks} & SC & \frac{S^2}{Ks} & 0 & 0 \\ -SCKs & C^2 & -S^2Ks & \frac{SC}{Ks} & 0 & 0 \\ -SC & -\frac{S^2}{Ks} & C^2 & \frac{SC}{Ks} & 0 & 0 \\ S^2Ks & -SC & -SCKs & C^2 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & \frac{L}{\gamma^2\beta^2} \\ 0 & 0 & 0 & 0 & 0 & 1 \end{pmatrix}, \quad (3.19)$$

where

$$C = \cos(Ks \cdot L)$$

$$S = \sin(Ks \cdot L)$$

Right parts of the particle motion equation are:

$$F_{solenoid} = (0, Ksp_x, 0, -Ksp_y, 0, 0) \quad (3.20)$$

At the exit of the solenoid the particle momentum components are transformed in accordance with:

$$\begin{aligned} \Delta p_x &= Ks \cdot y \\ \Delta p_y &= -Ks \cdot x \end{aligned} \quad (3.21)$$

At the exit of the solenoid:

$$\begin{aligned} \Delta p_x &= -Ks \cdot y \\ \Delta p_y &= Ks \cdot x \end{aligned} \quad (3.22)$$

3.4. Initialization of the ring structure

In this chapter we will use the following notation for the transformation matrix coefficient:

$$M = \begin{pmatrix} R_{11} & R_{12} & R_{13} & R_{14} & R_{15} & R_{16} \\ R_{21} & R_{22} & R_{23} & R_{24} & R_{25} & R_{26} \\ R_{31} & R_{32} & R_{33} & R_{34} & R_{35} & R_{36} \\ R_{41} & R_{42} & R_{43} & R_{44} & R_{45} & R_{46} \\ R_{51} & R_{52} & R_{53} & R_{54} & R_{55} & R_{56} \\ R_{61} & R_{62} & R_{63} & R_{64} & R_{65} & R_{66} \end{pmatrix} \quad (3.23)$$

The choice of the matrix (ring or optic element) to be used in each case is explained in the text below.

In the present version of the program initialization of the ring structure is performing under assumption that the horizontal and vertical degrees of freedom are independent. Therefore the program reads only uncoupled lattice functions from the output MAD file. In the case of ring structure initialization with input MAD file the tracking of Twiss parameters is correspondingly performed with the formulae for uncoupled motion.

3.4.1. Initialization using output MAD file

When imported the output MAD file, each module of the ring optic structure contains the following parameters: length L and for each plane the beta and alpha functions, dispersion and its derivative, phase advance.

To prepare the ring structure for procedure of the structure reduction the program calculates matrices of each optic element. The coefficient of the transformation matrix are calculated from the lattice parameters at the entrance and at the exit of the element in accordance with the following formulae:

$$\begin{aligned} R_{11} &= \sqrt{\frac{\beta_2^x}{\beta_1^x}} (\cos \mu_x + \alpha_1^x \sin \mu_x), R_{12} = \sqrt{\beta_2^x \beta_1^x} \sin \mu_x \\ R_{21} &= -\frac{(\alpha_2^x - \alpha_1^x) \cos \mu_x + (1 + \alpha_2^x \alpha_1^x) \sin \mu_x}{\sqrt{\beta_2^x \beta_1^x}}, R_{22} = \sqrt{\frac{\beta_1^x}{\beta_2^x}} (\cos \mu_x - \alpha_2^x \sin \mu_x) \\ R_{33} &= \sqrt{\frac{\beta_2^y}{\beta_1^y}} (\cos \mu_y + \alpha_1^y \sin \mu_y), R_{34} = \sqrt{\beta_2^y \beta_1^y} \sin \mu_y \\ R_{43} &= -\frac{(\alpha_2^y - \alpha_1^y) \cos \mu_y + (1 + \alpha_2^y \alpha_1^y) \sin \mu_y}{\sqrt{\beta_2^y \beta_1^y}}, R_{44} = \sqrt{\frac{\beta_1^y}{\beta_2^y}} (\cos \mu_y - \alpha_2^y \sin \mu_y) \end{aligned} \quad (3.24)$$

where $\mu_x = \mu_2^x - \mu_1^x$, $\mu_y = \mu_2^y - \mu_1^y$, and subscript 1 corresponds to the lattice parameters at the entrance of the module, subscript 2 – at the exit (the lattice parameters

at the exit of the module coincide with the lattice parameters at the entrance of the next module in the module array).

3.4.2. Initialization using input MAD file. Tracking of Twiss parameters

When the ring structure is initialized using input file at the first step the program calculates matrices of all optic elements and calculates the ring matrix by their multiplication. Thereafter Twiss parameters of the matched beam in the ring start position are calculated in accordance with the following formulae:

$$\begin{aligned}
\cos \mu_x &= (R_{11} + R_{22}) / 2 \\
\sin \mu_x &= \text{sign}(R_{12}) \cdot \sqrt{R_{12} \cdot R_{21} - \frac{(R_{11} - R_{22})^2}{4}} \\
\alpha_x &= (R_{11} - R_{22}) / (2 \sin \mu_x) \\
\beta_x &= R_{12} / \sin \mu_x \\
\gamma_x &= -R_{21} / \sin \mu_x \\
D_x &= \frac{(1 - R_{22})R_{16} + R_{12}R_{26}}{2 - R_{11} - R_{22}} \\
D'_x &= \frac{(1 - R_{11})R_{26} + R_{21}R_{16}}{2 - R_{11} - R_{22}}
\end{aligned} \tag{3.25}$$

in horizontal plane, and analogously in vertical plane

$$\begin{aligned}
\cos \mu_y &= (R_{33} + R_{44}) / 2 \\
\sin \mu_y &= \text{sign}(R_{34}) \cdot \sqrt{R_{34} \cdot R_{43} - \frac{(R_{33} - R_{44})^2}{4}} \\
\alpha_y &= (R_{33} - R_{44}) / (2 \sin \mu_y) \\
\beta_y &= R_{34} / \sin \mu_y \\
\gamma_y &= -R_{43} / \sin \mu_y \\
D_y &= \frac{(1 - R_{44})R_{36} + R_{34}R_{46}}{2 - R_{33} - R_{44}} \\
D'_y &= \frac{(1 - R_{33})R_{46} + R_{43}R_{36}}{2 - R_{33} - R_{44}}
\end{aligned} \tag{3.26}$$

Here R_{ij} are the coefficients of the ring matrix.

These parameters correspond to the lattice functions at the entrance of the first ring optic module. To calculate lattice functions in other optic elements the program provide the tracking of Twiss parameters in accordance with the following formulae:

$$\begin{aligned}
\beta_2^x &= \frac{(R_{11}\beta_1^x - R_{12}\alpha_1^x)^2 + R_{12}^2}{\beta_1^x} \\
\alpha_2^x &= -\frac{(R_{11}\beta_1^x - R_{12}\alpha_1^x)(R_{21}\beta_1^x - R_{22}\alpha_1^x) + R_{12}R_{22}}{\beta_1^x} \\
\gamma_2^x &= \beta_1^x R_{21}^2 - 2\alpha_1^x R_{22}R_{21} + \gamma_1^x R_{22}^2 \\
\mu_2^x &= \mu_1^x + \arctg\left(\frac{R_{12}}{R_{11}\beta_1^x - R_{12}\alpha_1^x}\right) \\
D_2^x &= R_{11}D_1^x + R_{12}D_1^{px} + R_{16} \\
D_2^{px} &= R_{21}D_1^x + R_{22}D_1^{px} + R_{26}
\end{aligned} \tag{3.27}$$

in horizontal plane and in vertical plane:

$$\begin{aligned}
\beta_2^y &= \frac{(R_{33}\beta_1^y - R_{34}\alpha_1^y)^2 + R_{34}^2}{\beta_1^y} \\
\alpha_2^y &= -\frac{(R_{33}\beta_1^y - R_{34}\alpha_1^y)(R_{43}\beta_1^y - R_{44}\alpha_1^y) + R_{34}R_{44}}{\beta_1^y} \\
\gamma_2^y &= \beta_1^y R_{43}^2 - 2\alpha_1^y R_{44}R_{43} + \gamma_1^y R_{44}^2 \\
\mu_2^y &= \mu_1^y + \arctg\left(\frac{R_{34}}{R_{33}\beta_1^y - R_{34}\alpha_1^y}\right) \\
D_2^y &= R_{33}D_1^y + R_{34}D_1^{py} + R_{36} \\
D_2^{py} &= R_{43}D_1^y + R_{44}D_1^{py} + R_{46}
\end{aligned} \tag{3.28}$$

Here R_{ij} are the coefficients of the module matrix. Phase advances at the exit of the first module are equal to zero in both planes.

3.4.3. Format of input file describing the accelerator structure

Example of the ASD file used for reduction of the accelerator structure is presented in the Fig. 3.3. Each string of the file contains number of the element for IBS calculation, distance to this element from the ring initial point, length of the module and the identifier of the module type. To indicate heating or cooling effect located in the point of IBS calculation special identifiers are used, for instance **ECOOOL** and **TARGET**. To include the effect into input MAD file the label of corresponding element has to coincide with one of the key words: **xECOOOL** or **xTARGET**.

	0, s = 0, L = 15 : ECOOL
	10, s = 9.5, L = 2 : SBEND
	20, s = 20, L = 1 :
0 ,	30, s = 29.5, L = 1 :
39 ,	40, s = 40.5, L = 0.5 : QUADRUPOLE
80 ,	50, s = 50, L = 0.5 : QUADRUPOLE
119 , QUAD	60, s = 60.5, L = 1 :
160 , BEND	70, s = 70, L = 1 :
199 ,	80, s = 79.5, L = 2 : SBEND
240 ,	90, s = 90, L = 1 :
280 ,	100, s = 99.5, L = 1 :
321 , QUAD	110, s = 110.5, L = 0.5 : QUADRUPOLE
360 ,	120, s = 120, L = 0.5 : QUADRUPOLE
402 , BEND	130, s = 130.5, L = 1 :
441 ,	140, s = 224.7333638, L = 0 : TARGET

Fig. 3.3. Example of the ASD file format. Left – initial ASD file. Right – file describing reduced ring structure.

3.5. Intrabeam scattering simulation

3.5.1. Calculation of the growth rates on the basis of analytical models

The usage of analytical model for IBS simulation is similar to that one in SimCool algorithm. In the position of IBS calculation in accordance with the lattice function of the corresponding module the characteristic times are calculated using one of the models developed in the BETACOOOL. When the growth rates are known one can calculate mean square of the scattering angle taking into account the module length:

$$\langle \theta^2 \rangle = \frac{\varepsilon}{\beta} \frac{T_{rev}}{\tau} \frac{L}{C}, \quad (3.29)$$

where β is the beta function in the position of IBS calculation, τ is characteristic growth time for corresponding degree of freedom, L is the module length, C is the ring circumference.

The variation of the particle trajectory angular deviation is calculated as in SimCool algorithm in accordance with

$$\Delta\theta = \sqrt{\langle \theta^2 \rangle} \xi \quad (3.30)$$

where ξ is the random value with Gaussian distribution at unit dispersion. After completion of the beam revolution in the ring the particle angle deviation will be averaged over the lattice functions in the points of IBS calculation.

3.5.2. Molecular dynamics

The particle momentum variation due to IBS process can be calculated by numerical solution of the particle motion equation in the real optic structure taking into account interactions between particles.

Since the calculation of the space-charge effect is time consuming, one can use two possibilities to reduce required particle number:

- molecular dynamics,
- macro particle simulation.

In the frame of Molecular Dynamics (MD) technique, the beam is divided into several cells and the motion equations are solved for the particles from one cell only. Another particles are taking into account using a periodic boundary condition for the distribution function in the longitudinal degree of freedom (Fig. 3.4).

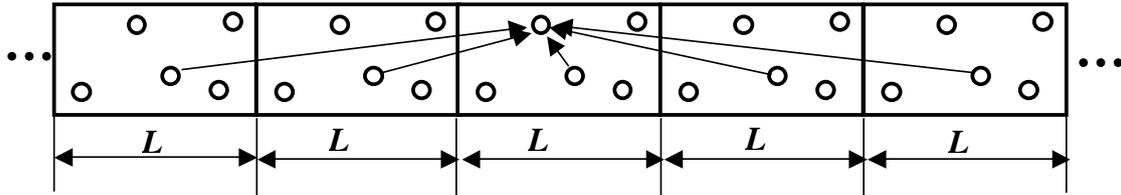


Fig. 3.4. The beam model used in Molecular Dynamics simulation.

The right part of the particle motion related with the interaction between particles has the following components:

$$F_{MD} = (0, F_{px}, 0, F_{py}, 0, F_{pz}). \quad (3.31)$$

They are calculated for interaction between two particles with the formulae;

$$F_{p,x} = \frac{1}{4\pi\epsilon_0} \frac{q^2 (X - X_i)}{m_0 c^2 \gamma_0^2 \beta_0^2} \left(\frac{1}{a^3} - \frac{2I_1}{rL^2} \right), \quad X \text{ is one of the transverse coordinates,}$$

$$F_{p,z} = \frac{1}{4\pi\epsilon_0} \frac{q^2 (z - z_i)}{m_0 c^2 \gamma_0^2 \beta_0^2} \left(\frac{1}{a^3} - \frac{2I_0}{sL^2} \right), \quad (3.32)$$

a is the distance between particles, $s = z - z_i$, $r = \sqrt{(x - x_i)^2 + (y - y_i)^2}$, L is the MD cell size. The total value of the force component acting on the given particle is calculated as a sum over all the particles inside the cell.

In the formulae the I_1 and I_0 are the following integrals:

$$I_1 = -\int_0^{\infty} \frac{k J_1(kr/L) \sinh(ks/L) - 1}{\exp(k) - 1} dk, \quad I_0 = -\int_0^{\infty} \frac{k J_0(kr/L) \cosh(ks/L) - 1}{\exp(k) - 1} dk, \quad (3.33)$$

J_1 is the Bessel function of 1st order, m_0c^2 – ion rest mass. The integral values were numerically calculated and are used in the program as table values.

The initial distribution of the particles inside the cell is generated randomly in accordance with the beam emittance and the ring lattice parameters in the initial position. Transverse particle co-ordinates and all three momentum components are generated in accordance with the Gaussian law, longitudinal co-ordinates are distributed uniformly inside the cell. Thereafter the particles in the sell are propagated through the ring module by module in accordance with the ring structure.

Annex. IBS growth rates calculation in accordance with Jie Wei model

In terms of normalized transverse emittance and longitudinal bunch area

$S = \pi m_0 c^2 \gamma \sigma_s \sigma_p / \beta^3 c A$ in phase space, Eq.(4.44) can be rewritten

$$\begin{bmatrix} \frac{1}{\sigma_p} \frac{d\sigma_p}{dt} \\ \frac{1}{\sigma_x} \frac{d\sigma_x}{dt} \\ \frac{1}{\sigma_y} \frac{d\sigma_y}{dt} \end{bmatrix} = \frac{Z^4 N}{A^2} \frac{r_0^2 m_0 c^2 L_c}{8 \beta^4 \gamma \epsilon_x \epsilon_y S} F(\chi) \begin{bmatrix} n_b (1 - d^2) \\ -a^2 / 2 + d^2 \\ -b^2 / 2 \end{bmatrix} \quad (A.1)$$

Except for the beam factors χ , d , a and b that depend on the ratio of the beam amplitudes in different dimension, the rates are linearly proportional to the density in the six-dimensional phase space, and are strongly dependent on the charge state of the particle.

The coupling between the horizontal and vertical motion averages the growth rates in the transverse dimension. If the motion is fully coupled within time periods much shorter than the IBS diffusion time, the average rates become:

$$\begin{bmatrix} \frac{1}{\sigma_p} \frac{d\sigma_p}{dt} \\ \frac{1}{\sigma_{x,y}} \frac{d\sigma_{x,y}}{dt} \end{bmatrix} = \frac{Z^4 N}{A^2} \frac{r_0^2 m_0 c^2 L_c}{8\beta^4 \gamma \epsilon_x \epsilon_y S} F(\chi) \begin{bmatrix} n_b(1-d^2) \\ (-\chi + d^2)/2 \end{bmatrix} \quad (\text{A.2})$$

In a typically circular accelerator, the transition energy γ_T is approximately equal to the average value of β_x / D_p in the regular cells. When the beam energy is high $\gamma \gg \gamma_T$, the growth in horizontal direction results mostly from the variation of the betatron closed orbit during the exchange of the particle momentum ($a^2 \ll d^2$). The growths in horizontal and longitudinal amplitudes are therefore proportional to each other.

When the vertical amplitude is no longer small so that $\chi > 1$ and $F(\chi) < 0$. According to equation (4.44), both horizontal and longitudinal amplitudes grow.

In a typical storage ring like the Relativistic Heavy Ion Collider (RHIC) the beams are stored at energies much higher than the transition energy. Due to the coupling and injection conditions, the horizontal and vertical betatron amplitudes are about the same. The growth rates can be explicitly written from Eq. (4.44) using the expression for $F(\chi)$:

$$\begin{bmatrix} \frac{1}{\sigma_p} \frac{d\sigma_p}{dt} \\ \frac{1}{\sigma_{x,y}} \frac{d\sigma_{x,y}}{dt} \end{bmatrix} = \frac{Z^4 N}{A^2} \frac{r_0^2 m_0 c^2 L_c}{16\beta^4 \gamma_T \epsilon_x \epsilon_y S} \begin{bmatrix} n_b(1-d^2)/d \\ d/n_c \end{bmatrix} \quad (\text{A.3})$$

Their dependence on the energy of the beam, which appears only in the form factor d , is usually weak. After the initial stage of the storage, the asymptotic configuration

$$n_b n_c \sigma_x^2 \approx D_p^2 \sigma_p^2, \quad \gamma \gg \gamma_T \quad (\text{A.4})$$

Will be approximately reached ($d \approx n_b n_c / (1 + n_b n_c)$).