PARTICLE TRACKING IN MATTER DOMINATED BEAM LINES*

T. Roberts[#], K. Beard, Muons, Inc., Batavia IL, USA S. Ahmed, D. Huang, D. M. Kaplan, L. Spentzouris, IIT, Chicago, IL, USA

Abstract

The G4beamline program [1] is a useful and steadily improving tool to quickly and easily model beam lines and experimental equipment without user programming. It has both graphical and command-line user interfaces. Unlike most accelerator physics codes, it easily handles a wide range of materials and fields, being particularly well suited for the study of muon and neutrino facilities. As it is based on the Geant4 toolkit [2], G4beamline includes most of what is known about the interactions of particles with matter. We are continuing the development of G4beamline to facilitate its use by a larger set of beam line and accelerator developers. A major new feature is the calculation of space-charge effects. G4beamline is open source and freely available at:

http://g4beamline.muonsinc.com

INTRODUCTION

As accelerator facilities become more complex and more expensive, accurate and comprehensive simulations of their performance are required long before construction begins. There are many choices and optimizations to be made, as well as new concepts to be explored, so flexible and user-friendly simulation programs become essential to streamline the design process. For future facilities such as muon colliders and neutrino factories, the muon cooling sections demand simulations that accurately compute the interactions of particles in matter, along with associated magnetic and RF fields. The Geant4 toolkit [2] is an excellent choice as the basis of such a program, as it is comprehensive, accurate, and actively supported by a vibrant collaboration. G4beamline [1] was conceived as a user-friendly interface between accelerator physicists and the C++ code of Geant4 to facilitate the rapid evaluation of new concepts and designs by physicists without the burden of C++ programming. An important aspect of G4beamline is that its description of the simulated system is far more comprehensible by physicists than the corresponding Geant4 C++ code would be, and no more complicated than is the system itself.

DESCRIPTION

An obvious aspect of G4beamline is that its user interface has been designed with physicist-users in mind. The system to be simulated is described in a single ASCII file using an object-oriented language specifically designed for this application. Most accelerator physicists can read and understand such a descriptive file without reference to the G4beamline documentation, and can

05 Beam Dynamics and Electromagnetic Fields

learn how to develop their own simulations with minimal effort. Extensive online help is available within the program to assist users in developing their simulations. Figure 1 shows the G4beamline graphical user interface (GUI) screen, including a hyperlinked index and the beginning of its Help text.



Figure 1: The G4beamline GUI Screen.

To facilitate the generation of histograms and plots, the G4beamline distribution includes the HistoRoot program, which provides a user-friendly graphical interface to ROOT [3]. While general ROOT programming can be used to create plots, most users find the interface shown in Figure 2 to be much more usable and efficient.

000	X HistoRoot
<u>F</u> ile <u>P</u> lot	Help
Use the	File menu to open file(s), select an NTuple and Plot type, then enter expressions below.
-NTuple	
- Plot Type	
C 2d His	stogram errorbars: sqrt(sum of weights squared)
C X-Y F	Plot
(no NTuple	selected) Enter expressions below using the fields of the NTuple (listed on left) and C arithmetic functions and operators. A right-click will insert from the list of fields of the NTuple.
	×
	y. weight
	errorbar X:
	errorbar Y:
	The sliders below impose range cuts on the NTuple:
	s1:
	\$2:
	\$3:
	\$4:
	Max Events: ParticleType: EventD:
	Create Plot

Figure 2: The HistoRoot GUI Screen.

^{*} Work supported in part by USDOE STTR Grant DE-FG02-08ER86281.

[#] tjrob@muonsinc.com

The major aspects of G4beamline are:

- Accurate and realistic simulations using the Geant4 toolkit
- A physicist-readable ASCII input file to specify the simulation, with auxiliary files for field maps, etc.
- A rich repertoire of beamline elements that can be combined to define new and customized elements
- A general set of initial beam specifications (including a cosmic ray muon "beam" and external files)
- Input and output of beam tracks using several formats including ASCII and ROOT [3] files, supporting easy interfacing to other programs (e.g., for partitioning of complex systems and for verification of results)
- Automatic tuning of many parameters (RF cavity timing and gradients, bending magnet fields, etc.)
- Included visualization of the simulated device or system using many viewers (OpenGL, VRML, Open Inventor, etc.) – provides a powerful visual check that the system implemented is the system desired, and lets you see tracks and their interactions with materials
- Support for parallel jobs on multiple CPUs
- The HistoRoot program, which makes it easy for non-experts to generate ROOT [3] histograms and plots

The basic structure of a simulation is to first define the beamline elements to be used (magnets, RF cavities, etc.), including their geometry, materials, and local fields. Then these elements are placed into the "world", usually along the nominal beam centerline; each placement can have a position, rotation, and its own field value. Parameters for the element can be defined in the input file or on the command line, so scripting is straightforward. Individual particles can be traced, beam profiles can be generated and displayed, and "virtual detectors" can be used to sample the beam at any point.

The tracking of particles through the simulated system is as accurate and realistic as the Geant4 toolkit implementation. The input file selects any one of the Geant4 physics lists, and can set values for the various Geant4 tracking accuracy parameters. This permits users to make trade-offs between CPU time and simulation accuracy. Similarly, G4beamline permits the specification of magnetic map parameters, permitting a trade-off between memory usage (and the CPU time needed to generate the map) and simulation accuracy.

While G4beamline can make it rather simple to specify a simulation, it cannot substitute for knowledge and experience about the problem domain or about particle tracking simulations in general. It is strongly suggested that visualization be used to verify the geometry of the simulation and the proper tracking of at least a handful of particles. Whenever possible, one should arrange to track through a simple geometry that can be compared to independent results, to make sure that what one thinks is happening actually does occur in the simulation.

SOFTWARE DEVELOPMENT

G4beamline is being developed using modern software development techniques. In particular, our methodology requires that feature documentation be written before the code, and that the documentation is contained within the code, so there is always comprehensive and up-to-date documentation available to users. There are two levels of documentation:

- User documentation describing how to use the code
- Internal documentation describing what the code does and how it works.

The first is intended for users and is contained in Help text within the code to implement always-available online help. The latter is intended only for developers, and is contained in structured comments that the doxygen [4] system converts into hyper-linked HTML. Keeping all documentation within the code makes it easy for developers to keep everything in sync; automated tools format it for presentation to users (e.g., the User's Guide).

RECENT FEATURES

Several recently added features make G4beamline more powerful and more usable by physicists:

- 1. The User's guide has been considerably enhanced and numerous tips and techniques have been discussed for the best performance of simulations and generation of results.
- 2. More Geant4 objects have been implemented, expanding the repertoire of G4beamline.
- 3. The Geant4 General Particle Source can now be used to generate beams.
- 4. The build structure has been completely revised, improving the installers and making it *much* easier for users to build G4beamline and add new code.
- 5. Visualization has been enhanced by adding the feature to generate movies of events showing the time evolution of tracks.
- 6. Space Charge: Current design projections for muon colliders indicate that space charge may be a problem in the final stage of cooling. The computation uses Lienard-Wiechert potentials, and its CPU time scales as the square of the number of macroparticles used. This includes radical revisions to the code of the Geant4 RunManager and EventManager to track particles in parallel.
- 7. Low energy physics: new physics lists are available that include the Geant4 low-energy electromagnetic models.
- Automated parallelization: Computer farms and multi-CPU systems are now common, and MPI [5] is used to take advantage of parallelism – a single set of output files contains the computations of multiple processes running on multiple CPUs.

05 Beam Dynamics and Electromagnetic Fields D06 Code Developments and Simulation Techniques



Figure 3: Comparison of G4beamline's space charge computation to a calculation in Reiser's textbook [6], showing transverse beam size as a function of propagation distance. The initial beam is a uniform cylinder, and would be focused to a point at Z=1 without space charge. The solid line is the calculation for a continuous charge density; the points are for computations using 100 (black), 1,000 (red), and 10,000 (blue) macro-particles.

SPACE CHARGE VALIDATION

Figure 3 shows a comparison of the G4beamline space charge computation to a calculation from Reiser's textbook [6] of how space charge forces affect the beam size as it propagates. The initial beam is a uniform cylinder with converging trajectories; the space charge defocusing is evident, as without it the beam would be focused to a point at Z=1. The values plotted are appropriately scaled, so the plot is unchanged for variations in beam momentum, bunch radius, total charge, and particle type. The points plotted are for the 95th percentile of macro-particle radii, which is why they are initially below the line. At large Z, some macro-particles have experienced an abnormally large interaction with other macro-particles, so the points are above the line, and higher-charge macro-particles are affected more strongly. Overall, the agreement with the calculation is quite good, but requires a large number of macro-particles for accurate agreement with the analytic result.

FUTURE PHYSICS INTERESTS

We are currently developing or planning the development of the following new features:

- 1. Improving the space charge computation: the current computation scales as N², which makes it extremely CPU intensive; we intend to implement a much faster algorithm based on Fast Fourier Transforms that will significantly reduce the CPU usage.
- 2. Collective effects in matter: at the end of a muon cooling channel, very high beam densities are expected, and the collective effects between the beam and the material in the absorbers are expected

05 Beam Dynamics and Electromagnetic Fields

D06 Code Developments and Simulation Techniques

to be important. This includes effects due to the polarization of the material, as well as the plasma generated by the ionization of the material by the beam. This requires an interdisciplinary approach that includes atomic, accelerator, and plasma physics.

3. Polarization physics: Many new muon facilities including muon colliders can benefit from using polarized muons. This requires modeling of polarization especially during the interaction with matter in a cooling channel.

SUMMARY

G4beamline is a highly flexible and user-friendly program for simulating beamlines both with and without matter. We are continually advancing the applicability of the code. In addition to the primary use of investigating many aspects of muon cooling for a muon collider or neutrino factory, applications include: target hall and test beam design evaluation, feasibility studies for new facilities and experiments, cosmic-ray muon tomography and detector design. Our commitment is to provide support; current funding for new features and our growing user base ensure that G4beamline will remain useful for the foreseeable future.

REFERENCES

- [1] G4beamline http://g4beamline.muonsinc.com
- [2] Geant4 Toolkit http://geant4.cern.ch
- [3] ROOT http://root.cern.ch
- [4] Doxygen http://www.doxygen.org
- [5] MPI http://www.open-mpi.org
- [6] Reiser, M., Theory and Design of Charged Particle Beams, Wiley, 2008.