Simulation Guide

Sarah Choate

09/18/2022

1 Introduction

This is a guide for running simulations on both stand alone Genie as well as a full detector simulation using art. The main focus of this project was to generate ν_τ simulations and compare with ν_μ and ν_e simulations. Therefore the main focus of this document will be generating ν_τ simulations with brief asides on how to generalize for ν_e or ν_μ simulations. The goal for both stand alone Genie and the full detector simulation is to produce exclusively charge current (CC) events. Due to the nature of the files produced from the full detector simulation, it is difficult to look at the particles produced and know if only CC events were produced. It is easier to tell what type of event was produced with stand alone Genie. For CC events, every event should have a charged lepton based on which flavor neutrino interacted (τ^-, e^- or μ^-) and its decay products, which can be found with their probabilities through the Particle Data Group (PDG). Throughout this document it will be assumed that we are only discussing the generation of CC events and that any commands discussed result in the production of only CC events.

The initial environment setup is the same across both methods. Begin by logging onto Fermilab (or NICADD). Setup the environment using the command: `source /cvmfs/dune.opensciencegrid.org/products/dune/setup_dune.sh`. Then setup the most recent version of dunesw. The most recent version can be found using the command **ups list -aK+ dunesw** and then most recent version can be setup using the command **setup dunesw $VERSION -q $QUALIFIER**, be sure the qualifier is “e20:prof”. Additional references are included in the readme of the github.

2 Stand Alone Genie

This process will include the initial neutrino interaction with the argon nucleus such that a charged lepton is produced, in this case a charged τ, that is then decayed according to the PDG.

2.1 Command for Running

In general, the command for running Genie is:

gevgen -n {number of events} -p {pdg of interacting particle} -t {pdg for target nucleus} -e {neutrino energy} -f {flux file} –cross-sections {cross section file} –event-generator-list "{type of events to generate}"

For ν_τ the pdg is 16 and for the argon nucleus the pdg is 1000180400. The energy for the neutrino is 0,100 and the type of events we want to generated are CC events. Then, specifically for the ν_τ, the command is:

gevgen -n {number of events} -p 16 -t 1000180400 -e 0,100 -f `/path/to/file/flux_dune_neutrino_FD.root`**,nutau_fluxosc –cross-sections** `/path/to/file/gxspl-FNALsmall.xml` **–event-generator-list "CC"**

Here, flux_dune_neutrino_FD.root is the root file which contains various flux data, nutau_fluxosc is the specific file contained in the root file used for this ν_τ simulation, but it also contains information for both ν_e and ν_μ fluxes. gxspl-FNALsmall.xml contains cross section information and can be used for ν_e or ν_μ as is. Neither of these files are required to run stand alone Genie, as if they are not included Genie will simply calculate the values as part of the simulation, but it will make the simulation take longer to run.

2.2 Generating a Usable Root File

The root file that is initially generated does not necessarily include all the information required for analysis of the simulation, it should be called something like gntp.0.gst.root. The type of file that is more likely to be useful is a rootracker format. The way to convert the file is through the command:

gntpc -i {input filename} -f rootracker

This will produce a file which is called something like gntp.0.gtrac.root which is more appropriate for analysis.

2.3 Running for ν_μ or ν_e

Looking at the general command, for ν_μ the pdg is 14 and for ν_e the pdg is 11. The same flux file can be used but rather than calling nutau_fluxosc, for ν_μ it would be numu_fluxosc and for ν_e it would be nue_fluxosc. The cross section file remains the same.

3 Full Detector Simulation

In order to produce a full simulation, there are three steps. First, the initial interactions are created using Genie, then the resulting particles are propagated through Geant allowing for more interactions, and finally the detector geometry and effects are added with detsim. These steps are all accomplished using various fcl files and the lar command. Full help and additional features not discussed here for the lar command can be found using **lar -h** in terminal.

3.1 FCL Files

This stands for Fermilab Hierarchical Configuration Language (FHiCL). These are files which contain information like particle types and detector geometry. They can be edited in order to utilize specific types of geometry or specific interaction types. In order to check what parameters are being applied to the simulation, the command **fhicl-dump name.fcl** can be used. This will print out all the information used in the fcl file without having to go through every included fcl file. For this project, it was necessary to edit fcl files as, by default, they produce both NC and CC events. This was problematic since CC events are much less likely than NC events, so the amount of events that would have needed to be generated in order to get a reasonable about of CC events was too much.

3.2 Genie Step

Running the Genie simulation is conceptually the same as running stand alone Genie except here we use larsoft rather than gevgen. In order to run Genie, a fcl file is required. Again, this can be a default fcl file or one that has been edited. For this project, we used an edited file. The command to run Genie with art in general is:

lar -n {number of events to generate} -c name-of-fcl-file.fcl

This will generate a file with the naming scheme name-of-fcl-file-used_gen.root

3.3 Geant Step

Once the initial events and interactions have been created with Genie, they are propagated using Geant 4. An additional fcl file is required to run this step called standard_g4_dune10kt.fcl which will not be edited. The art command to run is:

lar -n {number of events} -c standard_g4_dune10kt.fcl name-of-genie-file_gen.root

which will produce a file with the naming scheme name-of-fcl-file_gen_g4.root.

3.4 Detsim Setp

Finally, the detector effets are added through the detsim step. Again, an additional fcl file is required to run this step called standard_detsim_dune10kt.fcl which will not be edited. The art command to run is:

lar -n {number of events} -c standard_detsim_dunt10kt.fcl name-of-g4-file_gen_g4.root

which will produce a file with the naming scheme name-of-fcl-file_gen_g4_detsim.root. This is the file which is used for analysis.