ISBN: 978-3-95450-214-1 ISSN: 2673-5490 doi:10.18429/JACoW-IPAC2021-MOPAB004

JSPEC - A SIMULATION PROGRAM FOR IBS AND ELECTRON COOLING*

H. Zhang†, M.W. Bruker, Y. Zhang, S.V. Benson
Thomas Jefferson National Accelerator Facility, Newport News, VA, United States

Abstract

Intrabeam scattering is an important collective effect that can deteriorate the properties of a high-intensity beam, and electron cooling is a method to mitigate the IBS effect. JSPEC (JLab Simulation Package for Electron Cooling) is an open-source program developed at Jefferson Lab, which simulates the evolution of the ion beam under the IBS and/or the electron cooling effect. JSPEC has been benchmarked with BETACOOL and experimental data. In this report, we will introduce the features of JSPEC, including the friction force calculation, the IBS expansion rate and electron cooling rate calculation, and the beam-dynamic simulations for the electron cooling process; explain how to set up the simulations in JSPEC; and demonstrate the benchmarking results.

INTRODUCTION

Intrabeam scattering (IBS) [1] is one important problem that hadron collider designers need to consider. Due to small-angle collisions between the ions, the emittance and the momentum spread of the ion beam gradually increase and therefore the luminosity of the collider decreases. Electron cooling [2] is an experimentally proven leading method to reduce the ion beam emittance by overlapping the ion beam with a low-temperature electron beam while both beams co-move inside the cooler in the same velocity to allow thermal energy to transfer from the ion beam to the electron beam. It can be used to mitigate the IBS effect. JLab simulation package for electron cooling (JSPEC) is a program to simulate the effects of both IBS and electron cooling. We started to develop this program in 2014 in order to support the electron cooling scheme study and the electron cooler design for the then on-going electron-ion collider (EIC) project [3] at Jefferson Lab. Since 2018, we have refactored the old code and added some formulas and features [4]. The program is developed using C++ in consideration of efficiency and has been tested on both MS Windows 10 and Ubuntu 18.04 systems. Most computations are parallelized for shared-memory systems using OPENMP to take advantage of the multi-core processors widely available in desktop and laptop computers. JSPEC has been provided to the community as an open-source program. The source codes, the documents, and the examples are all available in the *github* repository [5]. An online JSPEC based on an earlier version has been developed by Radiasoft and is accessible through their cloud service SIREPO [6]. It allows one to run JSPEC and visualize the result inside a browser.

* Work supported by the U.S. Department of Energy, Office of Science, Office of Nuclear Physics under contract DE-AC05-06OR23177.
† hezhang@jlab.org

FEATURES

The basic feature of JSPEC is to calculate the emittance growth rate of the ion beam under the IBS and/or the electron cooling effect. The rate at time t is defined as $r_i(t) = \frac{1}{\epsilon_i(t)}\frac{\mathrm{d}\epsilon_i(t)}{\mathrm{d}t}$, where $i = x, y, s$, representing the horizontal, vertical, and longitudinal direction, and ϵ_i is the emittance in the respective direction. For the IBS rate, JSPEC provides the Martini model [7], the original Bjorken-Mtingwa model [8] calculated by Nagaitsev's method [9], and the complete Bjorken-Mtingwa model with vertical dispersion and non-relativistic terms included [10]. The electron cooling rate is calculated in the following way: a group of sample ions are generated and the friction force on each ion is calculated. The friction force works as a *kick* on the ion particle and changes its momentum. Hence the emittance of the ion beam is changed. The rate is calculated as the relative change of the emittance per unit time before and after the kick. JSPEC provides several formulas [11-14] for both the non-magnetized friction force and the magnetized friction force. Users can also choose two different formulas and use them in the transverse direction and the longitudinal direction, respectively.

Another important feature of JSPEC is to simulate the evolution of the ion beam under the IBS effect and/or the electron cooling effect. Three models have been implemented. The first is called the RMS dynamic model, in which the ion beam is represented by the macroscopic parameters, *i.e.* the emittances, the momentum spread, and the bunch length (for a bunched beam). For each time step Δt, the expansion rate R_i is calculated and the macroscopic parameters are updated as $\epsilon_i(t + \Delta t) = \epsilon_i(t)\exp(R_i \Delta t)$. This model assumes the ion beam maintains a Gaussian distribution throughout the simulation. The second is called the particle model, in which the ion beam is represented by sample particles. The IBS effect during one time step is represented as a random kick w.r.t. the IBS expansion rate. The electron cooling effect is also represented as a kick due to the friction force. The betatron oscillation and the synchrotron oscillation are modeled by a random phase advance. In this model, the ion beam can deviate from a Gaussian distribution. The third model is called the turn-by-turn model, which is identical to the particle model except that the time step is counted by "turns" and the sample ions are moved by a linear one-turn map. This model has a more accurate description of betatron and synchrotron oscillations.

In all the above calculations/simulations, the ion beam can be coasting or bunched and the electron beam can be DC or bunched. In JSPEC, there are several predefined

ISBN: 978-3-95450-214-1 ISSN: 2673-5490 doi:10.18429/JACoW-IPAC2021-MOPAB004

electron models with regular shape (round, elliptical, hollow) and regular distribution (uniform, Gaussian). Users can also define an electron beam with arbitrary shape and distribution by providing the 6D coordinates of the sample particles. JSPEC groups the particles according to their position and the nearby density inside boxes created hierarchically. The number of particles in each box should be below a predefined number. The local density and temperature in each box will be calculated and then be used to calculate the friction force on the ions inside the box. In many scenarios, the electron beam is assumed to be unaffected by the ion beam and hence the density and temperature only needs to be calculated once.

Besides the aforementioned features, JSPEC can also calculate the friction force map within a given parameter domain and the luminosity during the electron cooling or IBS expansion process given the parameters of the colliding beam.

BENCHMARK

JSPEC has been benchmarked with BETACOOL [11] for various scenarios. Here we choose two cases to present. Figure 1 shows a 30 GeV bunched proton beam cooled by a bunched electron beam, which is simulated using

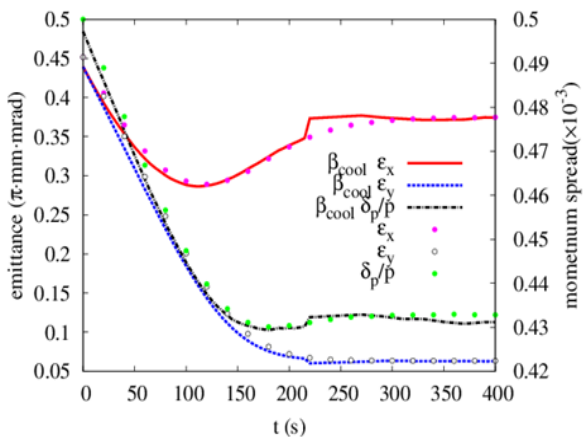

Figure 1: Emittance and momentum spread evolution of a 30 GeV proton beam cooled by bunched electron beam.

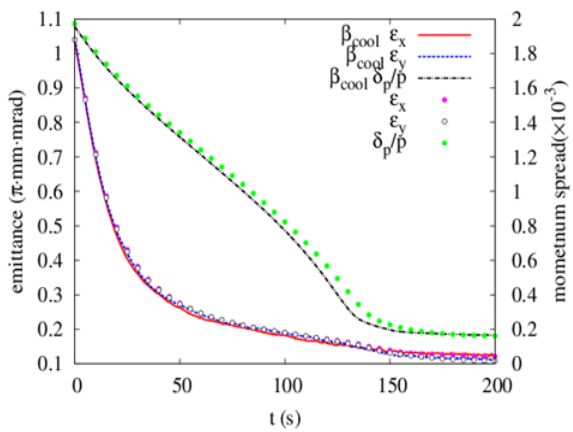

Figure 2: Emittance and momentum spread evolution of a 2 GeV coasting proton beam under DC cooling.

the RMS dynamic model. Figure 2 shows a nonrelativistic 2 GeV coasting proton beam cooled by a DC electron beam, simulated using the particle model. In both cases, the solid lines are BETACOOL results and the dots are JSPEC results. The two programs agree well. For the typical simulations we have done for the EIC project, a significant improvement of efficiency has been achieved even without using multiprocessing in JSPEC. Parallel computation will further improve the efficiency.

In JSPEC parallel computation is enabled for shared-memory structure by OPENMP. The users have the choice to compile a parallel JSPEC by adding the flag *OMPFLAGS=-fopenmp* to the *make* command. By default, the parallel JSPEC will use all the available threads. But JSPEC allows users to set the number of threads to use. Table 1 shows an example, in which the electron cooling process together with the IBS effect for a proton beam is simulated for 50,000 steps using 40,000 particles on a personal computer running an Intel i7-4820k CPU with four cores and eight hyperthreads. When four threads are used, the computation time is reduced by about 50%.

Table 1: Computation Using Multi-Threads

No. of threads	Time (s)	No. of threads	Time (s)
1	393	5	198
2	258	6	193
3	217	7	187
4	201	8	190

We also compared JSPEC simulations with experimental data. From 2016 to 2019, four pulsed-beam cooling experiments have been carried out by a collaboration of Jefferson Lab in the U.S. and Institute of Modern Physics in China. These experiments successfully demonstrated the cooling of heavy-ion beams with a pulsed electron beam [15]. We used JSPEC to simulate a few cases in the experiments and compared the results with the collected data. Figure 3 shows the cooling of the $^{86}Kr^{25+}$ beam with an energy of 5 MeV/nucleon using electron pulses with a length varying from 600 ns to 1000 ns. The cooling process starts at 0.1 s. The bunch lengths of the ion beams are measured during the cooling process in the experiments. The solid lines in the plot are the results from the simulation using the turn-by-turn model, the parameters used in which are listed in Table 2 [15]. The dots are experimental data. While the pulse length of the electron beam changes, the peak current remains constant, so a longer pulse length means a longer overlap between the two beams and a stronger cooling, which is observed through the larger slope of the experimental data and the simulation results at the beginning of the cooling process. In all the cases, the simulation agrees with the experiment reasonably well. Especially good agreement appears in the first half of the curve, when the electron cooling is overwhelmingly stronger than the IBS effect and the bunch length reduces almost linearly due to the strong cooling.

ISBN: 978-3-95450-214-1 ISSN: 2673-5490 doi:10.18429/JACoW-IPAC2021-MOPAB004

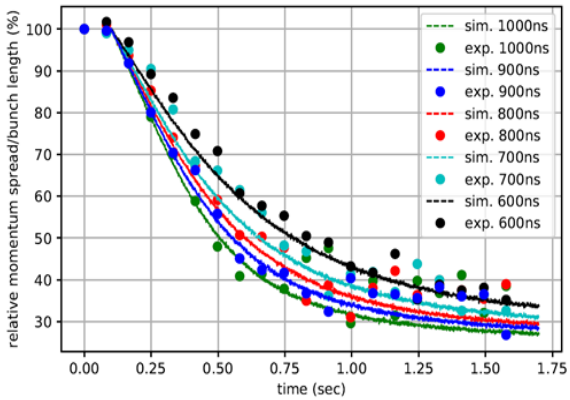

Figure 3: Cooling the $^{86}Kr^{25+}$ beam using pulsed electron beam, simulation (solid lines) and experiments (dots).

Table 2: Electron Cooling Simulation Parameters

e^- beam radius	15 mm
Cooler length	3.4 m
Magnetic field	0.1 T
β_x/β_y in the cooler	10/17 m
e^- beam peak current	30 mA
e^- beam temperatures	200/6 meV
Ion beam normalized emittance	0.6 mm mrad
Ion beam RMS bunch length	10.5 m
Ion beam RMS momentum spread	7×10^{-4}
Lorentz factor β/γ for both beams	0.103/1.005

HOW TO USE JSPEC

JSPEC has been tested on both Windows and Linux systems. It can be run from the command line by typing the executable file name followed by the input file name.

The JSPEC input file is in plain text format and composed of a few sections as shown in Fig. 4. Most sections are used to defined the elements and to set up models for the respective effect by given values to the keywords. In the last section (section_run), the elements are created and the simulation is carried out. A valid input has to include the above two types of sections. There are two optional sections: section_scratch, to define some variables used in the following sections and perform simple calculations, and section_comment, to write a long note that is not suitable for inline comments. For more useful examples, we suggest that the readers check out the *github* repository.

FUTURE WORK

Currently, the IBS models in JSPEC assume a Gaussian distribution of the ion beam. However, deviations from a Gaussian distribution have been observed in both experiments and simulations of the electron cooling process. IBS models that work for an arbitrary ion distribution are under construction. In these models, the ion beam is represented by sample particles, which are grouped by their position and nearby density in the same way we treat the arbitrary electron beam as mentioned before. Inside each group, there are two ways to simulate the interaction between the

```
section_ion #define the ion beam
        charge_number = 1
        mass = 938.272
        kinetic_energy = 3e4
        ...
section_ring #define the ring
        ...
section_cooler #define the cooler
        ...
section_scratch
        m = 938.272
        ke = 3e4
        gamma_ion = ke/m + 1
section_e_beam #use gamma_ion above
        gamma = gamma_ion
        ...
section_ibs #set up IBS calculation
        model = bm
        log_c = 24
        coupling = 0
section_ecool
        ...
section_simulation
        ...
section_run
        create_ion_beam
        create_ring
        create_e_beam
        create_cooler
        run_simulation
```

Figure 4: Structure of JSPEC input.

ions. One is the binary collision model, which randomly chooses the ions in pairs and calculates the change of momentum after the pair of ions collide with each other [16]. The other is to calculate the friction coefficient and the diffusion tensor using the local density and temperature and apply to the ions random kicks, which are statistically consistent with the friction coefficient and the diffusion tensor [17]. The assumption on the ion distribution is removed in these two models.

We are also developing a Python wrapper for JSPEC using Pybind11 [18], which will make it possible for JSPEC to run as a library in Python 3.x environment and to interoperate with other accelerator modeling programs with a Python interface and the abundant Python tools of numerical analysis, data visualization, machine leaning, *etc*.

SUMMARY

JSPEC is an open-source program for IBS and electron cooling simulations developed at Jefferson Lab, which has been benchmarked with other programs and experimental data. The source code, manuals, and examples can be found in the *github* repository. The models for IBS of an arbitrary ion distribution and a Python version are under active construction now.

ACKNOWLEDGMENT

The authors thank all the users for their valuable feedback and thank David Bruhwiler, Paul Moeller, Ilya Pogorelov, and Stephen Coleman at Radiasoft for fruitful discussions and for bringing JSPEC to SIREPO.

This material is based upon work supported by the U.S. Department of Energy, Office of Science, Office of Nuclear Physics under contract DE-AC05-06OR23177.

ISBN: 978-3-95450-214-1 ISSN: 2673-5490 doi:10.18429/JACoW-IPAC2021-MOPAB004

REFERENCES

[1] A. Piwinski, "Intra-beam scattering", in *Proc. 9th Int. Conf. on High Energy Accelerators*, Stanford, CA, May 1974, pp. 405-409. doi:10.5170/CERN-1992-001.226

[2] G. Budker *et al.*, "Experimental studies of electron cooling", *Part. Accel.*, vol. 7, pp. 197-121, 1976.

[3] S. Abeyratne *et al.*, "MEIC design summary", 2015. arXiv:1504.07961

[4] H. Zhang *et al.*, "Electron cooling simulation code development", Rep. JLAB-TN-21-002, Jefferson Lab, 2021.

[5] JSPEC, https://github.com/JeffersonLab/ElectronCooling

[6] SIREPO, https://www.radiasoft.net/sirepo

[7] M. Martini, "intrabeam scattering in the ACOL-AA machines", Rep. CERN-PS-8-4-9-AA, CERN, 1984.

[8] J. Bjorken and S. Mtingwa, "Intrabeam scattering," *Part. Accel.*, vol. 13, pp. 115-143, 1982.

[9] S. Nagaitsev, "Intrabeam scattering formulas for fast numerical evaluation", *Phys. Rev. ST Accel. Beams*, vol. 8, p. 064403, 2005. doi:10.1103/physrevstab.8.064403

[10] F. Zimmermann, "Intrabeam scattering with non-ultrarelativistic corrections and vertical dispersion for MAD-X", Rep. CERN-AB-2006-002, CERN, 2005

[11] I. Meshkov *et al.*, "BETACOOL physics guide", *Joint Institute for Nuclear Research*, Dubna, Russian Federation, 2007

[12] I. Meshkov, "Electron cooling: status and perspectives", *Phys. Part. Nucl.*, vol. 25, pp. 631-661, 1994. doi:10.1134/S1063779615060052

[13] V. Parkhomchuk, "New insights in the theory of electron cooling", *Nucl. Instrum. Methods Phys. Res., Sect. A*, vol. 441, pp. 9-17, 2000. doi:10.1016/S0168-9002(99)01100-6

[14] Y. Derbenev and A. Skrinsky, "The effect of an accompanying magnetic field on electron cooling", *Part. Accel.*, vol. 8, pp. 235-243, 1978.

[15] M. Bruker *et al.*, "Demonstration of electron cooling using a pulsed beam from an electrostatic electron cooler", *Phys. Rev. Accel. Beams*, vol. 24, p. 012801, 2021. doi:10.1103/PhysRevAccelBeams.24.012801

[16] A. Fedotov *et al.*, "IBS for non-Gaussian distributions", Rep. BNL-94081-2010-CP, BNL, 2010.

[17] P. Zenkevich *et al.*, "Modelling of electron cooling by Monte Carlo method", presented at the International Workshop on Cooling and Related Topics, Bad Honnef, Germany, 2001.

[18] Pybind11, https://pybind11.readthedocs.io