Nieves GENIE code Summary Joe Johnston

This document summarizes the changes made to GENIE to implement a model for CCQE events with RPA effects as developed by Nieves et al. The code has three main parts:

- Local Fermi gas model
- Andrew Furmanski's code to generate quasielastic events
- Nieves' xsec formulation

The following is a summary of the purpose of each of these parts. See the last section of this document for a complete list of files that were added and changed in order to implement Nieves' CCQE model.

Local Fermi Gas Model

Nieves uses a local Fermi gas model, where the Fermi momentum used to generate a nucleon momentum depends on the nucleon's position inside the nucleus. See the document provided for the LFG code review for more information. All following plots are made using the LFG model, unless otherwise specified.

Andrew Furmanski's Quasielastic Event Generation Code

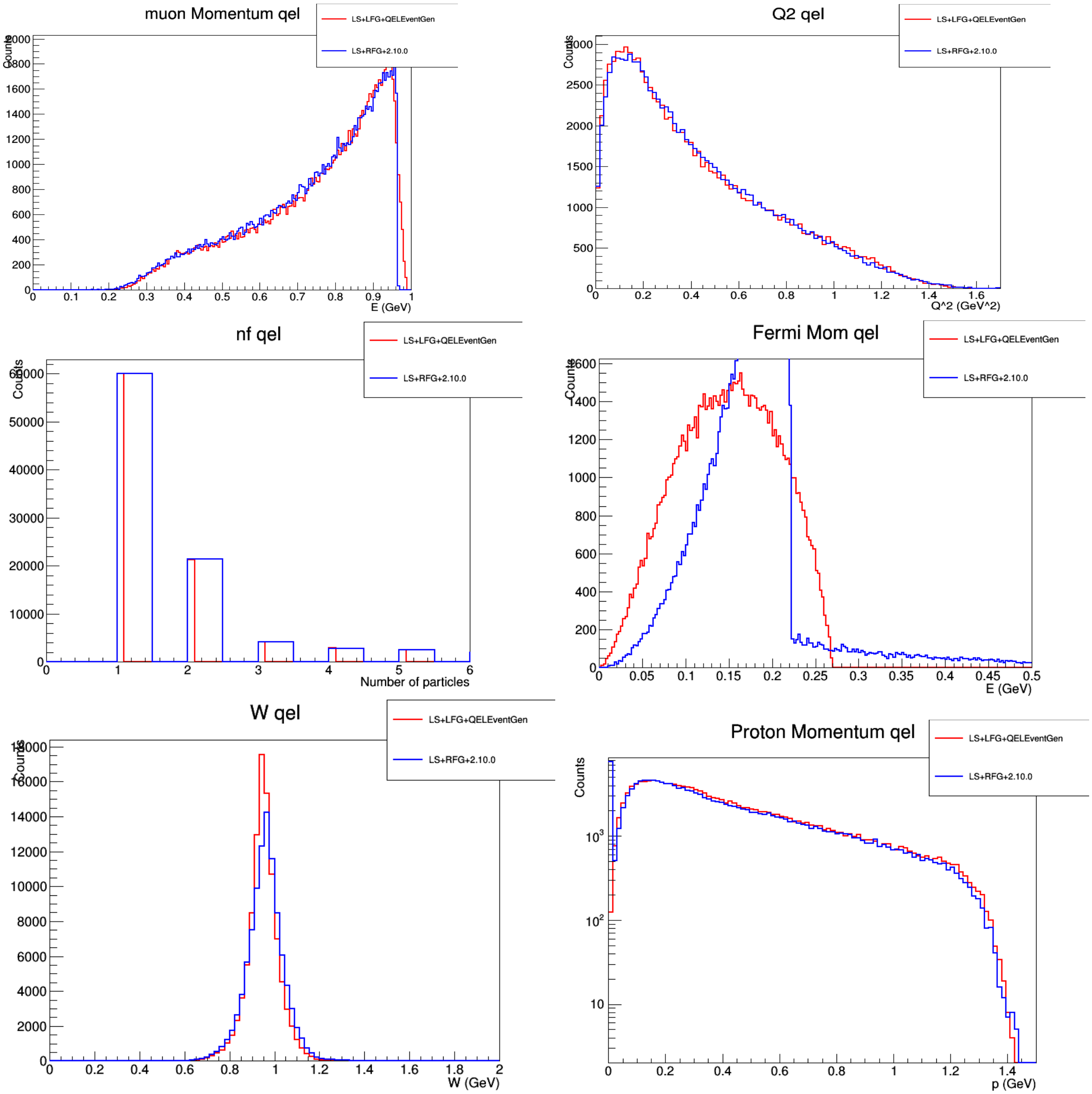
Andrew found a bug in GENIE that caused events to be weighted incorrectly. (See Andy's presentation for more details, including the equations that he uses to calculate the cross section). GENIE previously used the following four classes to generate QEL kinematics:

- FermiMover: Used the selected nuclear model in order to generate an initial nucleon.
- QELKinematicsGenerator: Boosted to the nucleon rest frame, then selected a Q2 value by using the rejection method with the LwlynSmith cross section equations for dSigma/dQ2. The allowed Q2 value depended on the initial nucleon momentum, so events were weighted incorrectly.
- QELPrimaryLeptonGenerator: Used the Q2 value and initial nucleon to generate an outgoing lepton.
- QELHadronicSystemGenerator: Generated the outgoing nucleon kinematics.

In order to fix the bug, Andy used the method he had put into NEUT for T2K. The four classes above are replaced by one QELEventGenerator class. This class generates all kinematics before calculating the cross section. Specifically, it generates an initial nucleon momentum, theta, phi, and binding energy, as well as the com frame outgoing lepton theta and phi. The outgoing nucleon kinematics can then be calculated and stored. The xsec method is then called, and returns a cross section differential in initial nucleon momentum, theta, phi, and binding energy, and com frame outgoing lepton theta and phi. Then the rejection method is then used to determine whether the generated kinematics should be stored.

Splines are still generated by integrating over Q2.

The plots below were made using Andy's QELEventGenerator class with the modified version of the LwlynSmithQELCCPXSec class (which is modified to be able to return a cross section differential in the initial nucleon kinematics and the outgoing lepton com theta and phi). Note that the relativistic Fermi gas model was used for the 2.10.0 runs.



The QELEventGenerator class is important because it fixes a bug. It is also beneficial when implementing Nieves' QEL formula, because Nieves' equations depend on the initial neutrino and outgoing lepton kinematics.

Nieves QEL Cross Section

The Llewellyn-Smith cross section that was previous in GENIE was of the form:

dSigma/dQ2 = Gfactor * LH

where LH is the contraction of the leptonic and nucleon tensors, and the cross section is calculated in the nucleon rest frame. The reason that we use Nieves' formulation is that he includes RPA effects in the nucleon tensor. That is, RPA effects occur because the nucleon is inside a nucleus, but Nieves determines how RPA effects a single nucleon. Thus it is easy to replace the cross section previously in GENIE by saying:

$$dSigma/dQ2 = Gfactor * LH_{RPA}$$

This dSigma/dQ2 formula is used in order to calculate splines. It is also used if events are generated using the old method of event generation that was already in GENIE.

Note that the formula for the full differential cross section that Andy used was also of the form:

$$\text{Full differential xsec} = \text{GfactorFullDiffXSec} * LH$$

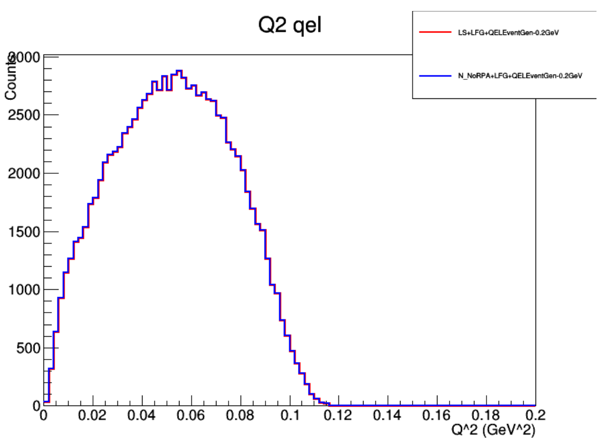
where LH is the same tensor contraction as Llewellyn-Smith. Thus we need only change the Gfactor in order to use Nieves' equations with the QELEventGenerator class. Essentially, the new XSec method follows the same steps as the LwlynSmith class, but uses a different tensor contraction. The other noticeable difference is that Nieves' formulation of the contraction requires that q be in the z direction, and that the initial nucleon be at rest. Thus I boost the kinematics and rotate before calculating the cross section. This full differential cross section formula is used when generating events with the QELEventGenerator class.

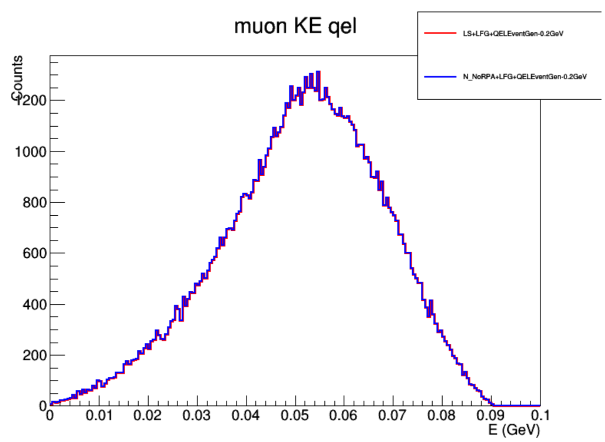
Another effect that Nieves adds is coulomb corrections. These account for the electric field of the nucleus interacting with the outgoing lepton. GENIE did already have code to apply coulomb corrections to the outgoing lepton, but these corrections are not accounted for in the Llewellyn-Smith CCQE model. Coulomb corrections now have an effect when calculating the cross section.

Validation

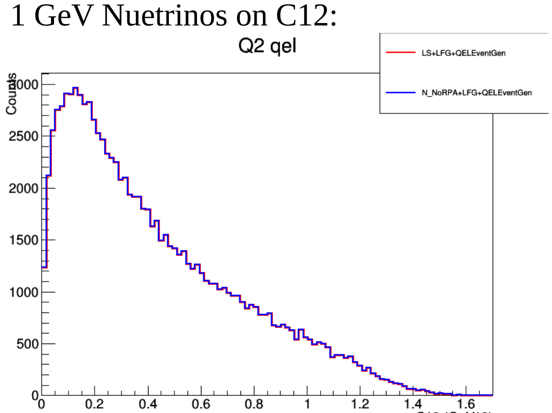
If RPA effects are turned off, Nieves' equations should be identical to LwlynSmith. The first test that I did was to print cross sections for the same kinematics using the Nieves class and LwlynSmith class, and I saw that these did give the same cross section. The following plots also demonstrate that without RPA effects, the two models are identical.

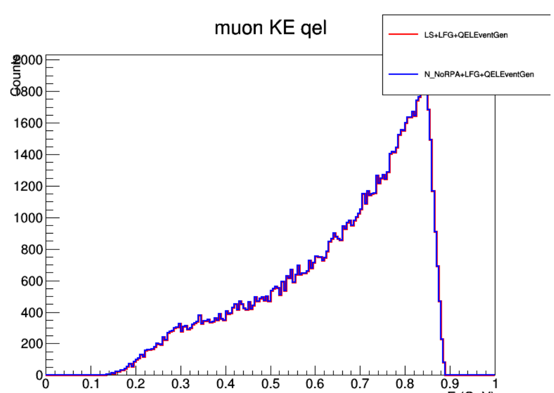
200 MeV Nuetrinos on C12:



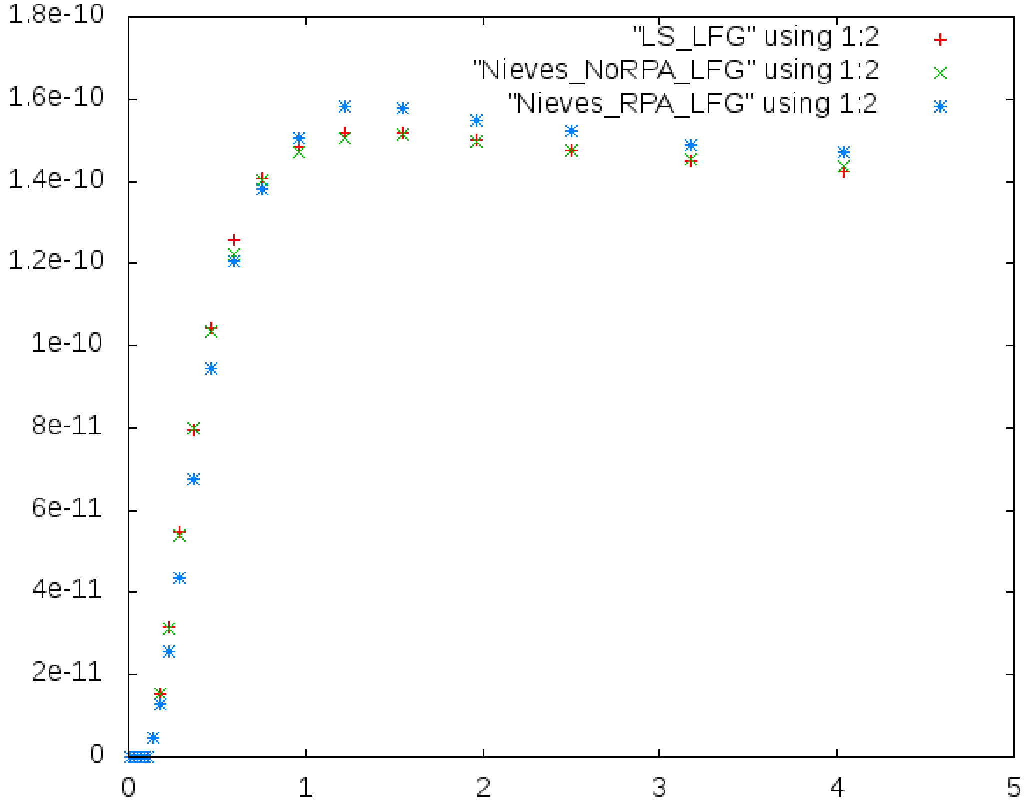


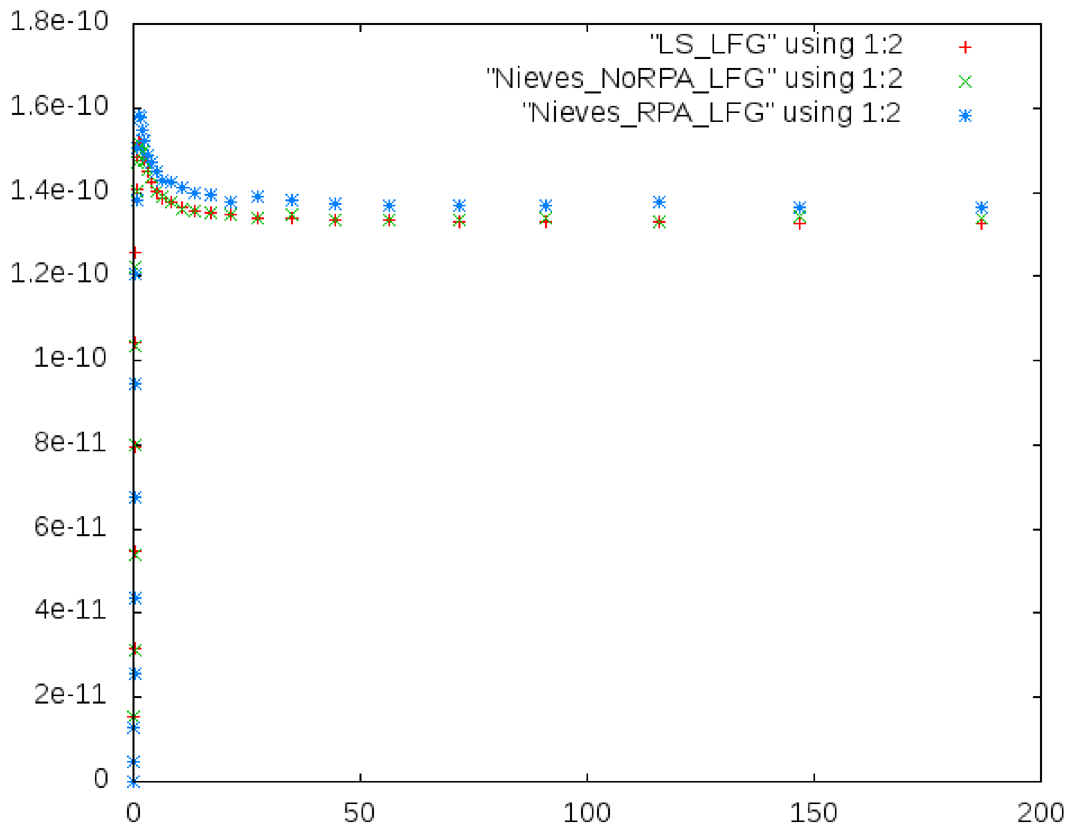
1 GeV Nuetrinos on C12:





Next I generated splines for Andy's version of LwlynSmith, Nieves without RPA, and Nieves with RPA. This spline for a C12 target shows that RPA and Coulomb effects become small at higher neutrino energies.

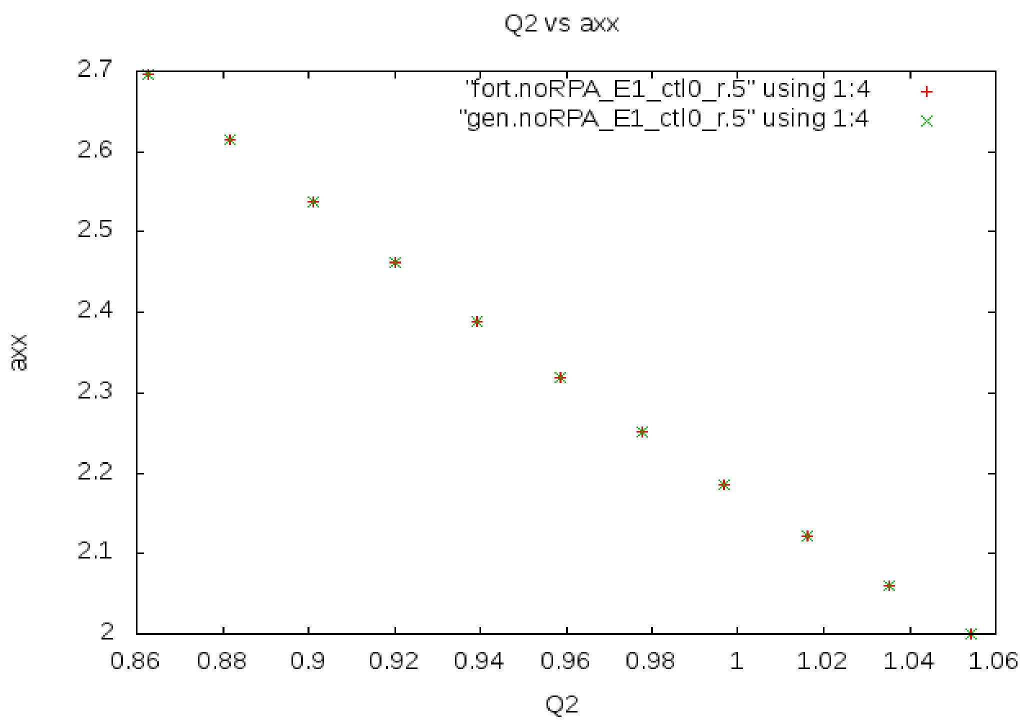




Validation for RPA effects could be done using fortran code that Nieves wrote in order to calculate dSigma/dTlepton/dOmegaLepton. Doing this calculation requires calculating elements of the leptonic tensor, so I wrote code to iterate over kinematics and print the resulting tensor in Nieves' fortran code and in my GENIE code. The following sample plot iterates over q2 for a given radius and letpon angle to show that I use the exact equations that Nieves uses to calculate the nucleon tensor elements. Similar plots were made for all relevant tensor elements.

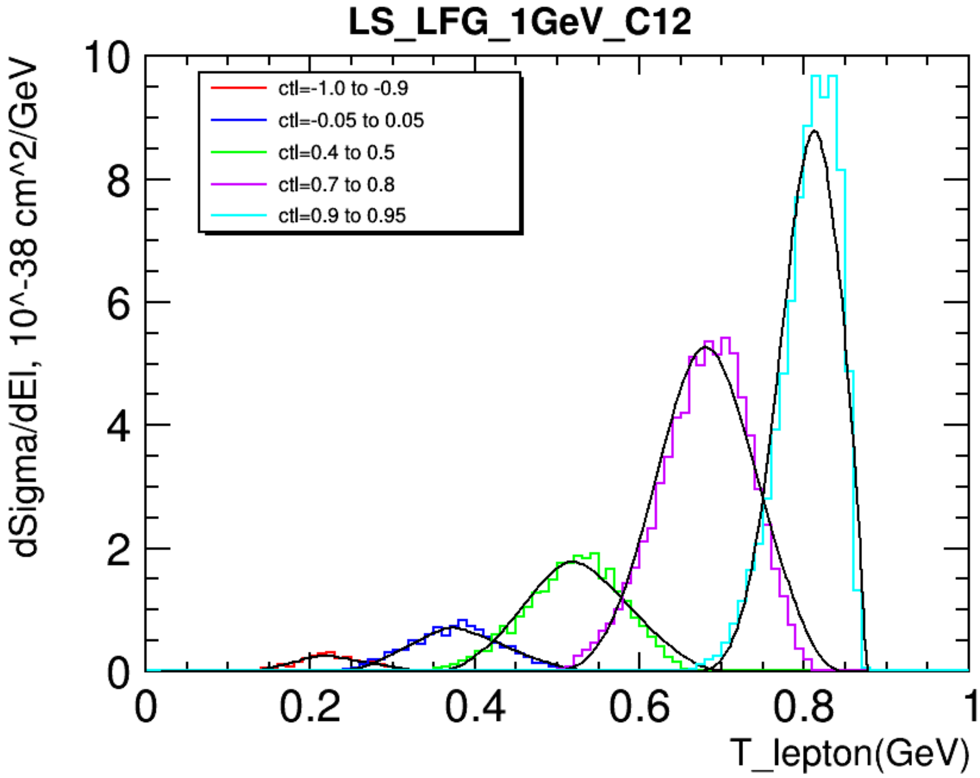
In order to make GENIE agree with Nieves' code, I had to use DipoleELFormFactorModel to calculate form factors. Further, the DipoleELFormFactorModel in GENIE sets Gen to 0, so I modified Nieves' code in order to set Gen to 0. With this change, I was able to produce plots showing that Nieves' code and genie were using the same form factors.

Nucleon tensor element agreement:

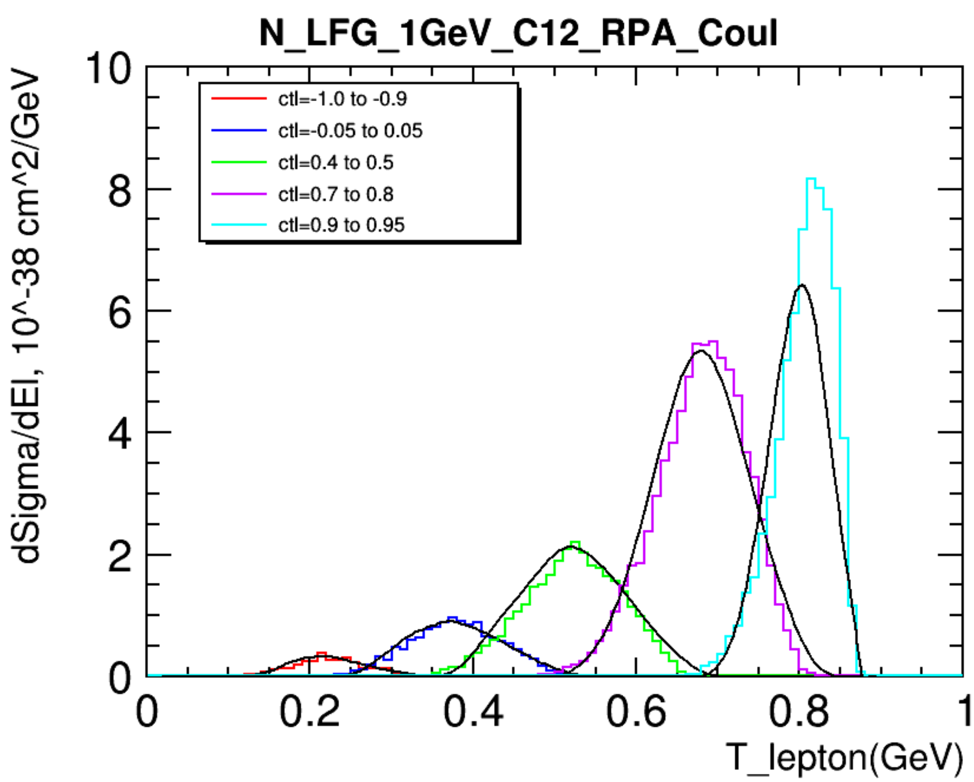


Finally, I modified Nieves' fortran code to integrate over ranges of the outgoing angle, so I had plots of dSigma/dTlepton for various angles. I could then take gevgen results, cut so the outgoing lepton angle is in the relevant range, bin according to Tlepton, and normalize using a spline in order to generate the same plots using gevgen. RPA effects can be turned off in Nieves' code, so I have plots of dSigma/dLepton with and without RPA. LS means LwlynSmith (With Andy's QELEventGenerator code):

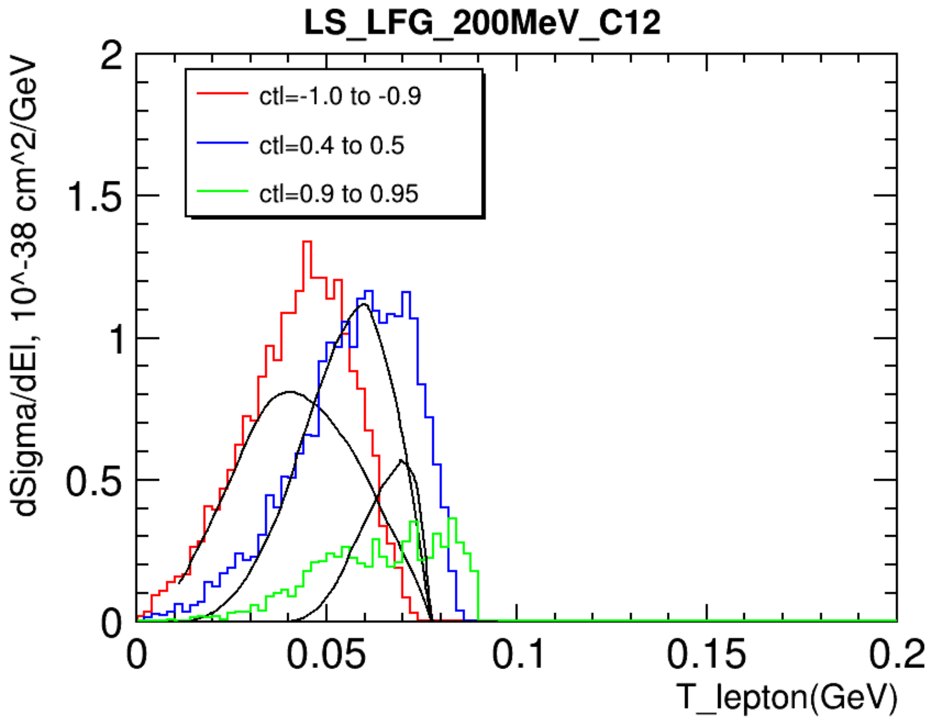
Andy's LwlynSmith output (histogram plot) vs Nieves' fortran code (solid black line). Note that as shown above, Andy's LwlynSmith code is identical to Nieves without RPA or Coulomb.



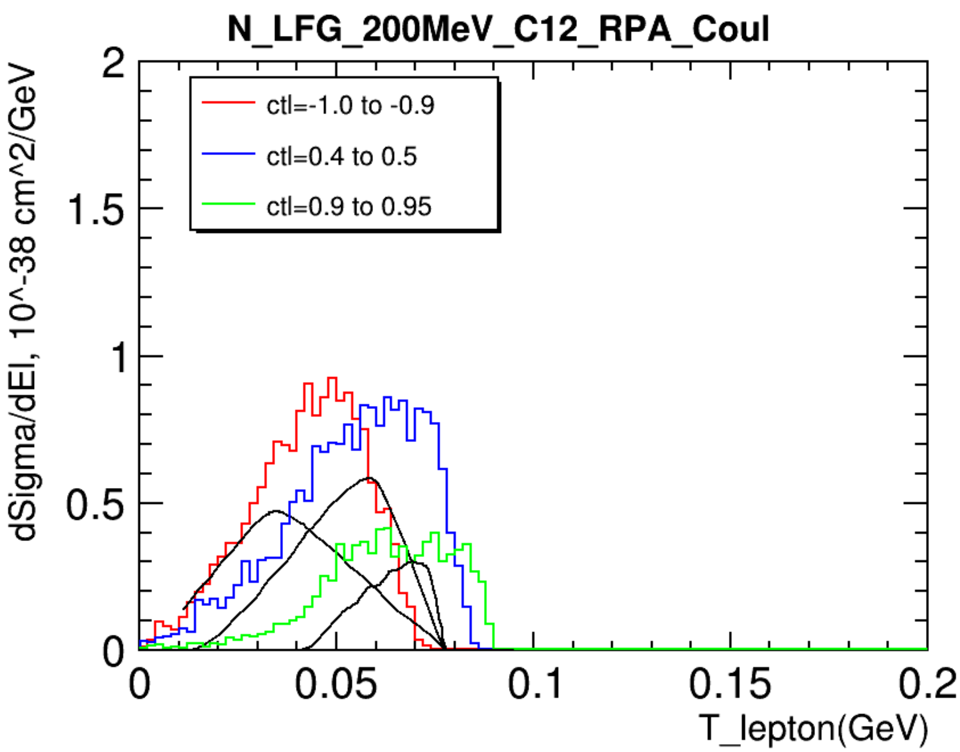
Nieves genie code + RPA + Coulomb vs Nieves' fortran code (solid black line)



LwlynSmith (Andy's code) vs Nieves' fortran code (Black line)

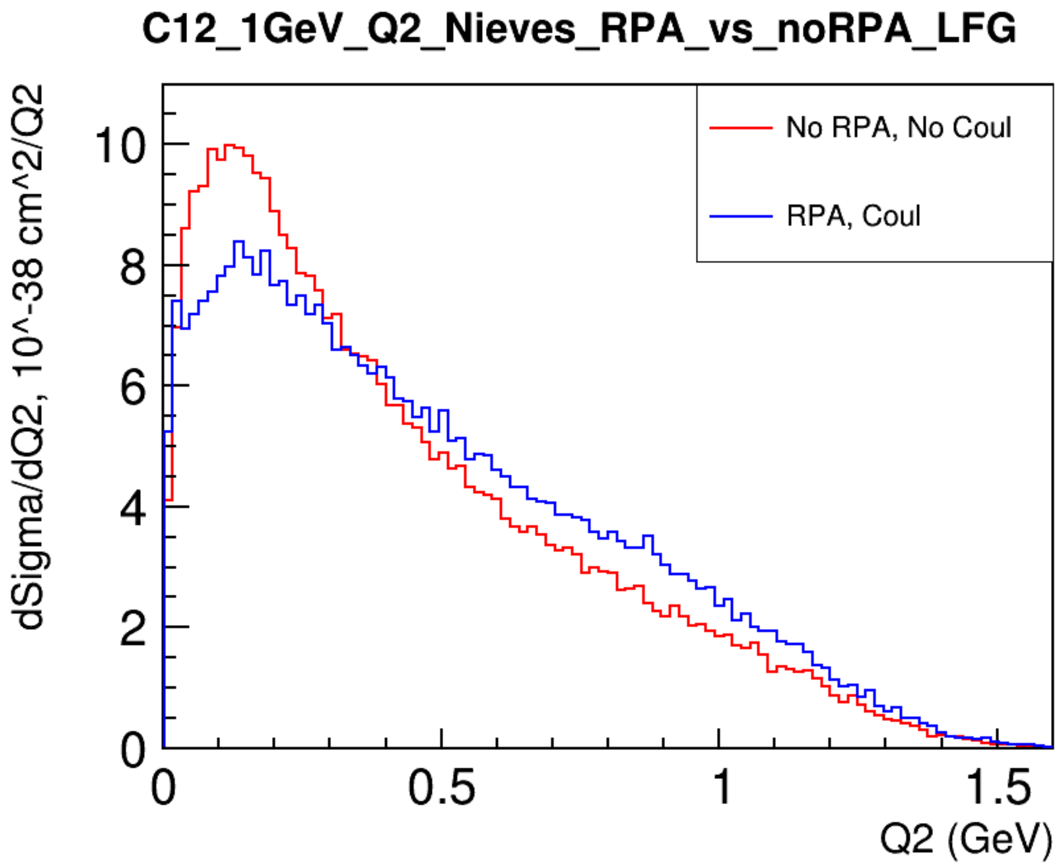


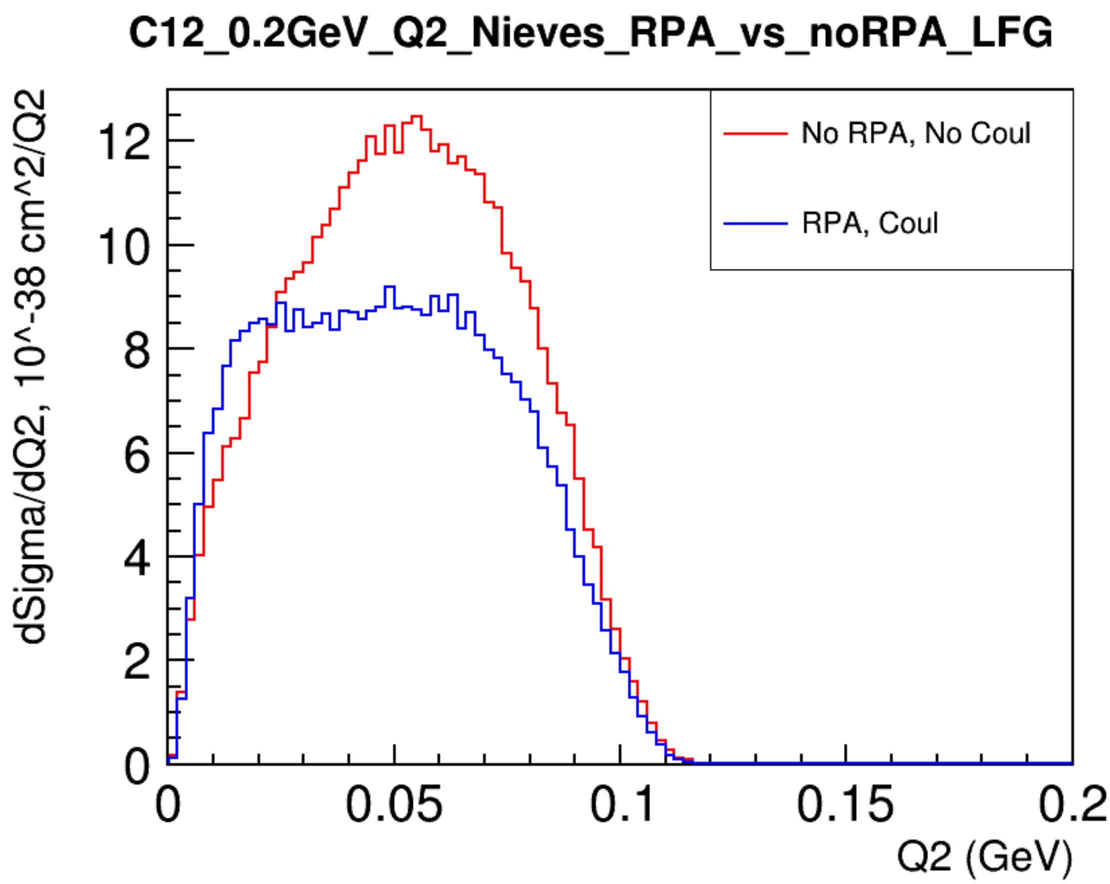
Nieves genie code + RPA + Coul vs Nieves' fortran code (Black line)



Agreement with Nieves' fortran code is not great at 200 MeV, because kinematic calculations surrounding the xsec calculation have a larger effect. However, I have shown that the calculated nucleon tensor elements are the same at 200 MeV between the two sets of code. See the plot below of dSigma/dQ2 at Enu=200MeV to see how RPA effects change the Q2 distribution.

Plots of dSigma/dQ2:





Full List of added and modified files for Nieves' QEL model with RPA

Nieves' model depends on the LFG nuclear model, so all files
from that model are used.

Files that must be added/modified in the GENIE code in order
to implement Nieves QEL model:

Added:
config/NievesQELCCPXSec.xml
src/LlewellynSmith/NievesQELCCPXSec.h
src/LlewellynSmith/NievesQELCCPXSec.cxx
src/LlewellynSmith/NievesQELException.h
src/LlewellynSmith/NievesQELException.cxx
config/QELEventGenerator.xml
src/QEL/QELEventGenerator.cxx
src/QEL/QELEventGenerator.h

Modified:
config/master_config.xml
config/UserPhysicsOptions.xml
config/ Messenger files
src/LlewelynSmith/LinkDef.h
config/EventGenerator.xml
src/Conventions/KinePhaseSpace.h
src/Interaction/InitialState.cxx
src/Interaction/InitialState.h
src/QEL/LinkDef.h
src/QEL/QELPrimaryLeptonGenerator.cxx
src/QEL/QELKinematicsGenerator.h
src/QEL/QELKinematicsGenerator.cxx
(Note that if the QELEventGenerator class replaces the old QEL
code in GENIE, then QELKinematicsGenerator will no longer be needed,
but the QELPrimaryLeptonGenerator class is still required because
it contains code that allows the NievesQELCCPXSec code to generate
an outgoing lepton for each Q2 when integrating over Q2 to generate
the spline)

Files from LFGNuclearModel:
config/LocalFGM.xml
config/LwlynSmithQELCCPXSec.xml
config/QELXSec.xml
src/CrossSections/QELXSec.h
src/CrossSections/QELXSec.cxx
src/EVGModules/FermiMover.cxx
src/EVGModules/PauliBlocker.h
src/EVGModules/PauliBlocker.cxx
src/Interaction/Target.h
src/Interaction/Target.cxx

src/LwlynSmith/LwlynSmithQELCCPXSec.cxx
src/Nuclear/LocalFGM.h
src/Nuclear/LocalFGM.cxx
src/Nuclear/LinkDef.h
src/Nuclear/NuclearModel.h
src/Nuclear/NuclearModelI.h
src/Nuclear/NuclearModelMap.h
src/Nuclear/NuclearModelMap.cxx
src/QEL/QELKinematicsGenerator.cxx
src/Utils/NuclearUtils.cxx

Summary:

config/NievesQELCCPXSec.xml:
src/LlewellynSmith/NievesQELCCPXSec.h
src/LlewellynSmith/NievesQELCCPXSec.cxx
src/LlewellynSmith/NievesQELException.h
src/LlewellynSmith/NievesQELException.cxx

Files to implement Nieves' QEL model. It can include RPA long range correlations in order to account for the intermediate W boson interacting with surrounding nucleons. It can also include Couloumb corrections. Gives identical results to LwlynSmithQELCCPXSec when the target is a free nucleon. The exception is thrown if an event is unphysical, such as when Coulomb effects would push the event below threshold.

config/master_config.xml
One line added to link to NievesQELCCPXSec.xml
One line added to link to QELEventGenerator.xml
(One line added to link to LFGNuclearModel.xml)

config/UserPhysicsOptions.xml
Comments added to mention NievesQELCCPXSec (and LocalFGM) as options

config/ Messenger files
updated to inclue NievesQELCCPXSec and QELEventGenerator (and LocalFGM) respectively use the same messenger settings as LwlynSmithQELCCPXSec and QELKinematicsGenerator (and BodekRitchie)

src/LwlynSmith/LinkDef.h
Updated to link NievesQELCCPXSec to the genie

src/QEL/QELKinematicsGenerator.h
src/QEL/QELKinematicsGenerator.cxx
QELKinematicsGenerator updated to catch NievesQELCCPXSecException.

src/QEL/QELPrimaryLeptonGenerator.cxx
QELPrimaryLeptonGenerator has a method that can be accessed by
NievesQELCCPXSec that genrates a lepton and stores it in the running
kinematics variables. If there is a stored lepton when QELPrimaryLeptonGenerator
runs, it accesses that stored leptona and adds it to the event record.
(Otherwise the parent PrimaryLeptonGenerator class is called to generate
a lepton).

config/EventGenerator.xml
src/Conventions/KinePhaseSpace.h
src/Interaction/InitialState.cxx
src/Interaction/InitialState.h
src/QEL/QELEventGenerator.cxx
src/QEL/QELEventGenerator.h
Replaces FermiMover, QELKinematicsGenerator, QELPrimaryLeptonGenerator,
and QELHadronicSystemGenerator in order to generate all kinematics
before calculating the cross section.
Adaptation of Andrew Furmanski's code.
One line added to KinePhaseSpace.h to include kPSFullDiffQE as a
choice for phase space.
A method is added to the Initial state class to calculate the center
of mass energy.

LocalFGM:
Local Fermi gas model, where the Fermi momentum depends on the position
in the nucleus. Nieves used this model when doing his calculation,
so the best data should be obtained by combining Nieves' QEL model
with the LFG nuclear model.
The code to implement Nieves' model also depends on several changes
made for the LocalFGM model, such as changes to QELXSec so that
the physics necessary to do an integral is moved to the XSec object
(in this case NievesQELCCPXSec), so I can have the Nieves code
always average over initial nucleons when calculating the integral.
See LFG_info.txt for more info about the changes added with the
LocalFGM model.
