Measuring the $e \rightarrow \mu^+\mu^-$ Cross-Section Using the EUDET JRA1 Beam-Telescope, a Simulation Study (MuGen Simulation User Guide)

A. Vincer , D. Cussans , D. Newbold , H. Heath[*]

January 6, 2011

Abstract

The feasibility of measuring the $e \rightarrow \mu^+\mu^-$ cross-section using the EUDET JRA1 telescope was investigated. A model of the EUDET telescope planes interspaced with sheets of Tungsten was written in GDML and the production rate and detector acceptance was modelled in Geant-4.

[*]H. H. Wills Physics Laboratory, University of Bristol, BS8 1TL, United Kingdom

1 Introduction

There is some interest in confirming the $e \rightarrow \mu^{+}\mu^{-}$ cross-section modelled by Geant-4. (The reaction proceeds by Bremsstrahlung followed by muon pair production). It has been proposed to measure the cross-section using a beam-telescope a production target upstream and plates of absorber in between the telescope planes to give muon-electron separation. A simple simulation was constructed based on one of the Geant-4 examples. The geometry is described using the GDML language.

2 Obtaining the Code

The code is based on the Geant-4 exampleN01 and is released under the same licence conditions. It is stored in a "subversion" repository accessible at https://svn.phy.bris.ac.uk/svn/uob-hep-eMuMu/trunk/ . The code can be "checked out" using the command:

```
svn co https://svn.phy.bris.ac.uk/svn/uob-hep-eMuMu/trunk eMuMu
```

Username = anonymous , password = anonymous for read only access. Please contact David.Cussans@bristol.ac.uk if you would like to commit code to the repository.

3 Prerequisites for compiling the code

The simulation requires several libraries to be present for successful compilation and execution. These are all freely available and listed below along with their location at the time of writing.

Name	Location
Geant4	http://geant4.cern.ch/
CLHEP	http://proj-clhep.web.cern.ch/proj-clhep/
Xerces	http://xerces.apache.org/xerces-c/
ROOT data analysis framework	http://root.cern.ch/drupal/
Doxygen (optional)	http://www.stack.nl/~dimitri/doxygen/
OpenGL (optional)	(depends on platform)
Qt4 (optional)	http://qt.nokia.com/products/

Unless more specific instructions are given below, follow the instructions provided with each library to compile and install it. If compiling on Linux and it is not possible to install to the system directories, or if you don't want to, it is simpler to install everything to the same folder, e.g. ~/g4-install

Please note that Geant4 requires CLHEP and Xerces so these must be compiled and installed before compiling Geant4. Also note that the OpenGL and Qt4 libraries are only required if certain visualisation drivers provided by Geant4 are desired. A list of visualisation drivers, accompanied by reasons for and against using them, may be found here. http://geant4.cern.ch/UserDocumentation/UsersGuides/ForApplicationDeveloper/BackupVersions/-V9.2.Rev00/html/ch08.html

OpenGL and Qt4, if needed, should be present before compiling Geant4.

When compiling Geant4 it is important to enable GDML support and all of the visualisation drivers you require. After successful compilation a script will be created in your-geant4-directory/.config/bin/your-system-type/ beginning with the word env. The extension depends on your system. The purpose of this script is to set the environment variables required for compilation and must be run at the beginning of each terminal session. Linux users could add this to their ~/.bashrc file.

ROOT can be compiled and installed at any time but is required to compile the simulation. Doxygen is only required to rebuild the documentation files from the simulation source. A Doxyfile is included. The documentation can be viewed using any browser by opening doxygen/html/index.html

4 Compiling the code

4.1 Environment Variables

The environment variables for ROOT can be set up using the thisroot.{sh,csh} script and Geant can be set up using the env.{sh,csh} script.

4.2 Compilation

Type make to compile changes to the code or make clean && make to do a complete rebuild. The resulting program, named MuGen, will be placed in the folder specified by the G4APPDIR environment variable. Note that the GNUmakefile may require editing if the above libraries were installed in a non-standard location.

5 Configuring the simulation

- **include/config.hh** contains a few simulation constants. These may be changed as required, but please note that PLANE_COUNT must match the number of planes specified in the geometry. The simulation must be recompiled if these are changed.

- **config.mac** is a list of commands to be executed once the simulation has been initialised. In addition to the standard Geant4 commands, the commands detailed in the Custom commands section may also be used.

- **vis.mac** is a list of visualisation specific commands to be executed if visualisation is enabled. See the Running the simulation section for more information.

6 Custom commands

Several commands have been created to configure the energy and distribution of incident particles. These may be used in config.mac and at a session prompt. See Running the simulation for more information.

Command	Notes
/gun/energy_MeV (value)	Set electron energy to (value) MeV
/gun/target_area_mm (width) (height) 0	Fire particles onto an area of (width) x (height) mm centered on the z axis
/gun/target_spot_mm (x) (y) 0	Fire particles onto the spot (x) (y) mm
/gun/spawn_z_mm (z)	Fire particles from a distance of (z) mm from the origin

7 Defining geometry

The geometry for the simulation is fully described in an XML based format called GDML. See http://lcgapp.cern.ch/project/simu/framework/GDML/doc/GDMLmanual.pdf for the user guide. The file detector.gdml is used for this purpose. Note that most attributes of the geometry can be altered by changing one or more of the quantity tags in the define tag at the beginning of this file. For example, the quantity target_x controls the size of the target in the x axis. Note that the default unit of length is mm, which is also true of Geant4.

8 Running the simulation

The simulation is run with the command $G4APPDIR/MuGen The following switches may be used to configure simulation behaviour. Note that switches 1 and 2 are interchangeable.

Switch 1	Switch 2	Meaning
-h	--help	Show the help message and exit. This also happens if an unrecognised switch is provided.
-c	--config	Specify an alternative path to config.mac
-d	--detector	Specify an alternative path to detector.gdml
-o	--output	Specify an alternative path to out.root
-s	--session	Start a session (interactive prompt) after simulation
-v	--visualisation	Enable visualisation

For example, to use a different configuration one could use $G4APPDIR/MuGen –c my_config.mac or, equivalently, $G4APPDIR/MuGen --config my_config.mac It's often useful to start a session when using visualisation, otherwise the program exits as soon as all particles have been simulated. $G4APPDIR/MuGen –s –v will do this.

9 Interpreting the results

Many histograms are written to the output ROOT file, but there are four distinct groups. Please note that incident electrons travel in the –z direction of a right-handed coordinate system.

Histograms that end in **chi2** hold information about track fitting. Within this group are histograms for pairs of particles (particle / anti-particle pair to be precise) and single particles. There are also histograms for tracks to pass through the first three, last three and all six detector planes, referred to as upstream, downstream and full respectively. For example, the histogram **e_minus_downstream_chi2** contains data on electrons to pass through the last three detector planes. The histogram **muon_pair_full_chi2** contains data for mu-/mu+ pairs where both tracks passed through all six detector planes. In all histograms there are entries for chi square and the energy of the track in MeV at the last plane it passed through.

Histograms that begin with **exit** give details about single particles and pairs of particles leaving the target. For single particles: dx_mm and dy_mm are the displacements in mm, in x and y respectively, of the particle from the beam at the point of exit. px and py are the components, in x and y respectively, of the unit momentum of the particle at the point of exit. For pairs of particles, consider electron/positron pairs as an example. phi_e is the opening angle between the particles, in radians. phi_e_e_3D is the angle between the average momentum of the particles and the beam direction. phi_e_e_xz is the projection of phi_e_e_3D in the xz plane. For muon pairs, the quantities are the same except e is replaced with mu i.e. phi_mu is the opening angle of the muon pair.

Information about particles hitting the detector plane is stored in histograms beginning with **plane**. The detector planes are numbered from zero, ascending downstream. Hence, **plane_2_single_mu+** gives details about single anti-muons to hit the third detector plane. E_MeV is the energy of the particle at the point of impact and not the energy deposited into the detector. x_mm and y_mm are the coordinates of impact. theta is the angle between particle momentum and beam momentum at the point of impact and theta_xz is the projection of theta into the xz plane.

Those histograms prefixed with **shower** detail the EM shower in the target. Note that, to save disk space, electrons, positrons and gammas with energy less than four times the muon rest mass are not stored. This value is defined as SHOWER_ENERGY_CUT in include/config.hh and may be altered. Energy in MeV and position of creation in z (called z_mm) are stored for each particle type. Note that z_mm is in world coordinates, i.e. the position of the target affects the results. Interpretation is greatly simplified if the target (of thickness T) is placed at z=T/2. z_mm is then identically equal to the depth of the creation point in the target.