ISBN: 978-3-95450-184-7 doi:10.18429/JACoW-IPAC2018-THPAK085

3D SPACE CHARGE IN BMAD

C. E. Mayes*, SLAC National Accelerator Laboratory, Menlo Park, CA, USA
R. D. Ryne, Lawrence Berkeley National Laboratory, Berkeley, CA, USA
D. C. Sagan, Cornell University, Ithaca, NY, USA

Abstract

We present a parallel fast Fourier transform based 3D space charge software library based on integrated Green functions. The library is open-source, and has been structured to easily be used by existing beam dynamics codes. We demonstrate this by incorporating it with the Bmad toolkit for charged particle simulation, and compare with analytical formulas and well-established space charge codes.

INTRODUCTION

Space charge is an important and often dominant effect in high brightness charged particle beams. When combined with external fields, the total effect on the evolution of a bunch distribution is complex, and can only be calculated by a direct numerical simulation. Because of this, many space charge codes have been developed which, in addition to the space charge calculation, offer full-featured lattice description and physics tracking capabilities [1–5].

To incorporate space charge into the Bmad toolkit for charged particle simulation [6], we have taken a different approach and developed a stand-alone software package called "Open Space Charge" (OpenSC) that aims only to calculate the internal fields in a bunch distribution, and nothing else. Bmad then uses these routines in tracking simulations. This modularity allows the space charge routines to be incorporated into other codes, without having to strip out unneeded components.

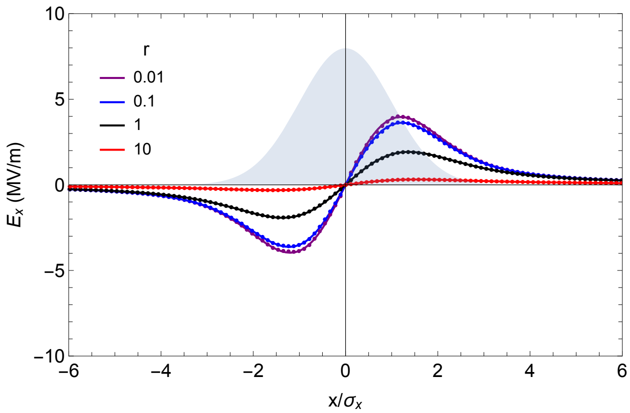
Here we describe the OpenSC package, and validate it against analytic formulas. We then show how it is incorporated in Bmad, and validate this against other space charge codes.

OPEN SPACE CHARGE PACKAGE

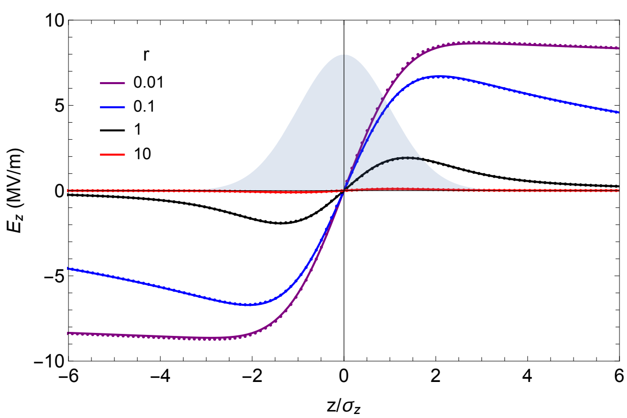
The OpenSC package is an open-source software library written primarily in Fortran 2008 for calculating space charge fields [7]. It was originally developed as a reusable Poisson solver with free-space boundary conditions for use within the Warp framework [8], and will be incorporated into the Particle-In-Cell Scalable Application Resource, PICSAR [9].

OpenSC currently implements free-space and rectangular conducting pipe methods using integrated Green functions (IGFs) as described in [3] and [10], respectively. The package provides high-level routines to:

- Deposit weighted charged particles on a 3D rectangular grid.



(a) Electric field along the line at $y = 0$, $z = 0$.



(b) Electric field along the line at $x = 0$, $y = 0$.

Figure 1: Electric field comparison using the OpenSC numerical calculation (dots) and the analytical formula Eq. 1 (lines) for a Gaussian bunch at rest for various aspect ratios $r = \sigma_z/\sigma_\perp$. The bunch charge is 1 nC, with $\sigma_x = \sigma_y \equiv \sigma_\perp = 1$ mm. This numerical calculation is with 10^6 particles on a $128 \times 128 \times 128$ computational grid, and uses an integrated green function (IGF) FFT method.

- Calculate the space charge fields on this grid (various methods).
- Interpolate the field to an arbitrary point within its domain.

Convolutions of the Green functions and the charge density are performed efficiently with fast Fourier transforms (FFTs).

* cmayes@stanford.edu

ISBN: 978-3-95450-184-7 doi:10.18429/JACoW-IPAC2018-THPAK085

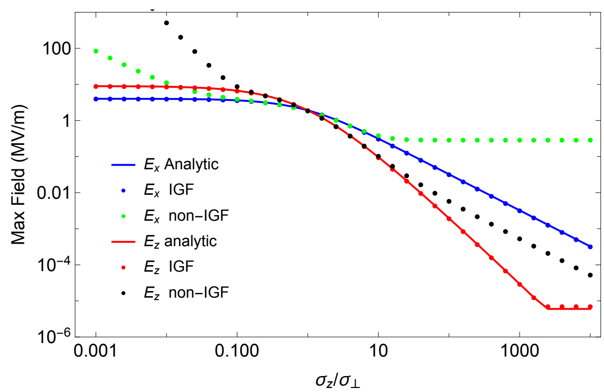


Figure 2: Maximum field comparison for the bunch described in Fig 1 between the analytic formula originating from Eq. 1, the OpenSC package integrated green function (IGF) FFT method, and the non-IGF method.

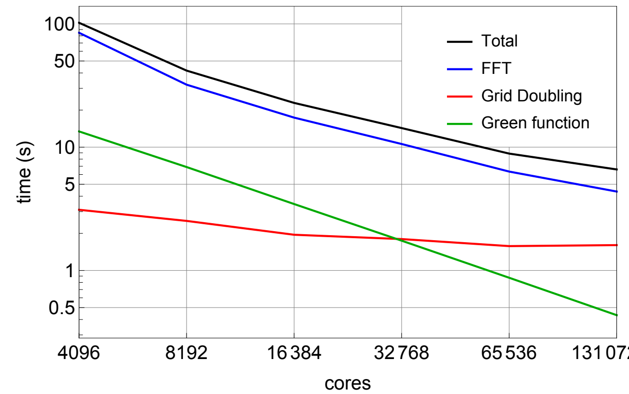


Figure 3: Strong scaling of the OpenSC space charge calculation on the Edison supercomputer at NESRC [12] for a 4096 × 4096 × 4096 computational grid. The code is parallelized with MPI. Times are shown for the dominant routines.

Validation

We validate the numerical calculation using an analytical formula. The electric potential ϕ at a Cartesian point (x, y, z) a due to a Gaussian charge distribution at rest can be calculated by

$$\phi = \frac{Q}{4\pi\epsilon_0}\sqrt{\frac{2}{\pi}}\int_0^\infty \frac{e^{\frac{-\lambda^2 x^2}{2(\lambda^2\sigma_x^2+1)}}\, e^{\frac{-\lambda^2 y^2}{2(\lambda^2\sigma_y^2+1)}}\, e^{\frac{-\lambda^2 z^2}{2(\lambda^2\sigma_z^2+1)}}}{\sqrt{(\lambda^2\sigma_x^2+1)(\lambda^2\sigma_y^2+1)(\lambda^2\sigma_z^2+1)}}\mathrm{d}\lambda \tag{1}$$

where Q is the total charge, ϵ_0 is the permittivity of free space, and σ_x, σ_y, σ_z are the standard deviations in each coordinate dimension [11]. Electric fields in this rest frame can be calculated by $E = -\nabla\phi$, and can be boosted to another reference frame using Lorentz transformations.

Figure 1 shows excellent agreement between the free space numerical calculation and the analytical formula for various aspect ratios of the bunch length to transverse size. This is important because for a bunch moving with relativistic factor γ in the z direction with bunch length $\sigma_{z,\text{lab}}$, the bunch length in the rest frame is $\sigma_z = \gamma\sigma_{z,\text{lab}}$.

Figure 2 shows the robustness of the IGF method for large aspect ratios, and compares this with the simpler non-IGF method. The non-IGF method fails when one of the bunch dimensions is about a factor of 10 greater than another.

Parallelization

The OpenSC package can be run in parallel using OpenMP and MPI. The OpenMP methods are useful for running on a local machine. The MPI methods are suitable for larger calculations on High Performance Computing (HPC) hardware. For these methods, the computational grid is domain decomposed. Figure 3 shows strong scaling of a single space charge calculation using up to 131,072 cores. The calculation is dominated by the parallel FFT.

BMAD SPACE CHARGE TRACKING

We exemplify the usage of the OpenSC package by incorporating it into Bmad. Bmad offers routines for a wide variety of accelerator physics effects, including nonlinear dynamics, incoherent and coherent synchrotron radiation, and intra-beam scattering. These effects are applied in a series of computational steps through the accelerator lattice.

To add space charge to Bmad tracking, we simply split an individual computational step into to two parts, and apply the space charge kick between them. This is straightforward due to the modularity of the Bmad code.

Validation

Figure 4 shows particle tracking results for Bmad, and compares them with Astra and ImpactZ. For Astra we used the cylindrically symmetric space charge calculation. The ImpactZ space charge method is an IGF FFT method similar to that in OpenSC. The figure shows that the transverse phase spaces are nearly identical, and the longitudinal phase spaces show excellent agreement.

CONCLUSION

Bmad now incorporates 3D space charge using the independent software library OpenSC, which can be run parallelized with OpenMP and MPI. The free-space space charge field calculation shows excellent agreement with analytical formulas, and tracking particles in these fields shows excellent agreement with well-established space charge codes. We are currently expanding the OpenSC package to include the effect of image charges on a cathode, so that a bunch can be simulated starting at an electron gun.

ISBN: 978-3-95450-184-7 doi:10.18429/JACoW-IPAC2018-THPAK085

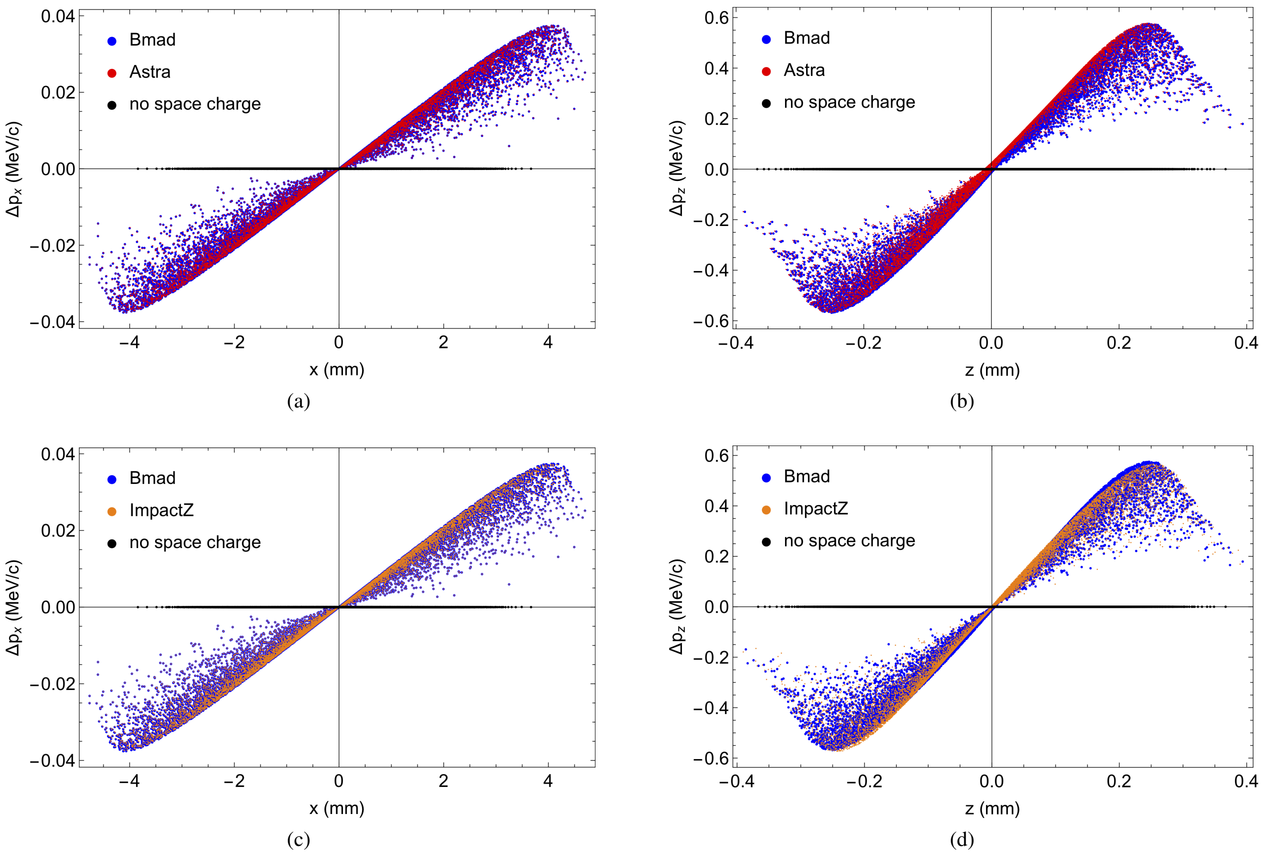


Figure 4: Phase space comparison at the end of a 1 m drift between Bmad [6] and Astra [1] (a,b), and between Bmad and ImpactZ [4] (c,d). Particles without space charge forces applied are also shown. The initial bunch distribution is Gaussian with $(\sigma_x, \sigma_y, \sigma_z) = (1, 1, 0.1)$ mm, 1 nC of charge, and 10 MeV total energy moving in the z direction. The initial momentum spreads are zero. All methods tracked the same initial 10^6 particles, of which we show here 10^4 sample particles.

ACKNOWLEDGEMENTS

This work was supported in part by the U.S. Department of Energy Office of Science, Office of Basic Energy Sciences under Contract No. DE-AC02-76SF00515 (under award field work proposal 10074), Office of High Energy Physics under Contract No. DE-AC02-05CH11231, and the National Science Foundation Grant NSF PHY-1416318. This research used resources of the National Energy Research Scientific Computing Center, a DOE Office of Science User Facility supported by the Office of Science of the U.S. Department of Energy.

REFERENCES

[1] ASTRA - A Space-charge TRacking Algorithm, `http://www.desy.de/~mpyflo/`

[2] GPT - General Particle Tracer, Pulsar Physics, `http://www.pulsar.nl`

[3] J. Qiang, S. Lidia, R. D. Ryne, and C. Limborg-Deprey, "Three-dimensional quasi-static model for high brightness beam Dynamics simulation," Phys. Rev. ST Accel. Beams, Vol. 9, 044204 (2006)

[4] J. Qiang, R. Ryne, S. Habib, V. Decyk, "An object-oriented parallel particle-in-cell code for beam dynamics simulation in linear accelerators," J. Comp. Phys. vol. 163, 434, (2000)

[5] A. Adelmann et al.,"The OPAL (Object Oriented Parallel Accelerator Library) Framework", Paul Scherrer Institut PSI-PR-08-02 (2018) `https://gitlab.psi.ch/OPAL/src/wikis/home`

[6] D. Sagan, "Bmad Reference Manual" `https://www.classe.cornell.edu/bmad/`

[7] R. Ryne & C. Mayes, Open Space Charge package `https://github.com/RobertRyne/OpenSpaceCharge`

[8] `http://warp.lbl.gov/`

[9] H. Vincenti, M. Lobet, R. Lehe, R. Sasanka and J-L Vay, "An efficient and portable SIMD algorithm for charge/current deposition in Particle-In-Cell codes," Comp. Phys. Comm. 210, 145-154 (2017) `https://picsar.net/`

[10] R. D. Ryne, "On FFT-based convolutions and correlations, with application to solving Poisson's equation in an open rectangular pipe," `https://arxiv.org/abs/1111.4971`

[11] G. Stupakov and G. Penn, *Classical Mechanics and Electromagnetism in Accelerator Physics*, Springer, ISBN:978-3-319-90187-9 (2018)

[12] National Energy Research Scientific Computing Center (NERSC) `http://www.nersc.gov`