ISBN: 978-3-95450-247-9 ISSN: 2673-5490 doi: 10.18429/JACoW-IPAC2024-WEPR69

PROGRESS & DEVELOPMENTS OF BDSIM

W. Shields*, John Adams Institute at Royal Holloway, University of London, Egham, UK
L. J. Nevay, CERN 1211 Meyrin, Switzerland
S. T. Boogert, Cockcroft Institute, Daresbury, UK

Abstract

Beam Delivery Simulation (BDSIM), is a C++ program that seamlessly models particle beam transport within an accelerator model that can encompass the beam line, the accelerator's environment, and any accompanying detectors. Based on a suite of high-energy physics software including Geant4, CLHEP, and ROOT, BDSIM transforms the optical design of an accelerator into a detailed 3D model. This facilitates the simulation of particle interactions with matter and the subsequent production of secondary particles. Widely utilized across diverse accelerators worldwide, BDSIM is ideal for simulating energy deposition and assessing charged particle backgrounds. Here, the latest BDSIM developments are shown including automatic rigidity scaling, a Gabor plasma lens beam line component, and progress on python bindings & interfacing with external tracking tools such as Xsuite.

INTRODUCTION

Modelling the transport of particles in an accelerator including their potential interactions with matter, beam losses, and energy deposition are essential for understanding the accelerator's performance and detector backgrounds. Such modelling, however, requires handling of the physics processes involved in particle matter interactions. The scale and complexity of this limits accurate accelerator loss studies for all particles and across all energies to only a small handful of codes including Geant4 [1–3], FLUKA [4], and MARS [5]. These codes, however, often require significant effort to construct accelerator models.

BDSIM is a C++ code that uses the Geant4 toolkit to programmatically construct 3D models of particle accelerators and their environment [6, 7]. BDSIM can rapidly generate accelerator models from simple optical descriptions of the lattice, constructing the most common accelerator components with scalable geometry in a variety of styles. BDSIM's simplicity coupled with its vast functionality has resulted in its widespread use in modelling and designing many accelerators. A light review of the BDSIM user community can be found in [8].

Here, we highlight some of the developments made available in the recently released v1.8.0 of BDSIM.

GENERAL DEVELOPMENTS

BDSIM is a relatively mature code and recent development has focused on bug fixes and small features as required. The muon-splitting feature was fixed for rare decays that produced muons where the weight was reset to 1.0 mistakenly. The implementation of the electric and electromagnetic fields RF cavities has been refined fixing behaviour for pill-box cavities.

Automatic Rigidity Scaling

BDSIM constructs the fields of magnetic beamline elements according to normalised strength parameters defined in the input in combination with the design particle's magnetic rigidity. In the case where acceleration in an *rf* element causes a net change in a particle's momentum, magnets downstream of the cavity are subsequently constructed with an incorrectly normalised strength. Previously, this could be corrected by manually supplying a *scaling* value to the element, however this could be a potentially time consuming task, more so if subsequent changes to upstream *rf* elements are made and scaling has to be re-calculated. The synchronous time for time-varying fields such as RF cavities assumed the speed of light as the particle velocity always.

Now, the rigidity and synchronous time are automatically calculated along the beamline and the scaling factors are no longer needed. The global synchronous time at the centre of an *rf* element is calculated automatically such that zero phase results in the peak E-field at the centre of the component for its position in the lattice.

Automatic rigidity scaling only occurs once during model construction, acceleration in circular machines is therefore limited at present to a single turn. Scaling can still be applied to an element on top of the automatic rigidity scaling. The user, however, should take care in calculating that the two scaling factors together will generate the expected element field strength. The automatic rigidity scaling can be turned off with the option *integrateKineticEnergyAlongBeamline*. Automatic rigidity scaling is only calculated from acceleration/deceleration in RF cavities. Deliberate changes in energy due to interaction with material, for example in a degrader, are not accounted for. Such changes would be too complex to calculate given the range of possible materials, beam energies, and geometries that would need consideration.

As BDSIM reuses components if they are defined in a sequence multiple times, we now cache components for reuse based upon their name and nominal rigidity at that point in the beamline. This is because if, say, a quadrupole is used later in the beamline after acceleration with the same k_1, the actual field gradient is different and so the component must be uniquely constructed to have a different field.

The calculated values of synchronous time and the nominal momentum and rigidity along the line are now given in the output so that the user can verify the behaviour.

---

* william.shields@rhul.ac.uk

Muon Cooler

A muon cooler element has been introduced to BDSIM. This is currently a straight element that includes multiple solenoid coils, RF cavities and optional absorbers inside one cell. Aside from the geometry, the implementation provides two solenoid field models where the magnetic field is calculated from the sum of the fields from each coil. For a given circular coil, a *sheet* and *cylinder* field model have been implemented. The geometry of a single muon cooling cell is shown in Fig. 1.

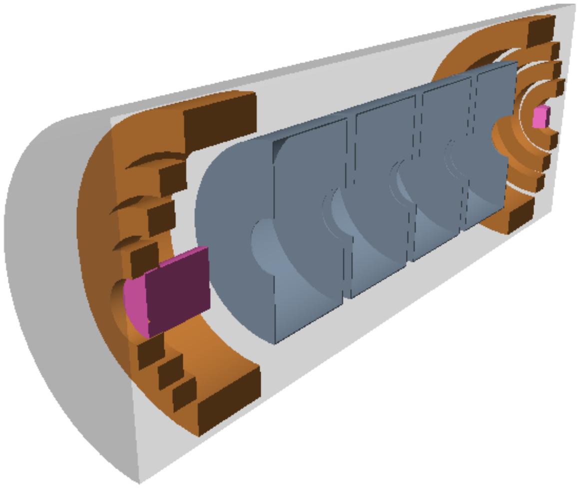

Figure 1: A muon cooler cell with coils (brown), two absorbers (pink), and RF cavities (grey) inside a single element.

The input allows the number of coils and their position to be chosen with great flexibility and is currently being tested with complete lattices. In future, angled channels with a combined dipole field will be added.

Gabor Lens Element

A Gabor lens is a device in which an electron plasma is confined in a Penning-Malmberg trap configuration, with an external magnetic solenoid field containing the plasma radially and an electric field containing it axially. The confined plasma generates a strong radial electric field with focusing forces comparable to that of the magnetic focusing of a solenoid. Whilst not a new concept, Gabor lenses have recently been explored as devices with potential for efficient capture of proton & ion beams in medical applications. LhARA, the Laser-hybrid Accelerator for Radiobiological Applications, proposes to develop and demonstrate Gabor lenses for the capture of 15 MeV protons generated from a laser-target interaction [9, 10]. Until recently, LhARA modelling in BDSIM simulated Gabor lenses as equivalent length solenoids, however the performance of the LhARA accelerator with Gabor lenses required demonstrating.

As such, a *gaborlens* element has been added to BDSIM as a beam line component. The current Gabor lens geometry is a simplified version of the experimental setup at Imperial College [11]. Figure 2 shows the external (top) and internal (bottom) perspectives of the BDSIM *gaborlens* element. The anode and electrodes are placed within the element's vacuum volume. The vacuum volume *apertureType* is *"circular"* and cannot be changed via element parameters or global options. The anode and electrode geometries have been parameterized to control their length, radius, and thickness. The anode, electrode, and coil geometry materials are hard-coded as copper. End caps 1 cm in length are constructed either end of the element. These are used practically in Gabor lenses for both vacuum & electrode grounding, we include these in BDSIM to provide semi-realistic geometry. This also ensures particles that have exited the beam pipe elsewhere in the model cannot axially enter the Gabor lens uninterrupted by material. The lens' plasma field is contained purely to within the beam pipe vacuum volume, therefore the Gabor lens field is 2 cm shorter than the total element length, the user must account for this when calculating a lens' focusing strength.

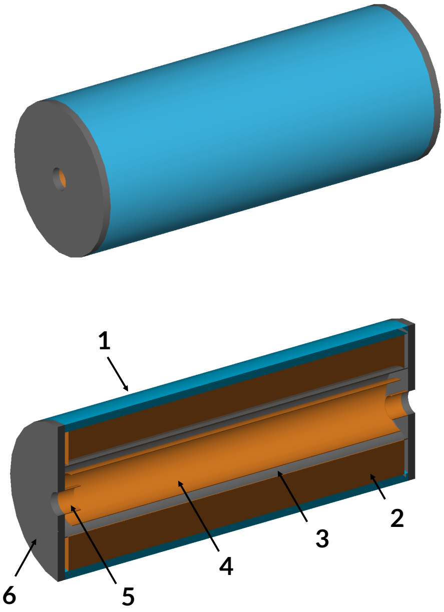

Figure 2: A Gabor lens element in BDSIM, externally (top), and internally (bottom). The internal structure of a Gabor lens comprises the outer container (1), copper representing coils for the solenoid confinement field (2), vacuum tube (3), anode (4), electrode (5), and end cap (6).

The *gaborlens* field is constructed only as the electric field that would be generated from a confined plasma. The confinement fields are not modelled at this stage though provision is made for their later inclusion. The strength of a Gabor lens is defined by the Gabor lens focusing parameter in Eq. (1) where e is the electron charge, ϵ_0 is the permittivity of free space, m_{ion} is the beam particle mass, γ is the beam particle Lorentz factor, p is the beam particle momentum, and n_e is the electron plasma density.

$$k_G = \frac{e}{2\epsilon_0}\frac{m_{ion}\gamma}{p^2}n_e \tag{1}$$

The Gabor lens strength can alternatively be defined with a solenoid-equivalent B field strength. Internally, BDSIM calculates the plasma electric field components with Eq. 2 where c is the speed of light. The field is only considered finite within the anode radius.

$$\begin{aligned} E_x &= -\frac{B^2c^2}{4m_{ion}}x \\ E_y &= -\frac{B^2c^2}{4m_{ion}}y \\ E_z &= 0 \end{aligned} \tag{2}$$

The B field value is calculated from the k_G strength assuming the plasma density is uniform and that the radial & axial confinement fields are in equilibrium and therefore the plasma density is optimal.

To validate the Gabor lens field, a Gaussian beam of 1000 protons at 15 MeV was tracked through a 0.857 m long Gabor lens element of strength $k_G = 0.804566$, equivalent to a 1 T solenoid magnet. Particle coordinates exiting the lens were compared to particles tracked through an external electric field map generated for an idealised Gabor lens field. The particle coordinate residuals are shown in Fig. 3, excellent agreement is observed. Simulations of the LhARA beamline with the implemented Gabor lens elements has been conducted in comparison to solenoids, excellent agreement is observed in both beam optics and 1D spatial and momentum profiles at the end of LhARA's Stage 1 beamline [12].

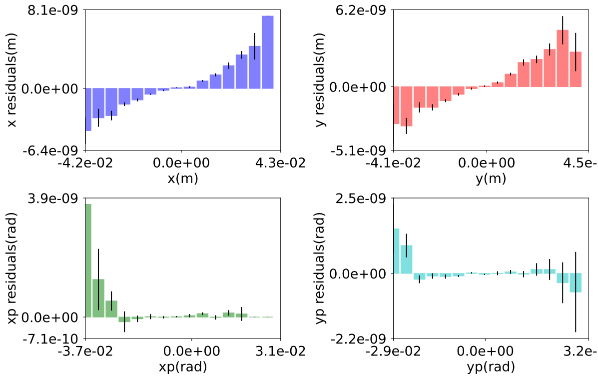

Figure 3: Particle coordinate residuals at the exit of a BDSIM *gaborlens* element compared to identical particles tracked through an idealised Gabor lens external field map.

Python Bindings with PYBIND11

Interfacing BDSIM with external tracking tools can offer potentially huge advantages. Sympletic tracking tools can offer stable long-term tracking that BDSIM is not designed for, and the external tools can benefit from BDSIM's geometry libraries and access to Geant4 physics processes.

Recent interest in this has seen the implementation of python bindings with the pybind11 package to couple BDSIM to Xsuite [13] for studies of FCC-ee collimation systems [14]. The study uses a toolchain in which the dedicated collimasim code acts as a interface by providing limited python bindings to BDSIM whilst also coupling to the Xtrack single particle tracking library within XSuite [15]. The code is limited to collimation studies only, it can create simple collimator models and exchange particle coordinates by extracting hit information from BDSIM samplers. More recently, this functionality has been made available via integration into the Xcoll package in within Xsuite.

The demonstrable success of this approach has lead to our own efforts to implement broader Python bindings with pybind11 directly in BDSIM. Developments are underway with bindings so far created primarily for BDSIM's parser to enable rapid model construction. This includes but is not limited to beam line elements, sequences, beam definition, apertures, fields, options, and model objects including placements, sampler placements, and scorer meshes.

The next stage will include bindings for extracting run data. This will include sampler data, hits and energy deposition, trajectories, collimator hits, aperture impacts, scored quantities, and runtime data. Python bindings will also be extended to BDSIM's analysis tools including rebdsim, rebdsimOptics.

SUMMARY & OUTLOOK

BDSIM v1.8.0 has recently been released with a modest number of new developments. Magnetic rigidity is now scaled automatically in models in which the beam is accelerated with cavities, removing the need for manual scaling. A muon cooler element has been added to BDSIM that includes solenoid fields and RF cavities. A Gabor plasma lens has been successfully implemented with tracking accuracy validated. The Gabor lens anode and electrode geometry will be updated to match the proposed LhARA lens configuration when the geometric design has been finalized. The electric axial confinement field is geometry dependent and will require an electrostatic solver to generate, this will be considered in future developments. The magnetic containment field will also be added.

REFERENCES

[1] S. Agostinelli *et al.*, "Geant4 - A Simulation Toolkit,"*Nucl. Instrum. Meth. A*, vol. 506, pp. 250–303, 2003.

[2] J. Allison *et al.*, "Geant4 Developments and Applications,"*IEEE Trans. Nucl. Sci.*, vol. 53, pp. 270–278, 2006.

[3] J. Allison *et al.*, "Recent Developments in Geant4,"*Nucl. Instrum. Meth. A*, vol. 835, pp. 186–225, 2016.

[4] G.Battistoni *et al.*, "Overview of the FLUKA code,"*Annals of Nuclear Energy*, vol. 82, pp. 10–18, 2015.

[5] N.Mokhov, *MARS15 ComputerSoftware Vers.00*, USDOE Office of Science (SC), High Energy Physics (HEP), 15 Jul. 2016.

[6] L.J. Nevay *et al.*, “BDSIM: An accelerator tracking code with particle-matter interactions,”*Comput. Phys. Commun.*, vol. 252, p. 107200, 2020. `doi:10.1016/j.cpc.2020.107200`

[7] BDSIM, `http://www.pp.rhul.ac.uk/bdsim`

[8] W. Shields, L. Nevay, and S. Boogert, “A review of the Beam Delivery Simulation (BDSIM) user community”, presented at the IPAC’24, Nashville, TN, USA, May 2024, paper WEPR70, this conference.

[9] The LhARA consortium, “The Laser-hybrid Accelerator for Radiobiological Applications,”Imperial College London, UK, Rep. CCAP-TN-01, 2020. `https://ccap.hep.ph.ic.ac.uk/trac/raw-attachment/wiki/Communication/Notes/CCAP-TN-01.pdf`

[10] G. Aymar *et al.*, “LhARA: The Laser-hybrid Accelerator for Radiobiological Applications,”*Front. Phys.*, vol. 8, 2020. `doi:10.3389/fphy.2020.567738`

[11] T. Nonnenmacher *et al.*, “Anomalous Beam Transport through Gabor (Plasma) Lens Prototype,”*Appl. Sci.*, vol. 11, p. 4357, 2021. `doi:10.3390/app11104357`

[12] W. Shields, “The Laser-hybrid Accelerator for Radiobiological Applications (LhARA): an update towards the conceptual design”, presented at the IPAC’24, Nashville, TN, USA, May 2024, paper THPR54, this conference.

[13] G. Iadarola *et al.*, “Xsuite: An Integrated Beam Physics Simulation Framework,”in *Proc. HB’23*, Geneva, Switzerland, Oct. 2023, pp. 73–80. `doi:10.18429/JACoW-HB2023-TUA2I1`

[14] A. Abramov *et al.*, “Development of Collimation Simulations for the FCC-ee”, in *Proc. IPAC’22*, Bangkok, Thailand, Jun. 2022, pp. 1718–1721. `doi:10.18429/JACoW-IPAC2022-WEPOST016`

[15] Collimasim, `https://gitlab.cern.ch/anabramo/collimasim`