G4BEAMLINE PARTICLE TRACKING IN MATTER-DOMINATED BEAM LINES

Thomas J. Roberts, Kevin Beard (Muons, Inc., Batavia, Illinois, USA 60510), Dazhang Huang, Shahid Ahmed, Daniel M. Kaplan, Linda Spentzouris, (Illinois Institute of Technology, Chicago, Illinois, USA 60616)

Abstract

The G4beamline program [1] is a useful and steadily improving tool to quickly and easily model beamlines and experimental equipment without user programming. 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 and matter. We are continuing the development of G4beamline to facilitate its use by a larger set of beam line and accelerator developers. It is open source and freely available at http://g4beamline.muonsinc.com.

INTRODUCTION

As accelerator facilities get 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, and 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 frequent evaluation of new concepts and design changes by physicists, without the complexities of C++ programming. An important aspect of G4beamline is that its description of the simulated system is far more understandable by physicists than the corresponding Geant4 / C++ code would be.

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 description file without reference to the G4beamline documentation, and can 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 an index and the beginning of its Help text.

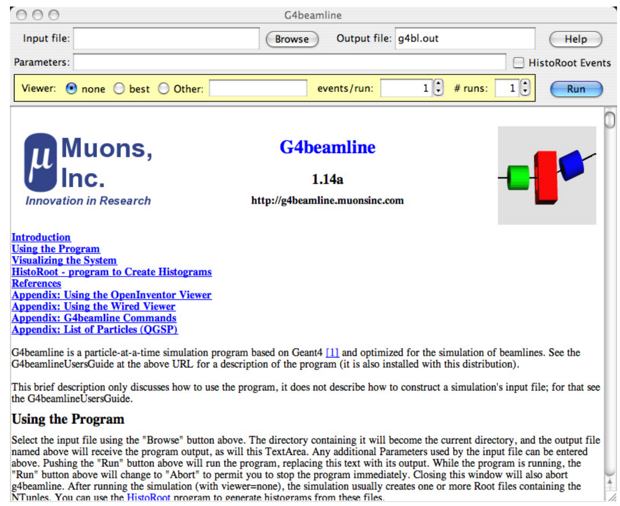

Figure 1. The G4beamline GUI screen.

To facilitate the generation of histograms and plots, the G4beamline distribution includes the historoot program that 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 more usable and efficient.

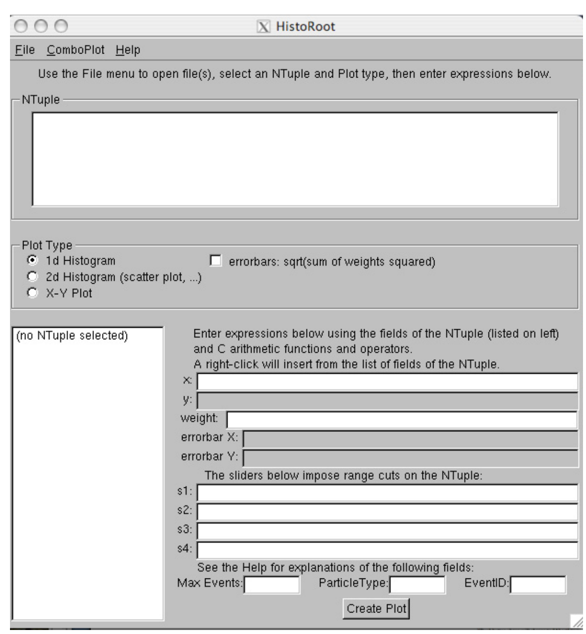

Figure 2. The historoot GUI screen.

Supported in part by DOE STTR grant DE-FG02-06ER86281.

The major aspects of G4beamline are:

- Accurate and realistic simulations using the Geant4 toolkit
- A physicist-readable ASCII 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)
- Beam tracks can be input and output using several formats including ASCII and Root [3] files – this permits easy interfacing to other programs (e.g. partitioning complex systems, verification of results)
- Many parameters can be automatically tuned (RF cavity timing and gradients, bending magnet fields, etc.)
- Visualization of the system is included, using many viewers (OpenGL, VRML, Open Inventor, etc.)
- 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 elements 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 implements. The input file selects from any of the Geant4 physics use cases, 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 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. Like all computer programs, G4beamline is prone to “garbage in, garbage out”, especially when used by unskilled users. It is strongly suggested that visualization be used to verify the geometry of the simulation and that a handful of particles are tracked properly through it. 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 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. By keeping all documentation in the code it is easier for developers to keep them 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 usable by physicists:

1. An event-list interface between historoot and G4beamline. The user can select an outlier event in a histogram or plot and easily re-run G4beamline with visualization to see what that particular event did.
2. More Geant4 objects have been implemented, expanding the internal repertoire of G4beamline.
3. A direct interface to the NIST material database gives users the ability to use many common materials without manually looking up their properties.
4. Rudimentary track fitting is implemented, intended for estimating required resolutions in proposals. This is not robust enough to serve as an analysis of real experiments, but combined with the inherent flexibility of G4beamline this becomes a powerful tool for the early design of experiments.
5. Do loops and if-then-else have been implemented to simplify repetitive tasks and input file options.

NEW FEATURES

We are currently developing the following new features:

1. Space Charge. Current expectations for muon colliders indicate that space charge may be a problem in the final stages of cooling. We are implementing computations of space charge based on several existing programs (Orbit [5], TREDI [6], Parmela [7]). We intend to extend this to computations inside absorbers. This includes radical revisions to the Geant4 G4RunManager and physics processes to track particles in parallel.
2. Polarization physics. Many new muon facilities, up to and including a muon collider, can benefit from the ability to use polarized muons. This requires modeling the polarization, especially in the interactions with matter in a cooling channel.

3. Very low energy physics. We are interested in simulating the inner workings of advanced detector systems, atomic traps, and the details of surface effects. This requires accurate modeling at the eV level and perhaps below.
4. Automated parallelization. Computer farms and multi-CPU systems are now common, and we intend to enhance G4beamline to automatically take advantage of such parallelism. At present the user must do this manually.

APPLICATIONS

At present, more than a dozen research groups are using G4beamline. In addition to the primary use of investigating many aspects of muon cooling for a muon collider or neutrino factory, applications include: analyzing test beamlines and target halls, feasibility studies for several new experiments and facilities, cosmic-ray muon tomography, and detector design. Out of these, we have selected two examples to display: the Mu2E experiment's beamline and detector [8], and the trapping of antiprotons in an atomic trap [9].

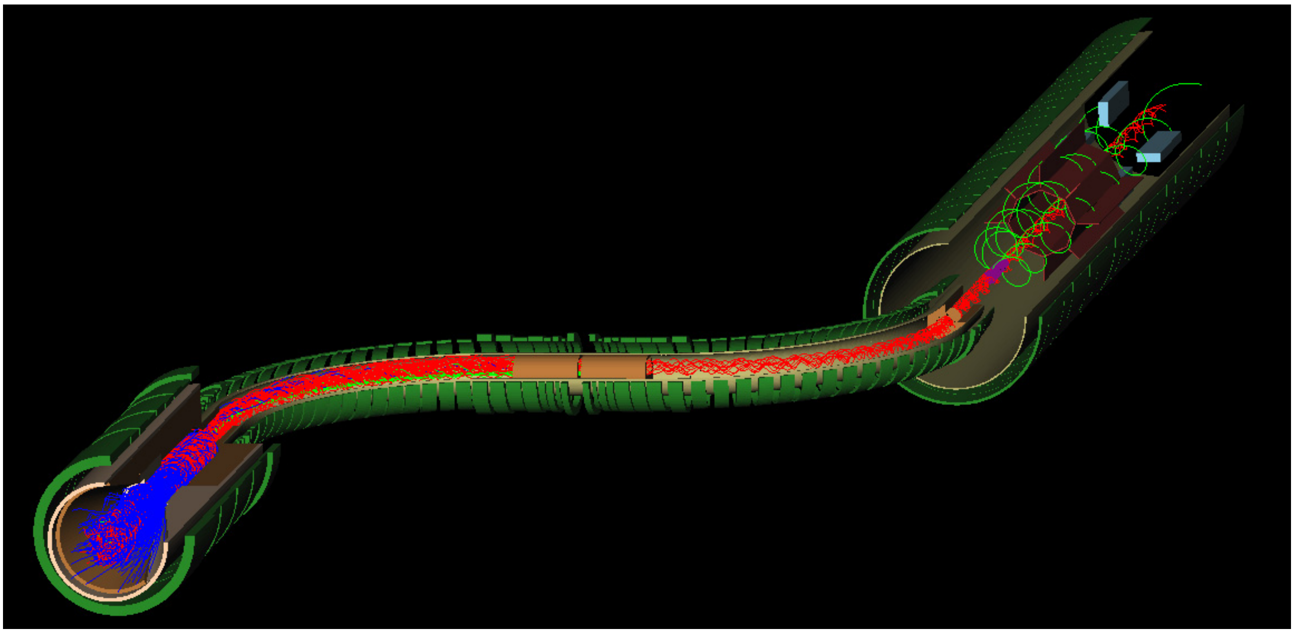

Figure 3. The Mu2E primary target, muon transport, stopping target, spectrometer, and calorimeter.

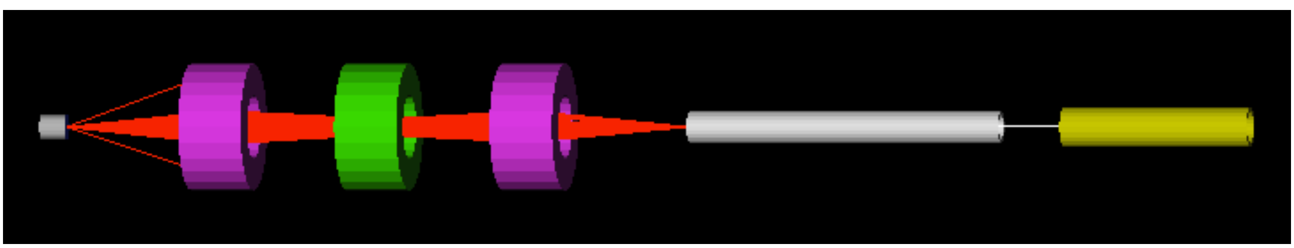

Figure 4. Antiprotons (1 GeV/c) incident on a lead degrader, a quad triplet (FDF in this plane), a pulsed linac antiproton decelerator, and the HiPat atomic trap (idealized).

SUMMARY AND FUTURE

G4beamline is a highly flexible and user-friendly program for simulating beamlines both with and without matter. Our commitment to supporting it, our current funding for new features, and our growing user base ensure that it 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] http://neutrons.ornl.gov/APGroup/Codes/orbit.htm
[6] TREDI – http://www.tredi.enea.it
[7] Parmela
[8] Mu2E – http://mu2e.fnal.gov
[9] Anti-hydrogen gravity experiment – http://capp.iit.edu/hep/pbar