Software Development for Beam Cooling Simulation Including General Collider Physics

Final report
2007

I.N.Meshkov, A.O. Sidorin, A.V.Smirnov, G.V.Trubnikov,V.N.Malakhov
Joint Institute for Nuclear Research, Dubna, Russia

Abstract

General attention at this stage of the work was devoted to development of the electron cooling model represented by the force vector and the tensor of diffusion coefficients for the real electron distribution function. The same model was realized for intrabeam scattering simulation at arbitrary shape of the distribution function.

The algorithms for the luminosity calculation and beam-beam effect simulation were improved.

A simplified model for simulation of longitudinal stochastic cooling at RHIC was developed.

Benchmarking of the electron cooling simulation was performed using results of dedicated experiments at Fermilab Recycler. In order to provide realistic comparison between simulation of antiproton beam dynamics and experiments new models of electron beam with parabolic density distribution and density distribution imported from an external file were developed. A new model of the ion synchrotron motion at rectangular RF burrier bucket was implemented. Results of the Fermilab experiment simulations will be a topic of independent report.

Contents

1. Local models for IBS and electron cooling

1.1. Overview

Main goal of the local model for IBS process is to simulate the distribution function evolution without additional assumption about its shape. Electron cooling application leads to formation of bi-Gaussian distribution in the ion beam. For simulation of IBS process in this case the "core-tail" model is used in BETACOOL now. However, this model includes a few free parameters that can not be calculated from the distribution as itself. The local model will give a possibility to benchmark the "core-tail" model and provide a choice of its parameters on the basis of the beam dynamics simulation.

The local model of the electron cooling is necessary to compare different electron distribution from the side of the cooling process efficiency.

To realise the local models for IBS and electron cooling the structure of the beam object in the code was modified. The bunch of the particle can be presented in the laboratory or in the particle rest frame. Corresponding modules for particle co-ordinate transformation from laboratory frame to beam frame and back were introduced. The friction force and diffusion tensor components are calculated in the particle rest frame. In the current version of the algorithm a kick of the ion momentum due to action of electron cooling is provided in the laboratory frame, kick of the ion momentum due to action of IBS – in the beam frame. All the others parts of the algorithms are the same. To reduce the simulation time in the case of IBS simulations the simplified optic structure of the ion ring is used. The total lattice is reduced to a few optic elements (10 – 20) that have the same properties from the side of IBS process.

The new models are based on the statement that intrabeam scattering (and ion scattering on electrons) is the local process and the ion interacts efficiently only with relatively small number of nearest particles. The particle density in the vicinity of the ion in this case is closed to uniform, the friction force and diffusion tensor components can be calculated using well known formulae from plasma physics.

In the frame of local algorithm the program finds in the total array of the particle a small local array and calculates local density and rms parameters of the particle distribution in this local array. The local parameters are used for calculation of the friction and diffusion components. To avoid systematic error in local density evaluation the program calculates number of particles inside a small cell surrounding the test ion. Dimensions of this cell are calculated from rms dimensions of the local array. Such an algorithm permits to adjust the cell dimensions to the local density of the particle distribution and provide an accurate calculation in a dense core and in tails of the distribution function.

The algorithm includes three parameters: total number of the particles in the array N, number of the local particles N_{loc} and dimensions of the cell used for local density evaluation.

The local particles are found in the beam frame; therefore for RHIC parameters the local array has a specific shape. Expected electron bunch length in the laboratory frame is about 1 mm, the ion bunch length is between 15 and 30 cm that corresponds in the beam frame to about 1 and more than 15 meters. The transverse dimensions of the electron and ion bunches are closed to each other in the cooling section and expected to be between 3 and 5 mm. The ion beam transverse dimensions in other optic elements are about 1-2 mm. Thus the bunch length is about two orders of magnitude larger than the transverse dimensions. Therefore the array of the local particle almost coincides with a longitudinal slice inside the bunch. The local rms transverse

sizes are close to the global ones, when the rms length of the local array is less than the total bunch length by the ratio between total and local particle number. The electron and ion bunches has approximately axial symmetry shape in the transverse plane.

To take into account these peculiarities of the particle distribution and provide fast and accurate algorithm the following procedure for the local density evaluation is used. The local density is calculated as a particle number located inside an elliptical cylinder of the length of $2\sigma_s$ and half-axis of the cross-section of $\alpha\sigma_x$ and $\alpha\sigma_y$ divided by the volume of this cylinder: $2\pi\alpha^2\sigma_s\sigma_x\sigma_y$. Where σ_s, σ_x, σ_y are the rms longitudinal and transverse dimensions of the local array and α is numerical coefficient less than unity.

For benchmarking of the local model an array of electrons can be generated in the program in accordance with the following distributions:
- Gaussian,
- bi-Gaussian,
- uniform in transverse plane and Gaussian along the bunch.

A few new procedures was developed for the benchmarking: the program can output local density and the friction force components in a given position inside electron bunch, which are calculated from the local array or analytically. This permits to adjust parameters of the model, such as the particle numbers in global and in local arrays and the cell dimensions, to obtain required accuracy of the simulations.

1.2. Friction and diffusion in an array of particles

Calculation of the friction force and diffusion tensor components related with the problem of coulomb scattering of a test particle of a mass m_t and velocity of $\vec{V}$ in an array of N_{loc} field particles of a mass m_f and velocities $\vec{v}_i$ (Fig. 1.1).

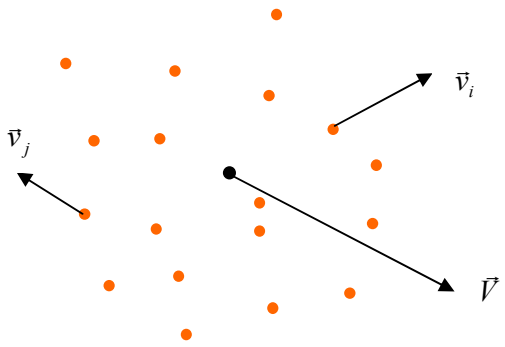


Fig. 1.1. Test particle (black circle) in the local cloud of field particles (colored circles).

Solution of this problem is well known from the plasma physics. At the distribution function of the field particles in the velocity space of $f(v)$ the friction force is equal to

$$\vec{F} = \frac{\langle \Delta \vec{p} \rangle}{\Delta t} = -\frac{4\pi n e^4 Z_t^2 Z_f^2}{\left(\frac{m_f m_t}{m_f + m_t}\right)} \int \ln\left(\frac{\rho_{\max}}{\rho_{\min}}\right) \frac{\vec{U}}{U^3} f(v) d^3 v \qquad (1.2.1)$$

and the diffusion tensor components are

$$D_{\alpha,\beta} = \frac{\left\langle \Delta p_\alpha \Delta p_\beta \right\rangle}{\Delta t} = 4\pi n e^4 Z_t^2 Z_f^2 \int \ln\left(\frac{\rho_{\max}}{\rho_{\min}}\right) \frac{U^2 \delta_{\alpha,\beta} - U_\alpha U_\beta}{U^3} f(v) dv \,. \qquad (1.2.2)$$

Here $\alpha, \beta = x, y, z$, the angular brackets mean averaging over the field particles, Z_t, Z_f are the charge numbers of the test and field particle, $\vec{U} = \vec{V} - \vec{v}$ is the relative velocity of the test and field particle. The minimum and maximum impact parameters are determined as in electron cooling simulation.

The distribution function of the field particles in the velocity space is given as a series of δ-functions:

$$f(v) = \frac{1}{N_{loc}} \sum_{j=1}^{N_{loc}} \delta\left(\vec{v} - \vec{v}_j\right). \qquad (1.2.3)$$

$$F_\alpha = \frac{4\pi n Z_t^2 Z_f^2 e^4}{\left(\frac{m_f m_t}{m_f + m_t}\right)} \frac{1}{N_{loc}} \sum_{j=1}^{N_{loc}} \frac{\left(V_\alpha - v_{j,\alpha}\right) L_{C,j}}{\left(\sqrt{\left(V_x - v_{j,x}\right)^2 + \left(V_y - v_{j,y}\right)^2 + \left(V_z - v_{j,z}\right)^2}\right)^3} \qquad (1.2.4)$$

n – mean local density of the field particles, N_{loc} – number of the local field particles, V_α is the component of the test particle velocity, $v_{j,\alpha}$ - velocity component of j-th field particle, $\alpha = x, y, z$.

The minimum impact parameter in the Coulomb logarithm is calculated as

$$\rho_{\min} = \frac{Z_t Z_f e^2}{\left(\frac{m_f m_t}{m_f + m_t}\right)} \frac{1}{\left|\vec{V} - \vec{v}_j\right|^2} \,. \qquad (1.2.5)$$

The dynamic shielding radius value required for the maximum impact parameter determination is calculated using rms velocity spread of the field particles.

The components of the diffusion tensor are

$$D_{\alpha,\beta} = 4\pi n Z_t^2 Z_f^2 e^4 \frac{1}{N_{loc}} \sum_{j=1}^{N_{loc}} \frac{\left(\left(\left(V_x - v_{j,x}\right)^2 + \left(V_y - v_{j,y}\right)^2 + \left(V_z - v_{j,z}\right)^2\right)\delta_{\alpha\beta} - \left(V_\alpha - v_{j,\alpha}\right)\left(V_\beta - v_{j,\beta}\right)\right) L_{C,j}}{\left(\sqrt{\left(V_x - v_{j,x}\right)^2 + \left(V_y - v_{j,y}\right)^2 + \left(V_z - v_{j,z}\right)^2}\right)^3} \qquad (1.2.6)$$

All the values are calculated in the particle rest frame. In the general case all the components of the diffusion tensor have nonzero values.

The presented formulae can be used for electron cooling simulation, when the electron bunch is presented as an array of particles, as well as for IBS simulation in the frame of Model Beam algorithm.

An universal procedures for the friction and diffusion calculation was developed and they is under benchmarking now in the frame of electron cooling simulation with a real electron distribution and IBS simulations at RHIC parameters. Preliminary results are presented in the chapter 1.6.

The local models are based on the assumption that in the small vicinity of the test particle the density distribution of the field particle is closed to uniform. The accuracy of the local density calculation determines mainly the accuracy of the model. The friction force and diffusion can be calculated, for instance, using analytical expressions (1.2.1) and (1.2.2) assuming Gaussian distribution of the field particles in the velocity space.

1.3. Local density calculation, accuracy of the model

In the frame of local algorithm the program finds in the total array of the particle a small local array and calculates local density and rms parameters of the particle distribution in this local array. The local parameters are used for calculation of the friction and diffusion.

The algorithm includes three parameters: total number of the particles in the array N, number of the local particles N_{loc} and dimensions of the cell used for local density evaluation.

One of the main sources of the error in calculations is the accuracy of the local density evaluation. To avoid systematic error in local density evaluation the program calculates number of particles inside a small cell surrounding the test ion. Dimensions of this cell are calculated from rms dimensions of the local array. Such an algorithm permits to adjust the cell dimensions to the local density of the particle distribution and provide an accurate calculation in a dense core and in tails of the distribution function.

The local particles are found in the beam frame; therefore for RHIC parameters the local array has a specific shape. Expected electron bunch length in the laboratory frame is about 1 mm, the ion bunch length is between 15 and 30 cm that corresponds in the beam frame to about 1 and more than 15 meters. The transverse dimensions of the electron and ion bunches are closed to each other in the cooling section and expected to be between 3 and 5 mm. The ion beam transverse dimensions in other optic elements are about 1-2 mm. Thus the bunch length is about two orders of magnitude larger than the transverse dimensions. Therefore the array of the local particle almost coincides with a longitudinal slice inside the bunch. The local rms transverse sizes are close to the global ones, when the rms length of the local array is less than the total bunch length by the ratio between total and local particle number. The electron and ion bunches has approximately axial symmetry shape in the transverse plane.

To take into account these peculiarities of the particle distribution and provide fast and accurate algorithm the following procedure for the local density evaluation is used. The local density is calculated as a particle number located inside an elliptical cylinder of the length of $2\sigma_s$ and half-axis of the cross-section of $\alpha\sigma_x$ and $\alpha\sigma_y$ divided by the volume of this cylinder:

$$V = 2\pi\alpha^2\sigma_s\sigma_x\sigma_y \ . \qquad (1.3.1).$$

Where σ_s, σ_x, σ_y are the rms longitudinal and transverse dimensions of the local array and α is numerical coefficient less than unity.

The ratio between local particle number and the particle number inside the cell can be estimated as a ratio between the volume of local array and volume of the cell. The volume of the local

array can be estimated assuming uniform distribution along the array and Gaussian distribution across it as:

$$V_{loc} \approx 4\sqrt{3}\sigma_s 2\pi\sigma_x\sigma_y . \tag{1.3.2}$$

Thus the local number and the particle number inside the cell are related with each other as

$$N_{loc} \approx \frac{7}{\alpha^2} N_{cell} \tag{1.3.3}$$

The N_{cell} determines stochastic fluctuations in the density evaluation and for 10% of the fluctuation expectation the N_{cell} has to be about 100.

Systematic error in the local density evaluation can be estimated on example of one dimensional Gaussian distribution. Lets assume that the particle array is a random population from the Gaussian distribution at variance σ:

$$n(x) = \frac{N}{\sqrt{2\pi}\sigma} \exp\left(-\frac{x^2}{2\sigma^2}\right). \tag{1.3.4}$$

Evaluation of the local density in the co-ordinate x is based on calculation of the particle number N_{box} inside a small vicinity a of this co-ordinate and the local density is assumed to be $N_{box}/2a$. This algorithm leads to transformation of the initial distribution $n(x)$ to new distribution $n'(x)$, that keeps the Gaussian shape, but the variance becomes to be larger (see Fig. 1.2):

$$n'(x) = \frac{\int_{x-a}^{x+a} n(y)dy}{2a} = \frac{N}{\sqrt{2\pi}\sigma'} \exp\left(-\frac{x^2}{2\sigma'^2}\right). \tag{1.3.5}$$

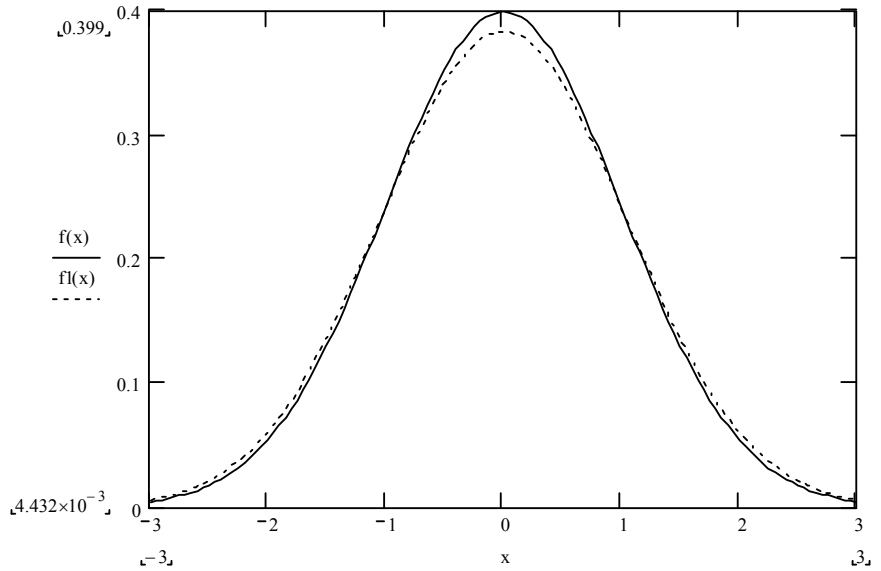


Fig. 1.2. Initial distribution (solid line) and "smoothed" distribution (dot line) when the averaging of the distribution is provided inside interval of 0.5 σ. x is plotted in σ units.

The variance σ′ of the new distribution is approximately equal to

$$\sigma'^2 \approx \sigma^2 + 0.35 \cdot a^2 .$$

Relative increase of the distribution width can be estimated from

$$\frac{\sigma'^2 - \sigma^2}{\sigma^2} \approx 0.35\left(\frac{a}{\sigma}\right)^2$$

under assumption that $\left(\frac{a}{\sigma}\right) << 1$. In this case $\sigma' + \sigma \approx 2\sigma$ and

$$\frac{\Delta\sigma}{\sigma} \approx 0.175\left(\frac{a}{\sigma}\right)^2 . \qquad (1.3.6)$$

This increase of the distribution width corresponds to systematic error in the local density evaluation: the density is underestimated in a center of the distribution and overestimated in tails. To keep the systematic error at small value one needs to minimize (a/σ), or in the case of local density calculation the parameter α. In one dimensional case the (a/σ) = 0.3 corresponds to relative mistake of about 1%.

1.4. Simulation of the diffusion processes in the Model Beam algorithm

The values of the friction and diffusion components are used for calculation of the momentum components variation. For electron cooling simulation the ion momentum components are changed after crossing the cooling section. The IBS process is simulated in each optic element of the ring. The model particle momentum variation after crossing an optic element providing a diffusion due to some physics process are calculated in accordance with Langevin equation:

$$P_i(t+\Delta t) = P_i(t) + \sqrt{\Delta t}\sum_{j=1}^{3} C_{i,j}\xi_j , \qquad (1.4.1)$$

where ξ_j are three Gaussian random numbers with unit dispersion. The coefficients $C_{i,j}$ have to be calculated from diffusion tensor coefficients.

In the general case the diffusion tensor components form a diagonal symmetric matrix:

$$\begin{pmatrix} D_{x,x} & D_{x,y} & D_{x,z} \\ D_{x,y} & D_{y,y} & D_{y,z} \\ D_{x,z} & D_{y,z} & D_{z,z} \end{pmatrix}, \qquad (1.4.2)$$

and depending on the process some of them can be equal to zero.

In the presence of the diffusion the mean values of the momentum component variation can be expressed via diffusion tensor components in accordance with the definition:

$$\left\langle \frac{d\left(P_i P_j\right)}{dt} \right\rangle = D_{i,j}, \qquad (1.4.3)$$

where triangular brackets mean averaging over the particles.

To find expressions for $C_{i,j}$ lets multiply the momentum variation for *i* and *j-th* particles:

$$P_i(t+\Delta t)P_j(t+\Delta t) = \left(P_i(t) + \sqrt{\Delta t}\sum_{k=1}^{3} C_{i,k}\xi_k \right)\left(P_j(t) + \sqrt{\Delta t}\sum_{k=1}^{3} C_{j,k}\xi_k \right) =$$

$$= P_i(t)P_j(t) + P_i(t)\sqrt{\Delta t}\sum_{k=1}^{3} C_{j,k}\xi_k +$$
$$+ P_j(t)\sqrt{\Delta t}\sum_{k=1}^{3} C_{i,k}\xi_k + \sum_{k=1}^{3} C_{i,k}\xi_k \sum_{k=1}^{3} C_{j,k}\xi_k \Delta t$$

and average this expression over the particles. Taking into account that

$$\left\langle P_j(t)\sqrt{\Delta t}\sum_{k=1}^{3} C_{i,k}\xi_k \right\rangle = 0\,,$$

$$\left\langle \xi_i \xi_j \right\rangle = \delta_{i,j}\,,$$

we obtain

$$\frac{\left\langle \Delta P_i P_j \right\rangle}{\Delta t} = \sum_{k=1}^{3} C_{i,k} C_{j,k}\,. \qquad (1.4.4)$$

($\delta_{i,j}$ is the Kronecker-Kapelli symbol.) The coefficients $C_{i,k}$ have to be chosen to obtain the same values of momentum variation (1.4.3), that gives the following system of equations:

$$\sum_{k=1}^{3} C_{i,k} C_{j,k} = D_{i,j}\,, \qquad (1.4.5)$$

or in the matrix form:

$$CC^T = D\,. \qquad (1.4.6)$$

It is a system of 6 non-linear algebraic equations, which has an infinite number of solutions. Here we describe more important specific cases and possible way of the system solution in the general case.

At **diagonal diffusion** tensor in the case when the momentum component variations do not correlate with each other, the simplest solution is:

$$C_{x,1} = \sqrt{D_{x,x}}\,,\; C_{y,2} = \sqrt{D_{y,y}}\,, C_{z,3} = \sqrt{D_{z,z}}\,, \qquad (1.4.7)$$

all the other coefficients are equal to zero.

The case, when the diffusion tensor has one non-zero off-diagonal coefficients was analyzed by Zenkevich and Boine-Frankenheim. This solution is necessary for kinetic IBS simulation based on Bjorken-Mtingwa theory in the case of zero vertical dispersion in the ring.

In presence of the horizontal and vertical dispersion the analytical expression for the diffusion tensor was derived by Venturini. In this case only $D_{x,y} = 0$ and the diffusion tensor has a form

$$\begin{pmatrix} D_{x,x} & 0 & D_{x,z} \\ 0 & D_{y,y} & D_{y,z} \\ D_{x,z} & D_{y,z} & D_{z,z} \end{pmatrix}, \tag{1.4.8}$$

the solution of the system (1.4.6) can be build by the following way. Total set of the equations can be rewritten by components as:

$$\begin{gathered} C_{x,1}C_{y,1} + C_{x,2}C_{y,2} + C_{x,3}C_{y,3} = D_{x,y} \\ C_{x,1}C_{z,1} + C_{x,2}C_{z,2} + C_{x,3}C_{z,3} = D_{x,z} \\ C_{y,1}C_{z,1} + C_{y,2}C_{z,2} + C_{y,3}C_{z,3} = D_{y,z} \\ C_{x,1}^2 + C_{x,2}^2 + C_{x,3}^2 = D_{x,x} \\ C_{y,1}^2 + C_{y,2}^2 + C_{y,3}^2 = D_{y,y} \\ C_{z,1}^2 + C_{z,2}^2 + C_{z,3}^2 = D_{z,z} \end{gathered} \tag{1.4.9}$$

Lets assume, that the random number ξ_1 correspond to scattering in the horizontal plane, ξ_2 – in the vertical plane and put $C_{x,2} = C_{y,1} = 0$. From the first equation of the system (1.4.9) it follows that $C_{x,3}C_{y,3} = 0$. Lets put $C_{x,3} = 0$. In this case

$$C_{x,1} = \sqrt{D_{x,x}}\,. \tag{1.4.10}$$

From the second equation of (1.4.9) follows

$$C_{z,1} = \frac{D_{x,z}}{\sqrt{D_{x,x}}}\,. \tag{1.4.11}$$

Then, for simplicity put

$$C_{y,2} = C_{y,3} = \sqrt{D_{y,y}/2}\,. \tag{1.4.12}$$

From the third equation of (1.4.9):

$$C_{z,2} = \frac{D_{y,z}}{\sqrt{D_{y,y}/2}} - C_{z,3}\,. \tag{1.4.13}$$

Substituting (1.4.11) and (1.4.13) into the last equation of (1.4.9) we obtain quadratic equation about $C_{z,3}$:

$$C_{z,3}^2 - \frac{D_{y,z}}{\sqrt{D_{y,y}/2}} C_{z,3} + \frac{\left(D_{x,z}\right)^2}{2D_{x,x}} + \frac{\left(D_{y,z}\right)^2}{D_{y,y}} - \frac{D_{z,z}}{2} = 0 .$$

which gives

$$C_{z,3} = \frac{D_{y,z}}{\sqrt{2D_{y,y}}} \pm \sqrt{\frac{D_{z,z}}{2} - \frac{\left(D_{x,z}\right)^2}{2D_{x,x}} - \frac{\left(D_{y,z}\right)^2}{2D_{y,y}}} . \qquad (1.4.14)$$

Fixing the sign "plus" in the last expression one can write total set of the coefficients:

$$\begin{aligned} C_{x,1} &= \sqrt{D_{x,x}} , \\ C_{y,2} = C_{y,3} &= \sqrt{D_{y,y}/2} , \\ C_{z,1} &= \frac{D_{x,z}}{\sqrt{D_{x,x}}} , \\ C_{z,2} &= \frac{D_{y,z}}{\sqrt{D_{y,y}/2}} - \sqrt{\frac{Det\, D}{2D_{x,x}D_{y,y}}} , \\ C_{z,3} &= \frac{D_{y,z}}{\sqrt{2D_{y,y}}} + \sqrt{\frac{Det\, D}{2D_{x,x}D_{y,y}}} , \end{aligned} \qquad (1.4.15)$$

all the other are equal to zero. Here *Det D* is the determinant of the matrix *D*.

In the general case when all the $D_{i,j} \neq 0$ the solution can be found as follows. To solve the equation (5.38) lets rotate the co-ordinate system around *z* axis by some angle φ that transforms the matrix *D* into the new matrix *G* having $G_{xy} = 0$. The transformation of the matrix coefficients is provided in accordance with

$$G = UDU^{-1}, \qquad (1.4.16)$$

where *U* is the following matrix

$$U = \begin{pmatrix} \cos\varphi & \sin\varphi & 0 \\ -\sin\varphi & \cos\varphi & 0 \\ 0 & 0 & 1 \end{pmatrix} \qquad (1.4.17)$$

having the property:

$$U^{-1} = U^T = \begin{pmatrix} \cos\varphi & -\sin\varphi & 0 \\ \sin\varphi & \cos\varphi & 0 \\ 0 & 0 & 1 \end{pmatrix}$$

After that the system of equations (1.4.6) can be rewritten as

$$AA^T = G, \quad A = UC . \qquad (1.4.18)$$

The coefficients of the matrix *G* are determined by the equations:

$$
\begin{aligned}
G_{xx} &= D_{xx}\cos^2\varphi + D_{yy}\sin^2\varphi + D_{xy}\sin 2\varphi,\\
G_{yy} &= D_{xx}\sin^2\varphi + D_{yy}\cos^2\varphi - 2D_{xy}\sin 2\varphi,\\
G_{zz} &= D_{zz},\\
G_{xy} &= G_{yx} = \frac{1}{2}\left(D_{yy} - D_{xx}\right)\sin 2\varphi + D_{xy}\cos 2\varphi,\\
G_{xz} &= G_{zx} = D_{xz}\cos\varphi + D_{yz}\sin\varphi,\\
G_{yz} &= G_{zy} = D_{yz}\cos\varphi - D_{xz}\sin\varphi.
\end{aligned}
\tag{1.4.19}
$$

The requirement $G_{xy} = 0$ determines the rotation angle φ:

$$
\text{if } D_{xx} = D_{yy}\ \ \varphi = \frac{\pi}{4}, \text{ in the opposite case}
$$

$$
\varphi = \frac{1}{2}\arctan\left(\frac{2D_{12}}{D_{11} - D_{22}}\right). \tag{1.4.20}
$$

In this case the equation (1.4.18) has the same solution (formulae 1.4.15):

$$
\begin{aligned}
&A_{x1} = \sqrt{G_{xx}}, \quad A_{x2} = 0, \quad A_{x3} = 0,\\
&A_{y1} = 0, \quad A_{y2} = A_{y3} = \sqrt{G_{yy}/2},\\
&A_{z1} = \frac{G_{xz}}{\sqrt{G_{xx}}}, \quad A_{z2} = \frac{G_{yz}}{\sqrt{G_{yy}/2}} - \sqrt{\frac{\det G}{2G_{xx}G_{yy}}},\\
&A_{z3} = \frac{G_{yz}}{\sqrt{2G_{yy}}} + \sqrt{\frac{\det G}{2G_{xx}G_{yy}}}.
\end{aligned}
\tag{1.4.21}
$$

The coefficients of the matrix C are calculated from the equation $C = U^{-1}A$:

$$
\begin{aligned}
&C_{x1} = A_{x1}\cos\varphi, \quad C_{x2} = -A_{y2}\sin\varphi, \quad C_{x3} = -A_{y3}\sin\varphi,\\
&C_{y1} = A_{x1}\sin\varphi, \quad C_{y2} = A_{y2}\cos\varphi, \quad C_{y3} = A_{y3}\cos\varphi,\\
&C_{z1} = A_{z1}, \quad C_{z2} = A_{z2}, \quad C_{z3} = A_{z3}.
\end{aligned}
\tag{1.4.22}
$$

The formulae (1.4.15) are used in Betacool for Kinetic model of IBS simulations (based on analytical expressions for the diffusion tensor components), the formulae (1.4.22) describes the Langevin coefficients in the local models of IBS and electron cooling.

1.5. Electron cooling simulation, benchmarking tools, friction models

Properties of numerical algorithm for local simulations were investigated on example of electron distribution analysis.

For benchmarking of the local model an array of electrons can be generated in the program in accordance with the following distributions:
- Gaussian,
- bi-Gaussian,

- uniform in transverse plane and Gaussian along the bunch.

A few new procedures was developed for the benchmarking: the program can output local density and the friction force components in a given position inside electron bunch, which are calculated from the local array or analytically. This permits to adjust parameters of the model, such as the particle numbers in global and in local arrays and the cell dimensions, to obtain required accuracy of the simulations.

For instance, the array of particles at Gaussian distribution in all degrees of freedom generated by the program is analyzed in the Fig. 1.3. The calculated local density is output as a function of horizontal and longitudinal co-ordinates at zero value of vertical co-ordinates.

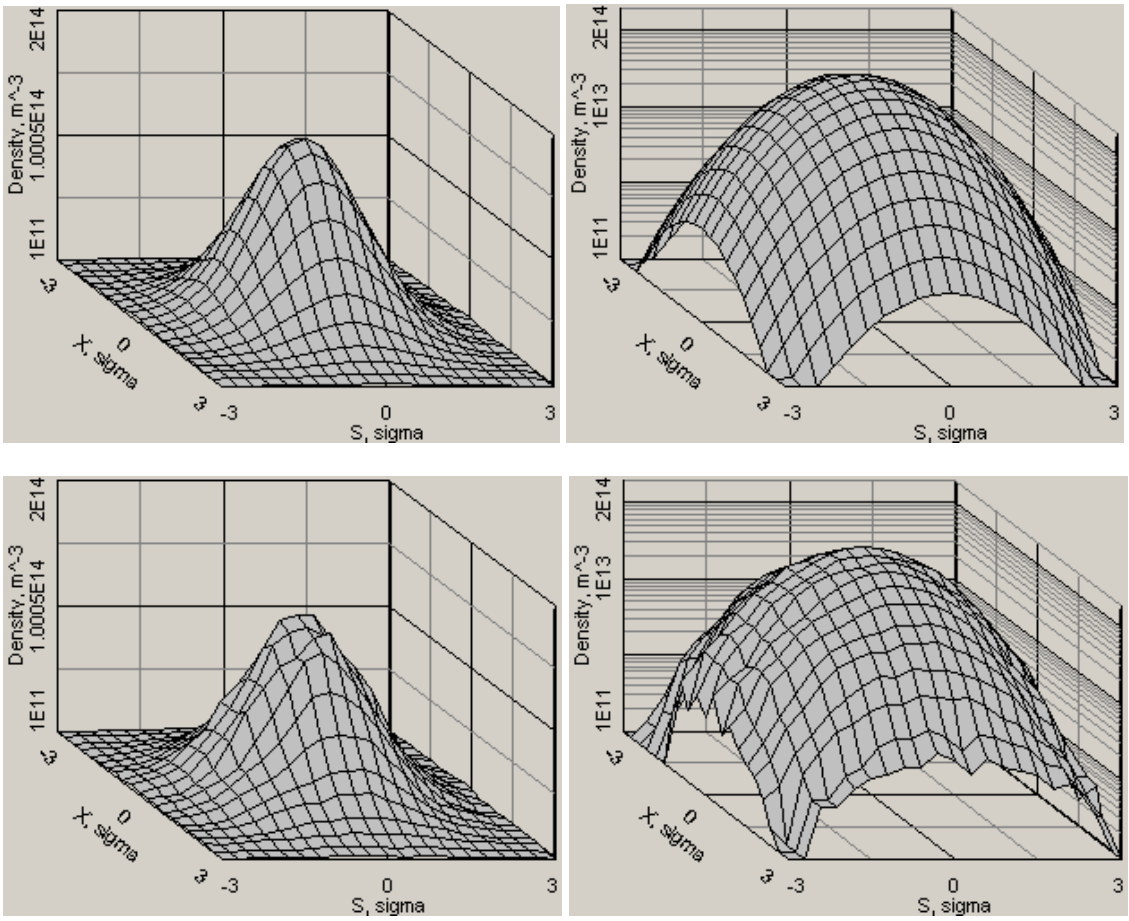


Fig. 1.3. Upper plots is analytical density distribution in linear (left plot) and logarithmic (right plot) scales. Lower plots are the same for array of 20000 particles, number of local particles is 4000, $\alpha = 0.5$ (number of particles inside cell is about 130 in accordance with estimation (1.3.3))

For generation of the electron array at different distributions the following visual form was developed in the interface (Fig.1.4).

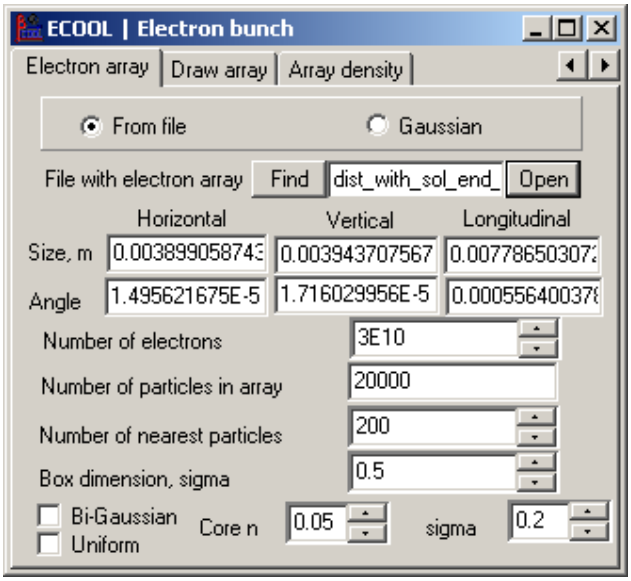
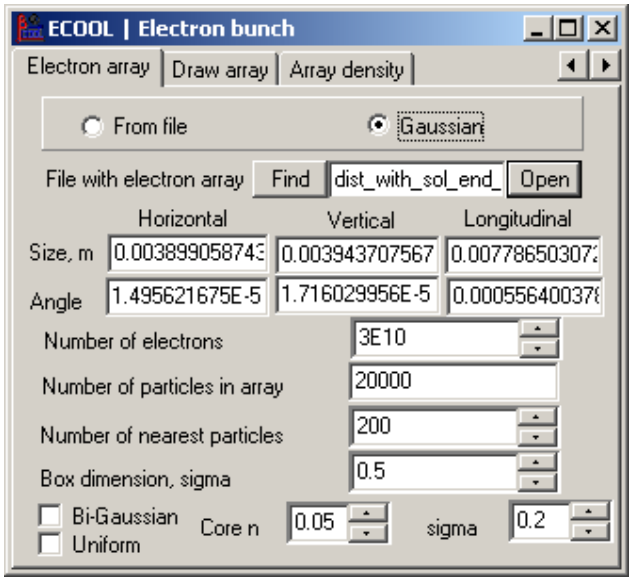


Fig. 1.4. The visual form **Electron array** developed for generation of different test electron distributions.

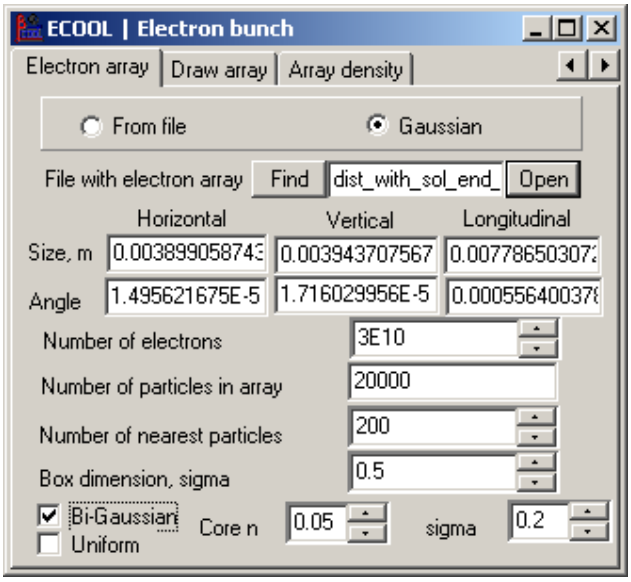
When the option "From file" is chosen the array is loaded from a file. In this case rms parameters and the "Number of particle in array" are output parameters. Input parameters of the model are "Number of nearest particles" and "Box dimension, sigma". The real particle number in array is output only after start of the program. If the number of nearest particles is large than particle number in array the local number will be reduced automatically to the array number.

If the radio button "Gaussian" is chosen, but check boxes "Bi-Gaussian" and "Uniform" are not checked:



the program will generate array of particles by itself. In this case rms parameters of the array and "Number of particles in array" are input parameters. The particle distribution over co-ordinates and velocities is Gaussian at input rms parameters.

If "Gaussian" is chosen, check boxes "Bi-Gaussian" is checked and "Uniform" is not checked:

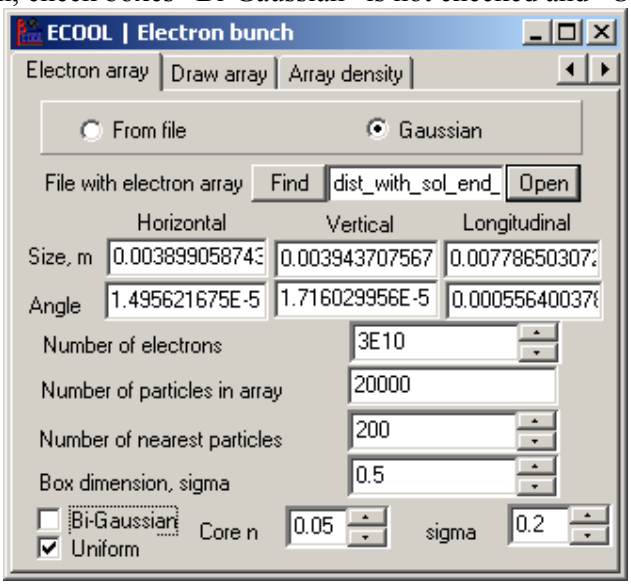


The program will generate the particle array in accordance with rms parameters and "Core n" and "sigma" parameters. The number of particles equal to

"Number of particles in array" * (1 – "Core n")

will be distributed in accordance with Gaussian low at input rms values. The other particles will be distributed in accordance with Gaussian low at rms parameters multiplied by "sigma" parameter.

If "Gaussian" is chosen, check boxes "Bi-Gaussian" is not checked and "Uniform" is checked:



the program will generate array of the particles with uniform distribution over horizontal and vertical co-ordinates inside ±2 multiplied by corresponding rms value. The longitudinal co-ordinate and all components of the particle velocity will be distributed in accordance with Gaussian low at input rms parameters.

Number of local particles determines accuracy of the friction force and diffusion calculation. The calculations of the force using velocity of the local particles is equivalent to calculation of the friction integral by Monte-Carlo method.

To investigate the accuracy of the friction force calculations were foreseen a few possibilities for the friction force calculation.

For the friction force calculation in the case of electron array the friction force model “Electron array” is used. To calculate the electron density and friction force from co-ordinates and velocities of the electrons in the visual form “Toepffer, 3D” the check box “From array” has to be checked, the check box “Analytic density” has to be not checked:

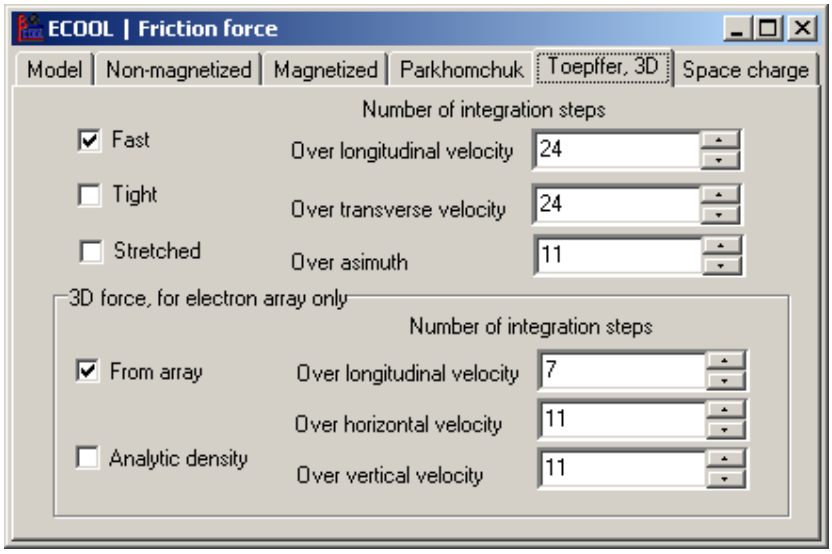
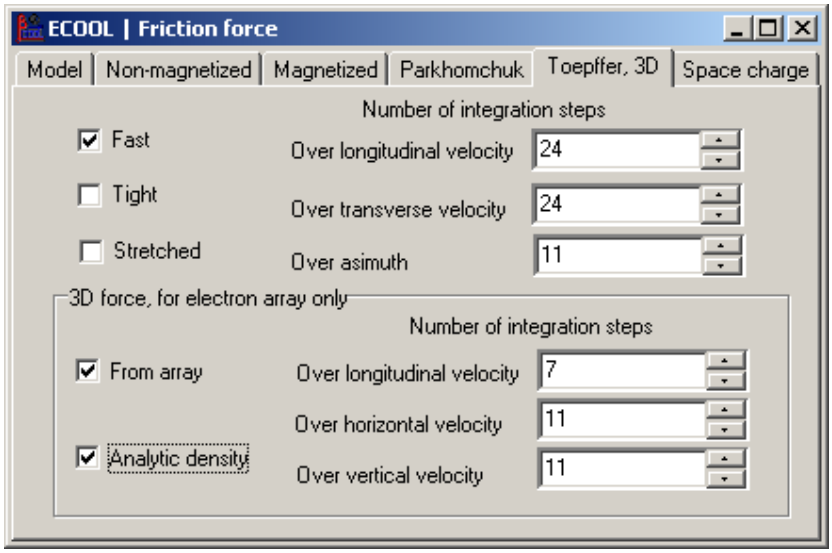
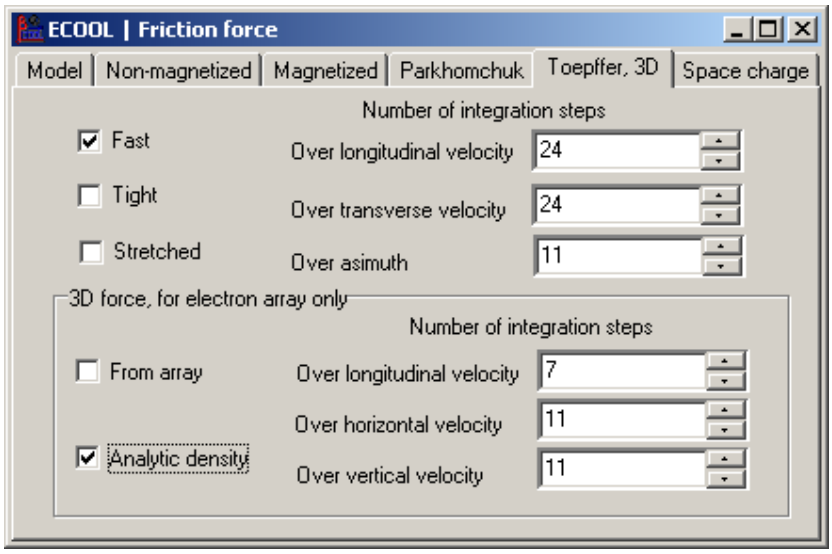


Fig. 1.5. The visual form Toepffer, 3D developed for input of the local model parameters for the friction calculation.

When the check box “Analytic density” is checked the local density of the electrons is calculated in accordance with the model of electron beam used for the array generation (Gaussian, bi-Gaussian or Uniform):



When the check box “From array” is not checked the friction force is calculated with the formulae for Maxwellian velocity distribution using rms velocity spread of the local array. The numbers of integration steps are input in the corresponding edit windows of the visual form.
At the combination of the check boxes presented below:



the program will calculate the friction force for analytic density and rms velocity spread of local electrons. For the Gaussian array this combination is equivalent to the electron beam model “Gaussian bunch”:

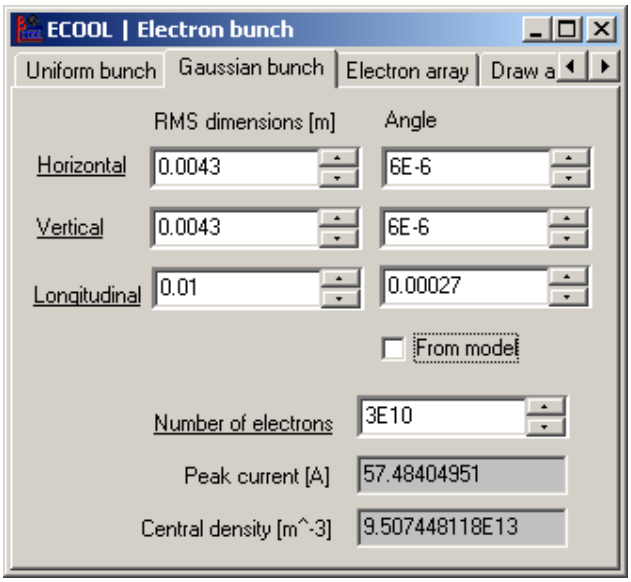


Fig. 1.6. Visual form **Gaussian bunch** used for benchmarking the friction force calculation.

When the check box “From model” in the visual form “Gaussian bunch” is not checked the velocity spread of the bunch will be calculated from the angular spread input in the corresponding edit windows. In this case the velocity spread can be different in all degrees of freedom and the friction force can be calculated using 3D force formula only. For this the force model “Electron array” has to be chosen, but the check box “From array” has to be not checked (if this box is checked the program will stop the calculations due to mistake). If the check box “From model” is not checked the electron rms velocities, temperatures and emitances will be calculated from the bunch rms parameters and output into the visual form “Model”:

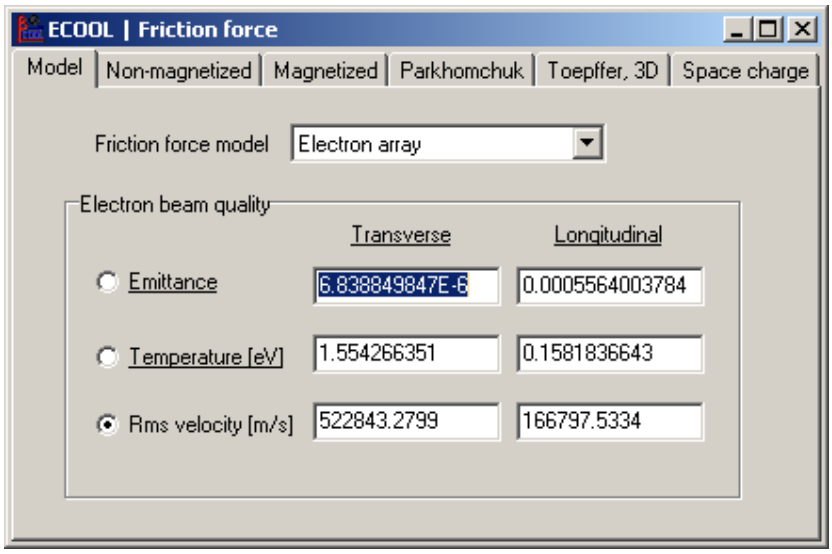
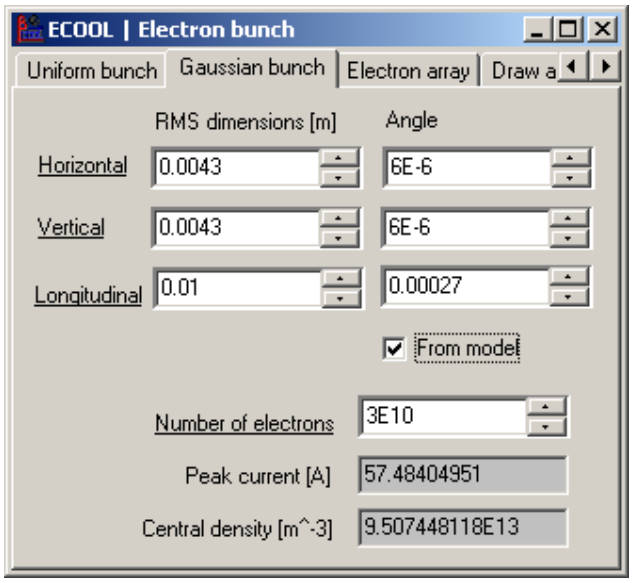


Fig. 1.7. Visual form **Model** used for input of the parameters for analytical calculation of the friction force.

When the check box "From model" in the visual form "Gaussian bunch" is checked:



the friction force will be calculated using the beam parameters from the visual form "Model" (Fig. 1.7), the rms angular spread of the bunch from the form "Gaussian bunch" will be ignored. In this case one can use for friction force calculation each model of the friction force, for instance "Non-magnetized".

If in the tab sheet "Non-magnetized" the check box "Enable" in the "Undulator" panel is checked:

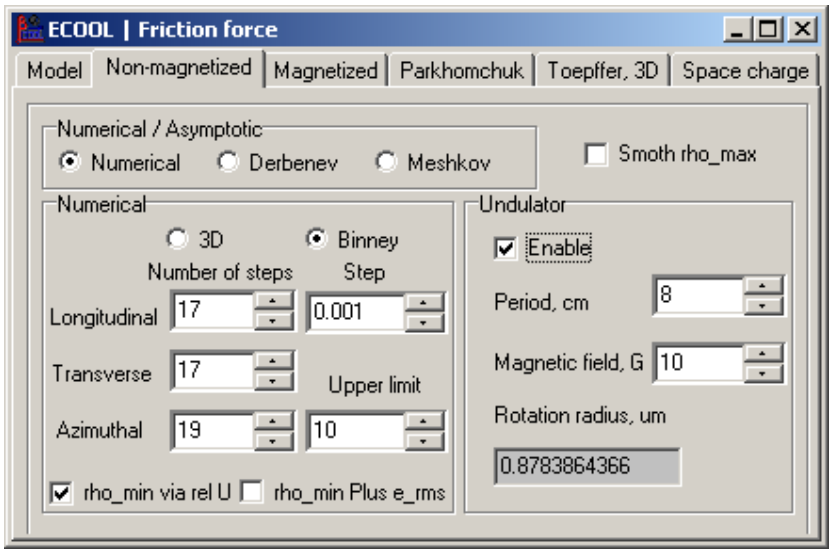


Fig. 1.8. Visual form **Non-magnetized** for input of the undulator parameters.

The undulator parameters will be used for calculation of the minimum impact parameter also in the case of friction force model "Electron array".

For visualization of the electron density in the case of electron model "Electron array" the following plot is used:

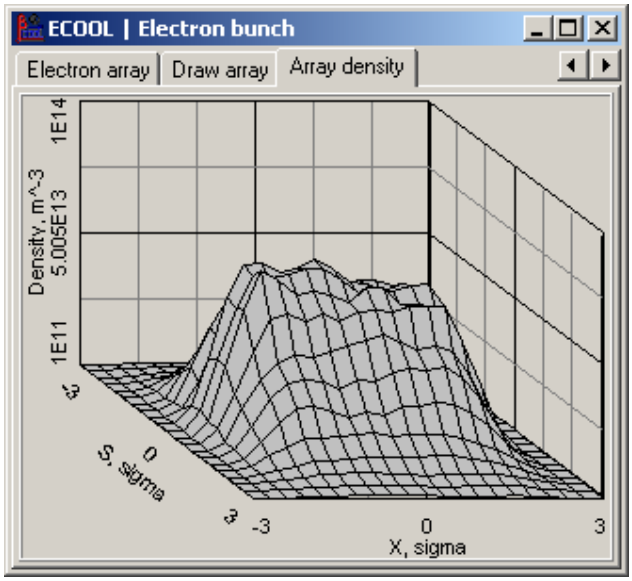


Fig. 1.9. Visual form **Array density** for output of the array density distribution.

The density is calculated as function of horizontal and longitudinal co-ordinates inside the bunch, the vertical co-ordinate is set to be zero.

Numbers of divisions along horizontal and vertical co-ordinates are input in the tab sheet "Draw array":

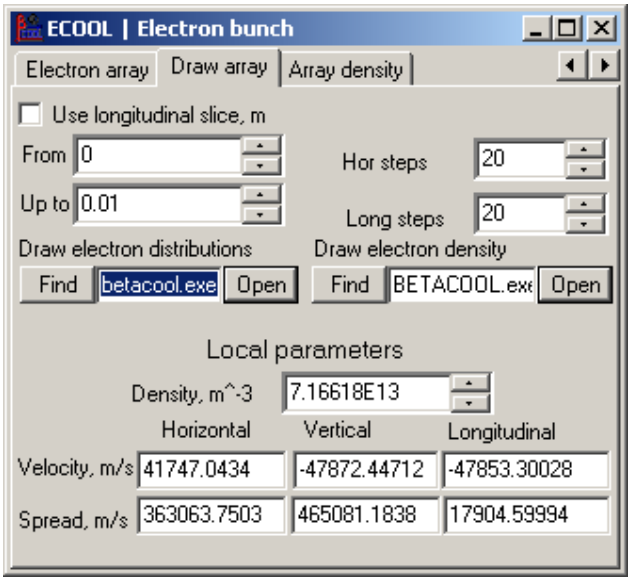


Fig. 1.10. Visual form **Draw array** for input of the parameters for plot in Fig. 1.9.

in the edit windows “Hor steps” and “Long steps”. The density is calculated using TBrowse component “Draw electron density”.

Local parameters of the electron array – density, mean velocities and the velocity spreads – are output taking into account displacement of the electron bunch. The electron bunch position is input in the tab sheet “Ebeam shifts” of the “Electron | Cooler” visual form:

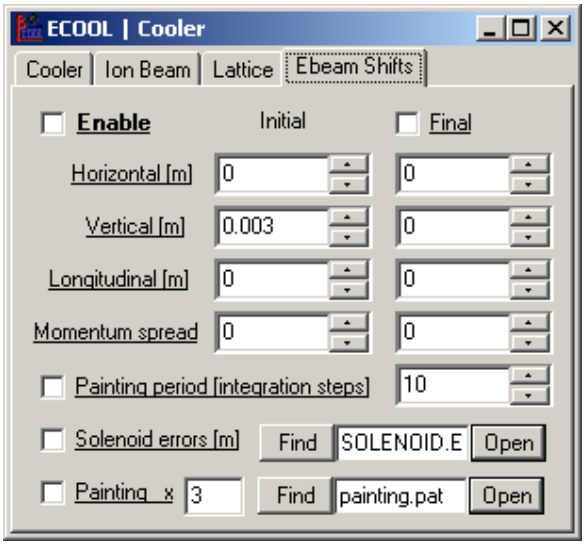


Fig. 1.11. Visual form **Ebeam Shifts** used for calculation of the array density in arbitrary position.

when the check box “Enable” is checked. The electron bunch position for the local parameters calculation and for visualization of the friction force components are input in the edit windows “Initial”.

A few examples of the friction force calculation using different models are presented in the Fig. 1.12 – 1.13.

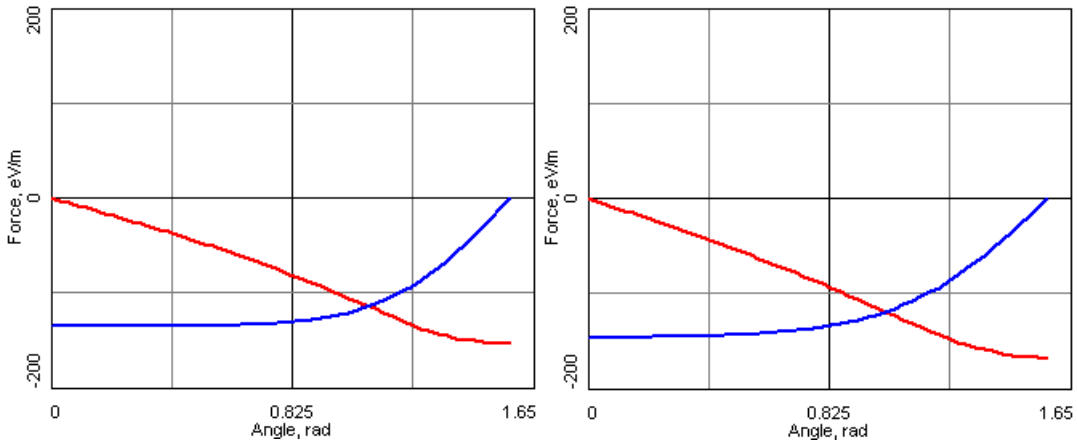


Fig. 1.12. Friction force as a function of the angle between the ion velocity and the electron beam axis calculated using analytical formulae for Gaussian electron array containing 20000 electrons. Left plot – using global rms parameters of the array, right plot – using rms parameters of 200 local electrons. The ion velocity is equal to $3 \cdot 10^5$ m/s. Red line – transverse component of the force, blue line – longitudinal one.

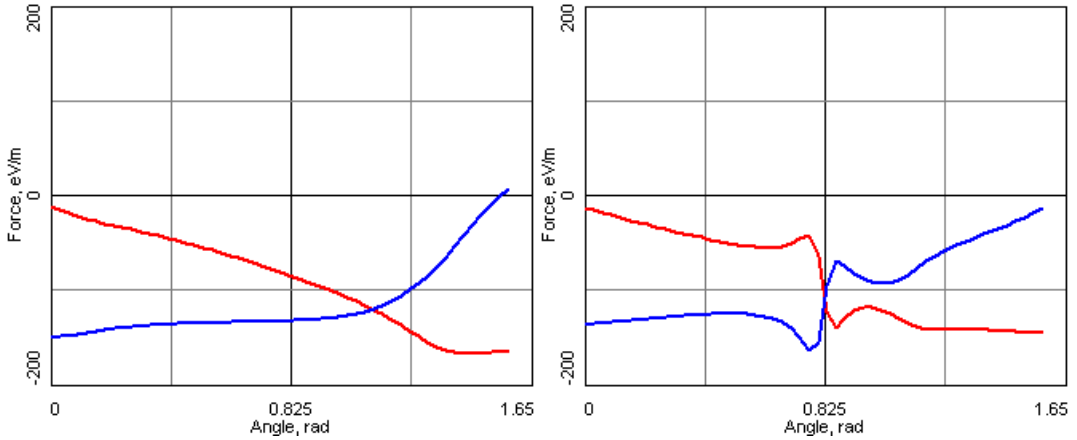


Fig. 1.13. Friction force as a function of the angle between the ion velocity and the electron beam axis calculated using velocities of 200 local electrons. Left and right plot are two random realization of the electron bunch. Number of electrons in the array is 20000.

1.6. Local simulations of IBS process

1.6.1. Reduction of the ring optic structure

For IBS simulation the friction and diffusion have to be calculated in each optic element of the ring. To keep the calculation time in reasonable range at sufficient particle number the total optic structure of the ring can be reduced to a few elements. This procedure has to be done without sufficient distortion of the optic structure from the side of its IBS properties.

For instance in the Fig. 1.14 the reduced RHIC lattice structure is presented. Its 15 elements was chosen to keep the same IBS growth rates as in the total structure. Results of the IBS simulation using Bjorken-Mtingwa model for total and reduced structures are coincide practically (Fig. 1.15.). For comparison in the Fig. 1.15 results of the simulations using Kinetic model (linear friction & constant diffusion) are presented. As one can see, for distribution closed to Gaussian one the Kinetic model provides acceptable accuracy at relatively high calculation speed. However the Kinetic model can not be applied at arbitrary ion distribution.

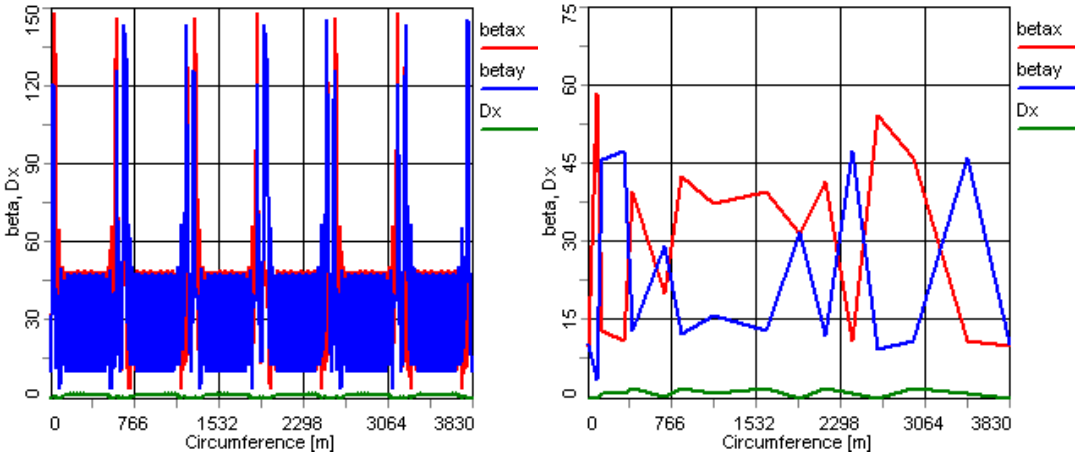


Fig. 1.14. Total RHIC optic structure containing 1900 elements (left) and the reduced structure (right) from 15 elements.

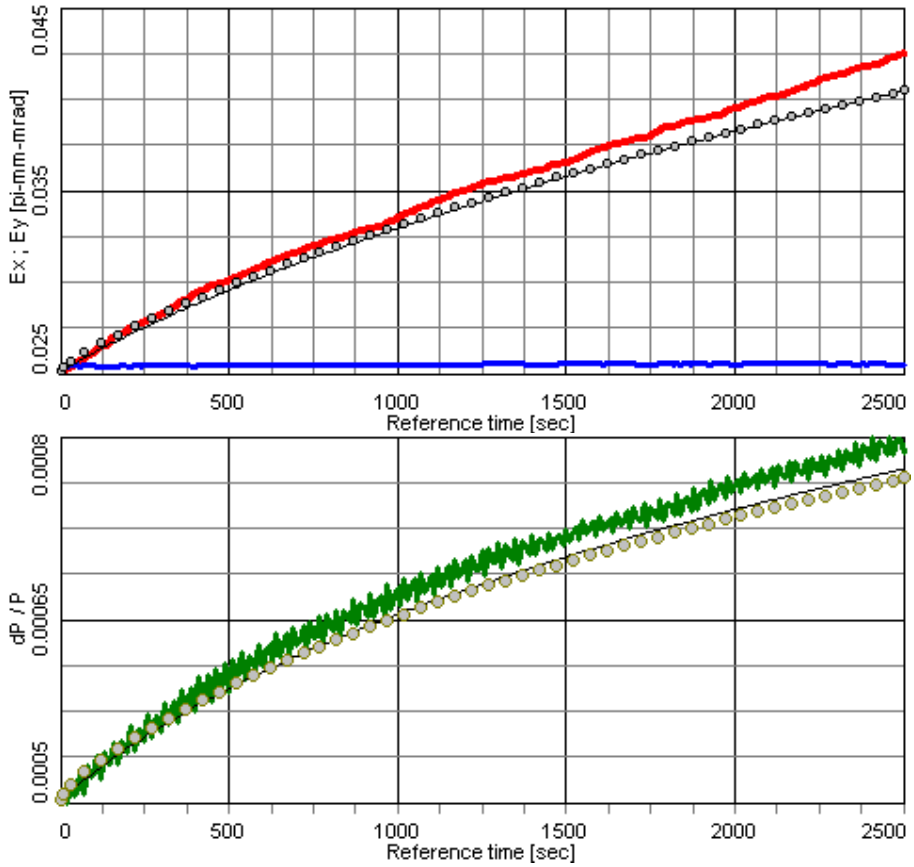


Fig. 1.15. Emittance (upper plot) and momentum spread (lower plot) evolution. Circles correspond to analytical IBS calculation for total RHIC optic structure. Black lines – reduced structure from Fig. 1.14. Colored lines – results of simulation using Kinetic model. Particle number is 4000, step over time is 10 sec. One revolution requires 2.5 seconds of calculation using Kinetic model.

1.6.2. Simulations at different structure of the diffusion tensor

RHIC is operated over the transition energy and at usual operation parameters the ion bunch has a flattened velocity distribution. In this case the main role in the IBS emittance growth plays D_{zz} component of the diffusion tensor. As one can see from the Fig. 1.15 at Gaussian distribution the emittance evolution is predicted with good enough accuracy even at constant diffusion tensor

components (independent on the particle co-ordinates and velocity). Any case the friction force and other components of the diffusion tensor (as well as their dependence on the particle velocity) can be important in prediction of the evolution of the distribution tails in the case of bi-Gaussian distribution formed at electron cooling application.

Therefore all the stages of the local IBS algorithm were checked step by step at the typical RHIC parameters (Fig. 1.16 – 1.19). As it was expected the account of the off-diagonal elements (Fig. 1.17 and 1.19) influences very slightly on the results of the simulations. The Model particle number in the simulations was chosen to have a calculation time on PC below 5 minutes per one revolution in the ring. In this case accuracy of the emittance and momentum spread calculation is about 2 – 3 %, which is clearly seen, for instance, in the difference of the initial emittance in the Fig. 1.19.

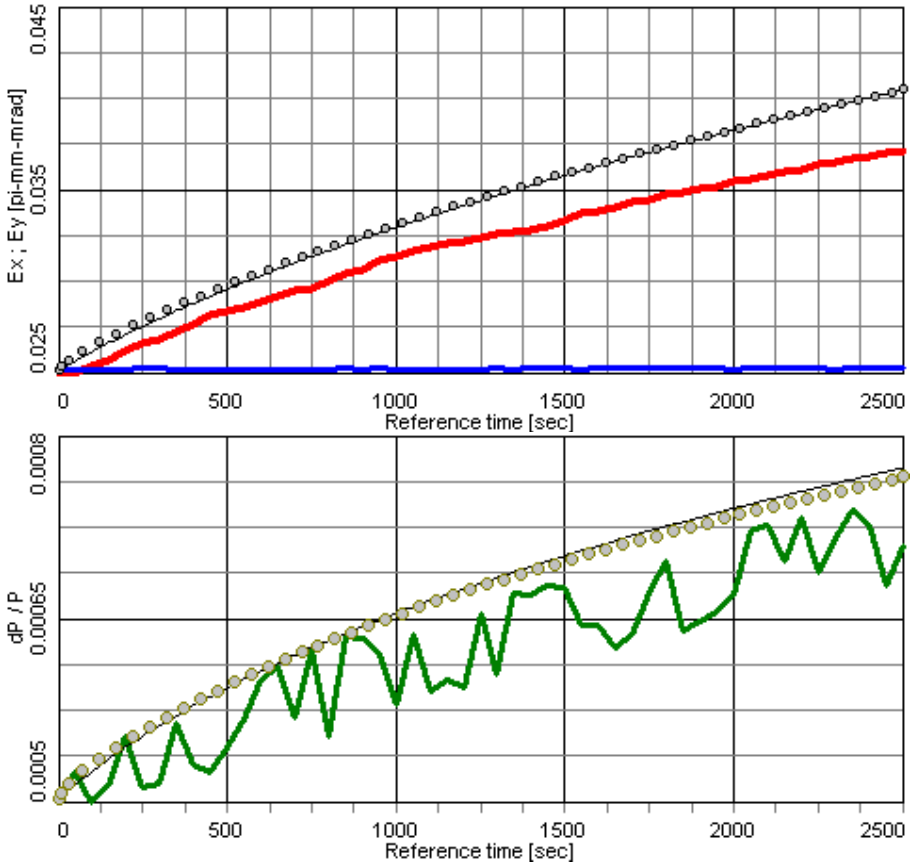


Fig. 1.16. Beam emittance (upper plot) and momentum spread (lower plot) evolution in time. IBS simulations are performed taking into account only diagonal elements of the diffusion tensor. Particle number is 2000. Integration step 50 seconds, 50 local particles. One revolution recuires 1 min 34 sec.

All the results underestimate slightly the growth rates in comparison with the analytical prediction. For the illustration of this fact the horizontal growth rate time-dependence is shown in the Fig. 1.18. The growth rates calculated from results of the local simulations predicts very well the tendency of the growth rate evolution; however the relative systematic error is almost independent on the rate value.

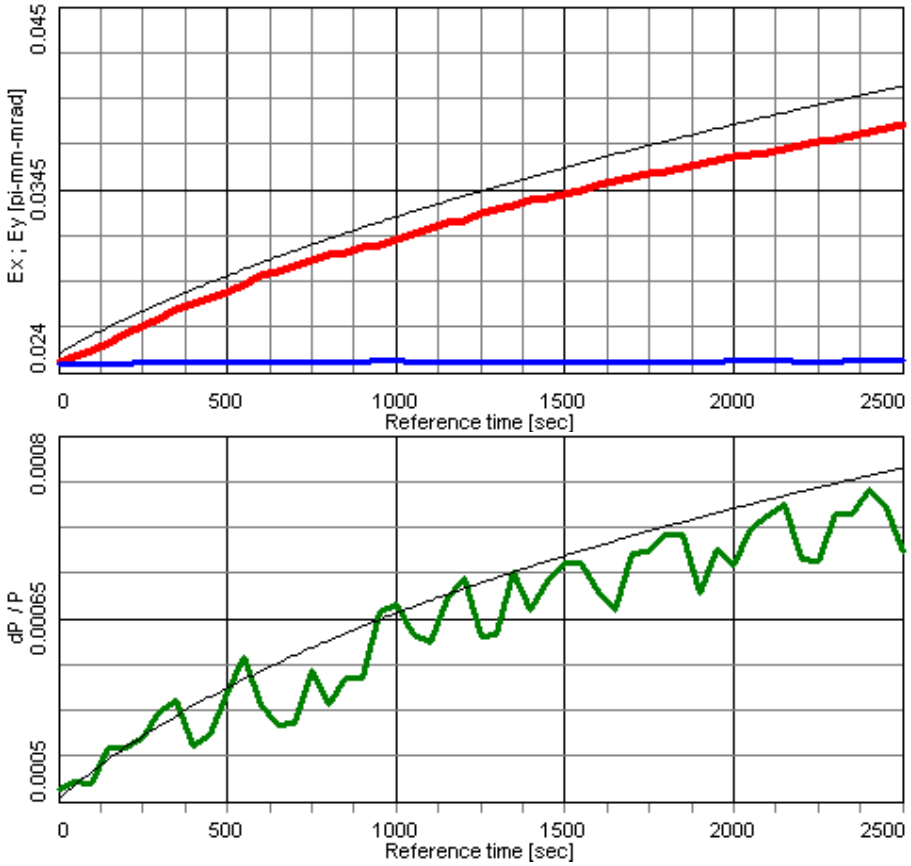


Fig. 1.17. Beam emittance (upper plot) and momentum spread (lower plot) evolution in time. IBS simulations are performed taking into account two off-diagonal elements of the diffusion tensor. Particle number is 2000, local number 100, step over time 50 sec, one revolution is about 3 min.

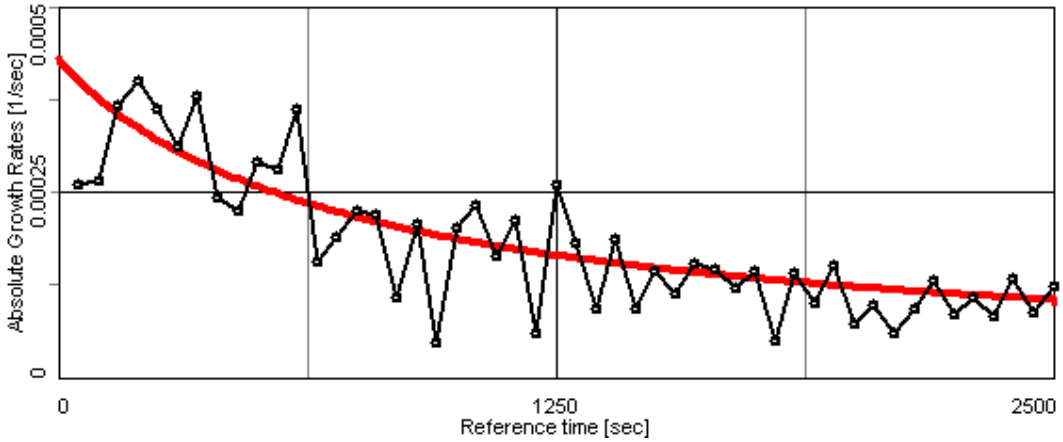


Fig.1.18. Horizontal heating rate as function of time. Red curve is result of rms dynamics simulation. Black line – Model Beam simulations at the parameters of Fig.1.17.

This systematic error originates from the algorithm of the local density calculation (see Chapter 1.3) and its value is closed to estimation (1.3.6). Correspondingly, to improve the accuracy of the calculations one needs to increase the number of the local particles and to decrease of the box dimension α. The optimum parameters have to be chosen as a compromise between the accuracy and calculation speed.

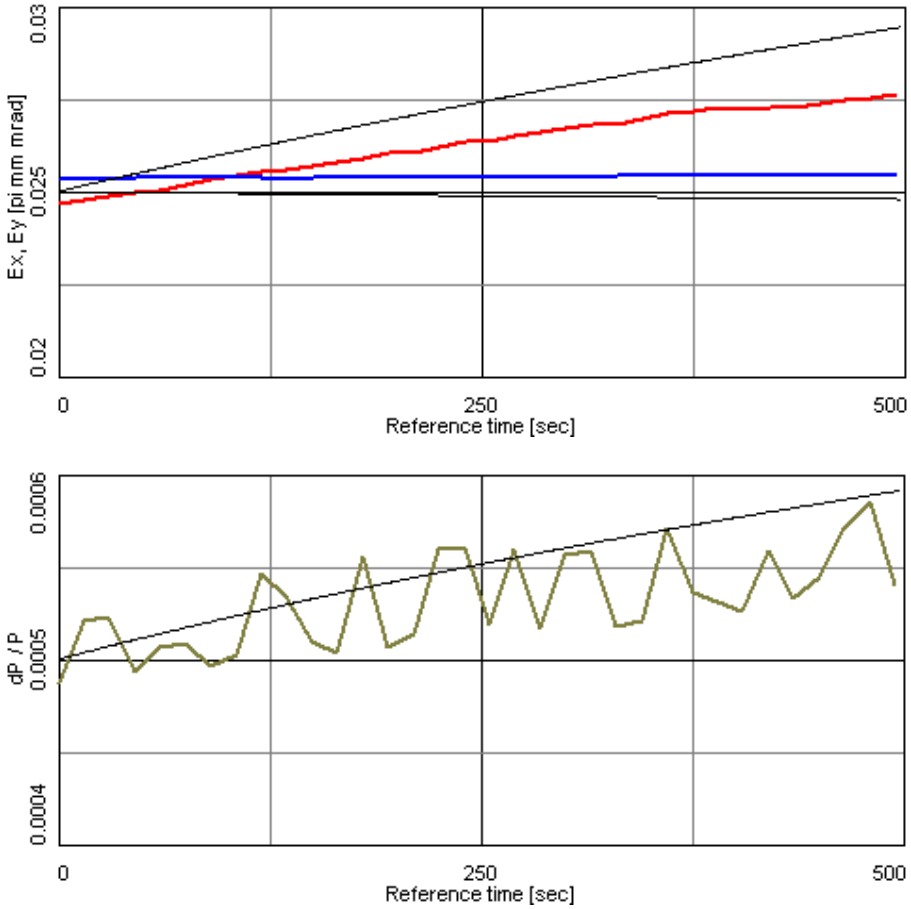


Fig. 1.19. Beam emittance (upper plot) and momentum spread (lower plot) evolution in time. IBS simulations are performed with all non-zero elements of the diffusion tensor. 2000 model particles, 300 local particles, 15 sec integration step, $\alpha = 0.3$. One revolution ~ 5 minutes.

2. Development of stochastic cooling simulation

First experiments at RHIC with longitudinal stochastic cooling application demonstrated its ability to compensate increase of the bunch length due to intrabeam scattering and exclude particle losses from the bucket. At these conditions one can expect increase of the emittance growth due to intrabeam scattering. To provide realistic simulation of the ion distribution function evolution due to common action of the longitudinal stochastic cooling and intrabeam scattering the development of the stochastic cooling simulation has been started.

Main peculiarities of the RHIC longitudinal stochastic cooling are sufficient non-linearity of a friction force and dependence of the diffusion on synchrotron amplitudes beginning from their zero value. The algorithm existing in Betacool presumes linear friction force and diffusion independent on the synchrotron amplitude. The development of the algorithm is realizing in three stages:

- at the first stage the model presumed cubic nonlinearity of the friction and constant diffusion was realized. Numerical parameters of the model are input into the program and used for fitting of experimental results. This model permits to estimate an influence of the longitudinal stochastic cooling on the transverse intrabeam growth rates and particle loss and simulate luminosity time dependence in the presence of the cooling. However it can not predict properties of the stochastic cooling at different parameters of the ion beam.
- the second stage presumes realization of more realistic model for the friction force calculation.
- and finally the diffusion will be calculated as a function of the particle synchrotron amplitude and the distribution shape.

In the frame of Model Beam algorithm the program solves Langevin equation for model particle from the particle array. The particle momentum during simulations is changed regularly by action of a friction force and randomly by diffusion. In the three dimensional case each component of the particle momentum is changed in accordance with the step of integration over time of Δt as:

$$P_i(t+\Delta t) = P_i(t) + F_i \Delta t + \sqrt{\Delta t} \sum_{j=1}^{3} C_{i,j} \xi_j \ ,$$

where F_i are the components of the friction vector, $C_{i,j}$ are connected with the diffusion tensor components in accordance with the equation:

$$\sum_{k=1}^{3} C_{i,k} C_{j,k} = D_{i,j} \ ,$$

and ξ_j are independent random numbers distributed in accordance with Gaussian low at unit variance.

In the case of longitudinal stochastic cooling we have one dimensional task and the equation simplifies. Denoting relative momentum deviation as $\delta = \frac{\Delta p}{p}$ one can write

$$\delta(t+\Delta t) = \delta(t) + F\Delta t + \sqrt{\Delta t} C \xi \ . \qquad (2.1)$$

If the diffusion due to thermal noise is negligible the coefficient C is determined by the Shottky noise diffusion.

Algorithm for realistic calculation of the friction force was proposed by M. Blaskevich. The shape of the friction force as a function of the revolution frequency offset can be calculated without account of the signal shielding in accordance with:

$$F = g \cdot \mathrm{Im}\left\{ \sum_{k=25}^{15+25} G\left[h \cdot k \cdot \Delta\omega\right] \right\}, \tag{2.2}$$

where the frequency offset is given by

$$\Delta\omega = -\omega_0 \eta \frac{\Delta p}{p}$$

with $\omega_0 = 2\pi/T_0$, T_0 is the revolution period and η is the momentum slip factor of the ring, h is the harmonic number.

In the general case the sum in the (2.2) has to be done over all the harmonic number satisfying the condition

$$\frac{f_{\min}}{f_{rev}} \le h \le \frac{f_{\max}}{f_{rev}}$$

Here $f_{\min}$ = 5 GHz and $f_{\max}$ = 8 GHz are the minimum and maximum frequencies of the cooling system band, f_{rev} is the revolution frequency. However the total bandwidth of the RHIC cooling system is divided by 15 bands corresponding to the cavity number. Therefore the sum can be calculated for central frequency of each band only. Central frequency of the first cavity corresponds to the harmonic number equal to 2560.

The form factor G is given by

$$G\left(h \cdot \Delta\omega\right) = \exp\left(ih \cdot \Delta\omega T_d\right)\left[1 - \exp\left(ih \cdot \Delta\omega T_0\right)\right]^2, \tag{2.3}$$

where the time delay $T_d = \frac{2}{3} T_0$ at RHIC cooling system.

Here $f_{\min}$ = 5 GHz and $f_{\max}$ = 8 GHz are the minimum and maximum frequencies of the cooling system band, f_{rev} is the revolution frequency. However the total bandwidth of the RHIC cooling system is divided by 15 bands corresponding to the cavity number. Therefore the sum can be calculated for central frequency of each band only. Central frequency of the first cavity corresponds to the harmonic number equal to 2560.At RHIC cooling system the time delay $T_d = \frac{2}{3} T_0$. $T_0 \approx 12.8\ \mu sec$, $T_d = 2/3\ T_0$, and n leads in the range $\frac{5GHz}{78kHz} \le n \le \frac{8GHz}{78kHZ}$, which can be written as $25 \le \frac{n}{2560} \le 40$. The coherent cooling force without signal shielding is

$$F = gF^*, \quad F^* = \mathrm{Im}\left\{ \sum_{k=25}^{40} G\left(2560 \cdot k\Delta\omega\right) \right\}. \tag{2.4}$$

To calculate the force (2.4), one can to simplify the expression (2.3):

$$\begin{aligned} &G(n\Delta\omega) = \exp\{in\Lambda\omega T_d\} + \exp\{in\Delta\omega(T_d + 2T_0)\} - 2\exp\{in\Delta\omega(T_d + T_0)\}, \\ &\mathrm{Im}\,G(n\Delta\omega) = \sin(n\Delta\omega T_d) + \sin(n\Delta\omega[T_d + 2T_0]) - 2\sin(n\Delta\omega[T_d + T_0]) = \\ &= 2\sin(n\Delta\omega[T_d + T_0])\{\cos(n\Delta\omega T_0) - 1\}. \end{aligned} \tag{2.5}$$

Using (2.5) and $T_d = 2/3\ T_0$, from (2.4) we have:

$$F^* = \sum_{k=25}^{40} 2\sin\left(\frac{5}{3}\cdot 2560\cdot k\cdot\Delta\omega\cdot T_0\right)\{\cos(2560\cdot k\cdot\Delta\omega\cdot T_0) - 1\}. \tag{2.6}$$

The results of the calculation are shown on the Fig. 2.1, with $\Delta p/p$ on x-axis and F on y-axis.

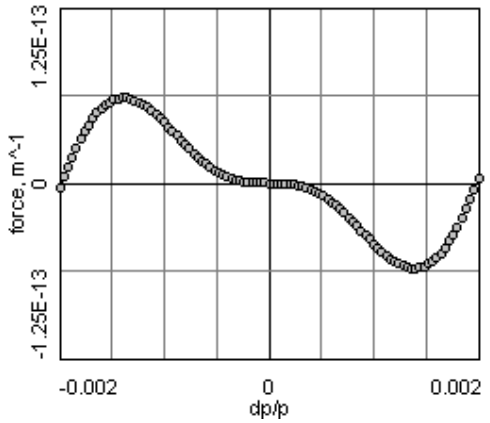


Fig. 2.1. The coherent friction force as function of the ion momentum deviation. Two filter system.

The formula (2.3) corresponds to the system included two filters in series. For the comparison in the Fig. 2.2. the friction force is plotted for the single filter system. In this case the form factor is given by

$$G(h\cdot\Delta\omega) = \exp(ih\cdot\Delta\omega T_d)[1 - \exp(ih\cdot\Delta\omega T_0)]. \tag{2.7}$$

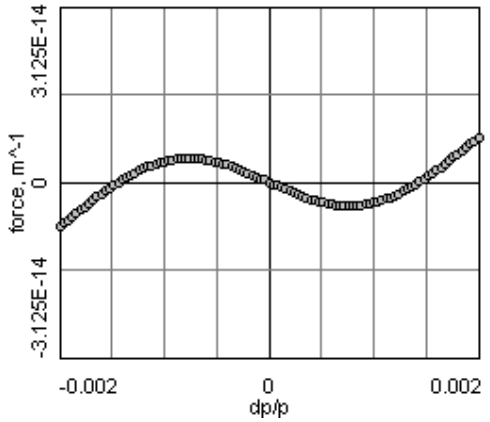


Fig. 2.2. The coherent friction force as function of the ion momentum deviation. Single filter system.

Diffusion in the case of two filter system calculated in the center of the bunch is shown in the Fig. 2.3.

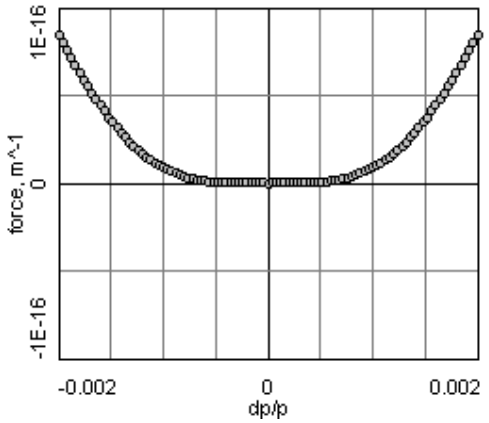
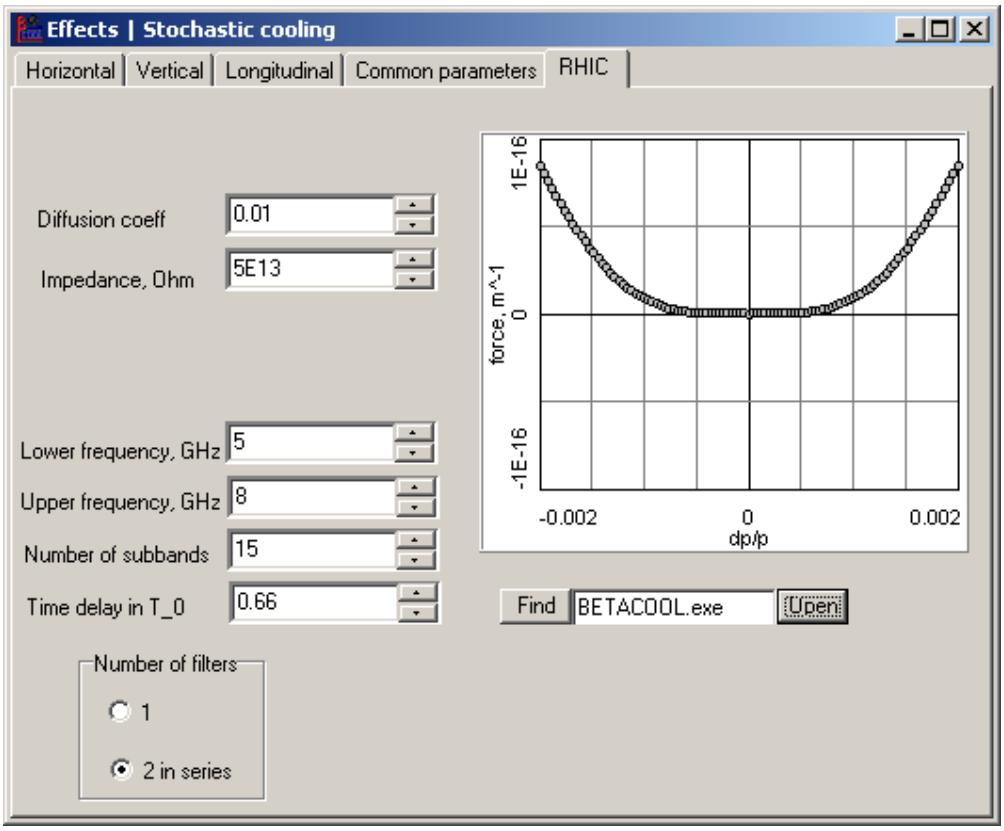


Fig. 2.3. The diffusion in the bunch center as function of the ion momentum deviation. Two filter system.

To input the parameters of the longitudinal stochastic cooling system the following visual form was developed:



3. Beam-beam effect and luminosity calculation

3.1. Beam-beam diffusion

When the beam-beam parameter ξ_i exceeds some threshold value the beam-beam instability is excited that leads to diffusional increase of the ion beam emittance. Theory explanation of the beam-beam effect given in [T. Katayama et al. "MUSES Conceptual design report" May 1999, unpublished] relates this diffusion with a non-linearity of the opposite bunch electric field and presence of a noise in the storage ring. The diffusion coefficient D can be expressed from the noise relative amplitude u as follows:

$$D = \pi^2 (\xi_i u)^2 \left(\frac{\sigma_\perp}{2\sigma_{1\perp}} \right)^2 . \tag{3.1.1}$$

The noise amplitude can be estimated by fitting of experimental results and it lies in the range 0.05 – 0.5.

The beam emittance after n revolutions in the ring is calculated as:

$$\frac{\varepsilon_n}{\varepsilon_0} = \sqrt{1 + Dn} , \tag{3.1.2}$$

where ε_0 is the initial value of the emittance. The heating rate

$$\frac{1}{\tau} = \frac{1}{\varepsilon_0} \frac{\Delta\varepsilon}{\Delta t} \tag{3.1.3}$$

can be calculated assuming that $Dn << 1$ ($n = \Delta t / T_{rev}$, where T_{rev} is the revolution period):

$$\varepsilon_n = \varepsilon_0 \sqrt{1 + Dn} \approx \varepsilon_0 + \frac{\varepsilon_0 D \Delta t}{2T_{rev}} . \tag{3.1.4}$$

That leads to

$$\frac{1}{\tau} = \frac{D}{2T_{rev}} . \tag{3.1.5}$$

In the case of different beam cross-sections in horizontal and vertical planes the calculations are similar in both planes. For instance the diffusion power in the horizontal plane is calculated as:

$$D_x = \pi^2 (\xi_x u)^2 \left(\frac{\sigma_x}{2\sigma_{1x}} \right)^2 ,$$

ξ_x is the horizontal beam-beam parameter, σ_x is the horizontal rms bunch size, σ_{1x} is the rms size of the opposite bunch.

The ion momentum variation in the frame of Model Beam algorithm is calculated in accordance with:

$$\Delta x' = \sqrt{\frac{2D\varepsilon_x \Delta t}{\beta_x T_{rev}}}\xi\,, \tag{3.1.6}$$

where ξ is the Gaussian random number at unit dispersion, β_x is the horizontal beta function in the collision point.

This algorithm was implemented into the code. The required modifications in the interface was done as it shown in the Fig. 3.1.

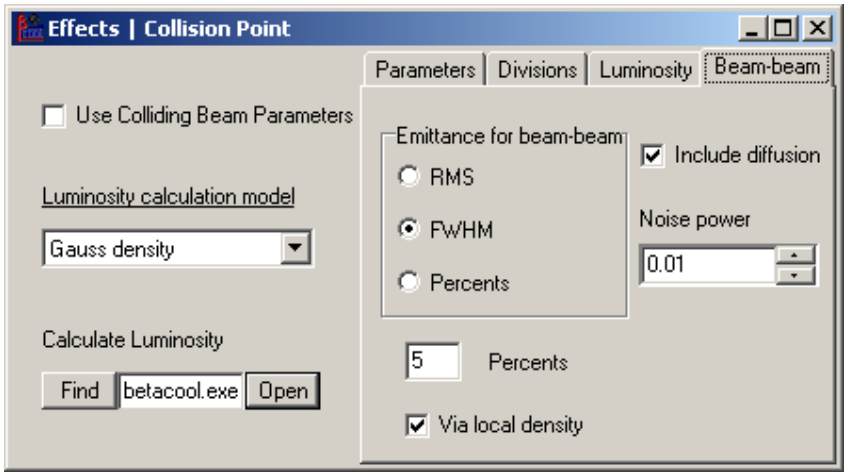


Fig. 3.1. Visual form **Collision point**, tab sheet **Beam-beam** for input the noise power.

3.2. Luminosity calculation for individual ion

For correct calculation of the particle loss in the interaction point the algorithm for luminosity calculation for individual ion is necessary. In the current version of the program there are a few algorithms for luminosity calculation based on evaluation of local areal density. For benchmarking of these algorithms and for fast luminosity calculation in the case of Gaussian or bi-Gaussian distribution a new algorithm was developed. Here the luminosity for individual ion interacting with a Gaussian bunch is described. In the case of bi-Gaussian distribution the luminosity is calculated for a superposition of two Gaussian bunches.

In the frame shown in the Fig. 4.1 the bunch is moving along negative direction of *z*-axis. The bunch velocity is equal *u*, the ion velocity - *v*.

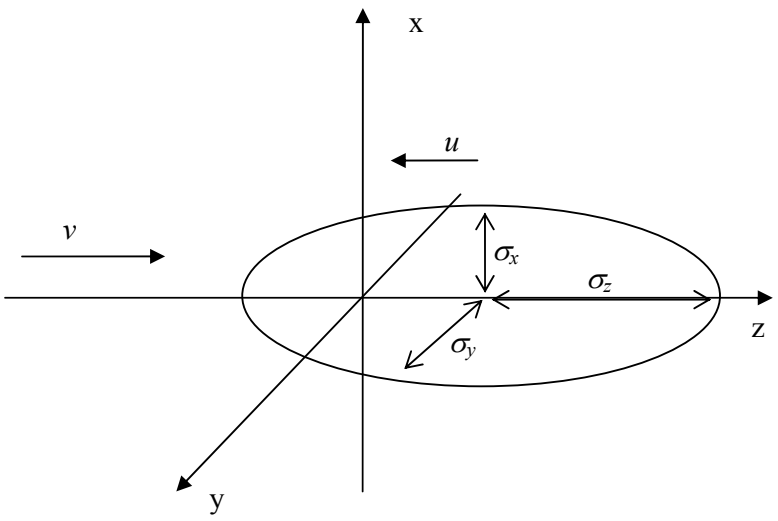


Fig. 3.2. Geometry of the collision. Collision point is located in the (0,0,0) point.

Particle distribution inside the bunch is given by:

$$\rho[x(t),y(t),z(t)]=\frac{N}{\sqrt{2\pi}^3\sigma_x(t)\sigma_y(t)\sigma_z(t)}\exp\left[-\frac{x^2(t)}{2\sigma_x(t)}-\frac{y^2(t)}{2\sigma_y(t)}-\frac{z^2(t)}{2\sigma_z(t)}\right]. \quad (3.2.1)$$

The luminosity ia determined as

$$L=\int_S \rho(\vec{r})ds, \quad (3.2.2)$$

where S is the ion trajectory. Rms dimensions of the bunch are changed in the vicinity of the collision point as:

$$\sigma_x^2(t)=\sigma_x^2(0)\left\{1+\frac{u^2t^2}{\beta_x^{*2}}\right\},\quad \sigma_y^2(t)=\sigma_y^2(0)\left\{1+\frac{u^2t^2}{\beta_y^{*2}}\right\},\quad \sigma_z(t)=\sigma_z(0), \quad (3.2.3)$$

where β_x^*, β_y^* are the beta-functions in the collision point. The integral (3.2.2) can be rewritten in the following form

$$L=\frac{(v+u)N}{\sqrt{2\pi}^3\sigma_x(0)\sigma_y(0)\sigma_z(0)}\times$$
$$\times\int_{-\infty}^{\infty}\frac{\exp\left\{-\frac{\left(x_0+x_0'z_0+x_0'vt\right)^2}{2\sigma_x^2(0)\left[1+u^2t^2/\beta_x^{*2}\right]}-\frac{\left(y_0+y_0'z_0+y_0'vt\right)^2}{2\sigma_y^2(0)\left[1+u^2t^2/\beta_y^{*2}\right]}-\frac{\left(z_0+[v+u]t\right)^2}{2\sigma_z^2(0)}\right\}}{\sqrt{\left(1+u^2t^2/\beta_x^{*2}\right)\left(1+u^2t^2/\beta_y^{*2}\right)}}dt. \quad (3.2.4)$$

Or at substitution $\xi = \frac{z_0 + (u+v)t}{\sqrt{2\sigma_z^2(0)}}$ it can be expressed as

$$L = C\int_{-\infty}^{\infty} \frac{\exp\left\{-\frac{A_x^2(\xi)}{B_x(\xi)} - \frac{A_y^2(\xi)}{B_y(\xi)}\right\}}{\sqrt{B_x(\xi)B_y(\xi)}} \exp(-\xi^2)d\xi, \tag{3.2.5}$$

where

$$\begin{aligned}
C &= \frac{N}{2\sqrt{\pi}^3 \sigma_x(0)\sigma_y(0)}, \\
A_x(\xi) &= x_0 + x_0' z_0 + x_0' \frac{v}{u+v}\left(\sqrt{2\sigma_z^2(0)}\xi - z_0\right), \\
A_y(\xi) &= y_0 + y_0' z_0 + y_0' \frac{v}{u+v}\left(\sqrt{2\sigma_z^2(0)}\xi - z_0\right), \\
B_x(\xi) &= 1 + \frac{u^2}{(u+v)^2 \beta_x^{*2}}\left(\sqrt{2\sigma_z^2(0)}\xi - z_0\right)^2, \\
B_y(\xi) &= 1 + \frac{u^2}{(u+v)^2 \beta_y^{*2}}\left(\sqrt{2\sigma_z^2(0)}\xi - z_0\right)^2.
\end{aligned} \tag{3.2.6}$$

The integral (3.2.5) is calculated using Gauss-Cristoffel method in accordance with:

$$L = \sum_{k=1}^{N} c_k F(x_k), \quad F(\xi) = C\frac{\exp\left\{-\frac{A_x^2(\xi)}{B_x(\xi)} - \frac{A_y^2(\xi)}{B_y(\xi)}\right\}}{\sqrt{B_x(\xi)B_y(\xi)}}, \tag{3.2.7}$$

where x_k are the roots of Hermit polinom of N-th order, c_k – coefficients, calculated in accordance with the formulae:

$$\begin{aligned}
c_i &= \int_{-\infty}^{\infty} h_i(x)\exp(-x^2)dx, \\
h_i(x) &= \frac{\prod_{j=1..N, j\neq i}(x - x_j)}{\prod_{j=1..N, j\neq i}(x_i - x_j)}.
\end{aligned} \tag{3.2.8}$$

At $N = 8$ relative mistake in the integral evaluation is less than 10^{-5} until 6σ inclination of the ion co-ordinates from expectation.