ISBN: 978-3-95450-184-7 doi:10.18429/JACoW-IPAC2018-THPAK025

RECENT DEVELOPMENTS IN BEAM DELIVERY SIMULATION -BDSIM*

L. J. Nevay†, A. Abramov, S. T. Boogert, H. Garcia-Morales,
S. M. Gibson, W. Shields, S. Walker
John Adams Institute at Royal Holloway, University of London, Egham, TW20 0EX, UK
J. Snuverink, Paul Scherrer Institut, 5232 Villigen PSI, Switzerland

Abstract

Beam Delivery Simulation (BDSIM) is a program to seamlessly simulate the passage of particles in an accelerator, the surrounding environment and detectors. It uses a suite of high energy physics software including Geant4, CLHEP and ROOT to create a 3D model from an optical description of an accelerator and simulate the interaction of particles with matter as well as the production of secondaries. BDSIM is used to simulate energy deposition and charged particle backgrounds in a variety of accelerators worldwide. The latest developments are presented including low-energy tracking extension, more detailed geometry, support for ion beams and improved magnetic fields. A new analysis suite that allows scalable event by event analysis is described for advanced analysis such as the trace back of energy deposition to primary particle impacts.

INTRODUCTION

When operating an accelerator, it is common to simulate and predict the possible beam loss throughout for safe operation; to minimise radioactivation; and to reduce possible backgrounds to experiments. Beam loss simulations are typically performed using multiple specialised codes resulting in simplifications at various stages and translation of data. Whilst a successful approach [1], a holistic approach is required for the most accurate simulation. BDSIM has been developed for this purpose [2–4]. It is based on the open source C++ physics library, Geant4 [5], that includes many advanced and validated particle physics models. BDSIM incorporates accelerator tracking routines in combination with a geometry library of common accelerator components to allow a 3D radiation transport model to be created in minutes. Presented here are the latest developments in V1.0.

TRACKING ALGORITHMS

BDSIM provides tracking routines compatible with the Geant4 tracking system that conventionally uses numerical integration. In the case of an accelerator with specific magnetic fields, specific solutions can be used for each field providing improved accuracy and performance. BDSIM provides a variety of routines and these have recently been augmented to include the effect of angled pole faces in dipoles as well as the fringe fields at the entrance and exit.

* Work supported by Science and Technology Research council grant "The John Adams Institute for Accelerator Science" ST/P00203X/1 and Impact Acceleration Account.
† laurie.nevay@rhul.ac.uk

A thick-lens matrix is used for passage through the body of a dipole and a *thin* element (1 μm long) is used to provide the edge based effects. The thin edge element has only a cylinder for geometry and is filled with the same vacuum as the beam pipe. In the Geant4 tracking scheme, it is not possible to enforce only one step in a volume or distinguish between primary and secondary particles. Therefore, to avoid the scenario of multiple kicks applied due to multiple steps in the volume, the integrator acts as a drift if the step length is less than half the thin element length. The Geant4 tracking engine typically takes a large step and then a second small step to meet the surface intersection tolerance. As there is no intervening geometry and only vacuum, this behaviour is highly consistent ensuring the edge effects are applied only once to a particle.

BDSIM constructs dipoles out of many small straight segments as opposed to a truly curved geometry. This allows a much wider variety of geometry to be constructed given the Geant4 primitive solids. However, a consequence of this is that the local coordinates of a volume are not the same as the required curvilinear frame, but are instead with respect to the axis of the solid (the chord between the entrance and exit). A coordinate transform is provided to convert between the local coordinate system of the solids and the curvilinear system using a simplified parallel geometry.

BDSIM was originally developed for high energy colliders where the ultra-relativistic regime was assumed. Recently, all of the tracking algorithms in BDSIM have been extended and validated to support sub-relativistic particles where the relativistic β may be significantly less than 1. Tracking comparison with PTC as part of MADX [6] show excellent agreement. Additionally, full support for the new Dormand Prince integrators in Geant 4.10.4 has been added.

A new scaling parameter has been added to all beam line elements allowing strength scaling to be easily applied in conversion. This is crucial where the beam energy is degraded by, for example, synchrotron radiation, or is increased in an accelerating cavity.

GEOMETRY & FIELDS

BDSIM provides a library of generic components for each component commonly found in an accelerator. This library of generic geometry provides magnet yokes with poles and has been recently extended to include the coils used to power the magnets as well as their curved end pieces. The end pieces are automatically constructed only where required, i.e. in the case of a split magnet end pieces are constructed

ISBN: 978-3-95450-184-7 doi:10.18429/JACoW-IPAC2018-THPAK025

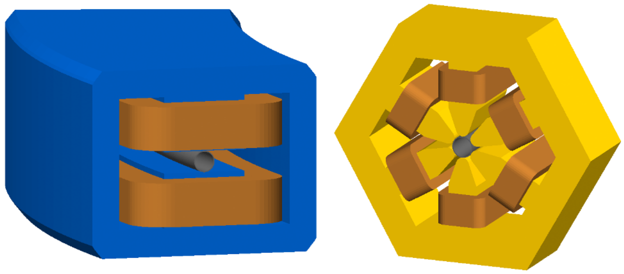

Figure 1: Example dipole and sextupole visualised in BDSIM showing yoke geometry and coils with curved end pieces.

for the incoming and outgoing faces only. If there is insufficient space between any two magnets and the end piece coils may touch, they are omitted to prevent geometrical overlaps that would cause incorrect geometry navigation in Geant4. Furthermore, the dipole geometry was extended to allow *h-style* dipoles as well as the regular *c-style* dipoles. The horizontal to vertical ratio as well as the coil to magnet overall size are controllable allowing the user to achieve close to ideal geometry quickly. Example geometry including poles and coils is shown in Fig. 1.

The BDSIM accelerator tracking integrators are based on a strength parameter, such as 'k1' for a quadrupole, and therefore typically ignore the field. However, a field for each element is provided so that the BDSIM integrators can fall back to a Geant4 numerical integrator in the case of non-paraxial or backwards travelling particles. The user may now select between sets of integrators for customised tracking. The accompanying fields for each accelerator element are calculated from the analytical description of the pure field for that magnet as opposed to a sampled field map. A new field has been introduced to describe a multipole field for the magnet yoke. This approximates the yoke field to the sum of the field from infinitely long current sources at the pole tips. The user may overlay field maps or specific geometry for their application if more detail is required. A sextupole example is shown in Fig. 2. The field is normalised at the pole tip radius of the geometry to that of the field used in the vacuum.

ANALYSIS SUITE

The output from a BDSIM simulation is stored in a ROOT [7] output file. The output has been extended to store the required information for strong recreation including the CLHEP pseudorandom number generator seed states for every event. This allows any individual event or series of events to be recreated in full.

A major change to the output files is the storage of the main simulation is stored event by event. This is a crucial structuring of the output that allows the correct statistical uncertainties to be calculated from the simulation that would not be possible if all 'hits' and trajectories were stored flatly.

The provided analysis utility *robdsim* has been replaced by an new one called *rebdsim* ('ROOT event BDSIM'). This

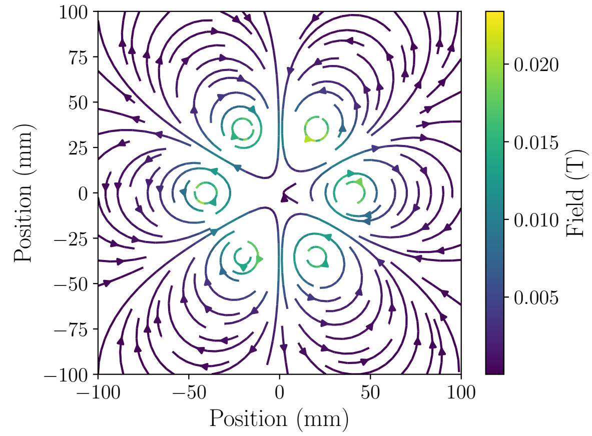

Figure 2: Yoke field from infinite current sources for a sextupole magnet.

accepts a simple text file specifying which analysis to perform and performs an event by event analysis producing a ROOT output file. This tool can create 1-3D histograms on any part of the data stored in the output either event by event or as a simple summation of all events. The statistical uncertainties for each bin are calculated using a numerically stable algorithm that allows scaling to arbitrary sizes without loss of precision. A typical analysis showing particle loss and energy deposition is shown for an example accelerator in Fig. 3.

In large scale simulations, a high throughput computing cluster that runs many independent instances of BDSIM can be used to rapidly increase the simulation scale. ROOT files can be easily 'chained' together to behave as one large file for analysis. However, an analysis of a large data set on the order of terabytes (e.g. from 10^6 events with $\sim 10^5$secondaries per event) may be prohibitively slow. To circumvent this, a separate tool *rebdsimCombine* allows the combination of the output from *rebdsim* even when each contains analysis of different numbers of events. The statistical uncertainties are calculated and the combination is again done in a numerically stable way equivalent to a single analysis.

The histograms are created through the ROOT `TTree::Draw` interface. This allows a 'selection', which is an expression to be used as a weighting. The expression can be numerical weight such as a variable also stored in the data, a Boolean expression or a combination of both. For example:

```
Eloss.S  Eloss.energy*(PrimaryFirstHit.S>340&&
    PrimaryFirstHit.S<350)
```

is the syntax used to histogram the energy loss S location with the weight of the deposition energy. Additionally, it's weighted by a Boolean expression that is equal to 1 only where the 'primary first hit' (the first primary trajectory point that is associated with a physics process and not transportation) is located between S = 340 m and 350 m. This demonstrates the ability to identify the energy deposition throughout an accelerator from losses in a particular location

ISBN: 978-3-95450-184-7 doi:10.18429/JACoW-IPAC2018-THPAK025

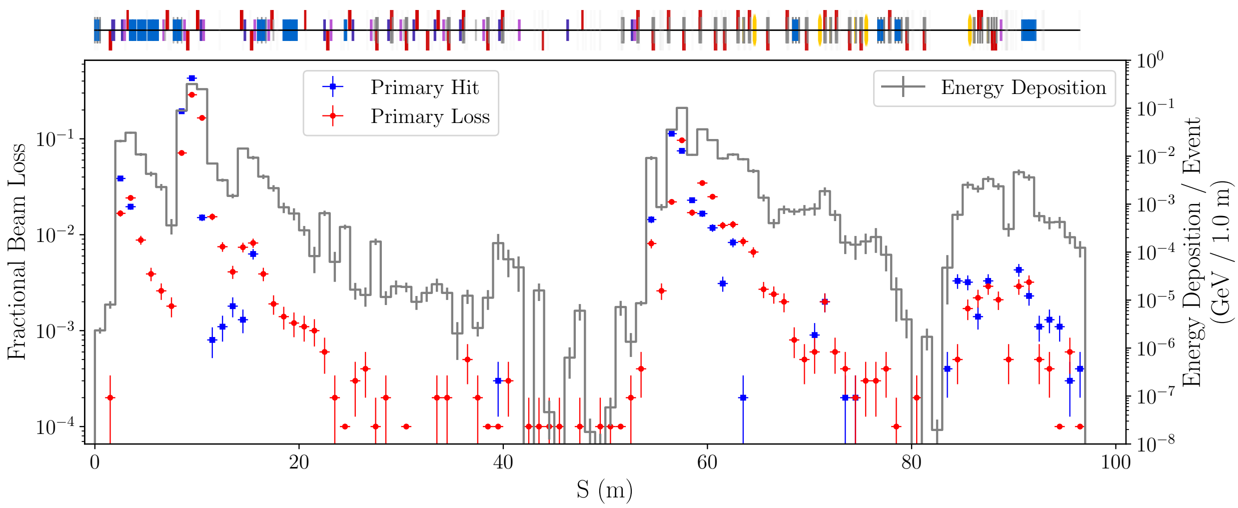

Figure 3: Example beam loss simulation of single pass halo in the Accelerator Test Facility 2 at KEK, Japan for demonstration purposes only. The primary impact point and absorption points are shown as a fraction of 10,000 primary particles as well as the energy deposition they caused. The machine components are shown at the top of the figure.

and provides powerful insight. For example, primary phase space leading to muons above 10 GeV reaching a certain element in the machine is a trivial analysis. Correlating beam losses, energy deposition and secondary particles in certain locations is easily accomplished without writing a complicated analysis. The per event analysis permits easy scaling from simulated rates to realistic rates in an accelerator.

BDSIM has been extended with several options to control the level of output data recorded. By default no trajectories are recorded as this would produce a large amount of data. The user may select which trajectories to store ranging from all particles to specific particles in a specific region and energy range. Trajectory linkages can be stored where, if a trajectory meets the criteria to be stored, the parent tracks all the way to the primary are also stored. Extra parameters such as the charge and rigidity of particles passing recording planes ('samplers') at the end of each element may also be stored.

The C++ classes used to store the output information in the ROOT files are compiled into a shared library that can be used to load and analyse the data in both the ROOT interpreter and Python using the Python ROOT interface, rootpy [7]. The provided *pybdsim* utility provides simple loading and plotting of data allowing any user-developed analysis.

ION SUPPORT

Geant4 provides the ability to simulate any subatomic particle and a selection of light nuclei. For ions, a single class provides a generic ion that can be used for any ion. BDSIM has been modified to support ion beam generation and physics lists allowing ion beam loss simulations. Options have been introduced to choose whether to store ion atomic number and mass number in the output.

As the tracking routines in BDSIM work for all rigidities and momenta, no modification was required to allow ion tracking in accelerators.

MODEL PREPARATION & VALIDATION

Included with BDSIM are a set of mature Python utilities: *pybdsim*, *pymadx*, *pymad8* and *pytransport*. In particular, *pybdsim* uses the other utilities to provide a standard conversion interface from MADX, MAD8 [6] and Transport [8] accelerator codes as well as standardised comparison plots and analysis. Such utilities provide simple high level ability to convert accelerator models in minutes as well as load, analyse and plot data generated from them.

SUMMARY & OUTLOOK

BDSIM tracking has been thoroughly validated against PTC in MADX. Models can be converted from several formats in minutes to provide a full 3D model with radiation transport of any particle species and BDSIM V1.0 was released in early 2018.

Further developments planned include support for the Geant4 crystal extensions allowing channelling and support for all MADX aperture shapes in collimators. Ongoing extensions to tracking include support for a general R-matrix element as well as RF-cavity fringe field effects. The R-matrix element will allow efficient simulation of insertion devices in light sources and a basic undulator field model will be added for the purposes of secondaries.

BDSIM provides a unique ability to create a radiation simulation from an optical description with minimal effort. The generic models can be easily extended to a much higher degree of detail to accurately predict experimental background and beam loss throughout an accelerator.

ISBN: 978-3-95450-184-7 doi:10.18429/JACoW-IPAC2018-THPAK025

REFERENCES

[1] R. Bruce *et al.*, "Simulations and measurements of beam loss patterns at the CERN Large Hadron Collider", Phys. Rev. Special Topics Accel. & Beams 17 (2014) 081004.

[2] BDSIM website: http://pp.rhul.ac.uk/bdsim

[3] L. J. Nevay *et al.*, "Beam Delivery Simulation: BDSIM Automatic Geant4 Models of Accelerators", IPAC '16, Busan, Korea, May 2016, paper WEPOY046.

[4] I. Agapov *et al.*, "BDSIM: A particle tracking code for accelerator beam-line simulations including particle-matter interactions", Nucl. Instrum. Methods A 606 (2009) 708.

[5] S. Agostinelli *et al.*, "Geant4 - a simulation toolkit", Nucl. Instrum. Methods A 506 (2003) 250-303.

[6] MAD-X website: http://madx.web.cern.ch/madx/

[7] I. Antcheva *et al.*, "ROOT — A C++ framework for petabyte data storage, statistical analysis and visualization", Computer Physics Communications, 180, Issue 12, (2009) 2499-2512.

[8] PSI Graphic Transport Framework by U. Rohrer, based on a CERN-SLAC-FERMILAB version by K.L. Brown et al.