

SIMULATION OF ELECTRON COOLING PROCESS IN STORAGE RINGS USING BETACOOOL PROGRAM

I.N.Meshkov, A.O.Sidorin, A.V.Smirnov, E.M.Syresin, G.V.Trubnikov
Joint Institute for Nuclear Research, Dubna, Russia. <http://lepta.jinr.ru/>
P.R.Zenkevich,
Institute for Theoretical and Experimental Physics, Moscow, Russia

ABSTRACT

An electron cooling method is widely used for ion beam parameter control in the storage rings. Presently there are about 20 storage rings in operation and under construction, which are equipped with electron cooling devices. The BETACOOOL program elaborated for simulation of electron cooling process is actively used in several research centres: CERN, Juelich-FZ, DESY, RIKEN, Fermilab. This report discusses new development of the BETACOOOL program aiming to include into computation algorithm the following processes: intrabeam scattering, ion scattering on residual gas atoms, interaction of the ion beam with internal target and some others. The BETACOOOL is programmed with object oriented method using C++ Builder4 and is based on BOLIDE package (Beam Optic Library & Interface Development Environment), which is dedicated to fast elaboration of the physics and mathematics applications.

1. PHYSICAL MODEL

The physical model which is investigated using BETACOOOL program [1] is based on the following general assumptions:

- 1) the ion beam has Gaussian distribution on all degrees of freedom, which does not change during the process.
- 2) algorithm for analysis of the problem consists in a solution of the equations for root mean square values of the beam phase space volumes of three degrees of freedom.
- 3) maximum of all distribution function coincide with equilibrium orbit.

The evolution of the ion beam parameters during its motion inside the storage ring is described by the following system of four differential equations:

$$\begin{cases} \dot{N} = N \sum_j \frac{1}{\tau_{life,j}}, \\ \dot{\varepsilon}_i = \varepsilon_i \sum_j \frac{1}{\tau_{i,j}}, \end{cases} \quad (1)$$

where N is particle number, ε_i is mean value of the beam phase space volume (index i corresponds to degree of freedom). For transverse degrees of freedom ε_i corresponds to the beam emittance, for longitudinal degree of freedom it is given by the following expression:

$$\varepsilon_{lon} = \begin{cases} \left(\frac{\Delta p}{p}\right)^2, & \text{coasting;} \\ \left(\frac{\Delta p}{p}\right)^2 + \frac{1}{\Omega_s^2} \left[\frac{d}{dt} \left(\frac{\Delta p}{p}\right)\right]^2, & \text{bunched.} \end{cases} \quad (2)$$

In Eq. (2) upper line corresponds to coasting beam, lower line to bunched beam with constant parameters (for changing synchrotron frequency it is necessary to use adiabatic invariant instead of energy and presently depression of the synchrotron tune due to action of the beam space charge is not taken into account during dynamics simulation). Characteristic times are functions of all three emittances and particle number, life time in the first equation is positive for particle injection and negative for particle losses. Index j in Eq.(1) is the number of process involved into calculations.

During numerical solution of the system (1) the parameters characterising beam stability are calculated also. There are incoherent betatron tune shift value, depression of the synchrotron tune, dimensionless parameters describing the beam from the side of longitudinal and transverse coherent instabilities [2]. Presented version of the program provides also luminosity calculation in the bunch-bunch collisions for head on and crossing angle modes by numerical evaluation of the overlapping integral for bunches with the Gaussian space charge density distributions. In nearest future we plan to include into the algorithm luminosity calculations in an experiment with internal target.

BETACOOOL structure is designed as Object Oriented Program that permits to include into calculation each process, which can be described by cooling or heating rates and characteristic times of the particle number variation. In the frames of the described physical model we can provide a common simulation the following processes:

General heating effects:

- interaction of the beam ions with the target atoms,
- intrabeam scattering of the ion beam,
- ion multi scattering on residual gas atoms,
- high order tune resonances of the ion betatron motion due to space charge tune shift.

The sources of the particle losses:

- nuclear scattering on the target and residual gas atoms,
- single scattering by large angles on the target and residual gas atoms,
- electron capture (recombination) in the electron cooling section.

Cooling processes:

- stochastic cooling (both – transverse and longitudinal),
- electron cooling – with single pass and circulating electron beams.

2. ELECTRON COOLING CALCULATION

Electron cooling rates are calculated through averaging of the action of friction force inside cooler over phases of betatron and synchrotron oscillations and over Gaussian distribution function of the particles in the space of invariants of the motion.

The cooling rates calculation was performed under the following general assumption: the ion coordinates are kept constant during ion interaction with cooling electron beam.

The motion of each particle is described by two Courant-Snyder invariants:

$$I_i = \frac{1 + \alpha_i^2}{\beta_i} i_\beta^2 + 2\alpha_i i_\beta i'_\beta + \beta_i i'^2_\beta, \quad i = x, z, \quad (3)$$

where x, z are a horizontal and vertical co-ordinates, α_i and β_i are alpha and beta functions in the cooling section. The invariant corresponded to synchrotron motion can be introduced in accordance with (2) as follows:

$$I_s = m\sigma_p^2, \quad m = \begin{cases} 1 - \text{coasting beam,} \\ 2 - \text{bunched beam.} \end{cases}$$

where σ_p is the r.m.s. momentum spread. The ion coordinates and momentum inside the cooling section can be calculated in accordance with

$$x_\beta = \sqrt{I_x \beta_x} \sin \varphi, \quad x'_\beta = \sqrt{\frac{I_x}{\beta_x}} (\cos \varphi + \alpha_x \sin \varphi)$$

The same expressions are used for z coordinate with substitution of corresponding alpha and beta function.

$$x = x_\beta + D(\Delta p / p), \quad x' = x'_\beta + D'(\Delta p / p),$$

$$\frac{\Delta p}{p} = \sigma_p \cos \phi, \quad s - s_0 = \sigma_s \sin \phi,$$

where s_0 is longitudinal coordinate of the bunch center, here $x' = p \not\propto p$.

A relative change of the ion momentum components after passing through the cooling section can be expressed with the following formula:

$$\begin{pmatrix} \frac{\delta p_x}{p} \\ \frac{\delta p_z}{p} \\ \frac{\delta p_s}{p} \end{pmatrix} = \vec{\Phi}(x, z, s - s_0, x', z', \Delta p / p), \quad (4)$$

where drag rates $\vec{\Phi}$ can be calculated in the BETACOOOL program using one of three different analytical formulae for friction force: at non magnetised [3] and at magnetised electron beam [3, 4] User of the program can make a choice of the formula which he prefers to use in calculations.

Under assumption, that the ion distribution over betatron and synchrotron phases is uniform in an interval $[0, 2\pi]$ (a stationary beam) we can calculate average invariant deviation for ions having the same invariants of motion:

$$\langle \delta I \rangle = \frac{1}{8\pi^3} \int_0^{2\pi} \int_0^{2\pi} \int_0^{2\pi} \delta I(I, \varphi_x, \varphi_z, \varphi_s) d\varphi_x d\varphi_z d\varphi_s \quad (5)$$

These values calculated at invariants of motion, which are equal to beam emittances and can be used for evaluation of so called "single particle" cooling rate:

$$\frac{1}{\tau_{cool,sp}} = \frac{1}{I T_{rev}} \langle \delta I \rangle, \quad (6)$$

where T_{rev} is the particle revolution period in the storage ring.

The cooling rates for ion beam with Gaussian distribution are calculated in BETACOOOL by averaging (5) over distribution function in the space of invariants:

$$\delta \varepsilon_i = \frac{1}{\varepsilon_x \varepsilon_z \varepsilon_s} \int_0^\infty \langle \delta I_i \rangle e^{-I_x/\varepsilon_x - I_z/\varepsilon_z - I_s/\varepsilon_s} dI_x dI_z dI_s$$

and cooling time value is equal :

$$\frac{1}{\tau_{cool,i}} = \frac{1}{\varepsilon_i} \frac{d\varepsilon_i}{dt} = \frac{1}{\varepsilon_i} \frac{\delta \varepsilon_i}{T_{rev}}. \quad (7)$$

It should be noted, that emittance used in Formula (7) as a parameter of distribution function contains 63% of particles and is twice higher than r.m.s. value and this has to be taken into account in calculations of the standard deviation values.

Both possibilities can be used in the simulations: cooling time can be calculated in accordance with (7) or, to speed up the calculations, in accordance with (6).

3. PROGRAM STRUCTURE

The BETACOOOL program is developed on the base of BOLIDE system using Borland C++ Builder 4 and works now under Microsoft Windows operation system. The BOLIDE version for Linux is under development presently. In the program only the standard C++ and BOLIDE commands are used for connection of algorithmic and interface parts. In future the program can be recompiled to Linux version without any modifications. The numerical algorithm is realised on the base of Object Oriented Programming method and the program structure consists of several basic objects. For input and visual presentation of parameters of each object special forms were developed. The tools of the BOLIDE system to load and save data into hard disc, for output the data into two and tree dimensional plots during the calculations and to control of the calculation process are used.

There are several objects in BETACOOOL program. The object "Ring" includes the general parameters of the storage ring, methods of import of

the parameters from external file and control of its validity, methods for calculation of required parameter. The parameters are divided by several groups in accordance with general systems of the storage ring: parameters of the stored ions, lattice parameters - mean and imported from external file, radio frequency and vacuum system parameters.

The object "Beam" includes general beam parameters: particle number, values of the emittances and momentum spread, methods for calculation of beam parameters characterising beam stability and luminosity, for general parameters visualisation in numerical and graphic form. The methods use only ring parameters as input.

Each process involved into calculation of the beam parameters evolution is performed as an independent object also. This object includes the parameters of the corresponding process or device and methods of these parameters optimisation and visualisation. For instance, object "Electron Cooling" includes parameters of the cooling section, parameters

of the electron beam and lattice functions in the cooling section. To optimise cooling time calculations user can choose physical model of the process and mode of the cooler operation. The dependence of friction force on ion angular and momentum deviation is output as a three dimensional plot.

All objects describing a beam parameters evolution are developed on the base of the ancestor class "Effect", which has a virtual function using "Beam" and "Ring" objects as input variables, and returning array of the rates (1): particle loss, two emittances and momentum spread variation. In each descendants class corresponding to concrete process this function is reloaded by its individual one. All variables of the class effect are automatically included into array of the effects, and class effect includes the method, which calculates the sum of the rates in the cycle. This ancestor class includes also methods for output of the rates in numerical and graphical format (see Fig. 1).

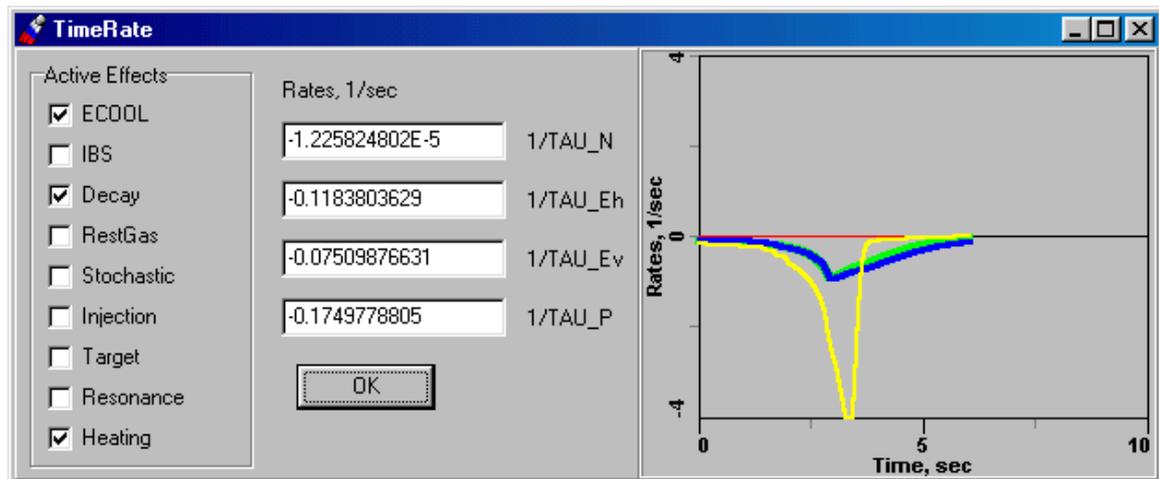


Fig.1. The form for visualisation of the effect array and output of the sum of the rates of the active effects.

CONCLUSION

BETACOOOL program presents now useful instrument for the calculation of the equilibrium parameters of circulating ion beam in storage rings. User can choose any processes, which are involved into calculation and compare their actions. The program structure permits very easy to include new processes inside the source code, which are described in the frame of the present physical model.

The power of C++ Builder compiler makes the friendly interface of BETACOOOL program. Our program products and their source codes are distributed as freeware programs. This program and other products can be downloaded from our homepage <http://nu.jinr.ru/~lepta/betacool.htm>.

ACKNOWLEDGEMENTS

This work is supported by RFBR grant #99-02-17716.

REFERENCES

- [1] A. Lavrentev, I.Meshkov "The computation of electron cooling process in a storage ring", JINR, E9-96-347 (1996).
- [2] P. Zenkevich, T.Katayama, A. Sidorin, "Problems of Ion Beam Stability for MUSES Storage Rings",RIKEN, AF-AC-18.
- [3] I. Meshkov, Fiz. Elem. Chastits At. Yadra 25, p. 1487 (1994).
- [4] V. Parkhomchuk "New insights in the theory of electron cooling", NIM A 441, p. 9-17 (2000).