# Guide to using extra GENIE weight calculators from MicroBooNE

**DRAFT**

S. Gardiner*

March 23, 2020

## 1 Introduction

To produce Monte Carlo samples in support of its 2020 analyses, MicroBooNE has adopted GENIE v3.0.4 as its neutrino interaction generator. Since the release of v3.0.4 in April 2019, two important issues in the code were found and fixed by the GENIE collaboration. Current releases of the GENIE Reweight product also lack support for varying the Nieves CCMEC model [1], which is included in the official GENIE tune `G18_10a_02_11a` used by MicroBooNE.

To address these issues, MicroBooNE has prepared two patched versions of the GENIE Generator and Reweight products. The first of these, "GENIE v3.0.4 MicroBooNE patch 01," makes three changes to the official release:

- Backporting an important bug fix in CCQE spline integration from Generator v3.0.6,
- Adding the ability to configure the binding energy prescription used in MECGenerator for the Nieves CCMEC model, and
- Adding several new calculators to Reweight.

The "patch 02" version extends "patch 01" by fixing a problem in decays of the P33(1600) and F17(1970) resonances (see GENIE docDB #153, now also fixed on the master branch) and adding several additional calculators to Reweight. Neither of these patched versions of GENIE v3.0.4 was intended for public distribution. They were created simply to resolve issues that arose as MicroBooNE has been pursuing its current round of MC production. To the extent that the GENIE collaboration feels that these modifications would be useful to the wider community, however, MicroBooNE is open to contributing them to an official release.

## 2 Weight calculators developed for MicroBooNE

The full list of new weight calculators in "patch 02" is as follows:

**NormCCMEC, NormNCMEC, NormEMMEC** Energy-independent MEC total cross section normalization, separated by interaction mode. These systematic tweak dials should work equally well for Nieves MEC as for GENIE empirical, SuSAv2, or any new MEC model that is implemented in the future.

**DecayAngMEC** Varies the angular distribution of the decay of the final-state nucleon cluster in MEC reactions between isotropic in the cluster rest frame (0, default MECGenerator behavior) and proportional to $\cos^2\theta$ (1), where the 3-momentum transfer is taken to be along the z-axis. The specific choice of alternate distribution was not theoretically motivated. This tweak dial is also model-independent.

**FracPN_CCMEC** Varies the fraction of initial pn nucleon pairs predicted by the Nieves CCMEC cross section.

*mailto:gardiner@fnal.gov

**FracDelta_CCMEC** Varies the relative contribution of diagrams involving an internal $\Delta$ line to the total Nieves CCMEC cross section.

**XSecShape_CCMEC** Varies the shape of the CCMEC differential cross section between Nieves (0) and GENIE empirical (1) in $(q^0, |\mathbf{q}|)$ space.

**RPA_CCQE** Linearly interpolates between the default Nieves RPA correction for CCQE (0) and the same calculation with RPA turned off (1).

**RootinoFix** Dummy knob. If configured, events affected by the resonance decay bug mentioned above are zeroed out. All other events are given a weight of one.

**NormCCCOH,NormNCCOH** Energy-independent normalization of the CC and NC coherent pion production total cross section. These tweak dials are model-independent.

**ThetaDelta2NRad** Varies the angular distribution for radiative decays of $\Delta^{0,+}$ using the same shapes as DecayAngMEC.

**NormCCNonCOHPi0, NormNCNonCOHPi0** Energy-independent normalization of CC and NC production of $\pi^0$, regardless of the underlying reaction mode.

**CoulombCCQE** Varies the strength of the electromagnetic potential used to apply Coulomb corrections in the Nieves CCQE model. This is an admittedly rough way of assessing an uncertainty on the Coulomb corrections (which are typically small).

Other than edits to `Generator/config/GSystUncertaintyTable.xml` (which stores the nominal 1-sigma uncertainties for Reweight knobs), only the implementation of `CoulombCCQE` required (minor) changes to the Generator product.

# 3 Patched GENIE code

MicroBooNE has recently set up a GitHub organization to host several code repositories. Among these are repositories which host the patched versions of the Generator (`https://github.com/uboone/Generator`) and Reweight (`https://github.com/uboone/Reweight`) products. In either repository, "patch 01" corresponds to the tag `v3_00_04_ub1` and "patch 02" to `v3_00_04_ub2`. Pre-built tarballs of these tags are available as ups products (used by Fermilab experiments in production) on SciSoft (see `https://scisoft.fnal.gov/scisoft/packages/genie/`).

# 4 nutools

NOvA and the Fermilab LArTPC experiments use the a shared framework called "nutools" for many software packages that they share in common. All Fermilab neutrino experiments use GENIE in production through the nutools interface. Each version of nutools is packaged with a particular build of GENIE as a ups product. For MicroBooNE, nutools `v2_27_06` is built against genie `v3_00_04_ub1`, and nutools `v2_27_07` is built against genie `v3_00_04_ub2`. For SBN, it is likely that a more up-to-date version of nutools will need to be linked to a patched GENIE build in order to incorporate the MicroBooNE modifications. No code changes to nutools are anticipated other than modifying the genie ups product dependency and related dependencies (e.g., dk2nugenie).

# 5 LArSoft

LArSoft is a software stack built on top of nutools with liquid-argon-specific packages for detector simulation and various other tasks. LArSoft includes GENIE reweighting support via a class called `GenieWeightCalc` used by the EventWeight module. The official implementation of the `GenieWeightCalc` class in turn relies on a nutools interface to pass information between GENIE and itself. To simplify maintenance of

the reweighting tools, the `GenieWeightCalc` class has been refactored by MicroBooNE to use the existing GENIE framework directly. The techniques used are very similar to, e.g., the GENIE command-line tool `grwght1p`. In particular, the LArSoft/nutools practice of using labels for the systematic knobs that are distinct from GENIE's (e.g., LArSoft's `ResDecayTheta` versus GENIE's `Theta_Delta2Npi`) has been abandoned in favor of using GENIE's consistently. The code for the refactored `GenieWeightCalc` class is available on the `v08_00_00_br` branch in the MicroBooNE fork of larsim (`https://github.com/uboone/larsim`). Although it will need to be ported into a newer version of larsim for use by SBN, doing so is expected to be straightforward. Simply replacing the existing `larsim/EventWeight/Calculators/GenieWeightCalc.cxx` and `larsim/EventWeight/Calculators/CMakeLists.txt` files with the MicroBooNE versions may be all that is needed.

# References


[1] J. Schwehr, D. Cherdack and R. Gran, *GENIE implementation of IFIC Valencia model for QE-like 2p2h neutrino-nucleus cross section*, `1601.02038`.