How to use NUISANCE flat-trees, and **many** examples

Clarence Wret, Callum Wilkinson,
Patrick Stowell (remote), Luke Pickering (remote),
Stephen Dolan (remote)

Pre-NuInt workshop
April 12 2024

Table of contents

- Getting the flat-tree files
- Brief introduction to generators and the experimental landscape
- Introduction to NUISANCE flat-trees
  - Simple TTree::Draw commands
  - Using the particle stack
- How can you utilise this to study the physics and model-dependence?

Introduction

- Flat-trees provided for GENIE, NuWro and NEUT
  - Thanks to Callum!
- **Flat-trees don't need generators** set up: **just need ROOT**
- Flat-trees contain:
  - Initial, pre-FSI, and post-FSI particle stack
  - Precalculated interaction-level ("theory") variables (e.g. q_0, W, Q^2)
  - True interaction mode from the generators
  - Cross-section scaling → Take number of events to xsec
  - Weights from reweighting procedures
  - ...and more!
- Can easily compare generator against generator using different selections

Introduction

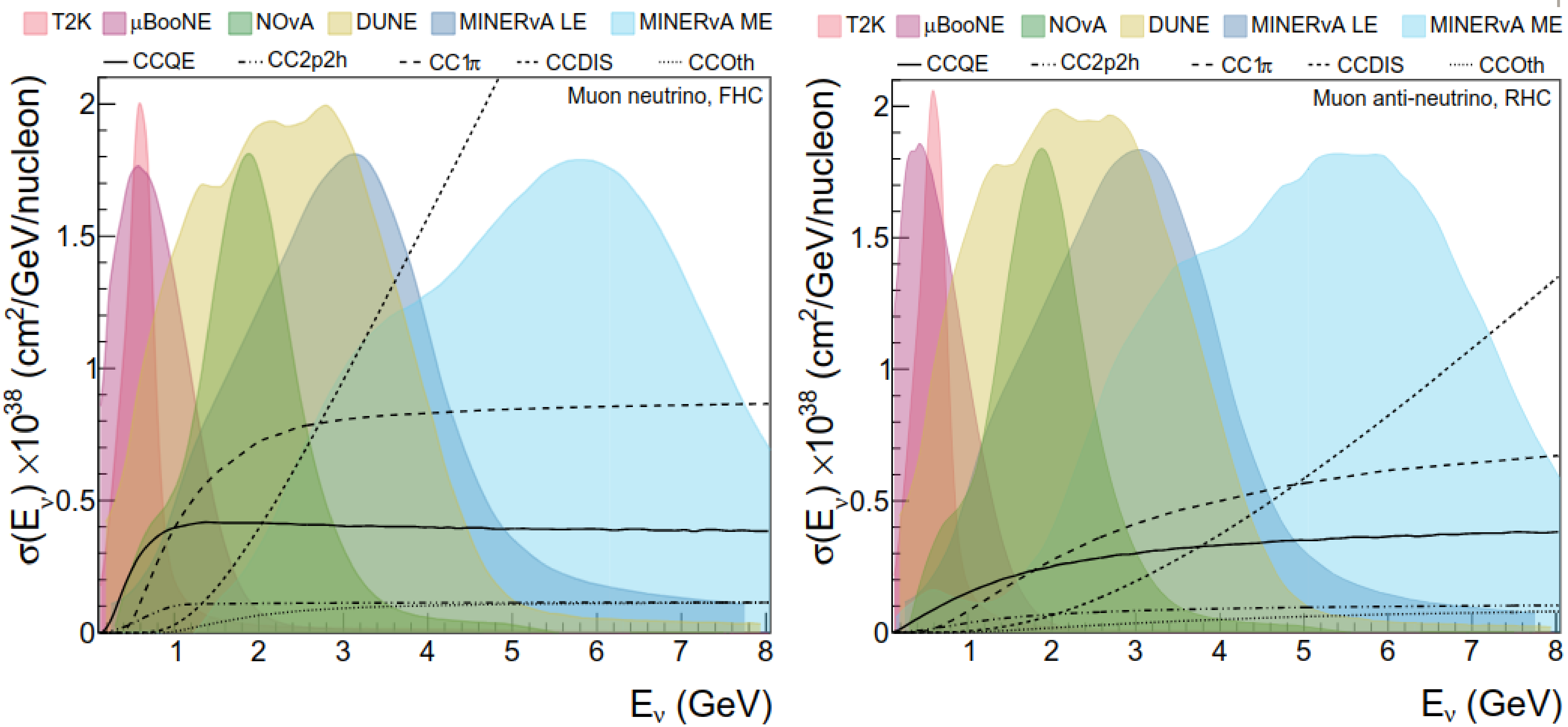
- Going to start slowly
  - Not sure how comfortable people are with ROOT
  - We are going to start from the ground up
- If you find the first few slides boring, don't worry, we'll get to the physics and fun things soon!
- If you have any questions, please don't hesitate to ask
- This tutorial is an in-depth version of the readme: https://github.com/NUISANCEMC/tutorials/blob/main/interactive/README.md#using-the-nuisance-flat-trees
- If you want to get in touch later, find us on email:
  - clarence.wret@physics.ox.ac.uk, cwilkinson@lbl.gov, luke.pickering@stfc.ac.uk, stephen.joseph.dolan@cern.ch, p.stowell@sheffield.ac.uk
  - Or connect to our slack: https://nuisance-xsec.slack.com

Purpose

- Purposes behind tutorial are multiple
  - Different generators should not be scary: democratise their usage!
  - Develop some intuition about neutrino interactions
  - Use generators to challenge and build the intuition
  - Enable you to investigate what different generators predict for your measurement
    - Without having to run entire productions
    - Without being a generator expert
    - Make measurements that help the generator community reject and build new models
- … and more

Introduction

- Experimental landscape for nuclear targets is between 0.5-8 GeV or so
- Many different neutrino interaction mechanisms
  - – We will explore some here today!



Very brief on the generators

- NuWro: LFG with RPA, Valencia 2p2h
- NEUT: SF for QE, Valencia 2p2h
- GENIE G18 10a: LFG, Valencia 2p2h
- GENIE CRPA: CRPA for QE, SuSAv2 for 2p2h?, LFG for rest?
- GENIE and NEUT use Rein-Sehgal with modifications, NuWro uses Delta-dominated
- SIS and DIS models have variations between generators

- We can discuss this more if anyone is interested
- For now, just consider them different models

How to get the files

- https://portal.nersc.gov/project/nuisance/IOP_review_2023/
- Files **with** “`*NUISFLAT.root`” are flat-trees
- Files **without** “`*NUISFLAT.root`” are generator output
- Can download via browser, or via `wget` (or `curl`, …):

```
#!/bin/bash
GEN="NEUT562"
NAME="T2KND_FHC_numu_C8H8_NEUT562_1M_0000_NUISFLAT.r
oot"
wget -r -l1 --no-directories --no-parent
https://portal.nersc.gov/project/nuisance/IOP_review
_2023/$GEN/$NAME
```

Generator

Number of events

Experiment

Flux

Target

Generator

Which file in batch

- Each file can be up to 250MB, make sure you have enough space
- …and let’s test the internet connection in the room!

How to get the files

- The full set of files can be downloaded here
  - https://github.com/NUISANCEMC/tutorials/blob/main/interactive/download_files_forNuSTEC2024.sh
- If you have not downloaded these already, it's probably too late to start the download
  - Transfer speeds are quite slow
- **We have a USB stick with these on, come down to the stage!**

Looking inside the trees

```
clarence@clarence-lin-hd:~/work/nustec_workshop_NuInt2024$
root -l T2KND_FHC_numu_C8H8_NEUT562_1M_0000_NUISFLAT.root
Loading rootlogon from:
/home/clarence/rootfun/rootlogonold.C...
root [0]
Attaching file
T2KND_FHC_numu_C8H8_NEUT562_1M_0000_NUISFLAT.root as _file0...
(TFile *) 0x57e864a896b0
root [1] _file0->ls()
TFile**     T2KND_FHC_numu_C8H8_NEUT562_1M_0000_NUISFLAT.root
 TFile*     T2KND_FHC_numu_C8H8_NEUT562_1M_0000_NUISFLAT.root
  KEY: TTree   FlatTree_VARS;9   FlatTree_VARS [current cycle]
  KEY: TTree   FlatTree_VARS;8   FlatTree_VARS [backup cycle]
  KEY: TH1D FlatTree_FLUX;1   FlatTree_FLUX
  KEY: TH1D FlatTree_EVT;1 FlatTree_EVT
```

Open the file

Look at the file contents

This is the flat-tree

Looking inside the trees

- Explore these at your leisure; have things like q_3, number of final state particles, momentum along x, y, z...

```
root [2] FlatTree_VARS->Print()
******************************************************************************
*Tree    :FlatTree_VARS: FlatTree_VARS                                       *
*Entries :  1000000 : Total =       453725185 bytes  File  Size =  242730867 *
*        :          : Tree compression factor =   1.87                       *
******************************************************************************
*Br    0 :Mode      : Mode/I                                                 *
*Entries :  1000000 : Total  Size=    4002701 bytes  File Size  =     882981 *
*Baskets :       24 : Basket Size=     488960 bytes  Compression=   4.53     *
*............................................................................*
*Br    1 :cc        : cc/B                                                   *
*Entries :  1000000 : Total  Size=    1001463 bytes  File Size  =     209466 *
*Baskets :       12 : Basket Size=     121856 bytes  Compression=   4.78     *
*............................................................................*
*Br    2 :PDGnu     : PDGnu/I                                                *
*Entries :  1000000 : Total  Size=    4002729 bytes  File Size  =      22216 *
*Baskets :       24 : Basket Size=     489472 bytes  Compression= 180.14     *
*............................................................................*
```

Mode listings

- Mode entry follows NEUT mode; exists for NEUT, GENIE and NuWro using the official conversions

1	CC QE
2	2p2h
11	CC 1π^{+}1p
12	CC 1π^{0}
13	CC 1π^{+}1n
16	CC coh
17	CC 1γ
21	CC Multi-π
22	CC 1η
23	CC 1K
26	CC DIS

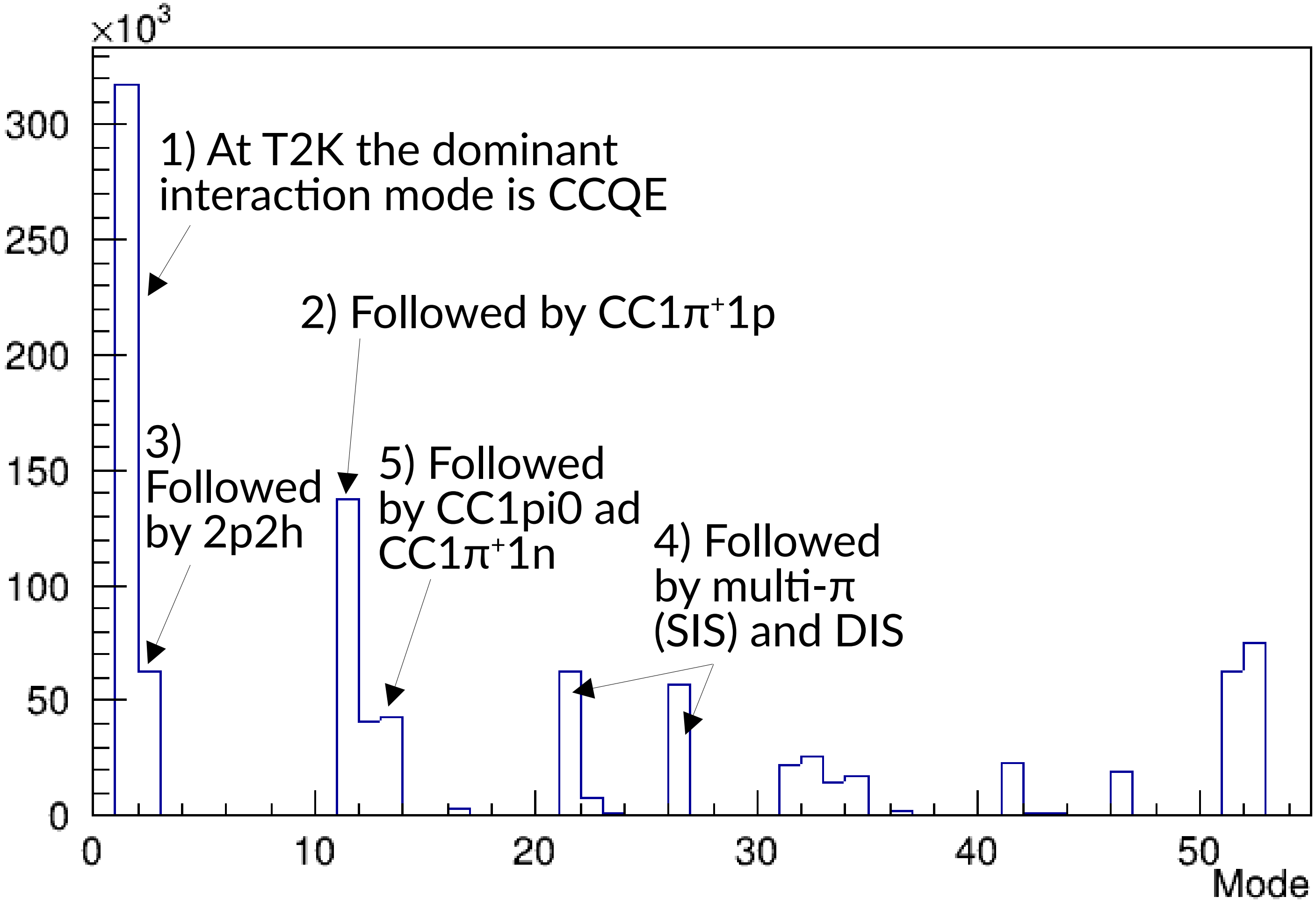
31	NC 1π^{0}1n
32	NC 1π^{0}1p
33	NC 1π^{-}1p
34	NC 1π^{+}1n
36	NC coh
38	NC 1γ 1n
39	NC 1γ 1p
41	NC multi-π
42	NC 1η 1n
43	NC 1η 1p
44	NC 1K^{0} 1Λ
45	NC 1K^{+} 1Λ
46	NC DIS
51	NC QE 1p
52	NC QE 1n

Mode listings

T2K

- Mode entry follows NEUT mode; exists for NEUT, GENIE and NuWro using the official conversions

```
root [4] FlatTree_VARS->Draw("Mode")
```



1) At T2K the dominant interaction mode is CCQE

2) Followed by CC1π^+1p

3) Followed by 2p2h

4) Followed by multi-π (SIS) and DIS

5) Followed by CC1pi0 ad CC1π^+1n

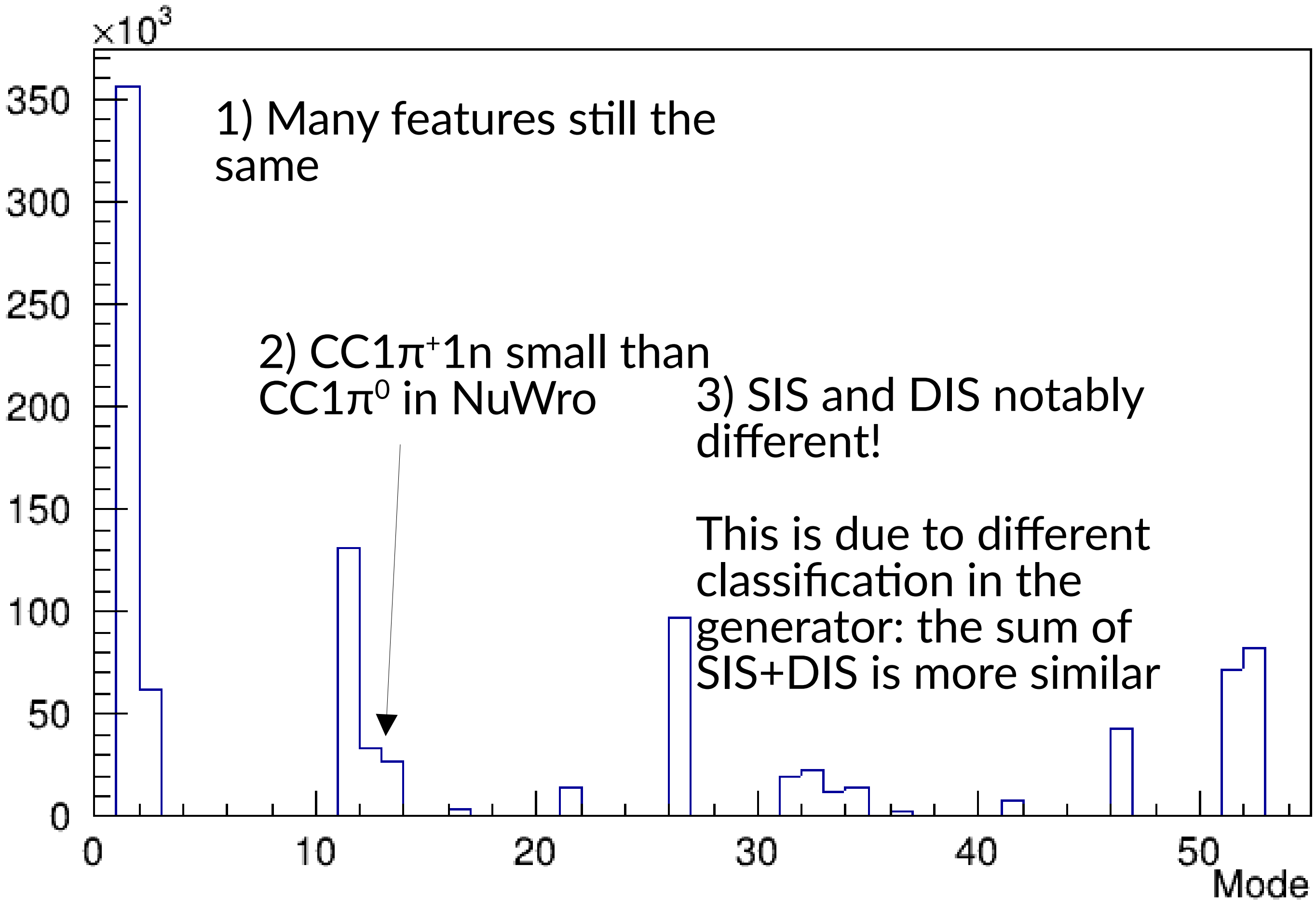
1	CC QE
2	2p2h
11	CC 1π^+1p
12	CC 1π^0
13	CC 1π^+1n
16	CC coh
17	CC 1γ
21	CC Multi-π
22	CC 1η
23	CC 1K
26	CC DIS

Mode listings

T2K

- Now try different generator: NuWro

```
root [4] FlatTree_VARS->Draw("Mode")
```



1) Many features still the same

2) CC$1\pi^{+}$1n small than CC$1\pi^{0}$ in NuWro

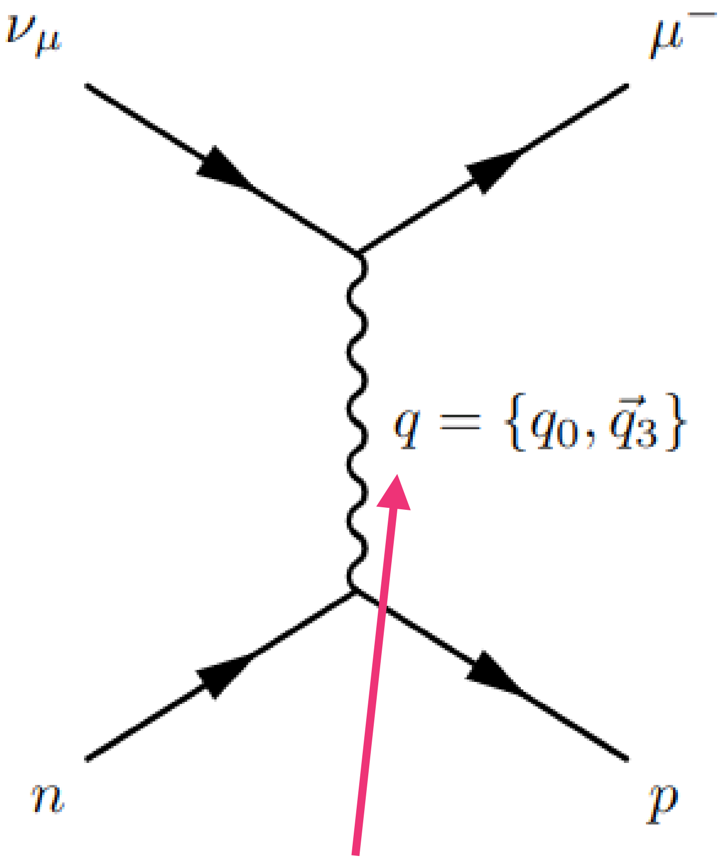
3) SIS and DIS notably different!

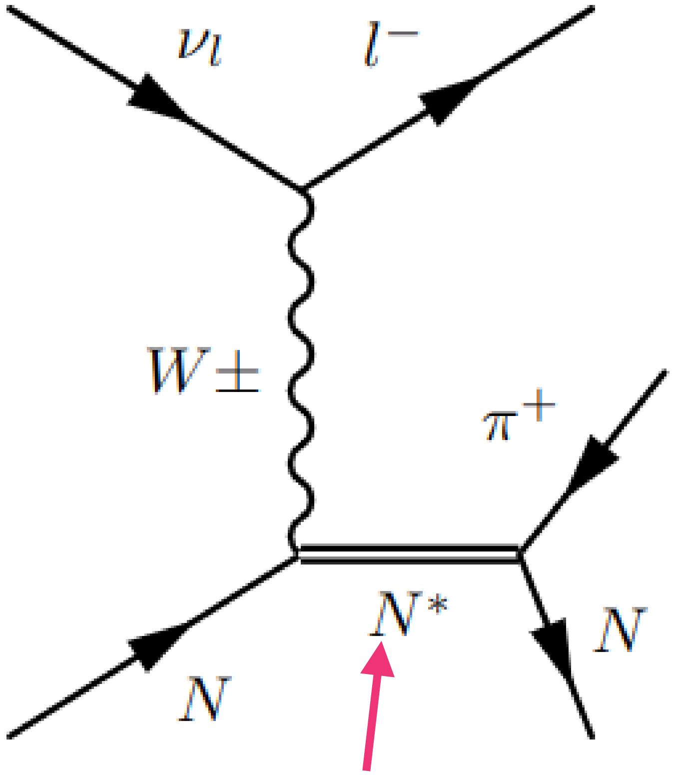
This is due to different classification in the generator: the sum of SIS+DIS is more similar

1	CC QE
2	2p2h
11	CC $1\pi^{+}1p$
12	CC $1\pi^{0}$
13	CC $1\pi^{+}1n$
16	CC coh
17	CC 1γ
21	CC Multi-π
22	CC 1η
23	CC 1K
26	CC DIS

Defining variables

- First let's define some kinematic variables commonly used in scattering





$Q^2 = -q^2$

q_0 also referred to as ω (energy transfer)

W is hadronic mass of N* resonance

e.g. 1.232 GeV/c^2 for Delta

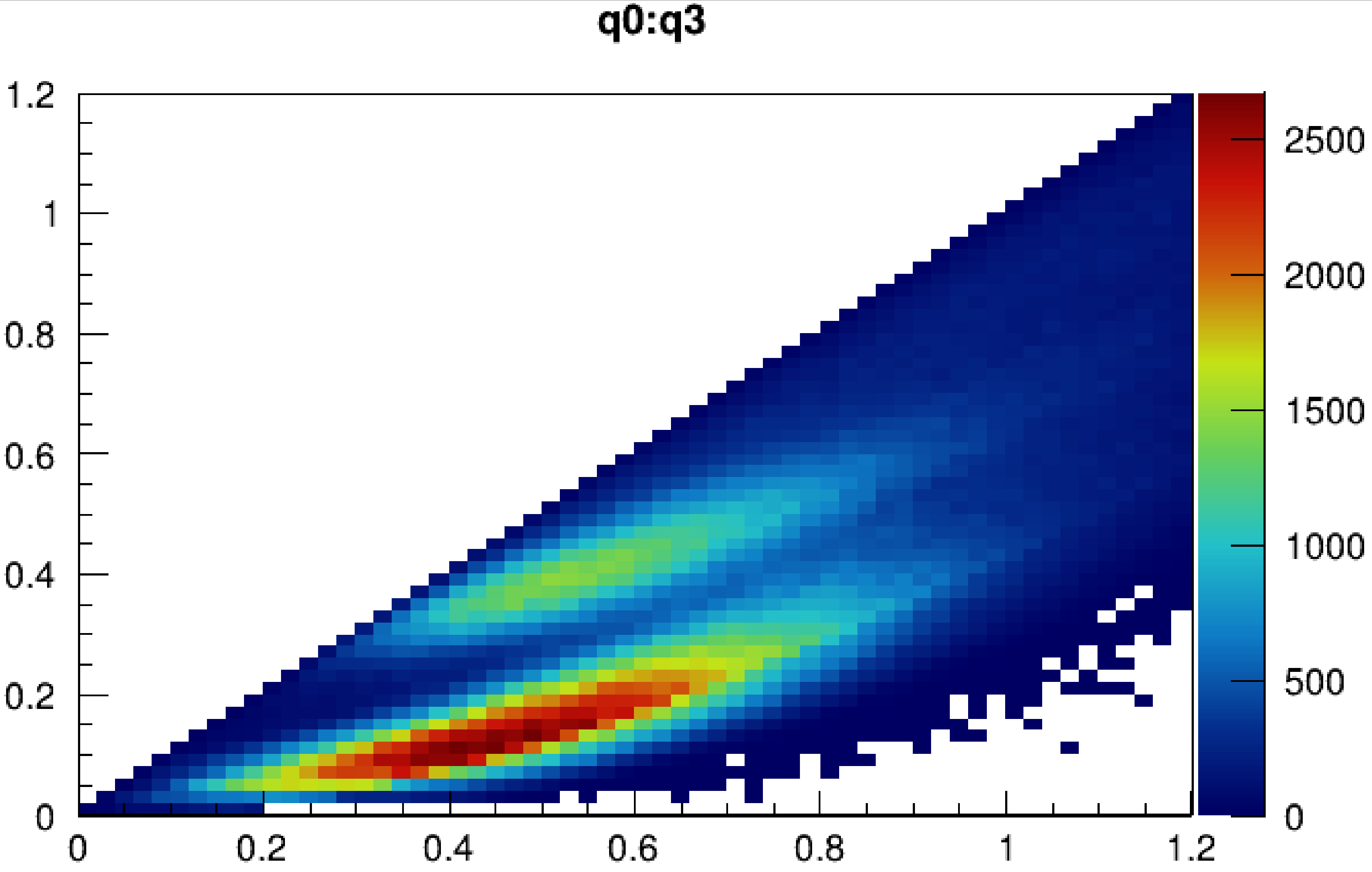
But keep in mind, never interact off **a single resonance: always multiple interfering!**

Simple draws

T2K

- Plot event distribution in q_0, q_3 for all events

```
root [10] FlatTree_VARS->Draw("q0:q3>>h1(60, 0, 1.2, 60, 0, 1.2)",
"", "colz")
```

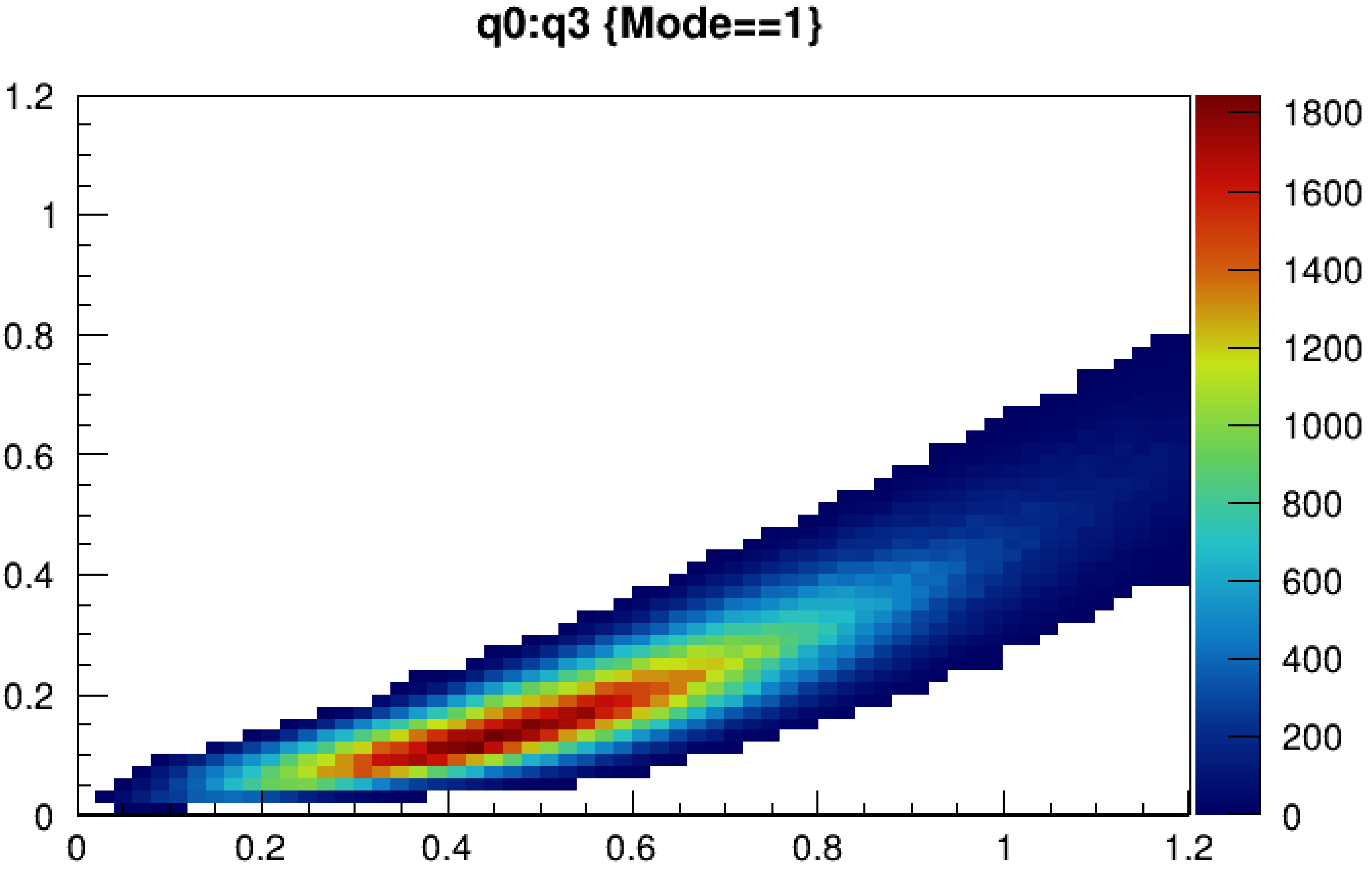


Simple draws

T2K

- Same, but for CCQE events only

```
root [12] FlatTree_VARS->Draw("q0:q3>>h1(60, 0, 1.2, 60, 0, 1.2)", "Mode==1", "colz")
```



Exercises

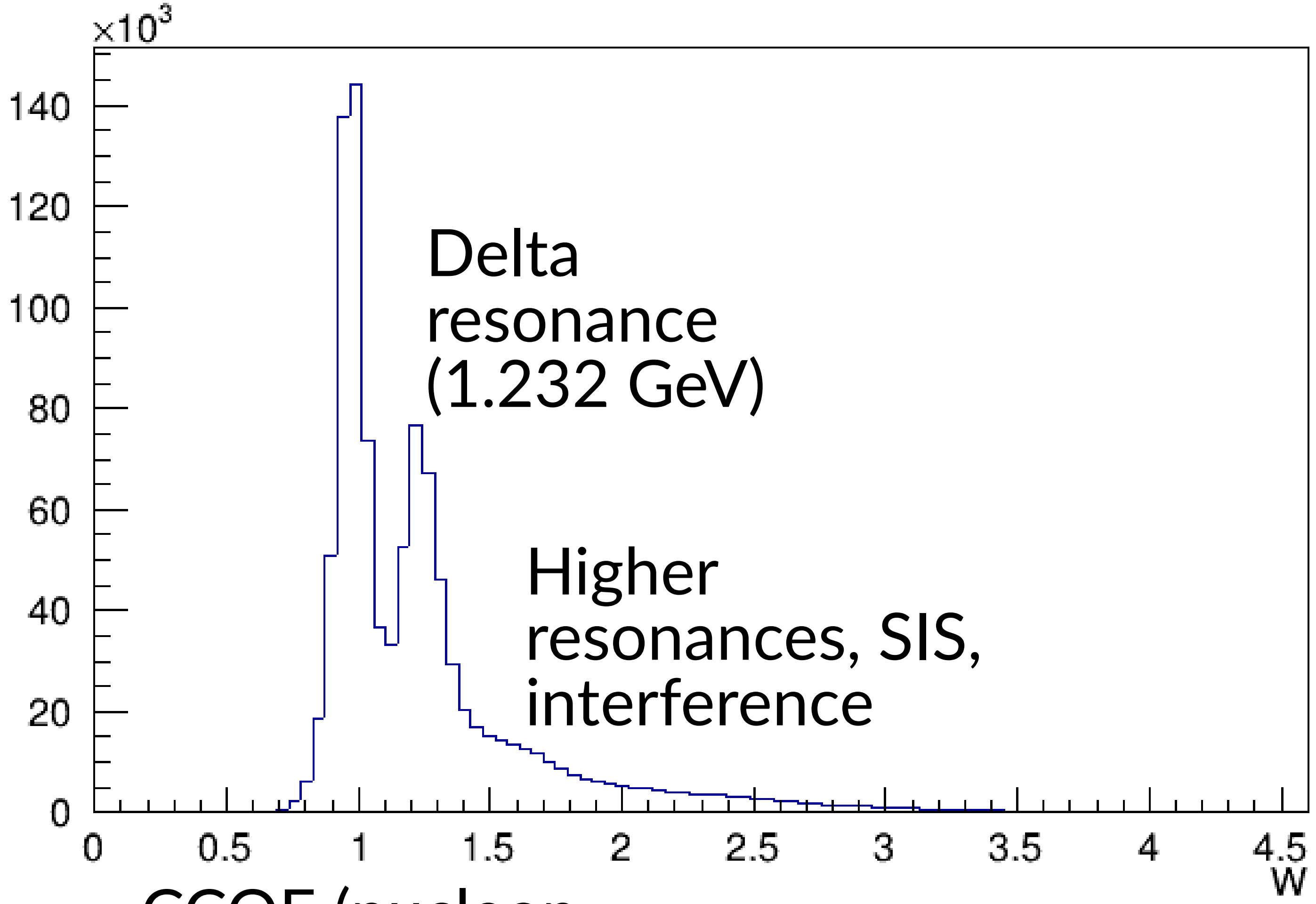
- Zeroth exercise
  - What is q_0 and q_3 here? What is their meaning? (hint: look at previous slide defining the variables)
- First exercise
  - Use the previous slides and the mode definition to draw q_0 vs q_3 for true 2p2h events
  - What do you observe compared to CCQE events?
- Second exercise
  - q_0 vs q_3 looks very different for all events compared to CCQE events, can you explain why?
- Third exercise
  - Compare q_0 vs q_3 for 2p2h events and CCQE+resonant events (hint, you'll need a or statement when drawing)
  - Do you notice any similarities between the two? Why?

Hadronic mass

- What about W, the reconstructed hadronic mass?

```
root [6] FlatTree_VARS->Draw("W")
```

T2K



Delta resonance (1.232 GeV)

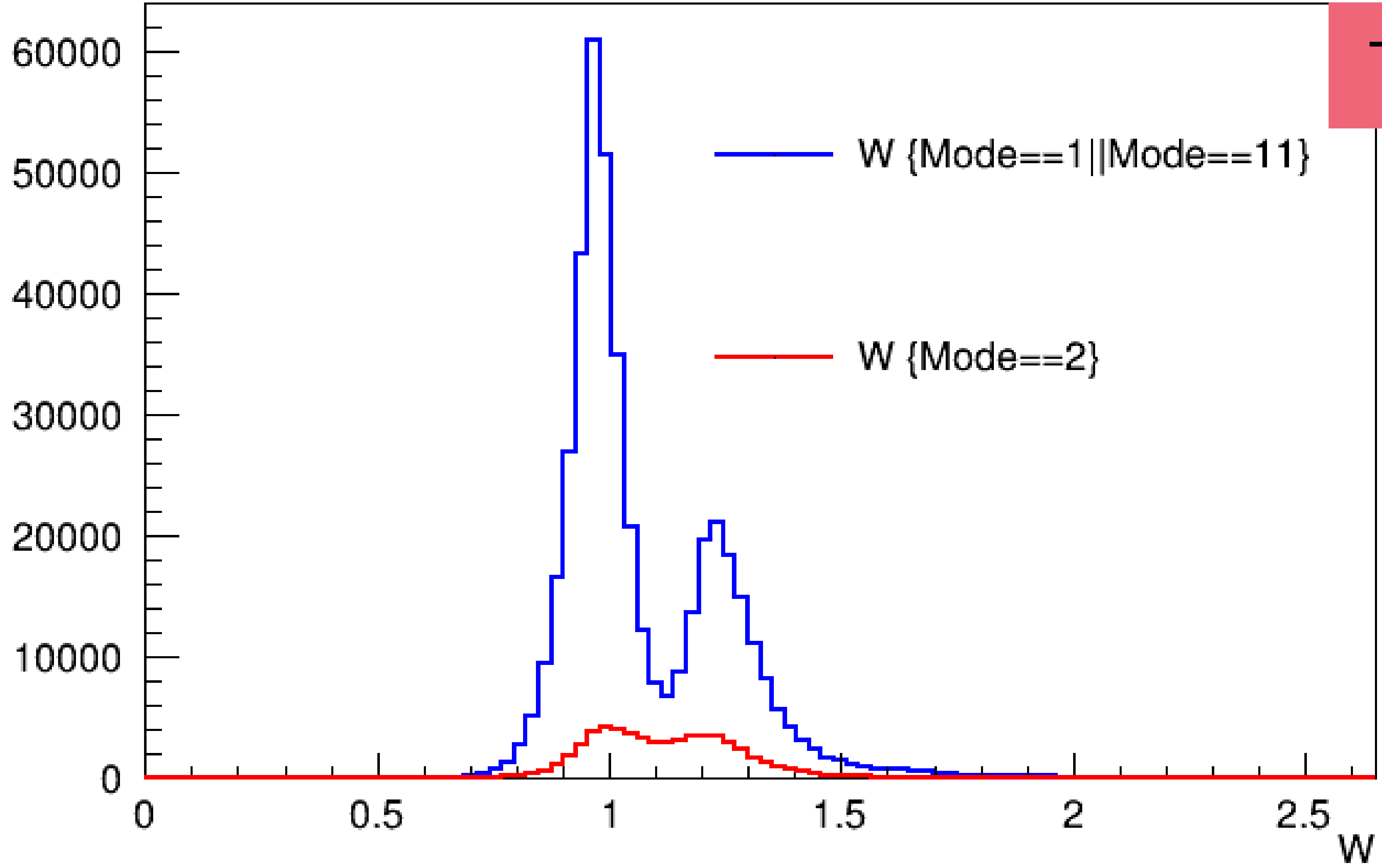
Higher resonances, SIS, interference

CCQE (nucleon, 0.938 GeV)

DIS often defined as W>2.0

Hadronic mass exercise

- Plot CCQE+RES together and 2p2h separately
- Positions of peaks similar → hints at Delta and nucleon part of 2p2h model



T2K

Clarence Wret

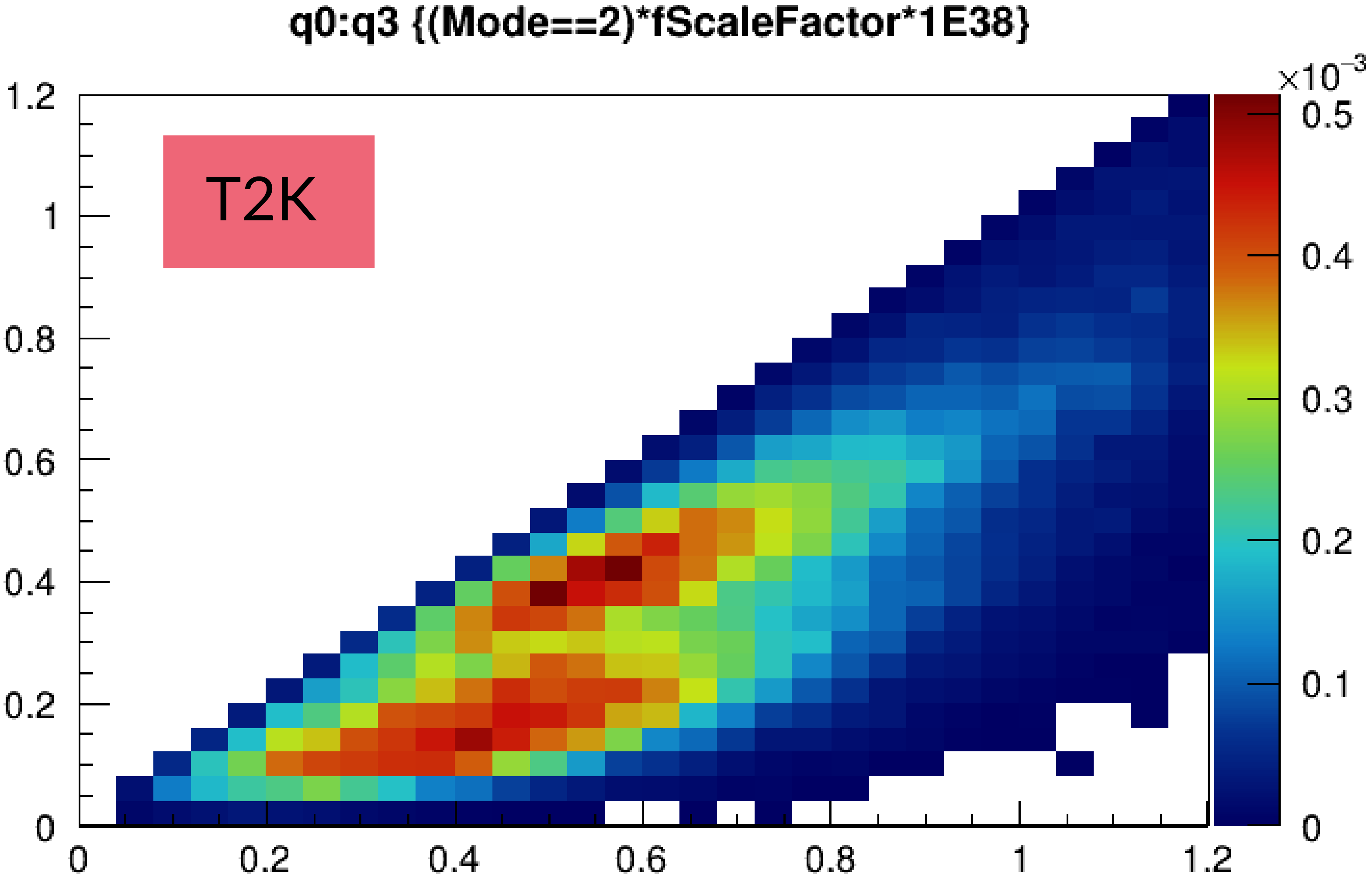
Scaling to a cross-section

- NuWro, GENIE, NEUT all generate events in proportion to cross section (GiBUU does not, it has weights!)
- `fScaleFactor` in the NUISANCE flat trees takes you from **event rate to cross section**: the same for all events for these generators
- Pre-calculated for you, just needs applying to your distribution
- **Beware**: `fScaleFactor` is often very small (~1E-45), so scaling a `TH1F` (with floats) will produce empty plots!!!
  - `TTree->Draw` often writes to a `TH1F`
  - **Might miss entries!**
  - **Either scale entire distribution by fScaleFactor, or scale by fScaleFactor*1E38**

Simple draws with xsec

- Plot event distributions for 2p2h events, now scaling to a differential cross section in units of 1E-38

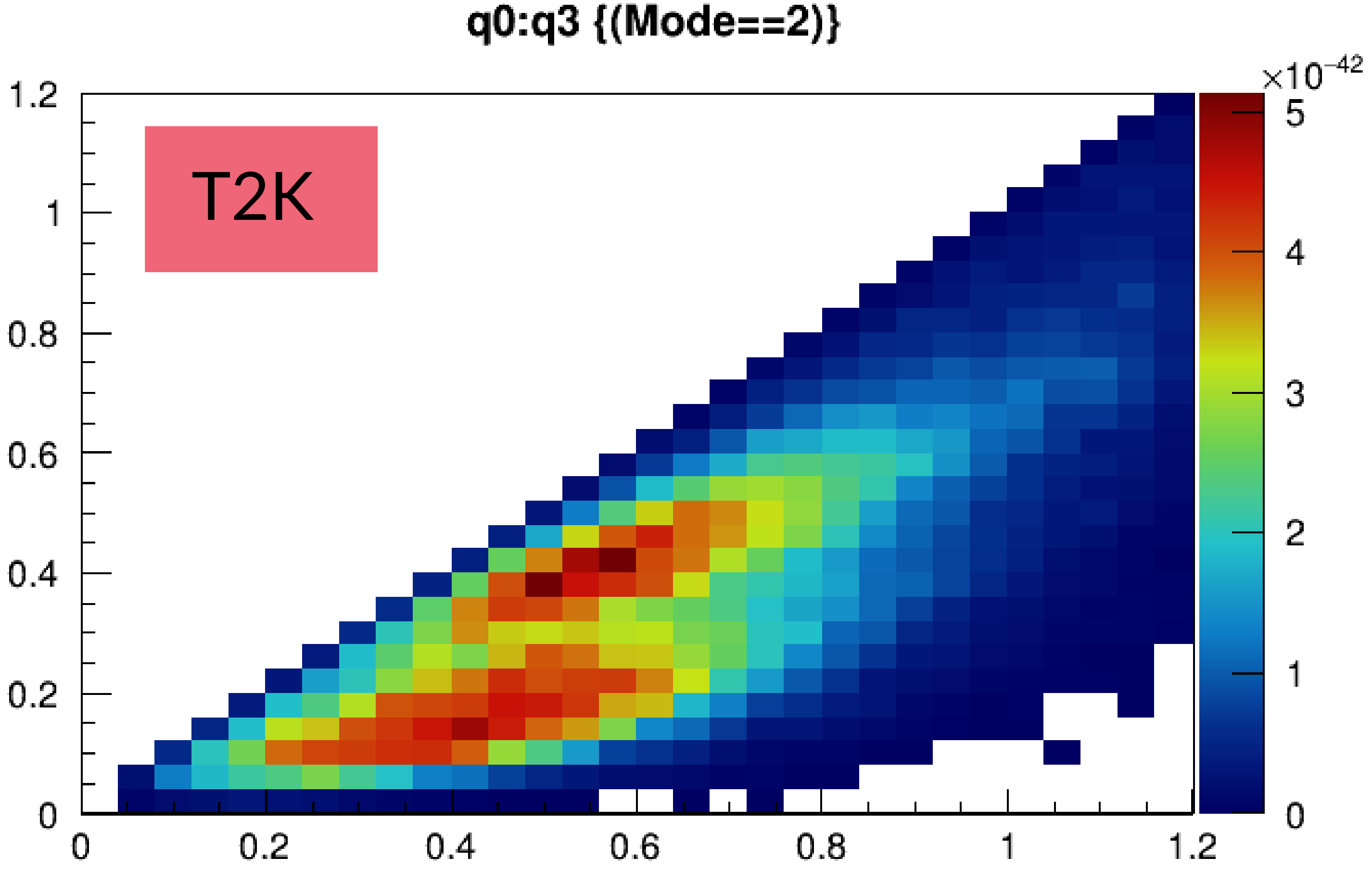
```
root [30] FlatTree_VARS->Draw("q0:q3>>h1(30,0,1.2,30,0,1.2)",
"(Mode==2)*fScaleFactor*1E38", "colz")
```



Simple draws with xsec

- Or with the alternative method, in real units

```
root [31] FlatTree_VARS->Draw("q0:q3>>h1(30,0,1.2,30,0,1.2)",
"(Mode==2)", "colz")
root [32] TH1D *h1 = (TH1D*)gDirectory->Get("h1")
root [33] double scalefactor = FlatTree_VARS-
>GetMaximum("fScaleFactor")
root [34] h1->Scale(scalefactor)
```



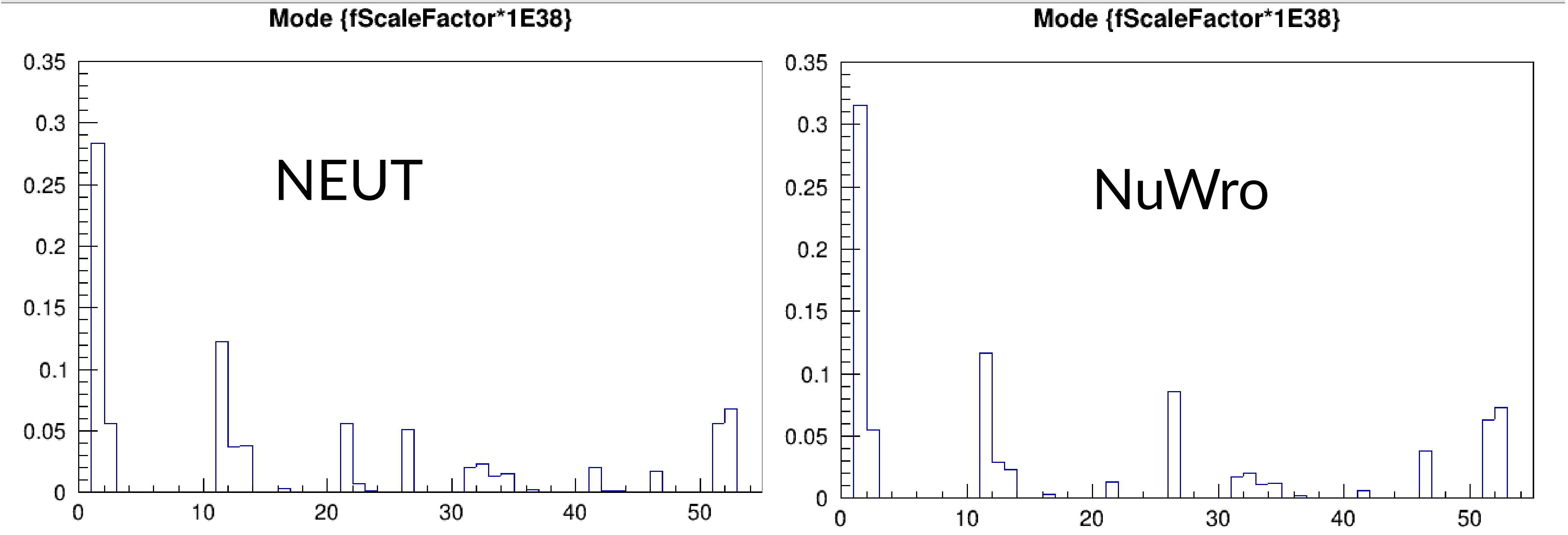
Simple draws with xsec

- Can repeat exercise for mode histograms, comparing NEUT and NuWro
  - We now directly compare the cross section

```
root [5] FlatTree_VARS->Draw("Mode", "fScaleFactor*1E38", "hist")
```

  - Make comparison easier by setting same y-axis range

T2K



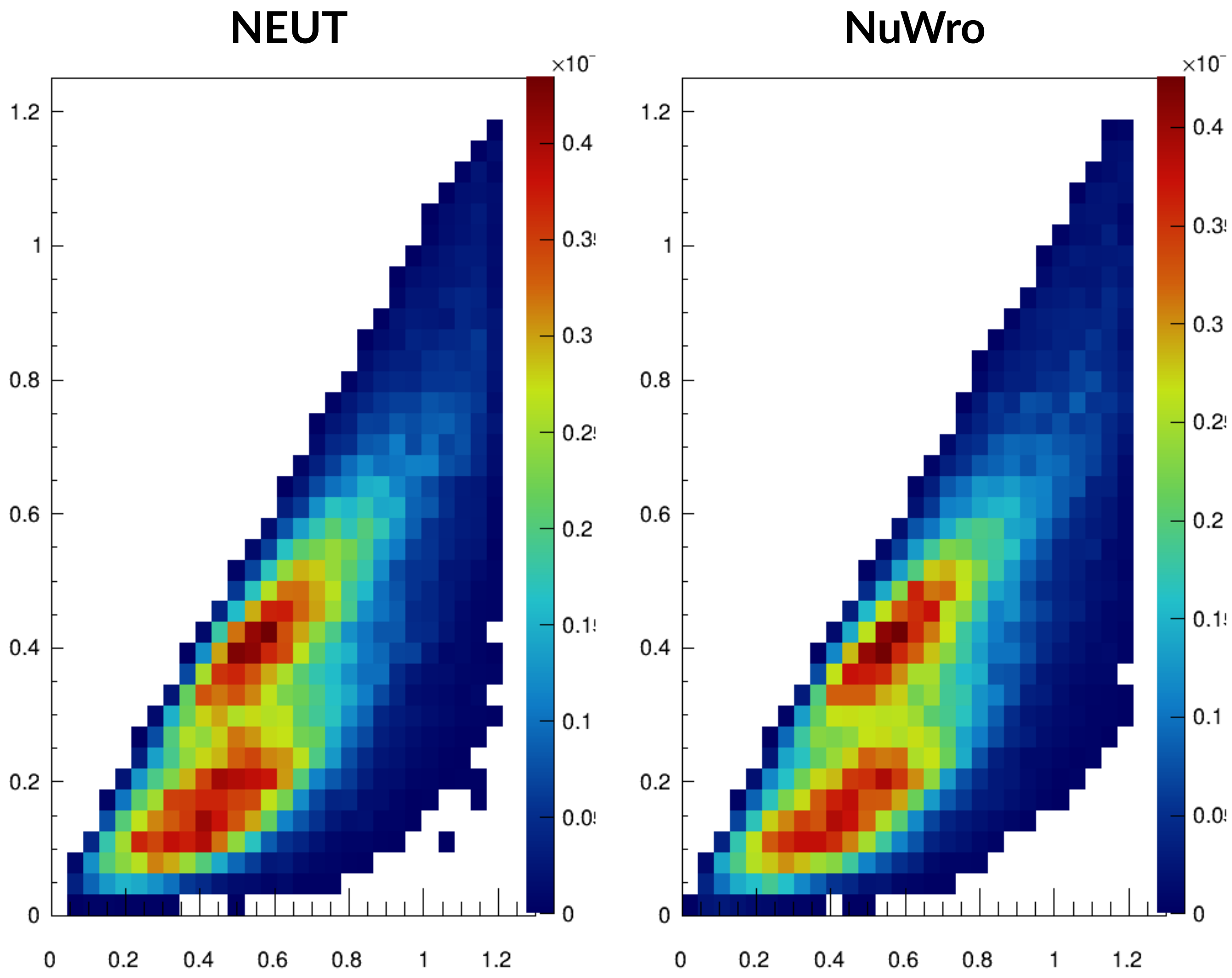
Comparing some generators

- Now draw different generators to your hearts content!

```
root [0] TFile *neut = new
TFile("T2KND_FHC_numu_C8H8_NEUT562_1M_0000_NUISFLAT.root")
root [1] TFile *nuwro = new
TFile("T2KND_FHC_numu_C8H8_NUWRO_LFGRPA_1M_0000_NUISFLAT.root")
root [2] neut->cd()
root [3] FlatTree_VARS->Draw("q0:q3>>h1(30, 0, 1.2)",
"fScaleFactor*1E38*(Mode==2)", "colz")
root [14] TH1D *h1 = (TH1D*)gDirectory->Get("h1")
root [6] nuwro→cd()
root [8] FlatTree_VARS->Draw("q0:q3>>h2(30, 0, 1.2)",
"fScaleFactor*1E38*(Mode==2)", "colz")
root [17] TH1D *h2 = (TH1D*)gDirectory->Get("h2")
root [20] TCanvas *canv = new TCanvas("canv", "canv", 1024, 1024)
root [21] canv->Divide(2)
root [22] canv->cd(1)
root [23] h1->Draw("colz")
root [24] canv->cd(2)
root [25] h2->Draw("colz")
```

Comparing some generators

- 2p2h model in NEUT and NuWro are **very** similar, which makes perfect sense!
  - Both use Valencia 2p2h



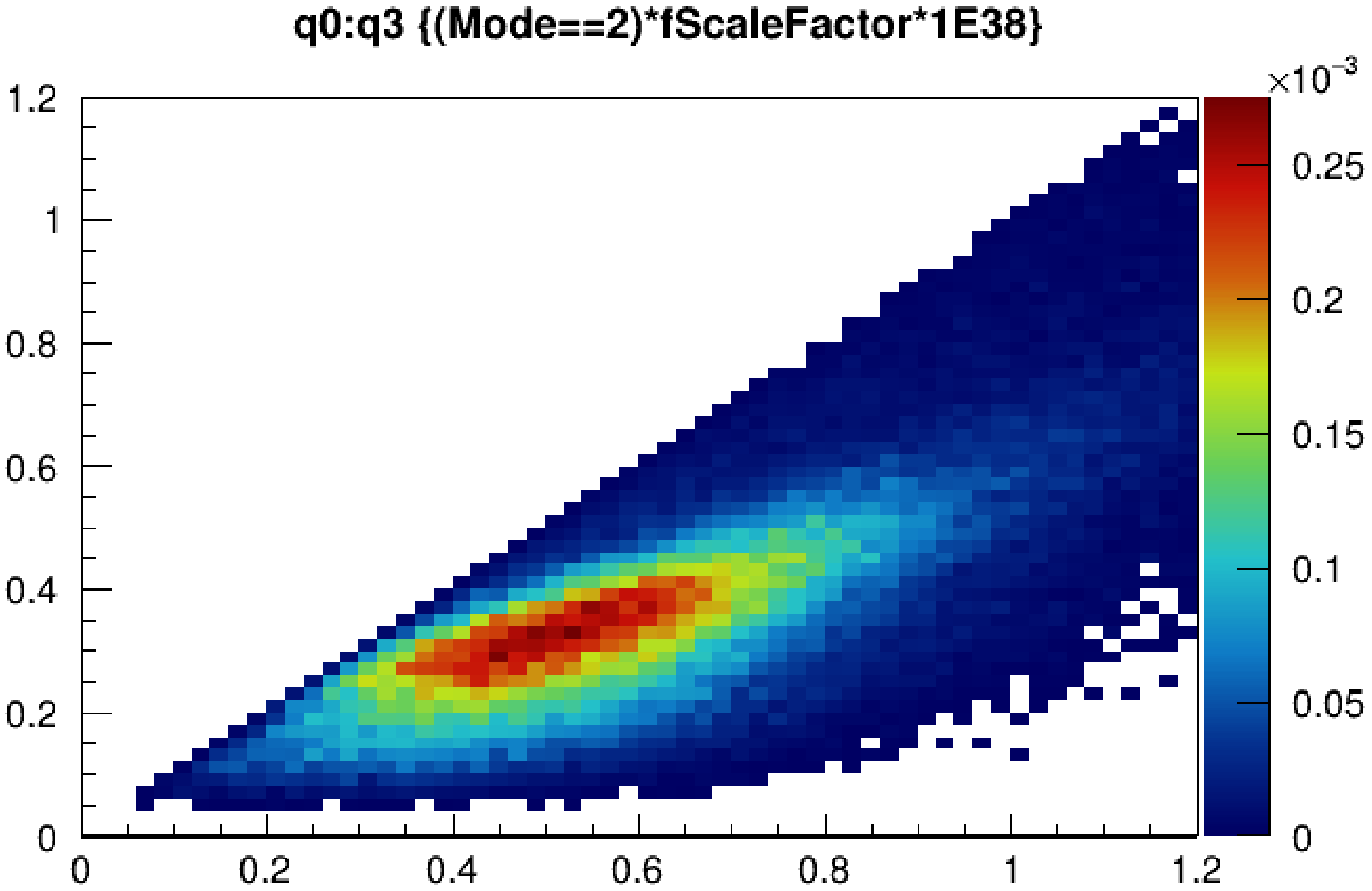
T2K

Comparing some generators

- If instead we use GENIE CRPA, the 2p2h distribution looks very different

T2K

```
root [7] FlatTree_VARS->Draw("q0:q3>>h1(60,0,1.2,60,0,1.2)",
"(Mode==2)*fScaleFactor*1E38", "colz")
```



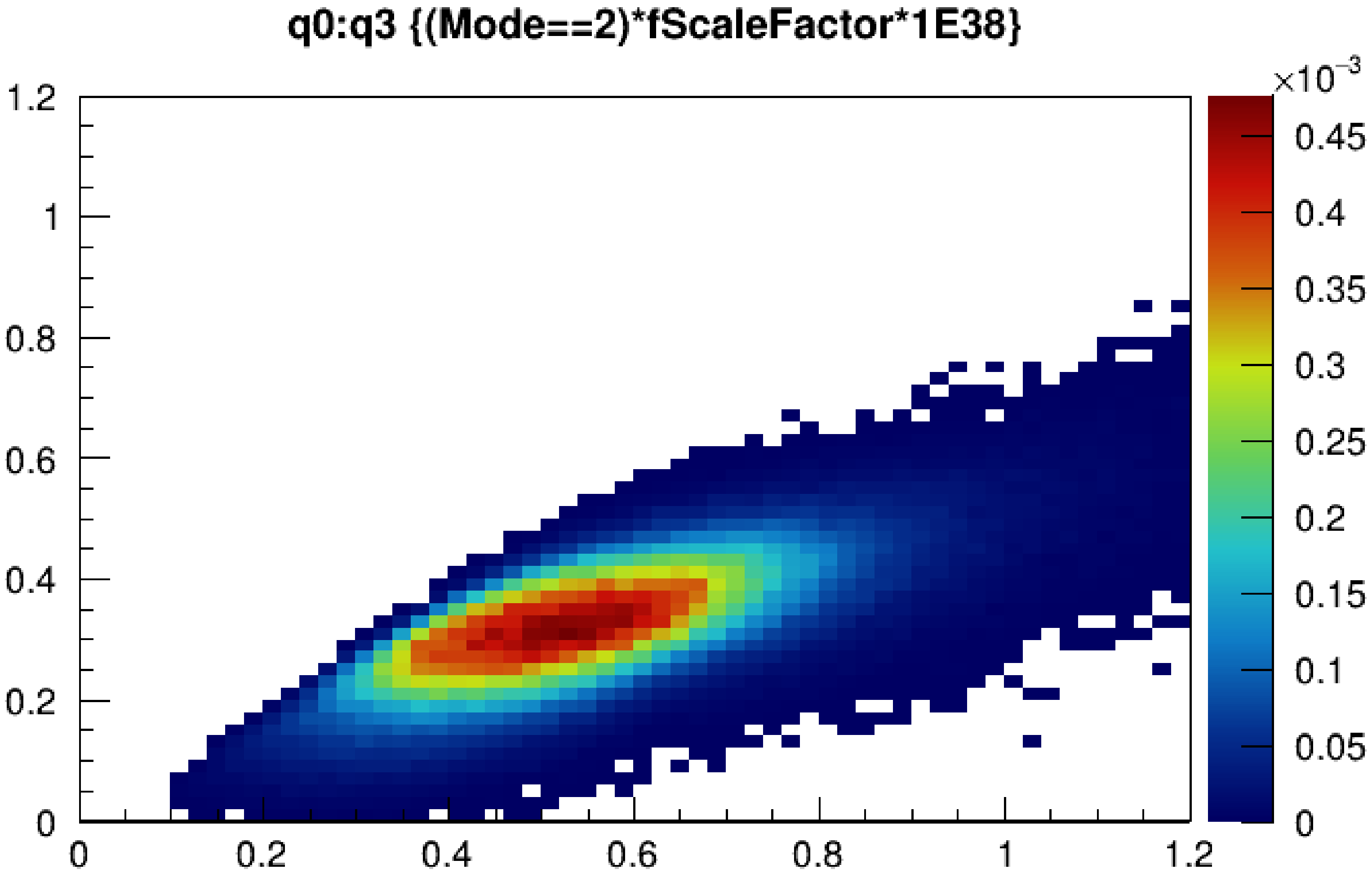
Clarence Wret

Comparing some generators

- Interestingly, good similarity to “Empirical MEC” (aka “Dytman MEC”)

T2K

```
root [7] FlatTree_VARS->Draw("q0:q3>>h1(60,0,1.2,60,0,1.2)",
"(Mode==2)*fScaleFactor*1E38", "colz")
```



Clarence Wret

Hadronic mass exercise

- Let's now try to plot hadronic mass (W) broken down by mode (Mode) and in total for the **DUNE** flux, for modes below 30 (CC)
- DUNE is higher energy → the hadronic mass phase space is populated differently to T2K

Hadronic mass exercise

- Let’s now try to plot hadronic mass (W) broken down by mode (Mode) and in total for the DUNE flux, for modes below 30 (CC)

```
void plot_w(std::string filename) {
  TFile *file = new TFile(filename.c_str(), "open");
  TTree *tree = (TTree*)file->Get("FlatTree_VARS");
  int nModes = 30;
  double scale = tree->GetMaximum("fScaleFactor");

  TH1D **w = new TH1D*[nModes];
  for (int i = 0; i < nModes; ++i) {
    tree->Draw(Form("W>>h%i(50,0.5,3)", i), Form("Mode==%i", i));
    w[i] = (TH1D*)gDirectory->Get(Form("h%i", i));
    w[i]->SetTitle(Form("W (Mode==%i);W (GeV/c^{2});d#sigma/dW
(cm^{2}/(GeV/c^{2})/nucleon)", i));
  }
  tree->Draw("W>>h00(50, 0.5, 3)", "Mode<30");
  TH1D *h0 = (TH1D*)gDirectory->Get("h00");
  h0->Scale(scale, "width");
```

Loop over interaction modes

Make a draw that depends on the iterator

Draw the total

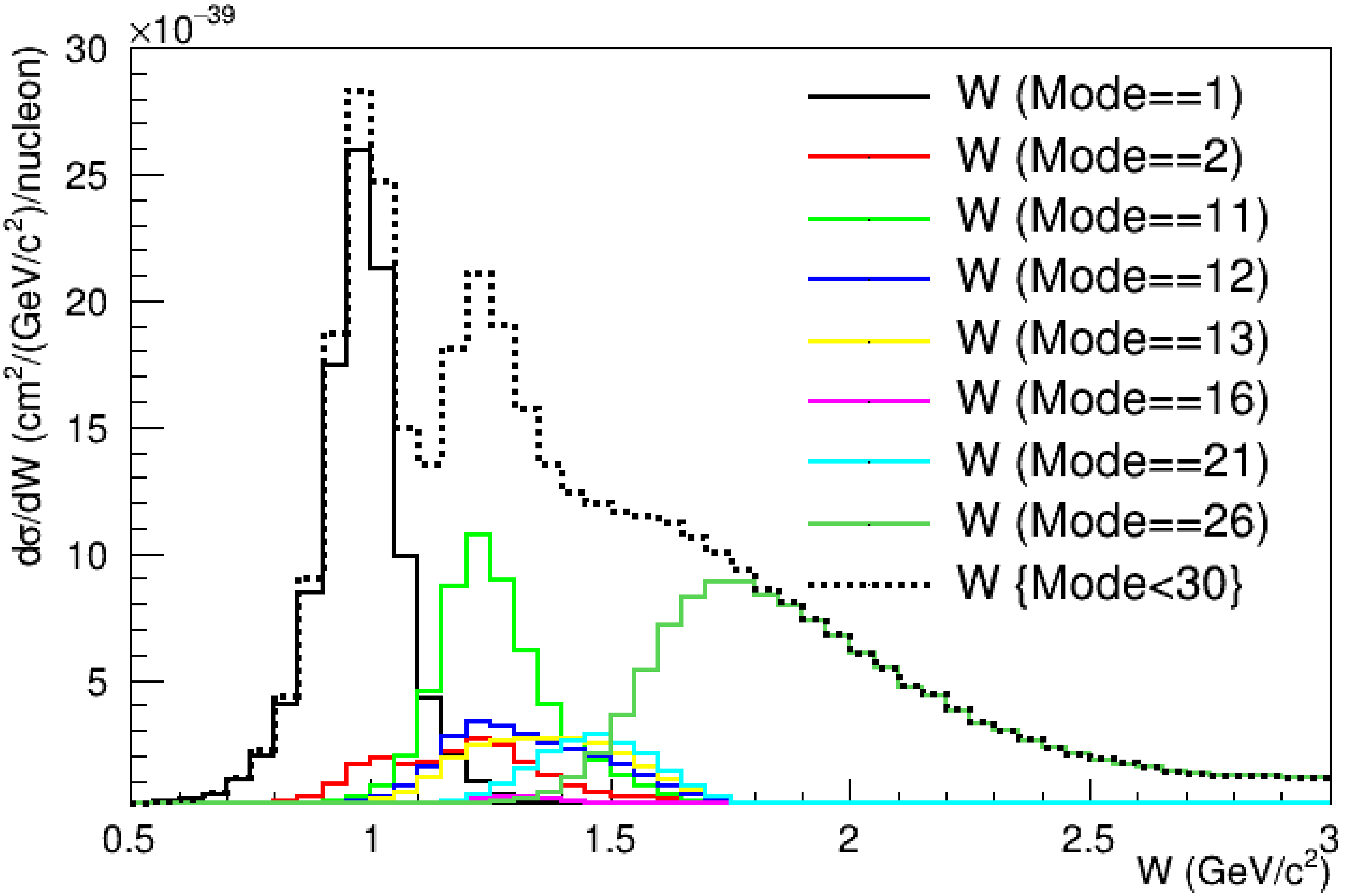
Hadronic mass exercise

- And the rest just prettifies the plot

```
  bool drawn = false;
  double max = 0;
  int first = 0;
  int nhists = 0;
  for (int i = 0; i < nModes; ++i) {
    if (w[i]->Integral() == 0) continue;
    w[i]->Scale(scale, "width");
    max = w[i]->GetMaximum() > max ? w[i]->GetMaximum() : max;
    if (!drawn) {
      first = i;
      w[i]->Draw("hist");
      drawn = true;
    }
    else w[i]->Draw("hist,same");
    w[i]->SetLineColor(nhists+1);
    w[i]->SetMarkerSize(0);
    w[i]->SetLineWidth(2);
    if (nhists == 9) w[i]->SetLineColor(11);
    nhists++;
  }
  h0->Draw("same,hist");
  h0->SetLineColor(kBlack);
  h0->SetLineWidth(3);
  h0->SetLineStyle(kDashed);
  w[first]->GetYaxis()->SetRangeUser(1E-40, 30E-39);
}
```

Hadronic mass exercise

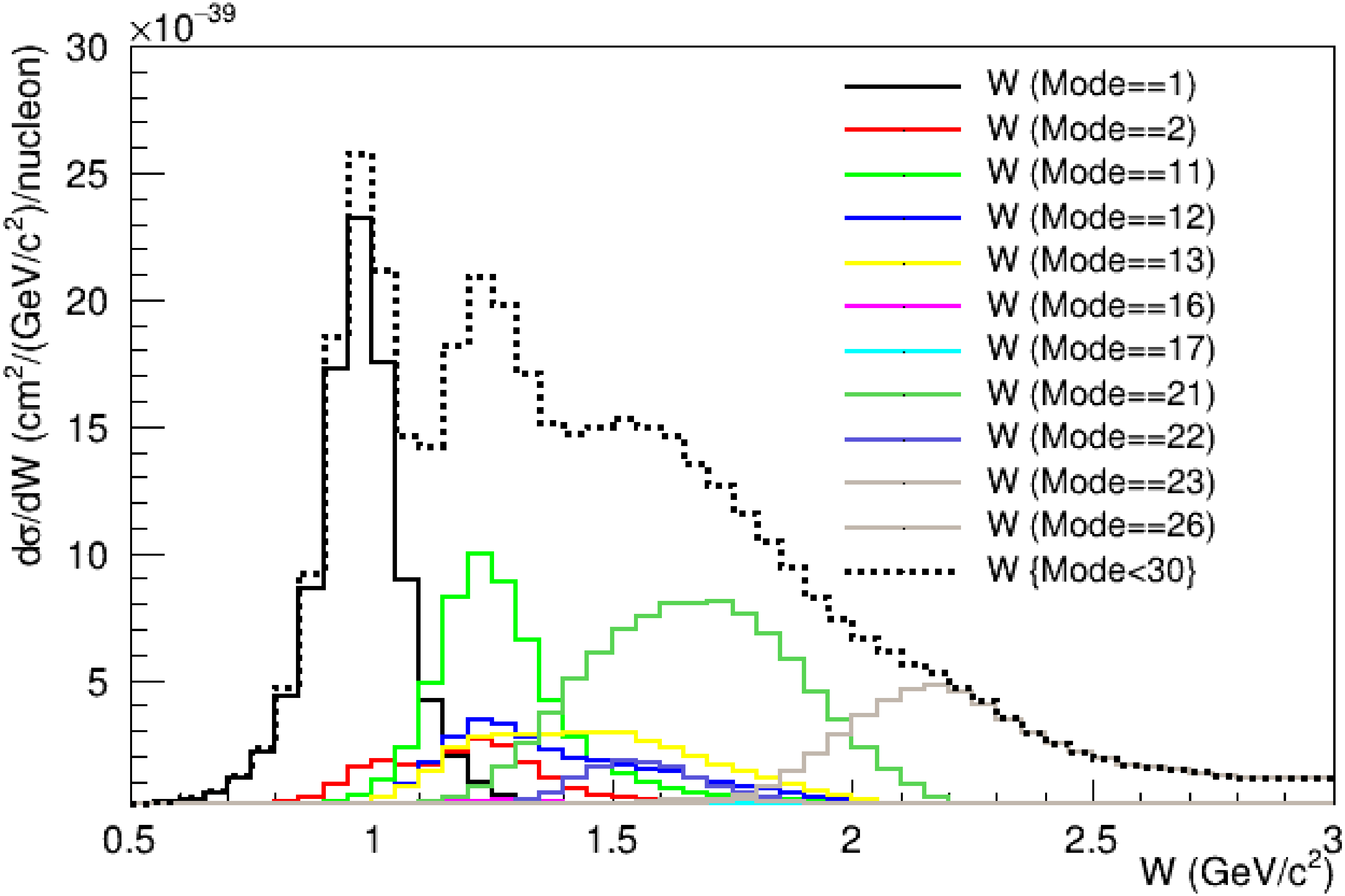
- DUNE flux, NuWro



Hadronic mass exercise

- DUNE flux, NEUT

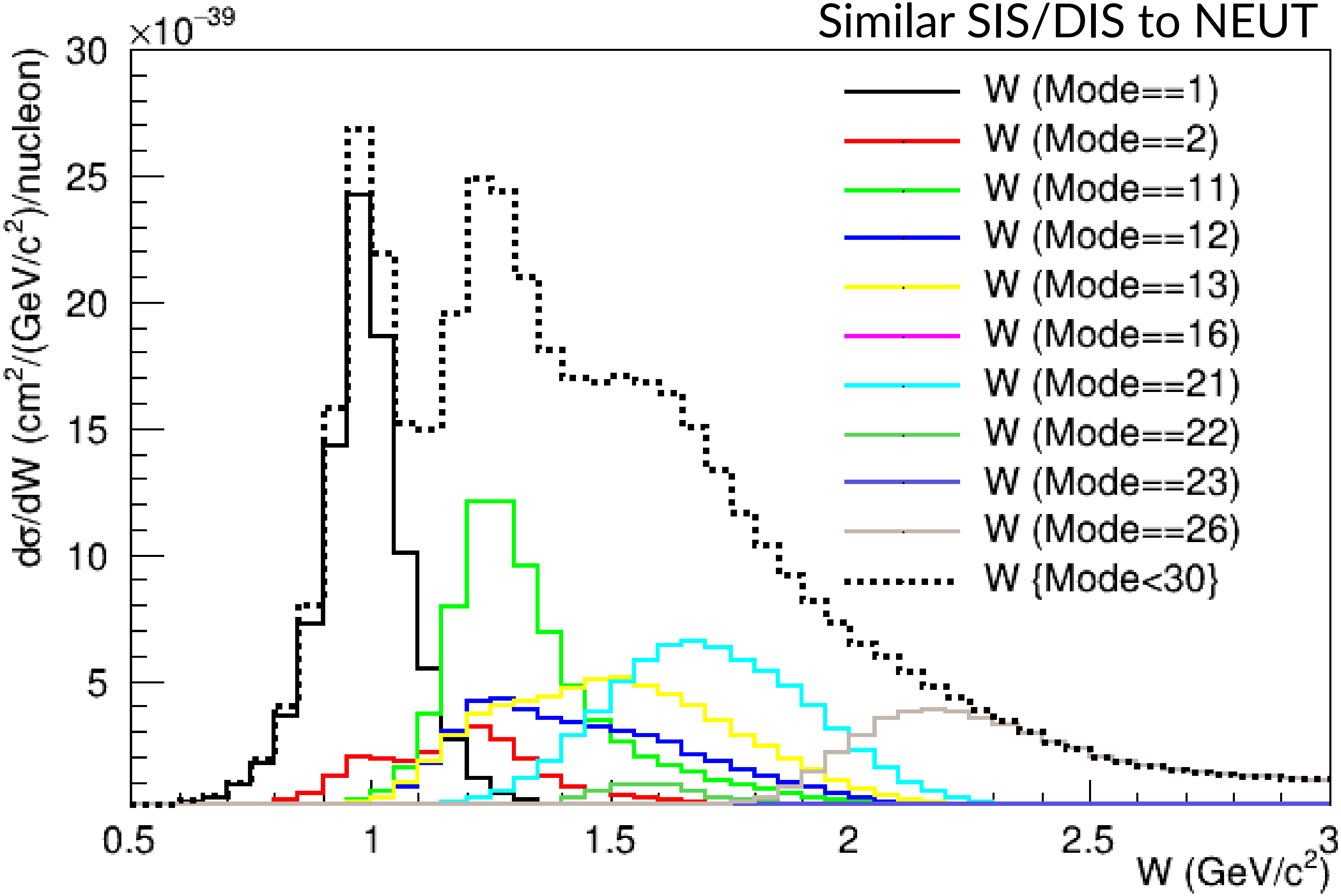
NuWro's SIS quite narrow
NuWro's QE larger



Hadronic mass exercise

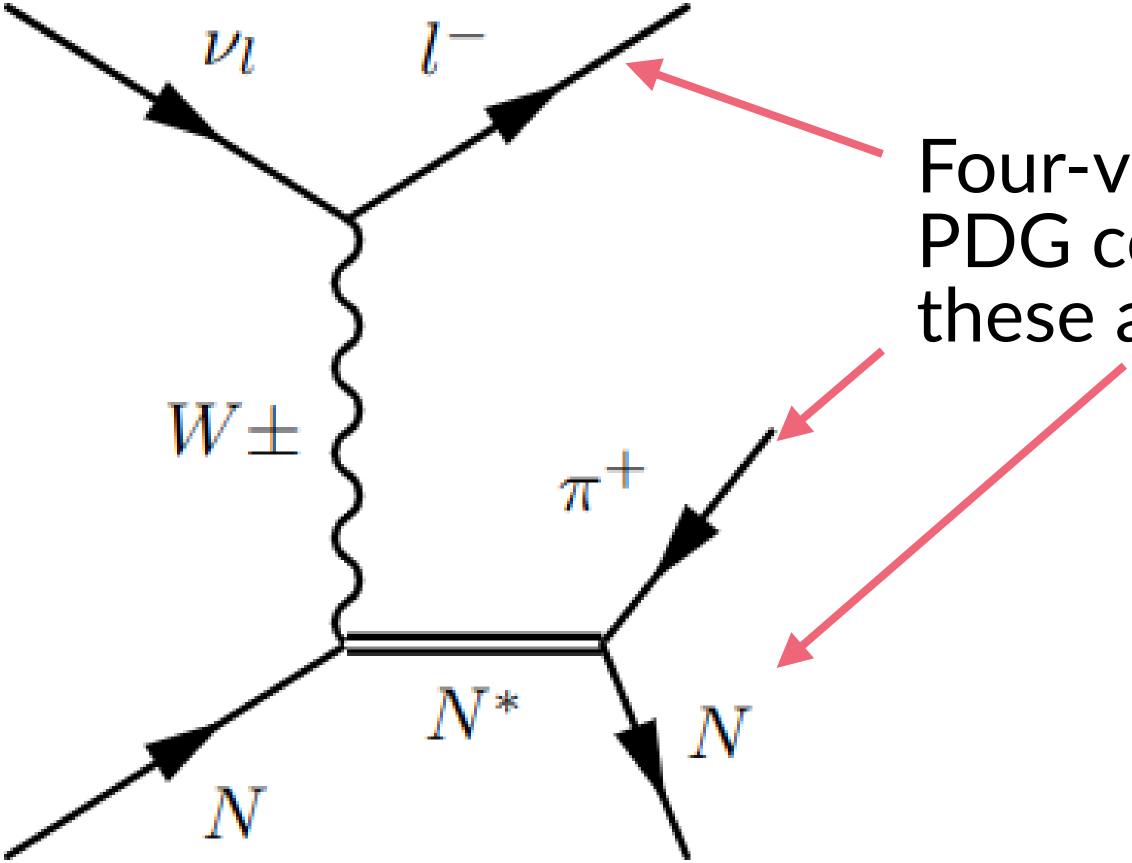
- DUNE flux, GENIE G18 10a

GENIE CC1pi+1p and CC1pi+1n larger
Similar SIS/DIS to NEUT



Inspecting the particle stack

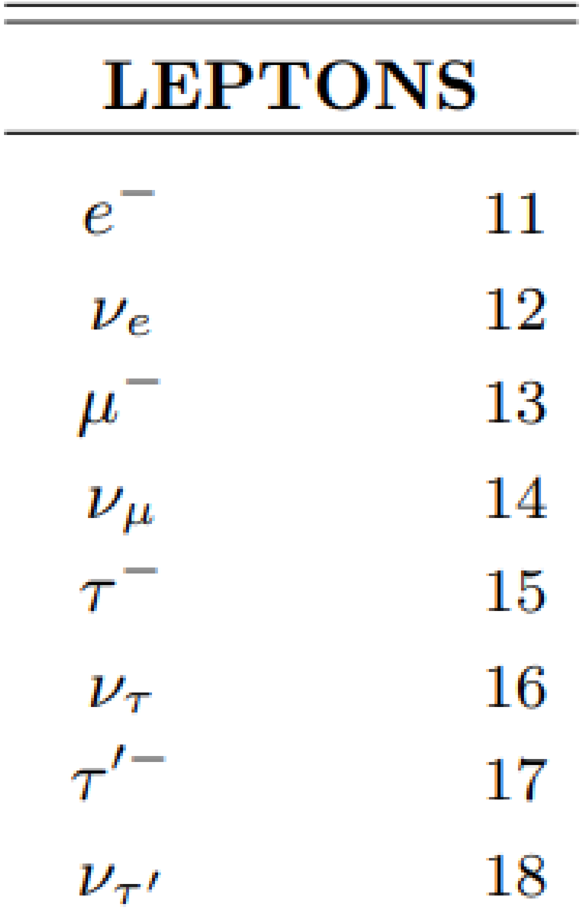
- At this point we're getting comfortable with simple draws directly from the TTree
- Let's investigate the particle stacks, through which you can explore generators deeper



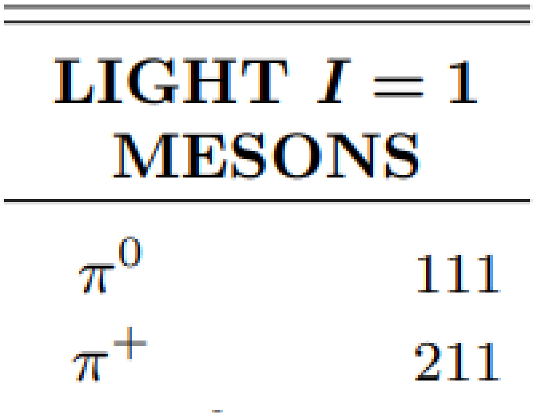
Four-vectors and PDG codes of these are saved

PDG codes

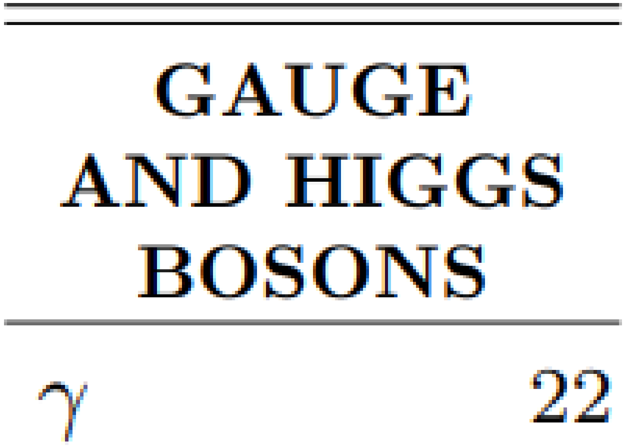
- PDG codes: https://pdg.lbl.gov/2020/reviews/rpp2020-rev-monte-carlo-numbering.pdf
- Here are the important ones for today:



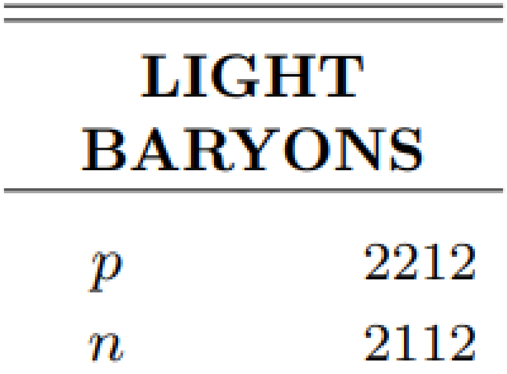
LEPTONS	
e^-	11
ν_e	12
μ^-	13
ν_μ	14
τ^-	15
ν_τ	16
τ'^-	17
$\nu_{\tau'}$	18



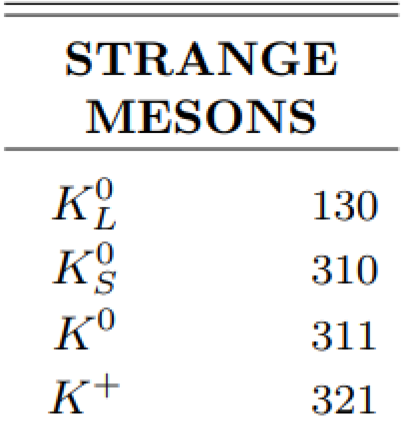
LIGHT $I = 1$ MESONS	
π^0	111
π^+	211



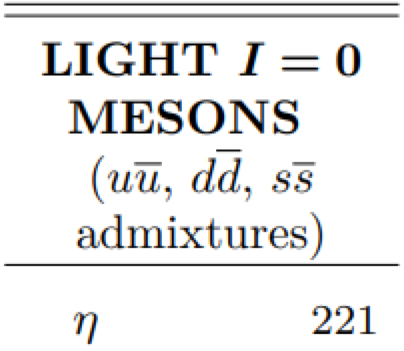
GAUGE AND HIGGS BOSONS	
γ	22



LIGHT BARYONS	
p	2212
n	2112



STRANGE MESONS	
K_L^0	130
K_S^0	310
K^0	311
K^+	321



LIGHT $I = 0$ MESONS ($u\bar{u}$, $d\bar{d}$, $s\bar{s}$ admixtures)	
η	221

Drawing pion momentum

- The following examples use ROOT and reading a TTree
- We'll start slowly
- There are other ways of doing this too, but I think this is the fastest in terms of reading a TTree
  - You can adapt it to use pyROOT instead, or different TTree interface
- If this is completely new, see the ROOT documentation on TTree: https://root.cern.ch/root/htmldoc/guides/users-guide/Trees.html
- We'll start with drawing the pion momentum of single pion production events, then lots more

Drawing pion momentum

- Following code is committed to tutorial repository:
  - https://github.com/NUISANCEMC/tutorials/blob/main/interactive/flat_pion_mom.cpp

- We’ll walk through it in detail, and then try to build on it for other studies

- All of these scripts are simple “root macros”
- Run by doing:

```
root -l ‘flat_pion_mom.cpp(“FLATTREE_FILE.root”)’
```

- Where “`FLATTREE_FILE.root`” is one of the flat tree files you’ve downloaded earlier (or got on the USB stick)

Drawing pion momentum

```
void flat_pion_mom(std::string filename) {
  TFile *file = new TFile(filename.c_str(), "open");
  TTree *tree = (TTree*)file->Get("FlatTree_VARS");
  int nentries = tree→GetEntriesFast();

  const int nmax = 100;
  float E[nmax];
  float px[nmax];
  float py[nmax];
  float pz[nmax];
  int pdg[nmax];

  int Mode;
  int nfsp;
  tree->SetBranchStatus("*", false);
  tree->SetBranchStatus("E", true);
  tree->SetBranchAddress("E", &E);
  tree->SetBranchStatus("px", true);
  tree->SetBranchAddress("px", &px);
  tree->SetBranchStatus("py", true);
  tree->SetBranchAddress("py", &py);
```

Setting things up as usual when doing TTree analyses

(There are other ways of doing this too, but I think this is the fastest)

Please ask questions if this isn't clear

The energy, momentum and PDG code of the outgoing particles

Drawing pion momentum

```
  tree->SetBranchStatus("pz", true);
  tree->SetBranchAddress("pz", &pz);
  tree->SetBranchStatus("pdg", true);
  tree->SetBranchAddress("pdg", &pdg);
  tree->SetBranchStatus("Mode", true);
  tree->SetBranchAddress("Mode", &Mode);
  tree->SetBranchStatus("nfsp", true);
  tree->SetBranchAddress("nfsp", &nfsp);
  tree->SetBranchStatus("fScaleFactor", true);
  double scale = tree->GetMaximum("fScaleFactor");
  tree->SetBranchStatus("fScaleFactor", false);

  int nModes = tree->GetMaximum("Mode");

  TH1D **ppi = new TH1D*[nModes];
  for (int i = 0; i < nModes; ++i) {
    ppi[i] = new TH1D(Form("ppi_%i", i), Form("ppi_%i;p_{#pi}
(GeV/c);d#sigma/dp_{#pi} (cm^{2}/(GeV/c)/nucleon)", i), 50, 0,
1);
  }
  TH1D *ppit = new TH1D("ppit", "ppit;p_{#pi}
(GeV/c);d#sigma/dp_{#pi} (cm^{2}/(GeV/c)/nucleon)", 50, 0, 2);
```

Continuing to set up branch addresses

True interaction mode, number of final state particles..

The histograms we'll be writing to

Let's plot them per interaction mode, and total

Drawing pion momentum

Loop over the particles

```
  for (int i = 0; i < nentries; ++i) {
    tree->GetEntry(i);
    if (i % (nentries/5) == 0) std::cout << i << "/" << nentries 
<< " (" << (double(i)/nentries)*100 << "%)" << std::endl;
    int nPions = 0;
    int PiIndex = 0;
    int nMuons = 0;
    for (int j = 0; j < nfsp; ++j) {
      if (pdg[j] == 211) {
        nPions++;
        PiIndex = j;
      }
      if (pdg[j] == 13) nMuons++;
    }
    if (nPions != 1) continue;
    if (nMuons != 1) continue;
    double ptot = sqrt(px[PiIndex]*px[PiIndex]
+py[PiIndex]*py[PiIndex]+pz[PiIndex]*pz[PiIndex]);
    ppi[Mode]→Fill(ptot);
    ppit->Fill(ptot);
  }
```

Get the ith entry

Now all of our particle stacks, PDG codes etc are filled!

Loop over the final state particles check when
* it's a pion
* it's a muon

Require 1 muon and 1 pion in the final state

Calculate the momentum and fill histogram

Drawing pion momentum

```
  bool drawn = false;
  double max = 0;
  int first = 0;
  int nhists = 0;
  for (int i = 0; i < nModes; ++i) {
    if (ppi[i]->Integral() == 0) continue;
    ppi[i]->Scale(scale, "width");
    max = ppi[i]->GetMaximum() > max ? ppi[i]->GetMaximum() : max;
    if (!drawn) {
      first = i;
      ppi[i]->Draw("hist");
      drawn = true;
    }
    else ppi[i]->Draw("hist,same");
    ppi[i]->SetLineColor(nhists+1);
    ppi[i]->SetMarkerSize(0);
    if (nhists == 9) ppi[i]->SetLineColor(11);
    nhists++;
  }
  ppit->Scale(scale, "width");
  ppi[first]->GetYaxis()->SetRangeUser(0, ppit->GetMaximum()*1.2);
  ppit->Draw("same,hist");
  ppit->SetLineColor(kBlack);
  ppit->SetLineWidth(3);
  ppit->SetLineStyle(kDashed);
  ppit->SetMarkerSize(0);
}
```

Mostly make the histograms easy to read

Scale to a cross section

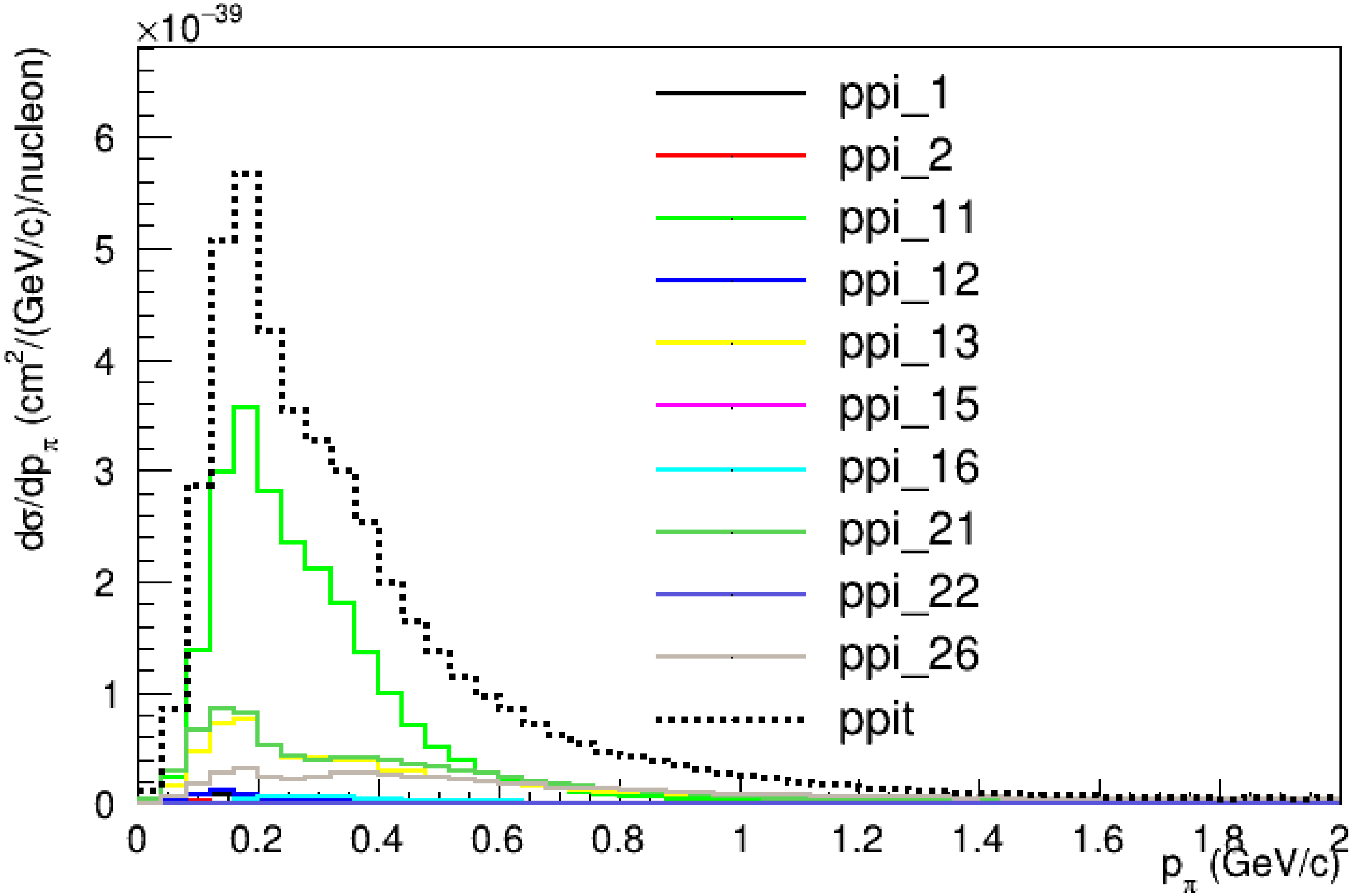
Get maximum, set ranges, and so on

Set the colour to be something readable

Make the total histogram “pretty”

Drawing pion momentum

- The output is: CC1μ^-1π^+ events in pion momentum, broken down by true interaction mode
- Single pion production dominated by CC1π^+1p, then CC1π^+1n, then multi-pi production, then DIS, then CC1π^0



Clarence Wret

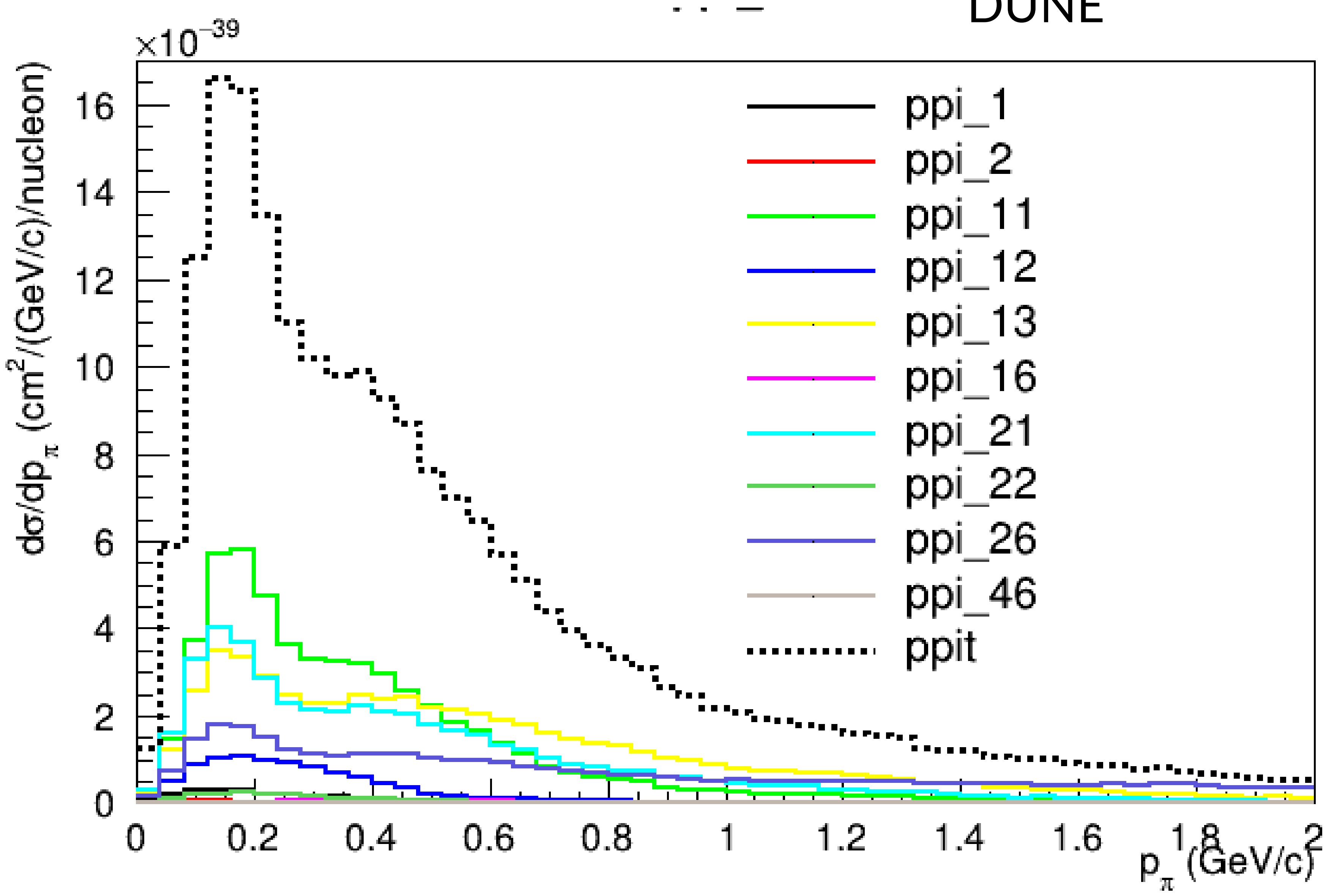
Some questions for you

- How can CC1π^0, CCQE and 2p2h states give 1π final states?
- Do the other generators look the same?
- What about for DUNE?

Some questions for you

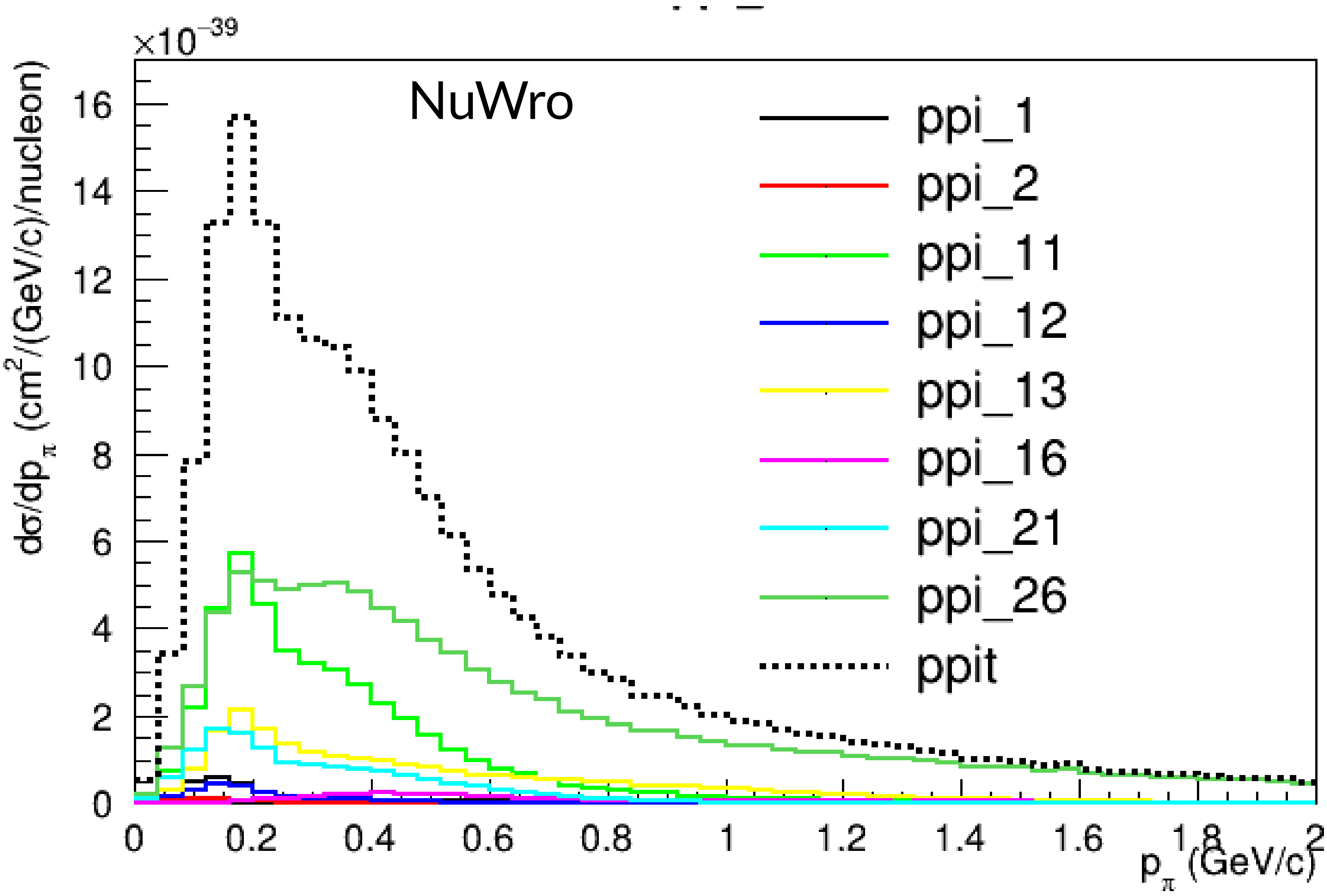
- What about for DUNE?

GENIE G18 10a
DUNE



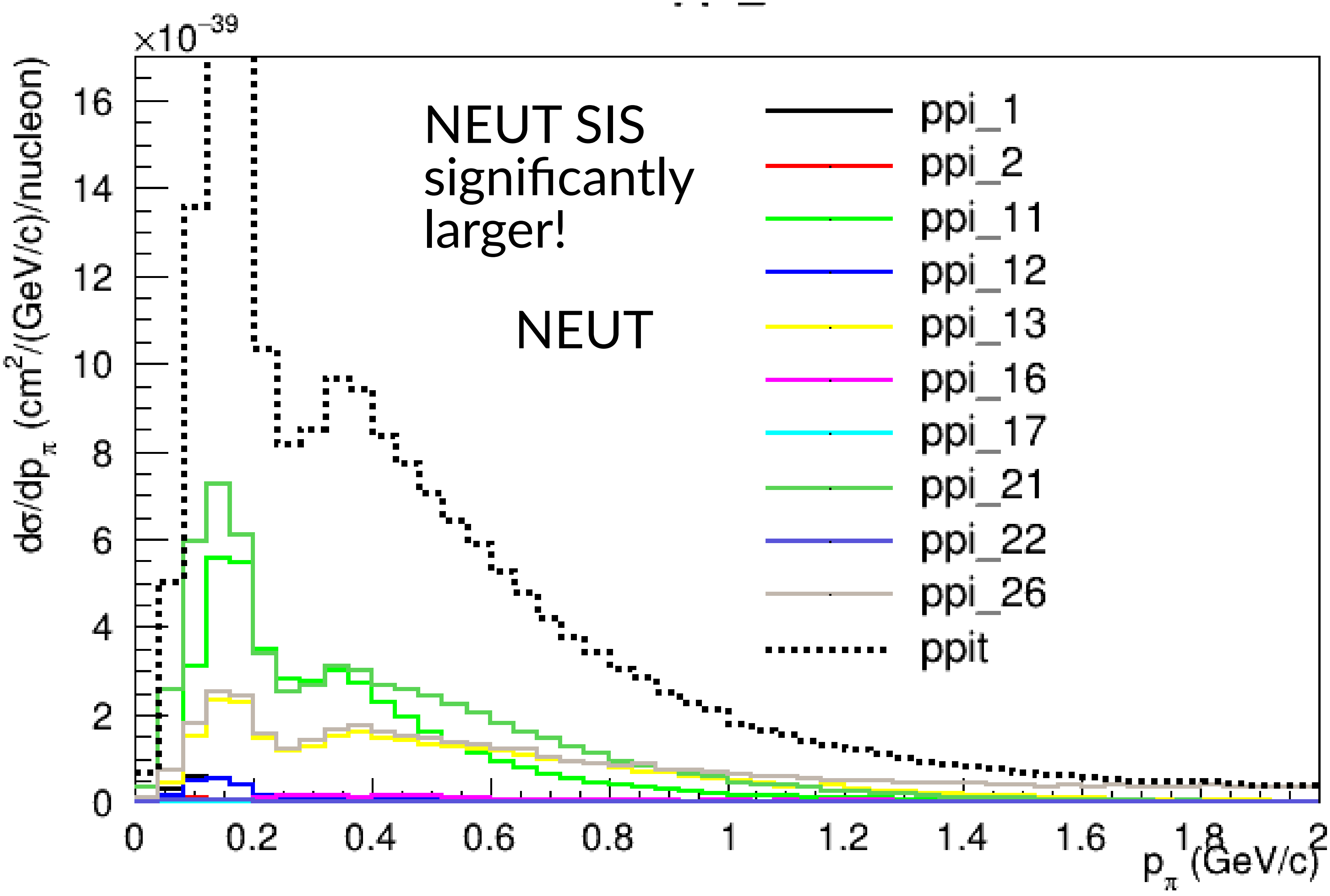
Some questions for you

- What about for DUNE?



Some questions for you

- What about for DUNE?



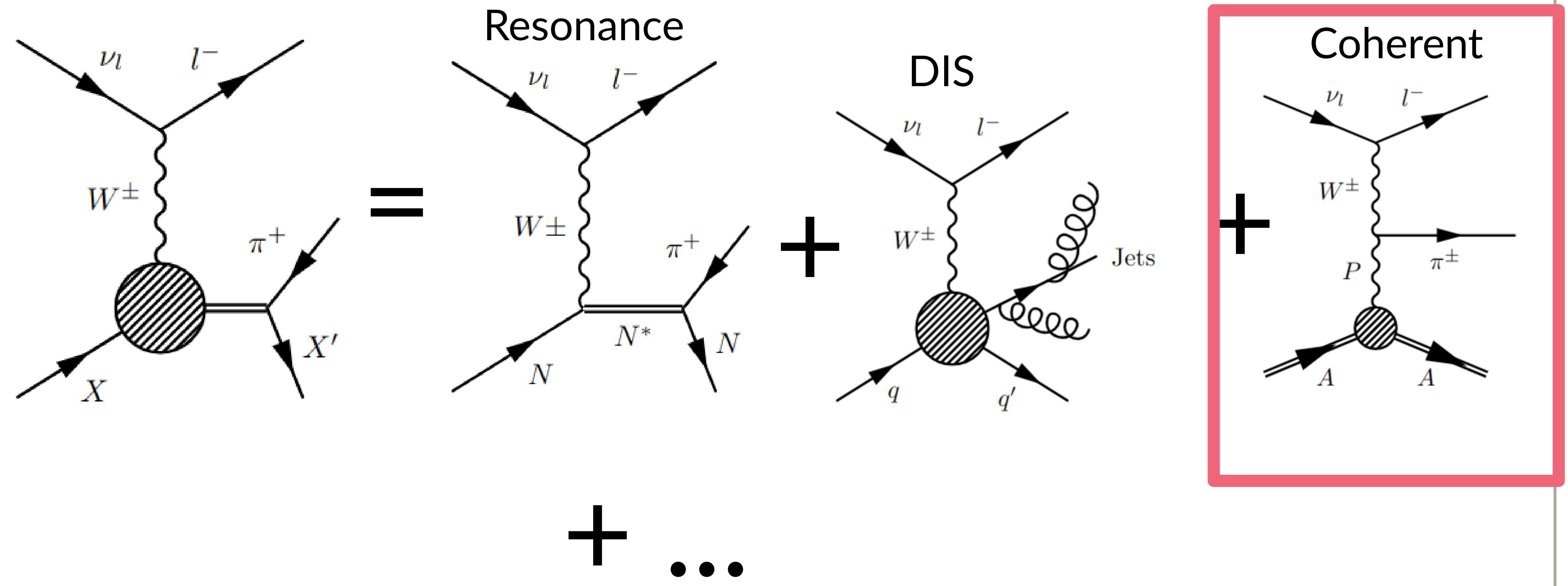
Isolating CC coherent

- Modifying previous code, what do you need to measure a CC coherent event?
  - Hint 1: Does the pion have any particular kinematics?
  - Hint 2: Deborah mentioned this in yesterday's talk
  - Hint 3: You will have to change variable from pion momentum to something else to do with the pion and another particle

Isolating CC coherent

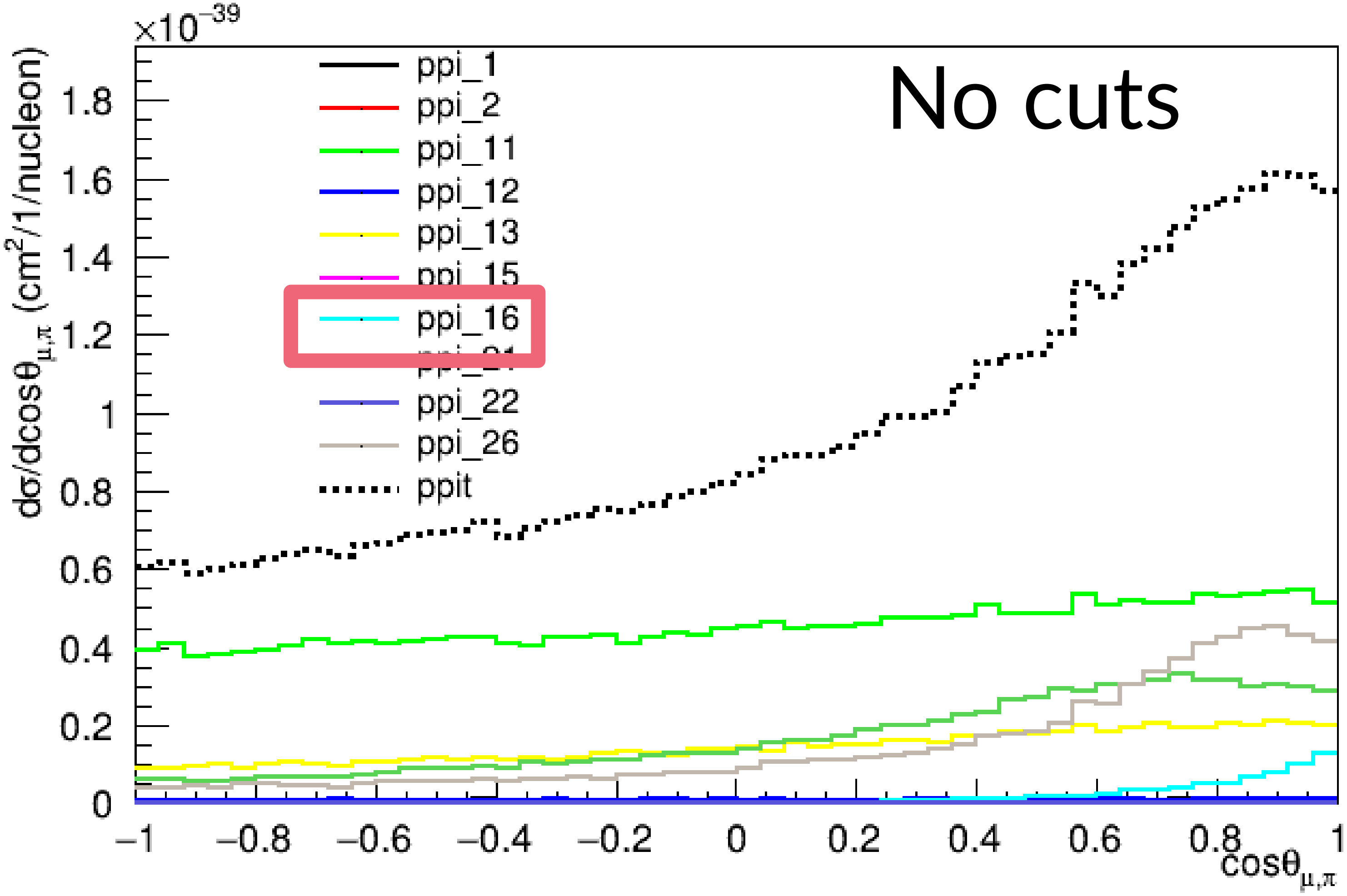
- Modifying previous code, what do you need to measure a CC coherent event?
  - Can you think of putting in some cut on particle momentum to improve coherent contributions?

Our 1mu- 1pi+ selection



Isolating CC coherent

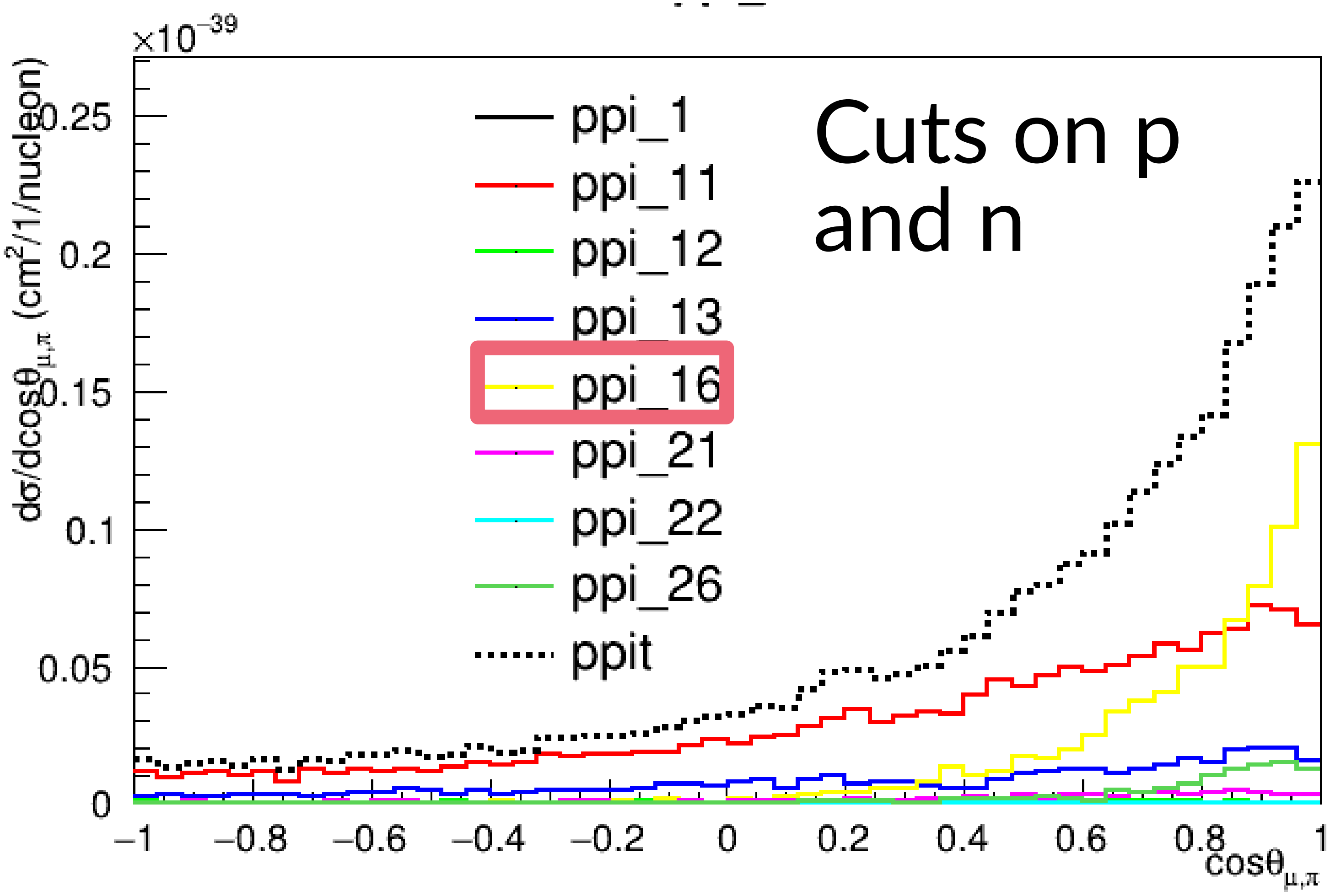
- Modifying previous code, what do you need to measure a CC coherent event?
  - Can you think of putting in some cut on particle momentum to improve coherent contributions?



No cuts

Isolating CC coherent

- Modifying previous code, what do you need to measure a CC coherent event?
  - Can you think of putting in some cut on particle momentum to improve coherent contributions?



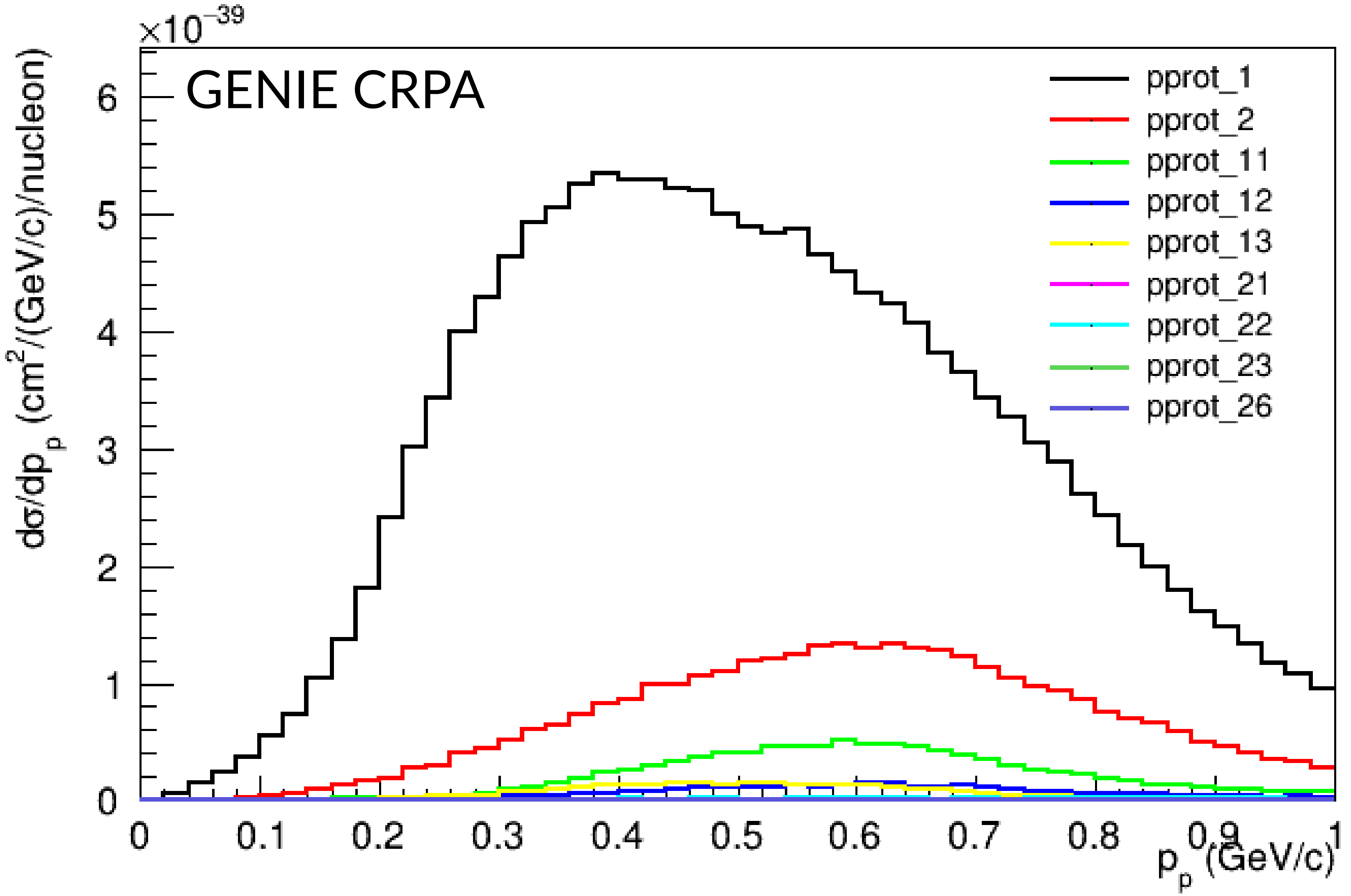
Cuts on p and n

Highest momentum proton

- Try modifying the script to produce the highest momentum proton in a CC0π selection

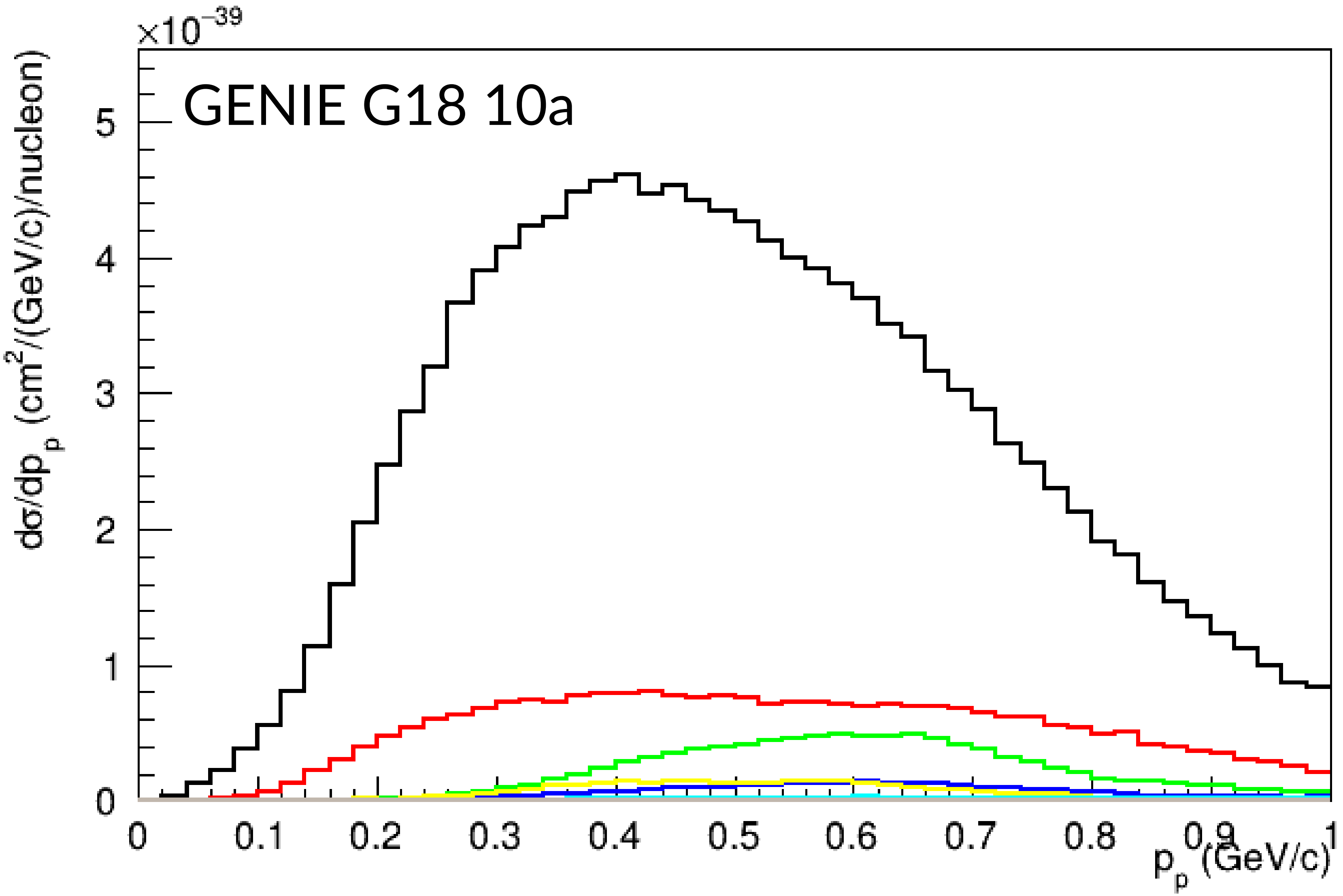
Highest momentum proton

- Try modifying the script to produce the highest momentum proton in a CC0π selection



Highest momentum proton

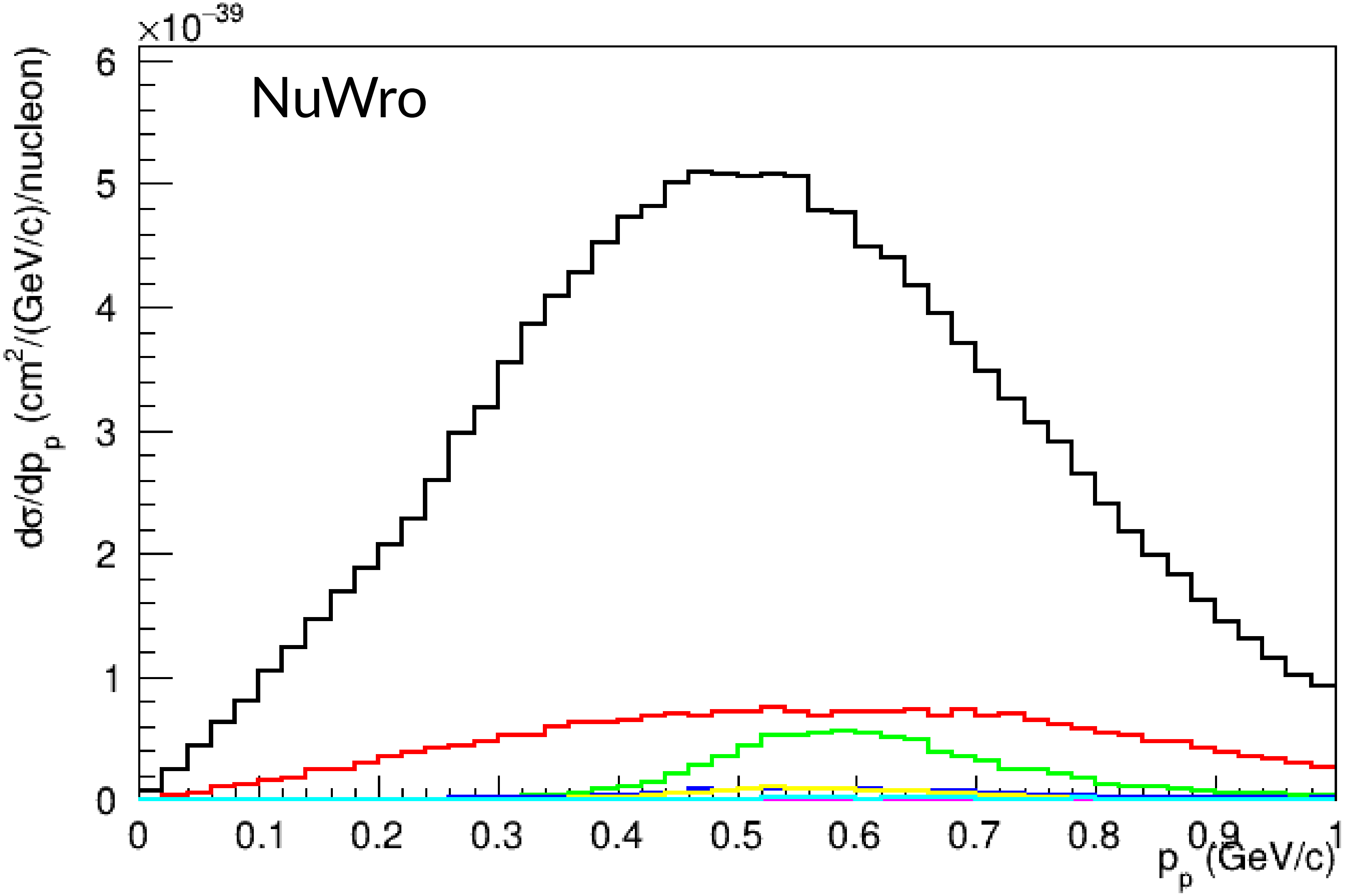
- Try modifying the script to produce the highest momentum proton in a CC0π selection



Clarence Wret

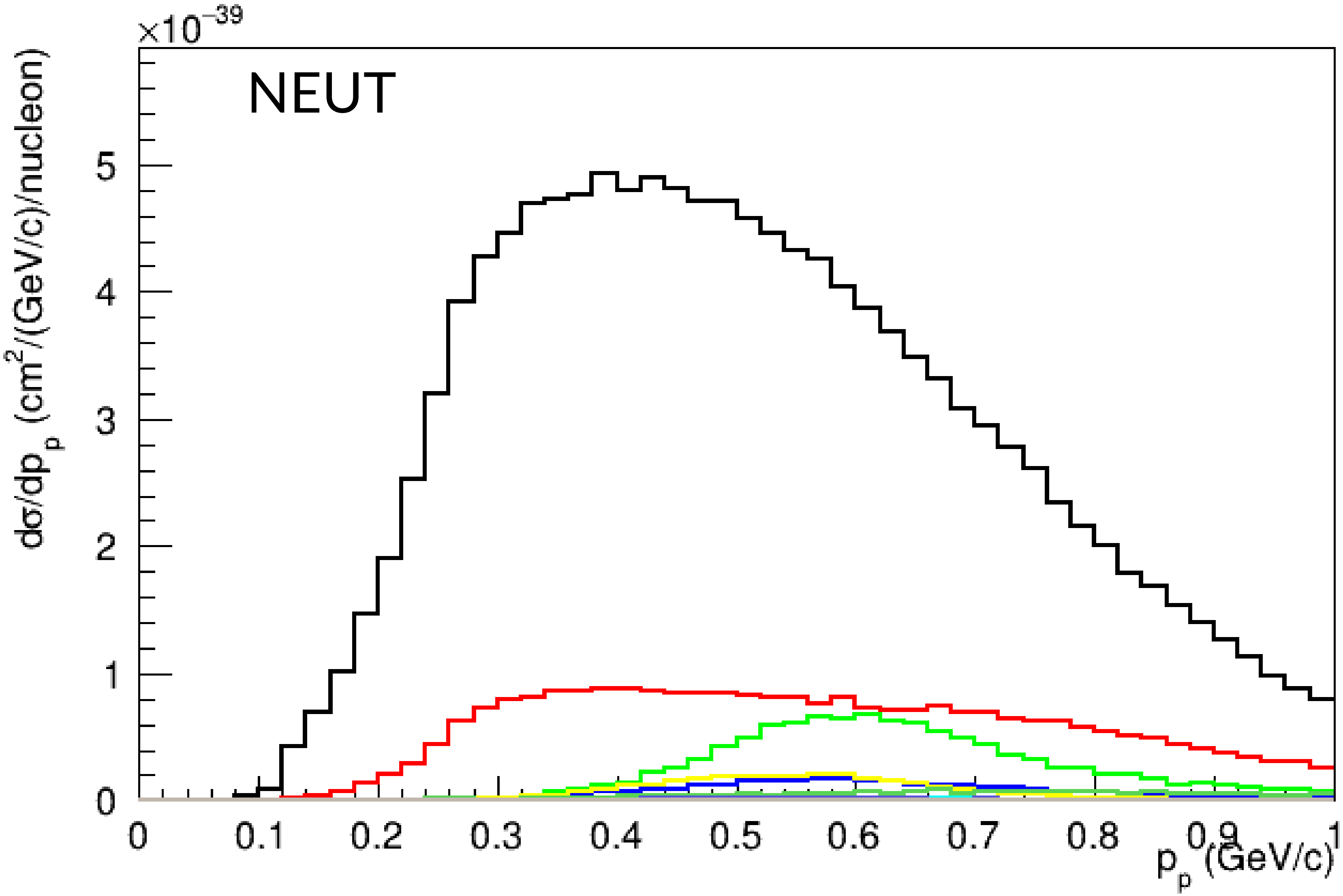
Highest momentum proton

- Try modifying the script to produce the highest momentum proton in a CC0π selection



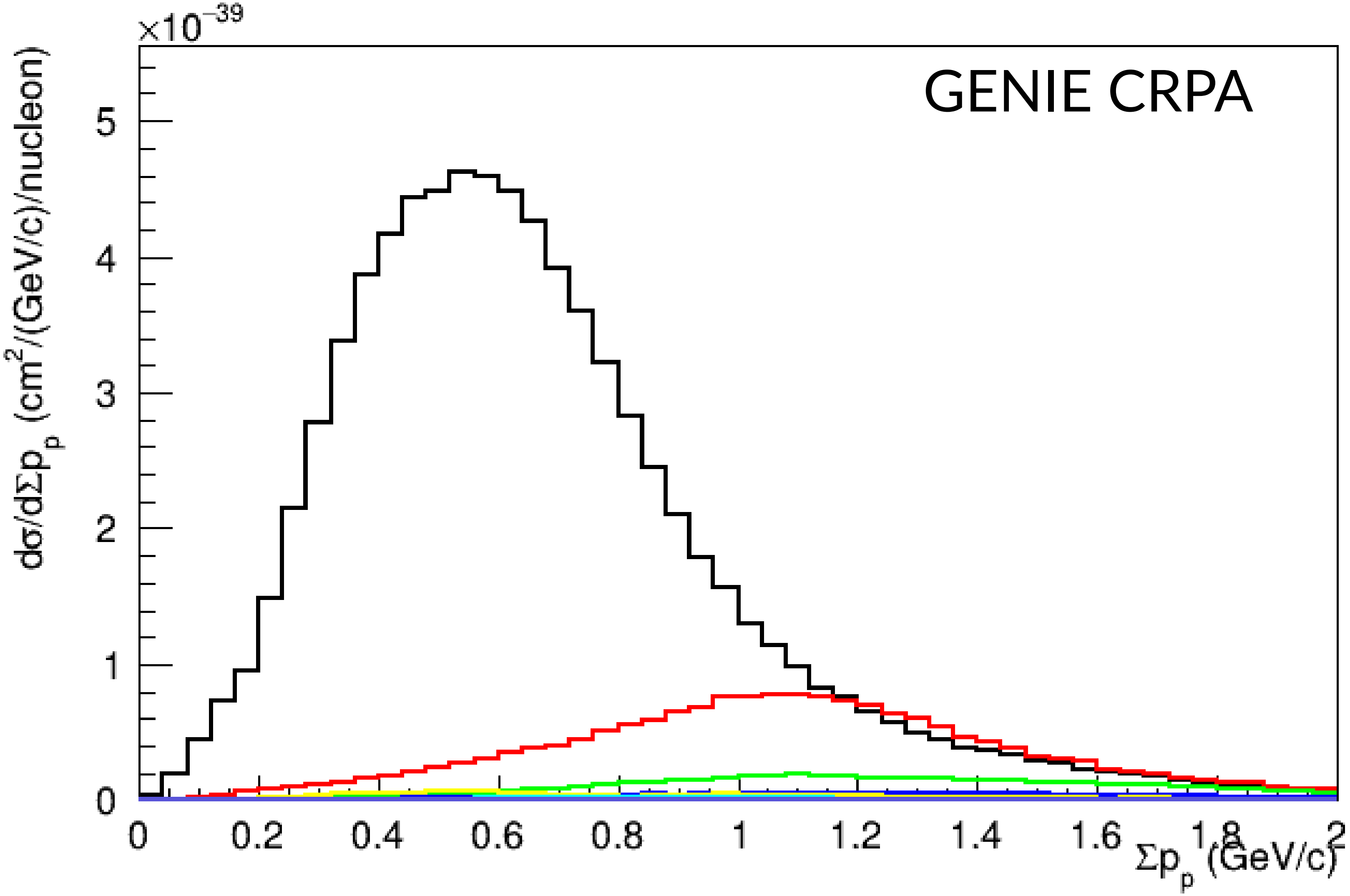
Highest momentum proton

- Try modifying the script to produce the highest momentum proton in a CC0π selection



Sum of proton momentum

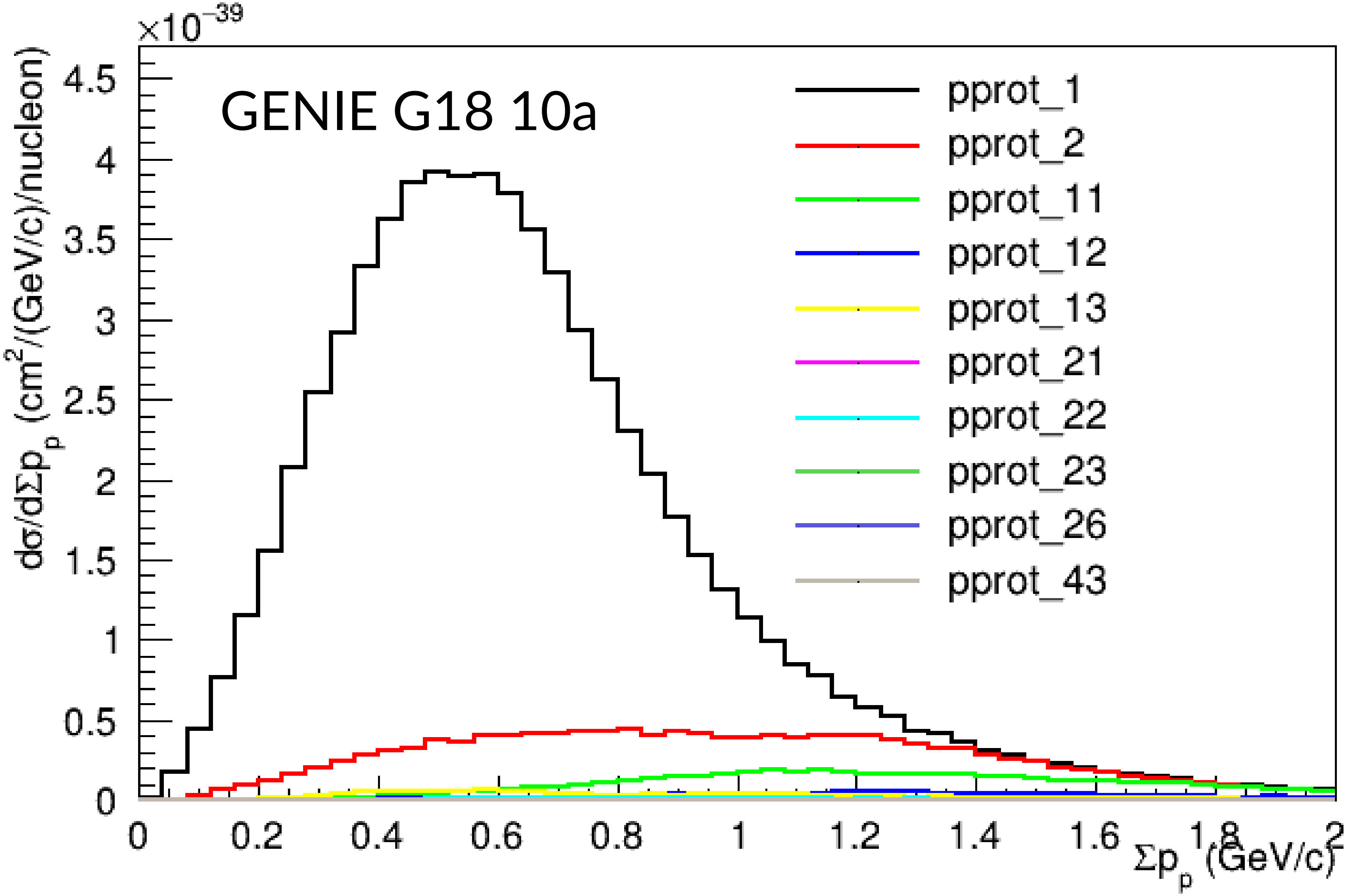
- Looked a bit different; maybe it’s because multi-nucleon systems: try sum of proton momentum
- Sum of proton momentum in a CC0π selection



Clarence Wret

Sum of proton momentum

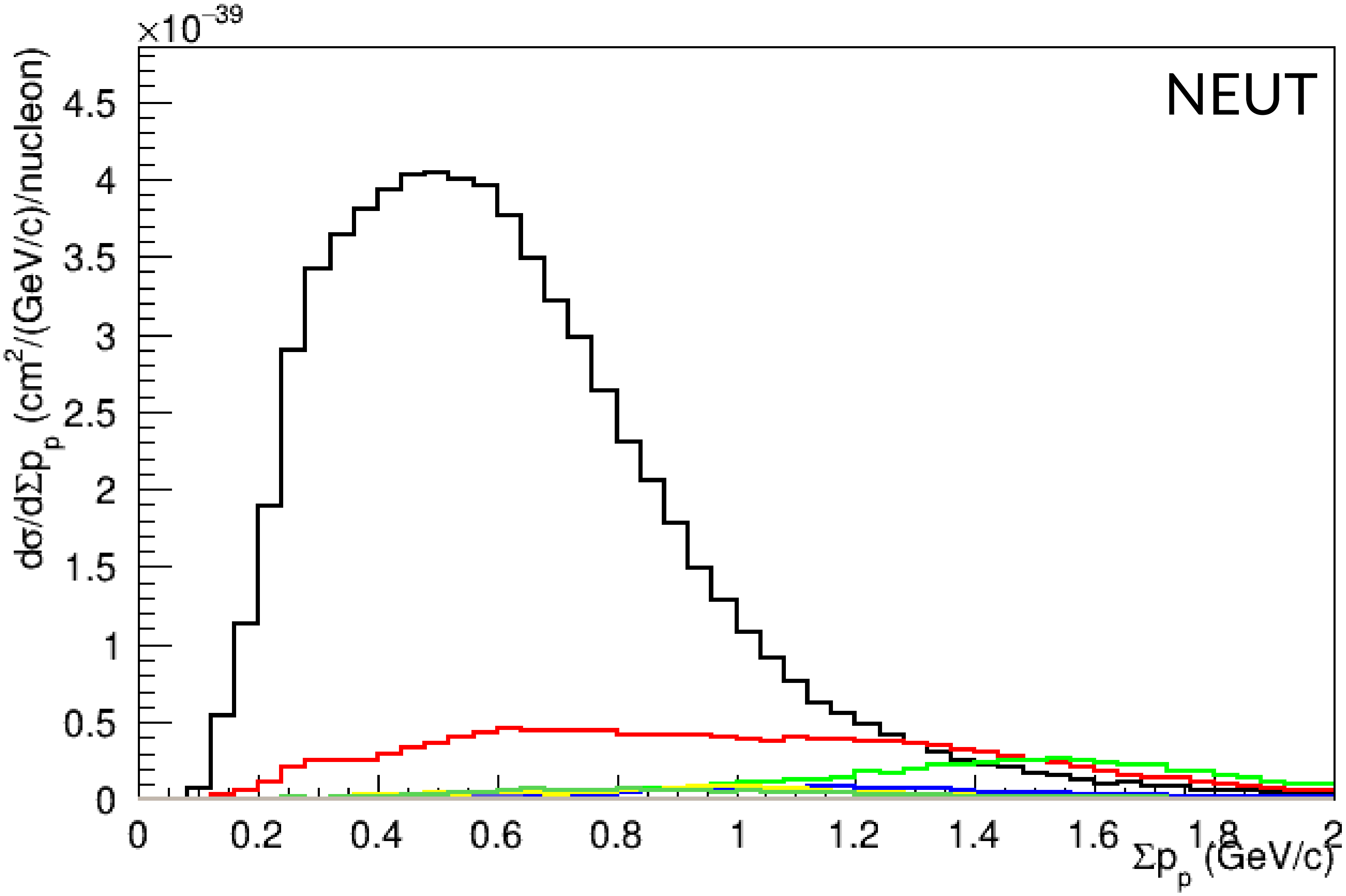
- Sum of proton momentum in a CC0π selection



Clarence Wret

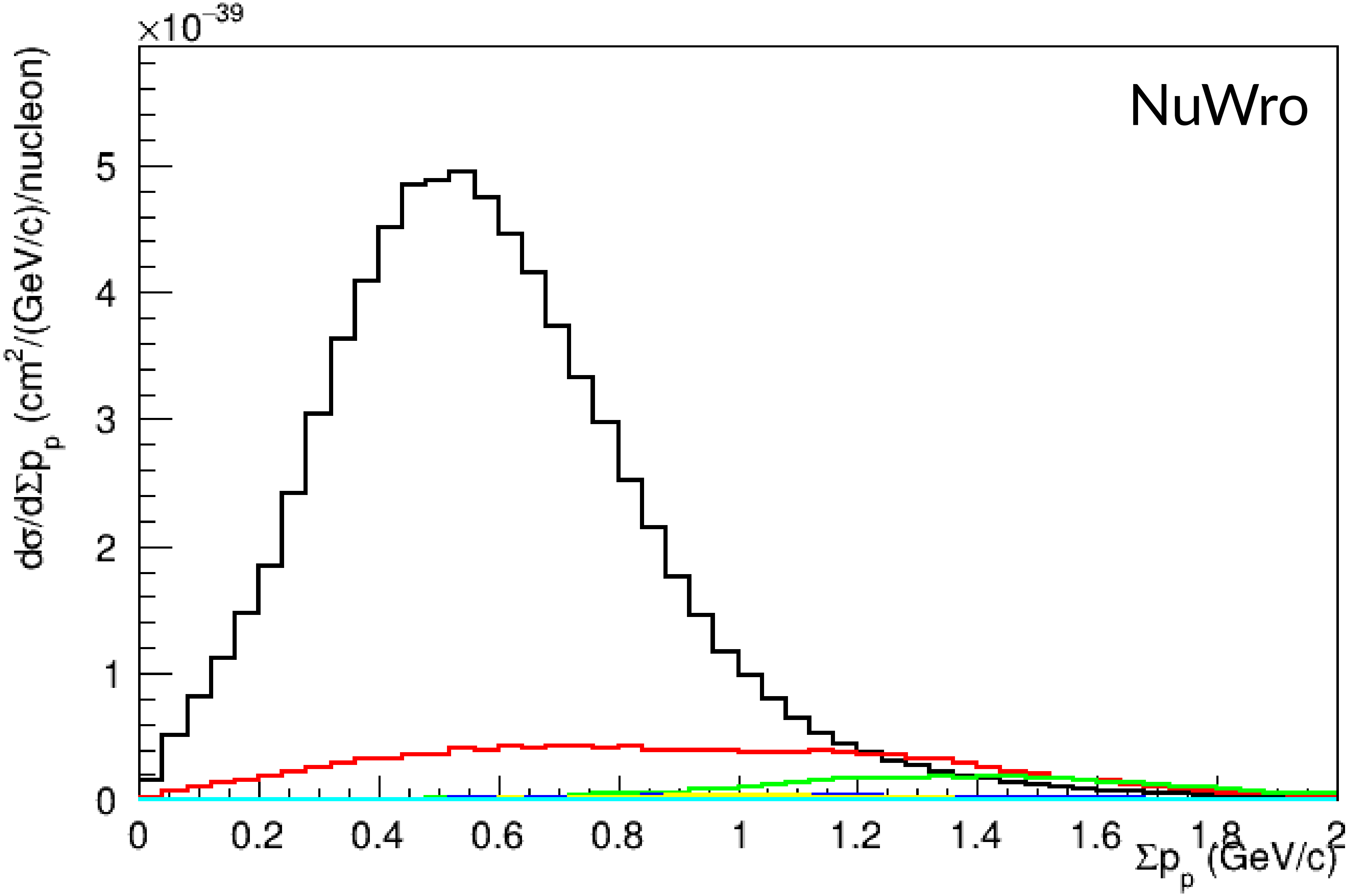
Sum of proton momentum

- Sum of proton momentum in a CC0π selection



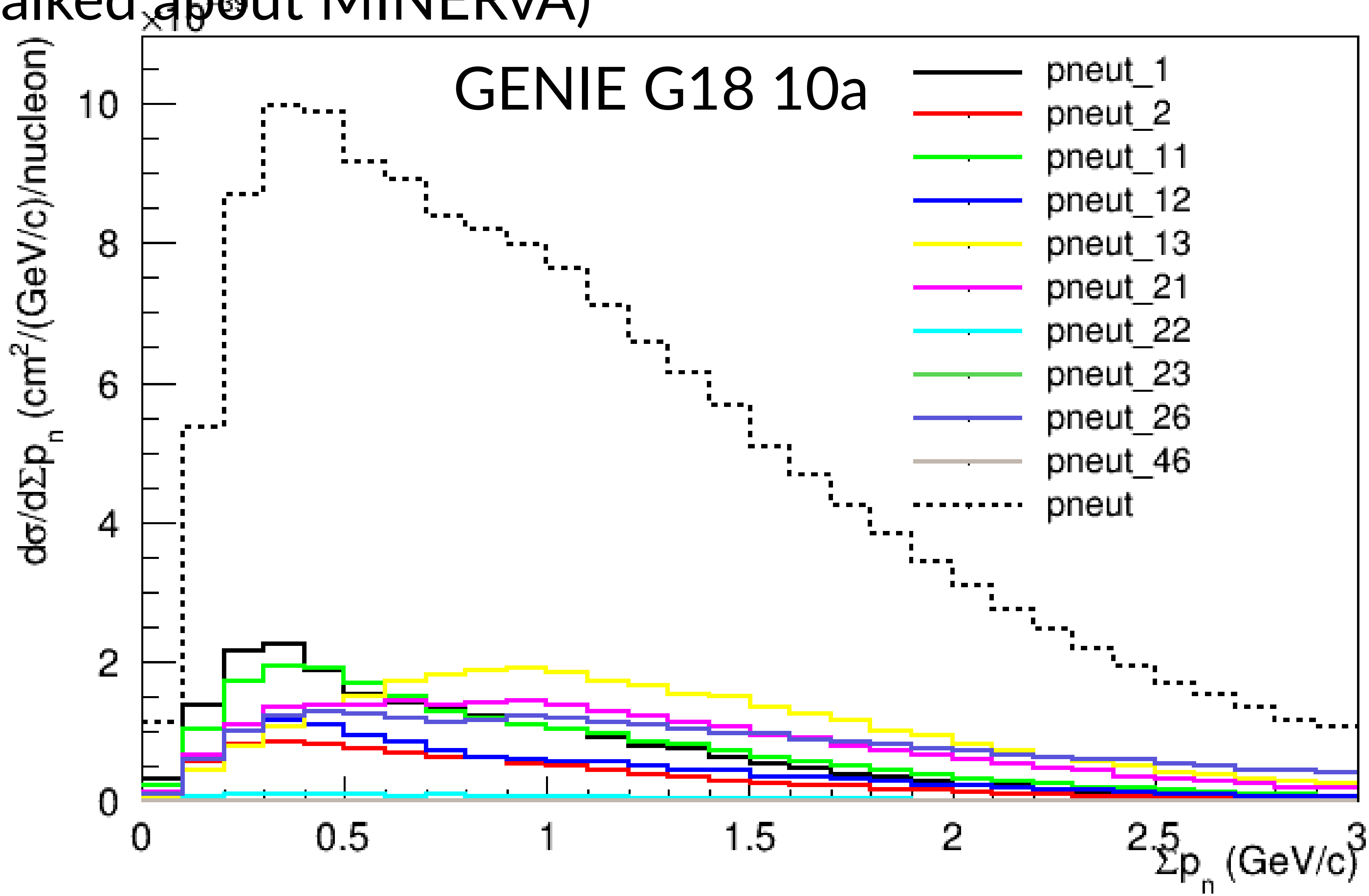
Sum of proton momentum

- Sum of proton momentum in a CC0π selection



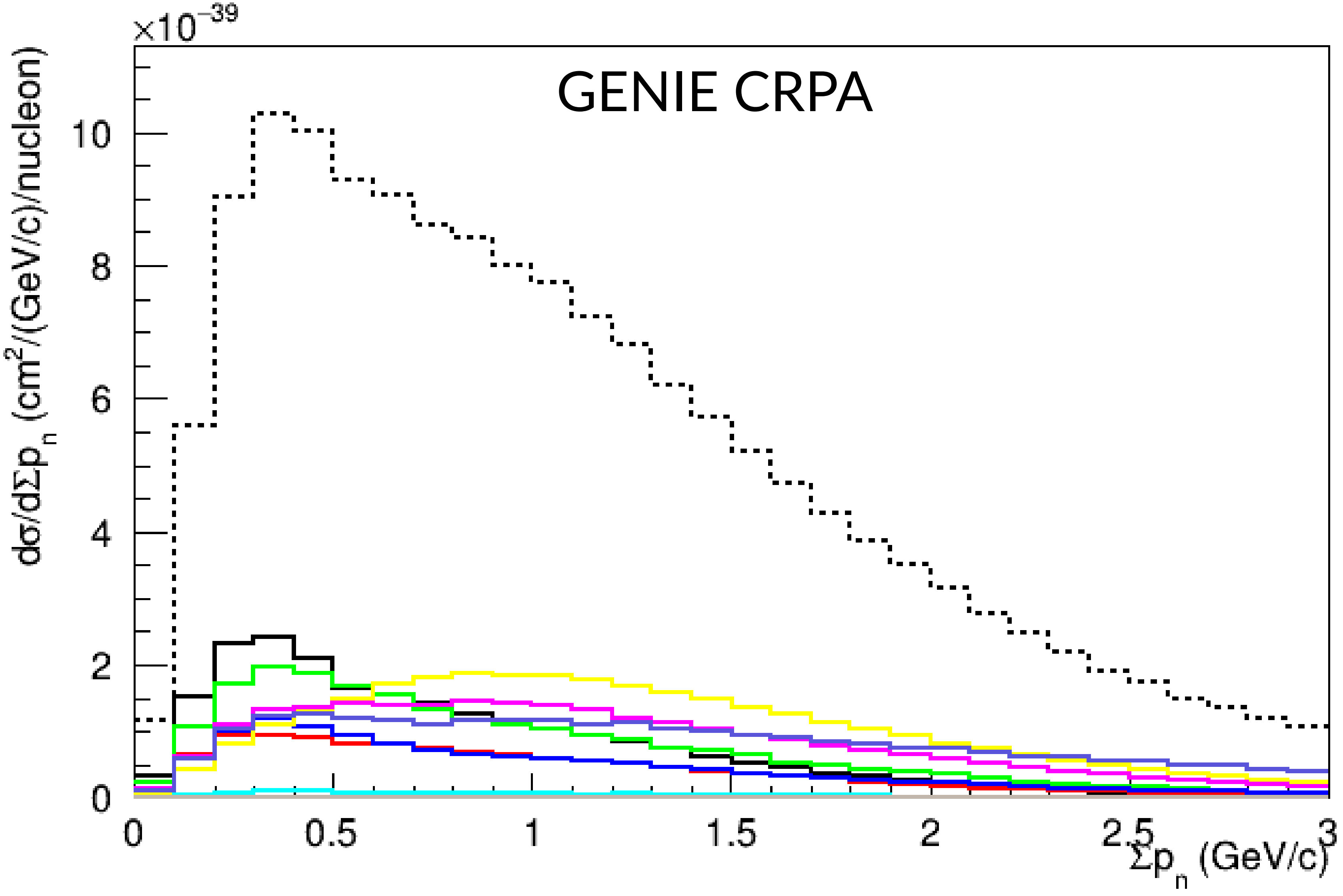
Sum of neutron momentum

- Deborah mentioned this yesterday
  - Energy carried away by neutrons in DUNE (Deborah talked about MINERvA)



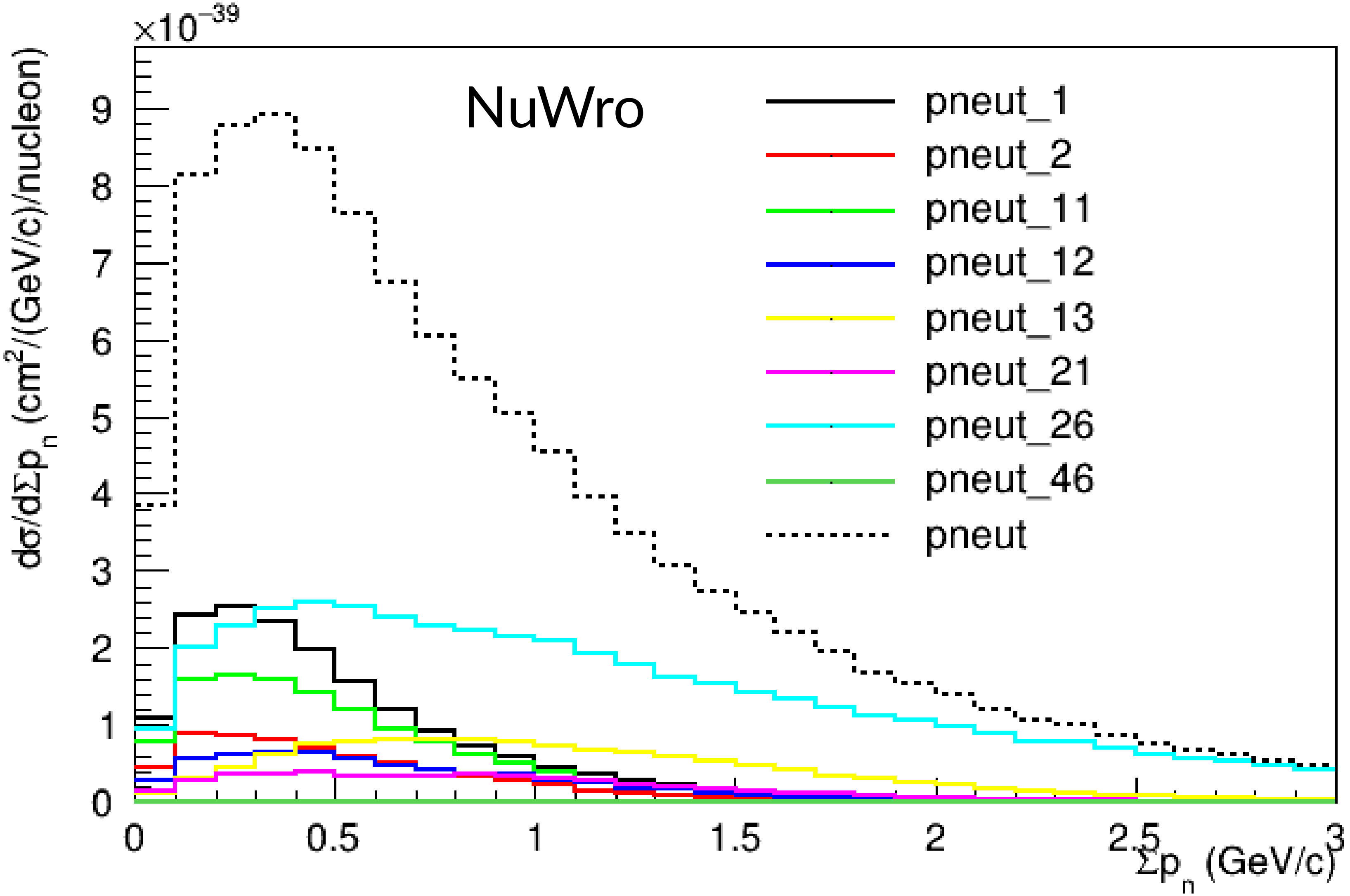
Sum of neutron momentum

- Energy carried away by neutrons in DUNE



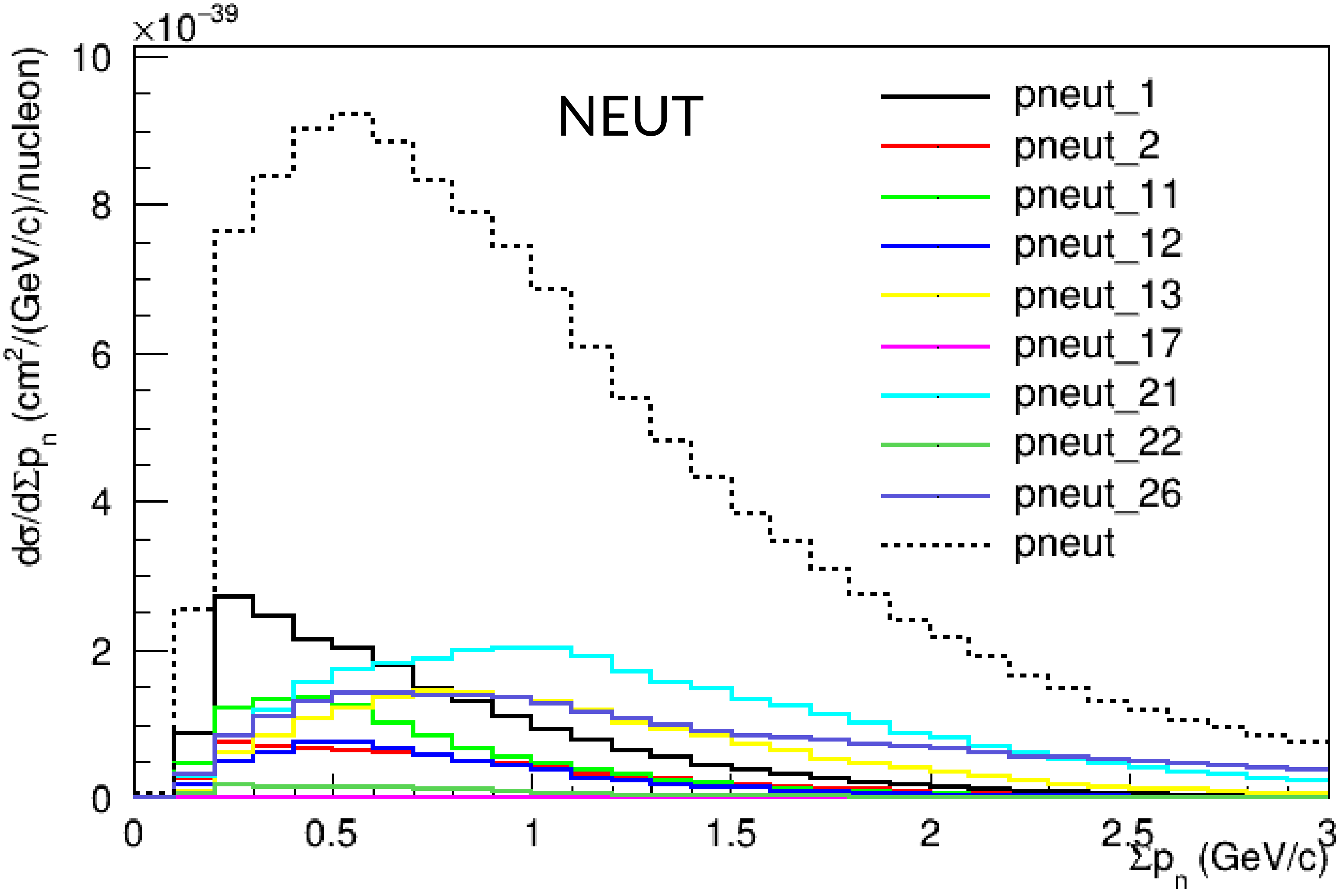
Sum of neutron momentum

- Energy carried away by neutrons in DUNE



Sum of neutron momentum

- Energy carried away by neutrons in DUNE



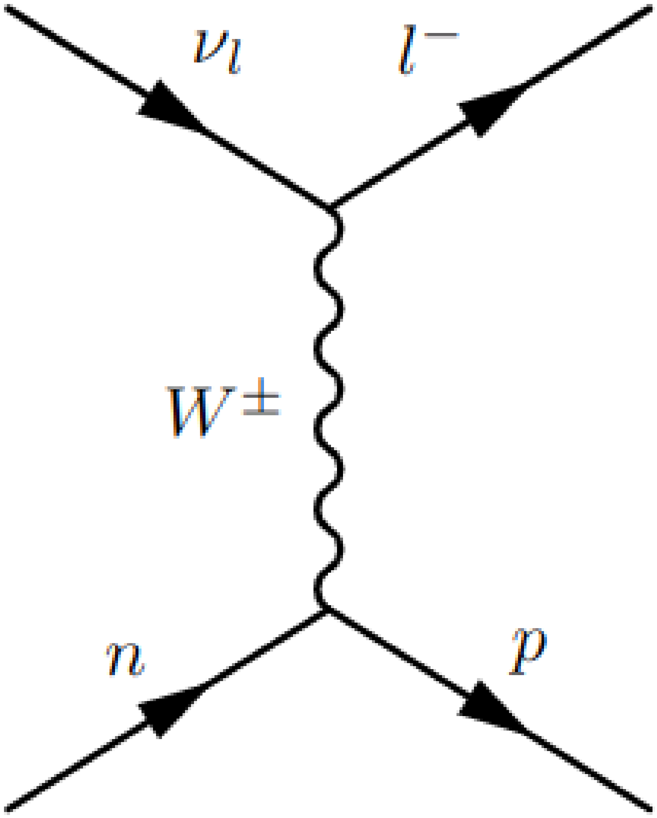
Energy estimation using CCQE

- Yesterday, Deborah (and others!) mentioned*:
  - How can we measure the neutrino energy using only one single outgoing particle?
  - Can show on whiteboard for completeness
  - Make a CC0π selection, assume 4 limb CCQE interaction
    - Incoming neutrino and neutron
    - Outgoing lepton and proton
  - Assume initial state is at rest
- Each approximation leads to individual biases
  - 4 legged diagram: what about FSI or missed particles?
  - Initial state at rest; on average this is true, but smears out
- Let's do that exercise here!

https://indico.fnal.gov/event/59963/contributions/288516/attachments/176666/240192/NuSTEC_School_SP_v2.pdf#page=23

Energy estimation using CCQE

- Can do the derivation if there is time



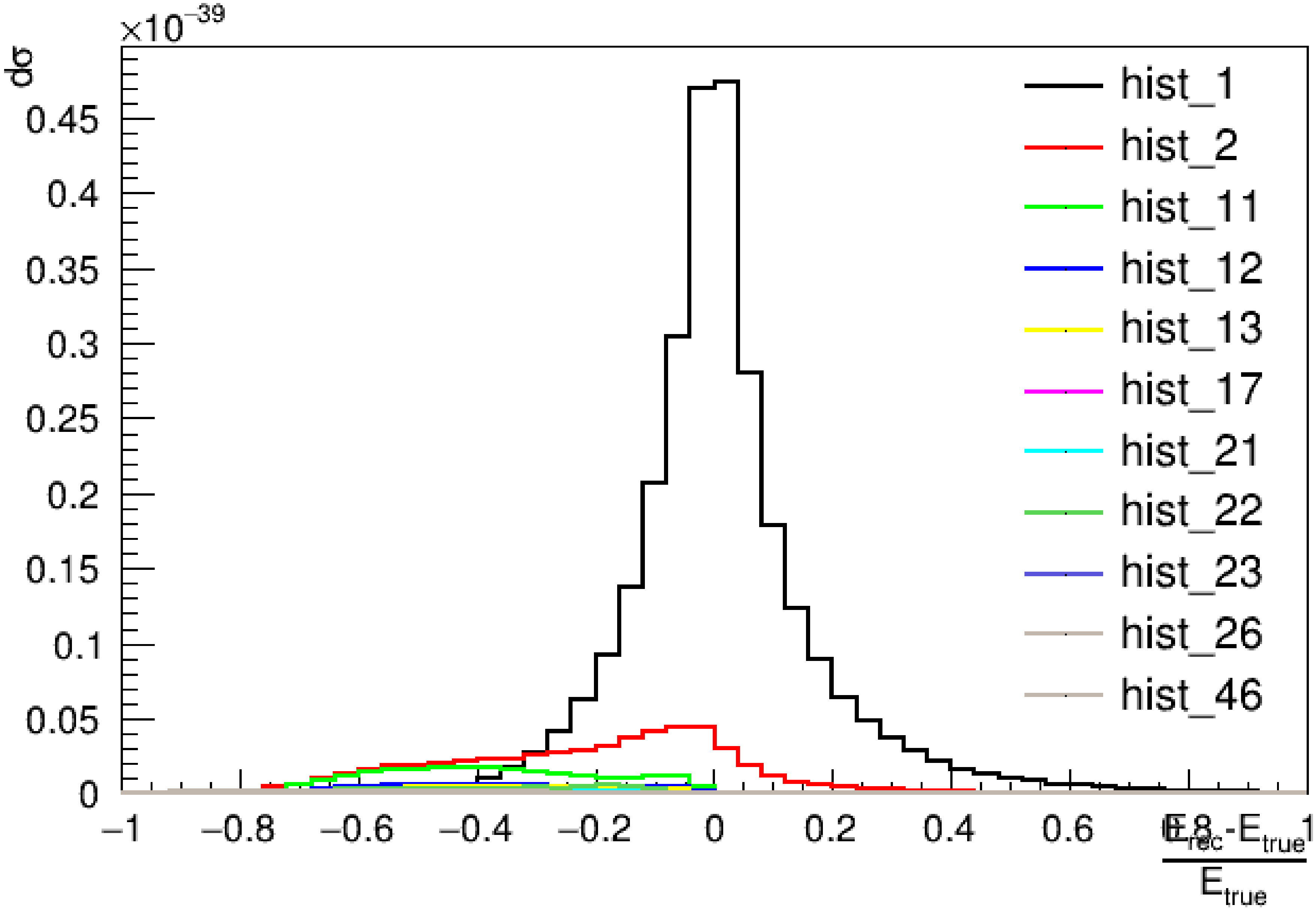
$$E_\nu^{\mathrm{CCQE}} = \frac{2m_N E_l - m_l^2 + m_{N'}^2 - m_N^2}{2\left(m_N - E_l + p_l \cos\theta_{\nu,l}\right)}$$

Energy estimation using CCQE

- Now let’s use NUISANCE to see how it works
- Let’s use (Enu_QE-Enu_true)/Enu_QE as the estimator of the bias
  - Expect 0 for no bias
- Enu_QE and Enu_true are pre-calculated in the NUISANCE trees for you
  - You can calculate it yourself too if you want
    - use the outgoing lepton four-vector
    - The neutrino was shot along z (have this four-vector too if you don’t trust me)

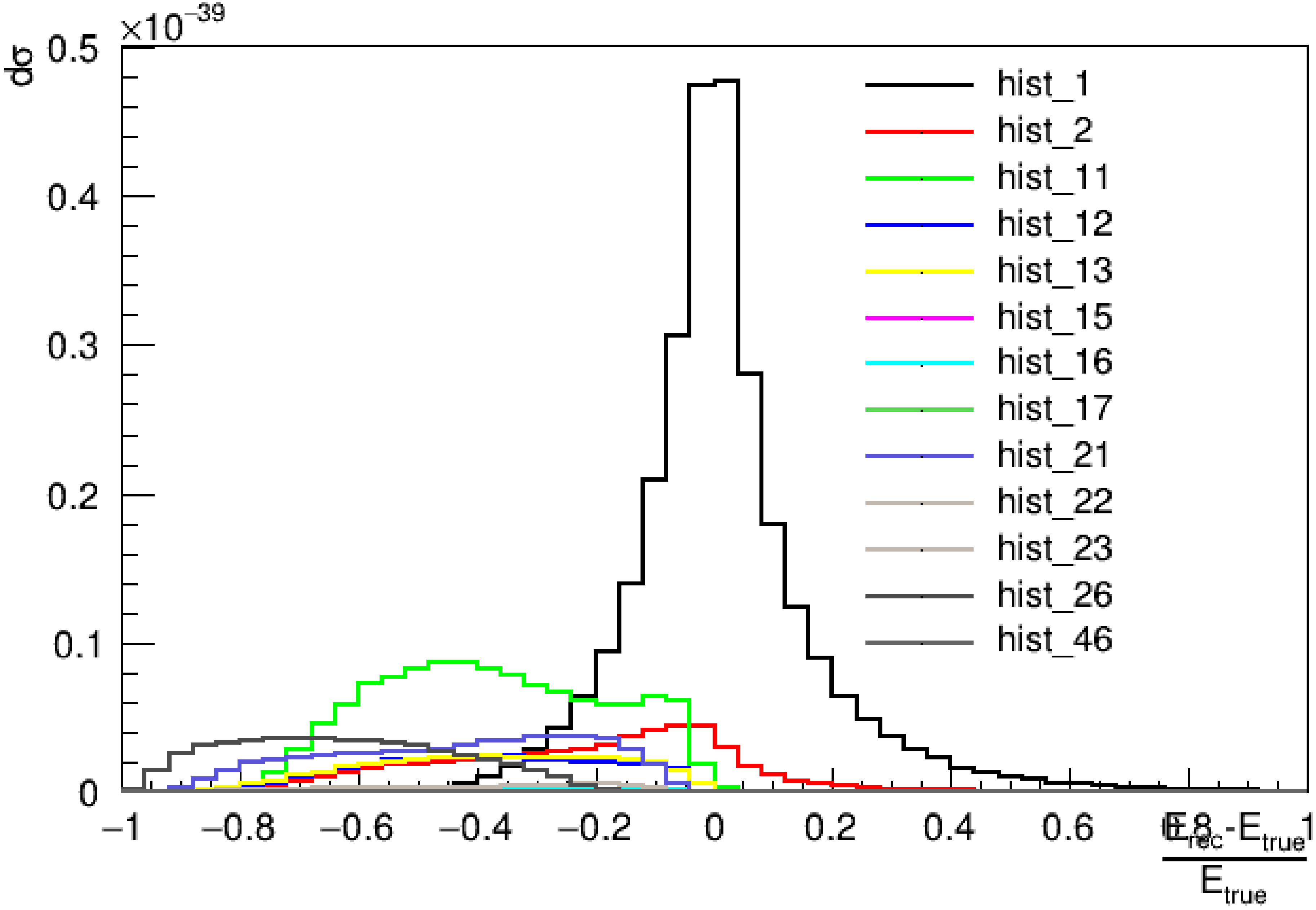
Energy estimation using CCQE

- With CC0π restriction



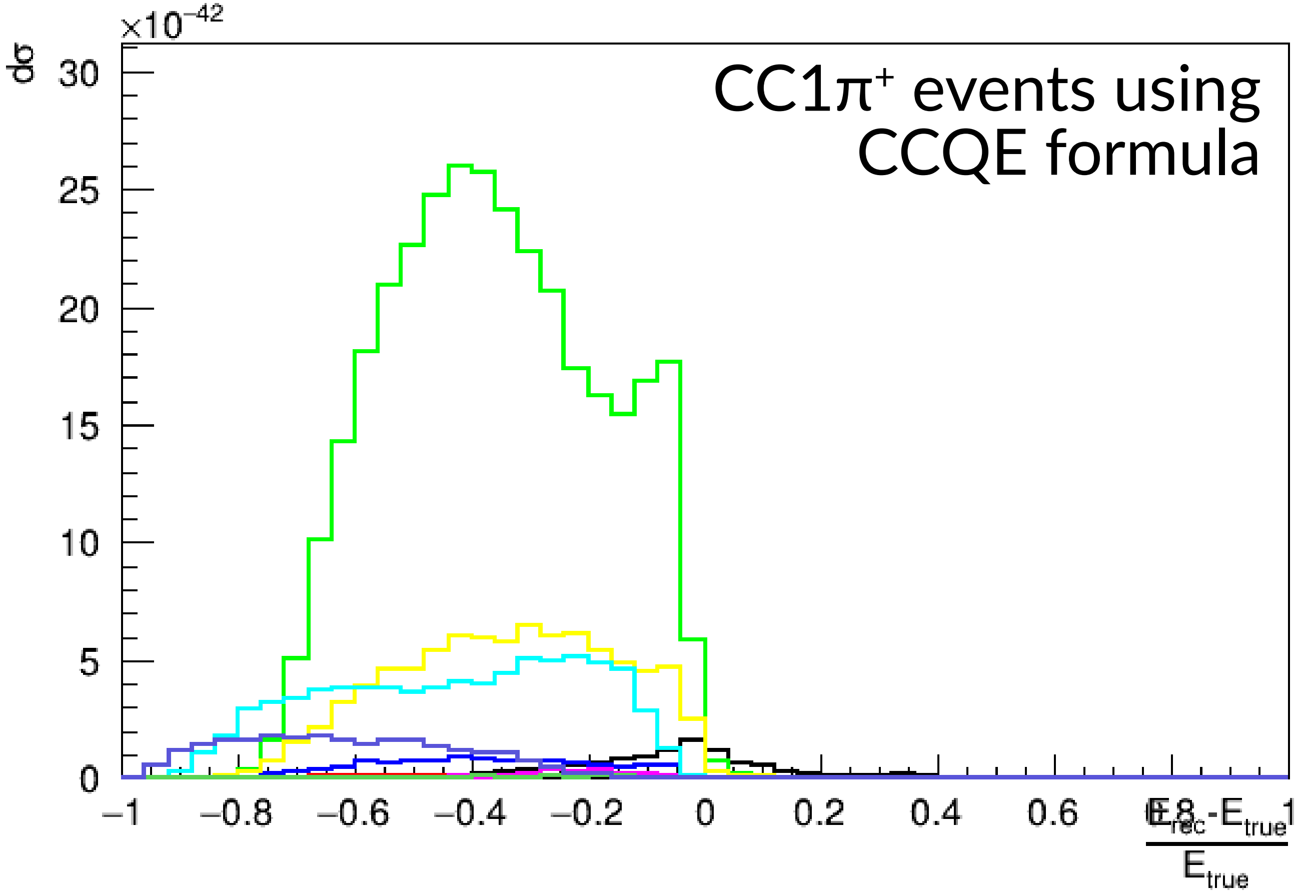
Energy estimation using CCQE

- Without CC0π restriction



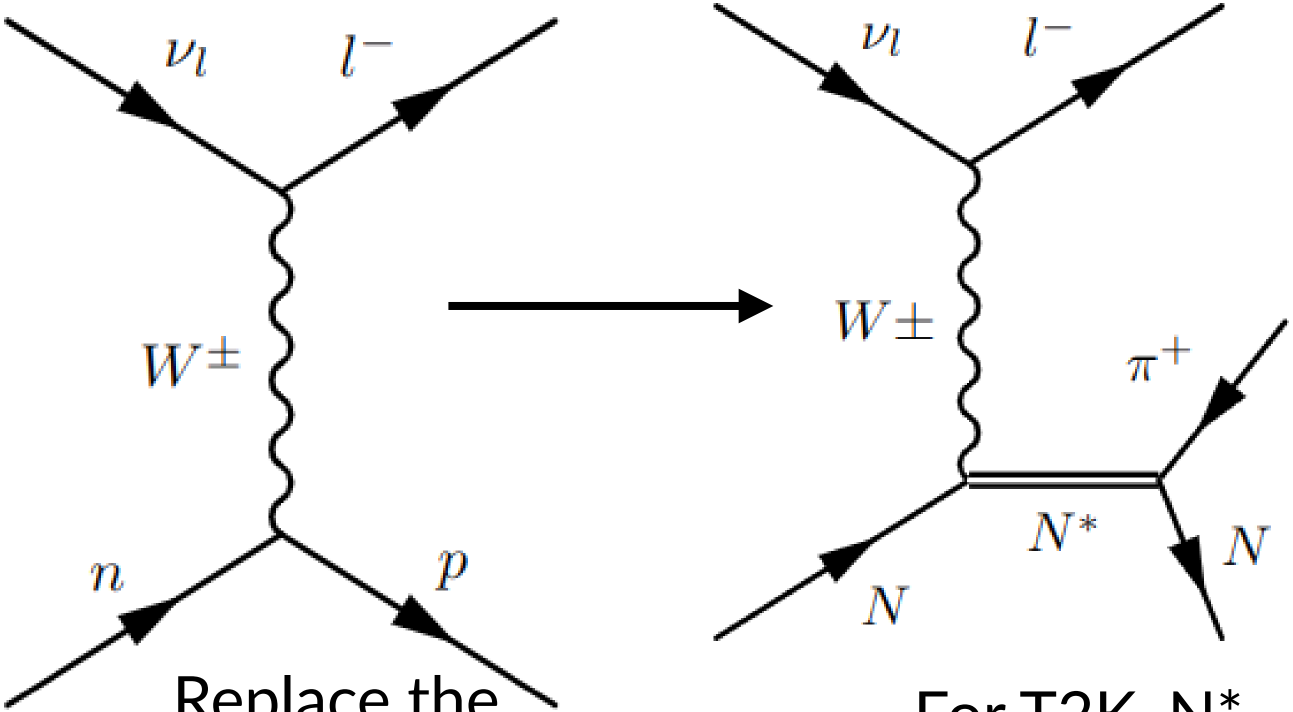
Energy estimation for 1π

- Added bonus, what if we have a CC1π$^+$ selection instead of CC0π?
- And we have little information on the pion kinematics: only lepton information
  - Low momentum → let's place a cut of 200 MeV/c on pion momentum
- CCQE formula is poor for these events
- Can we improve the estimator?



Energy estimation for 1π

- This scenario is very similar to T2K's "below Cherenkov ring" pion selection
  - Selects a muon or electron in final state
  - Delayed Michel electron without primary ring → low momentum pion

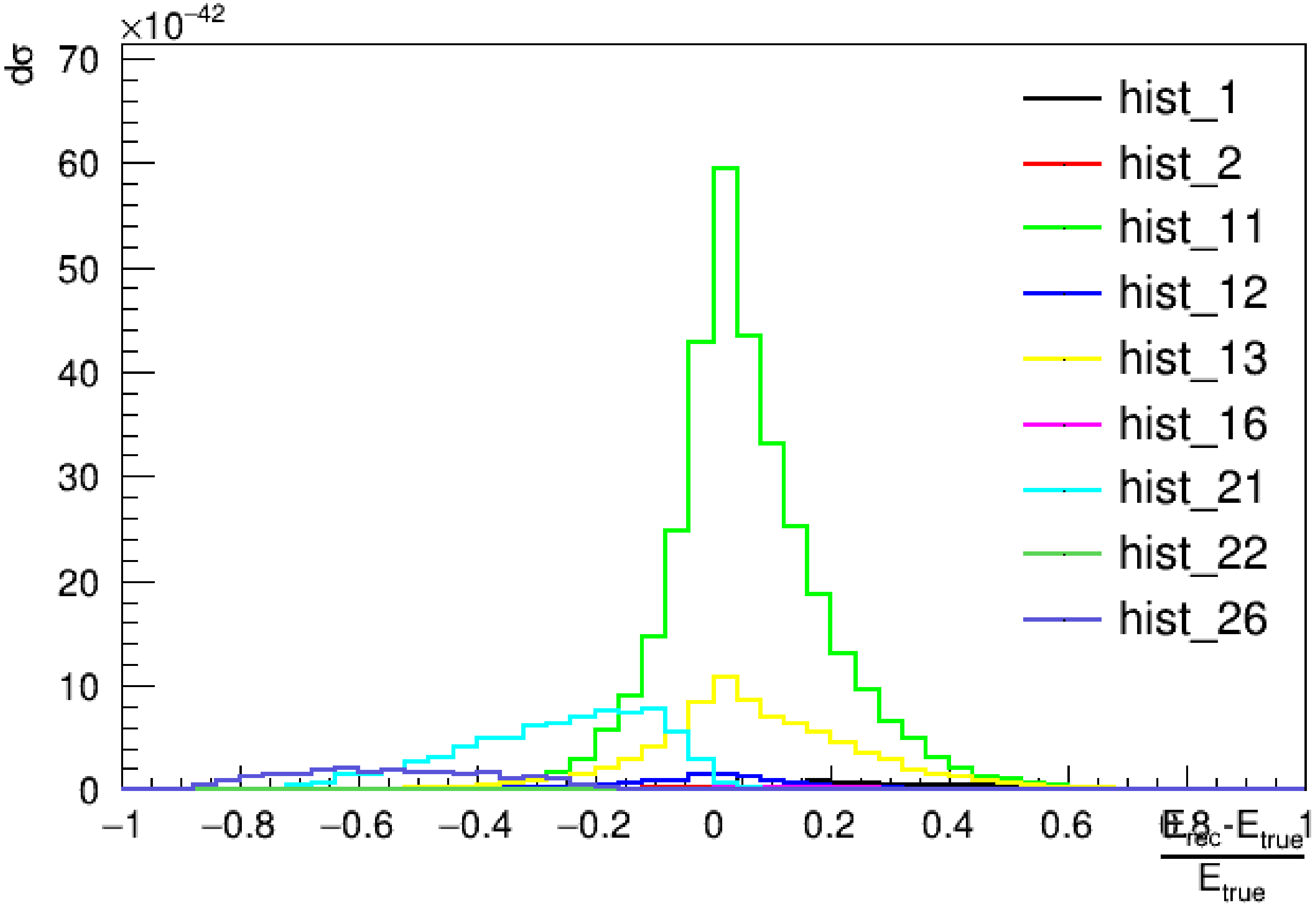


Replace the outgoing proton with the N* mass

For T2K, N* mostly Δ(1232)

For wider band beams, and higher energy, approximation is more cumbersome
→ May need to reconstruct W

Energy estimation for 1π

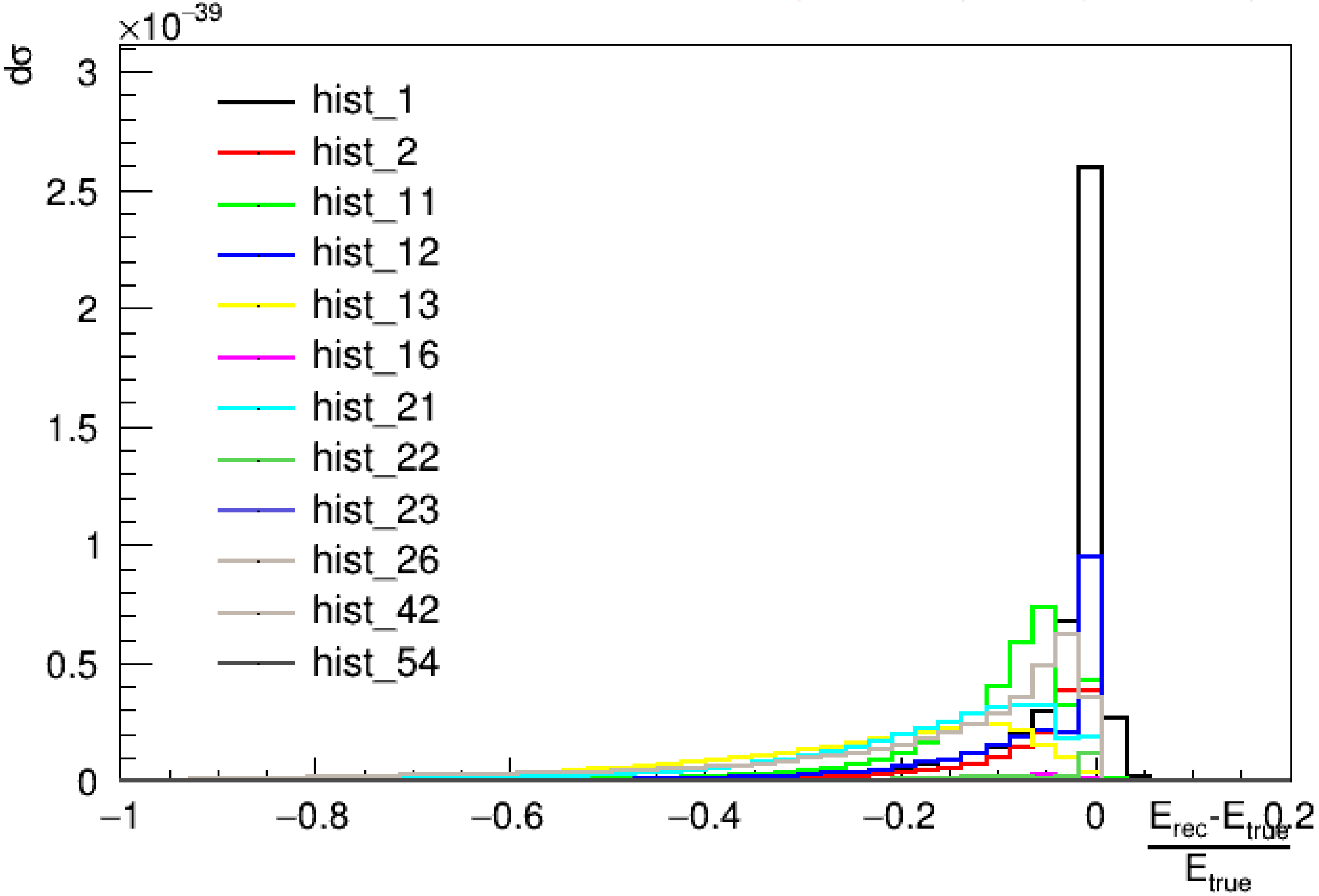


- Much better!
- Now we see an offset in SIS and DIS, which makes a lot of sense
  - These processes do not proceed via a Delta resonance
  - As Alexis said yesterday, SIS/DIS is not resonant process

Energy estimation for CC-inc

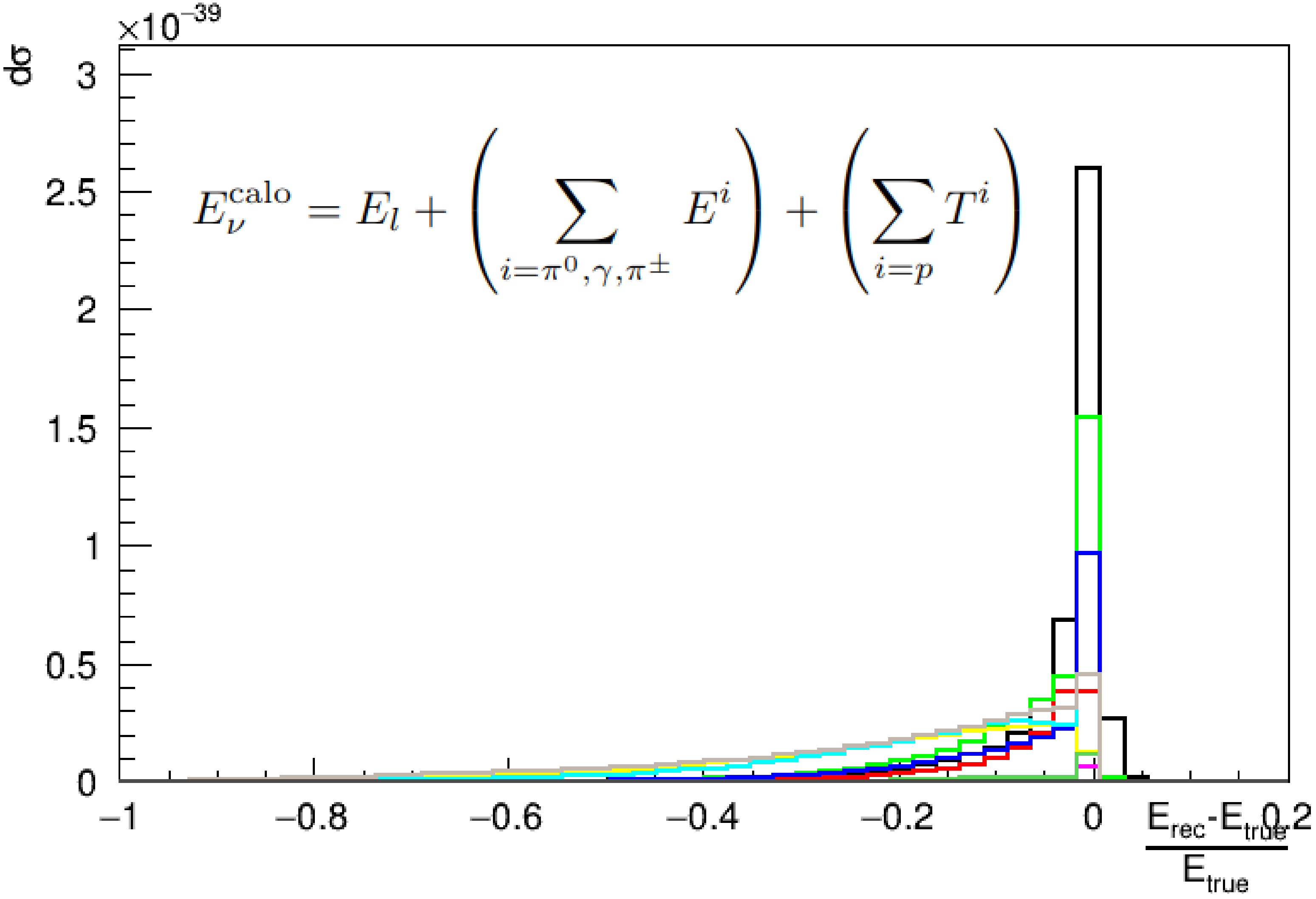
- Calorimetric reconstruction in DUNE
- Add up proton and charged pion KE, and total E of π^0 and photon

$$E_\nu^{\mathrm{calo}} = E_l + \left(\sum_{i=\pi^0,\gamma} E^i \right) + \left(\sum_{i=p,\pi^\pm} T^i \right)$$



Energy estimation for CC-inc

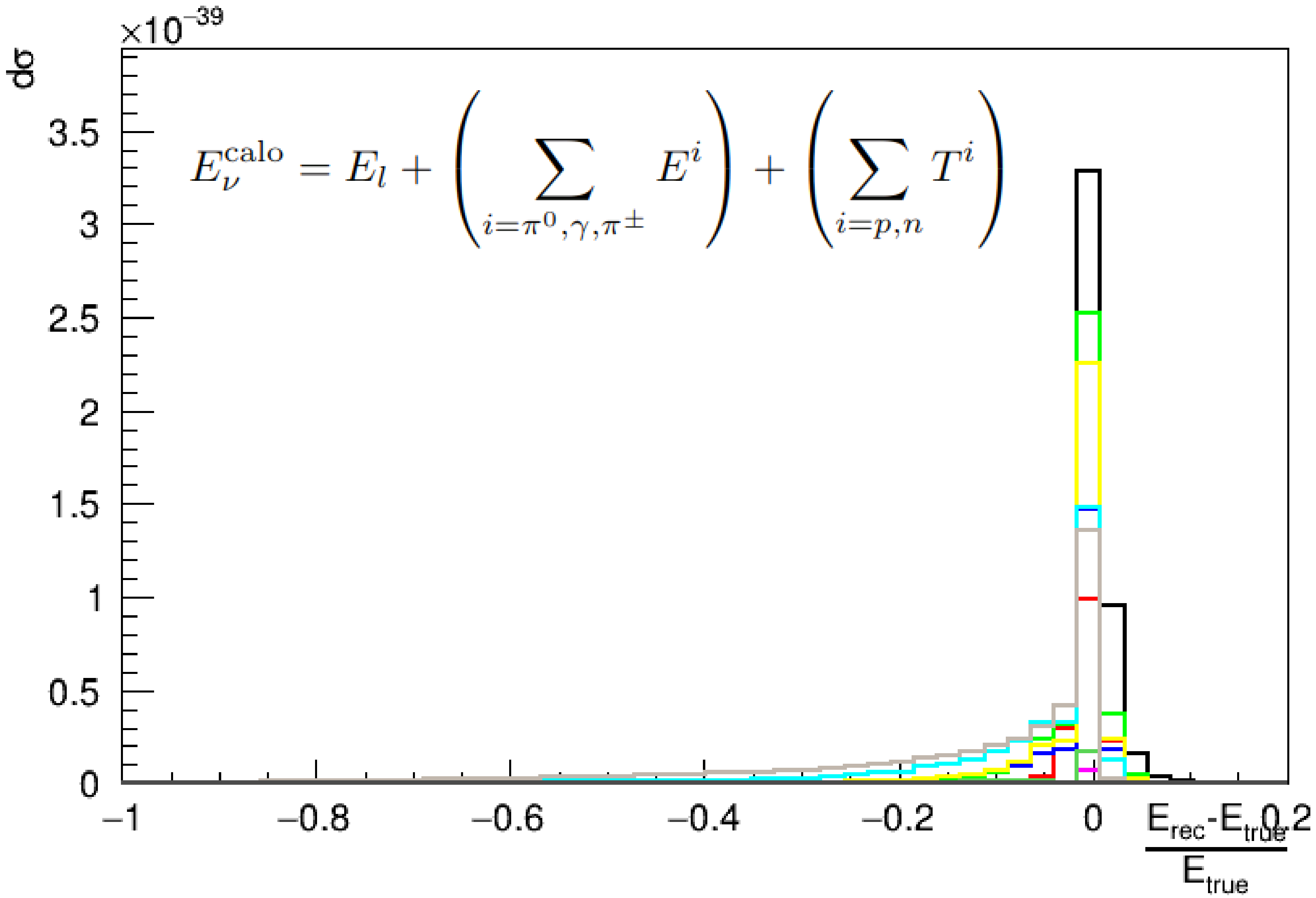
- Calorimetric reconstruction in DUNE
- Improve by correctly identifying proton/pion → add charged pion total energy



$$E_\nu^{\mathrm{calo}} = E_l + \left(\sum_{i=\pi^0,\gamma,\pi^\pm} E^i \right) + \left(\sum_{i=p} T^i \right)$$

Energy estimation for CC-inc

- Calorimetric reconstruction in DUNE
- Add in ability to measure neutrons

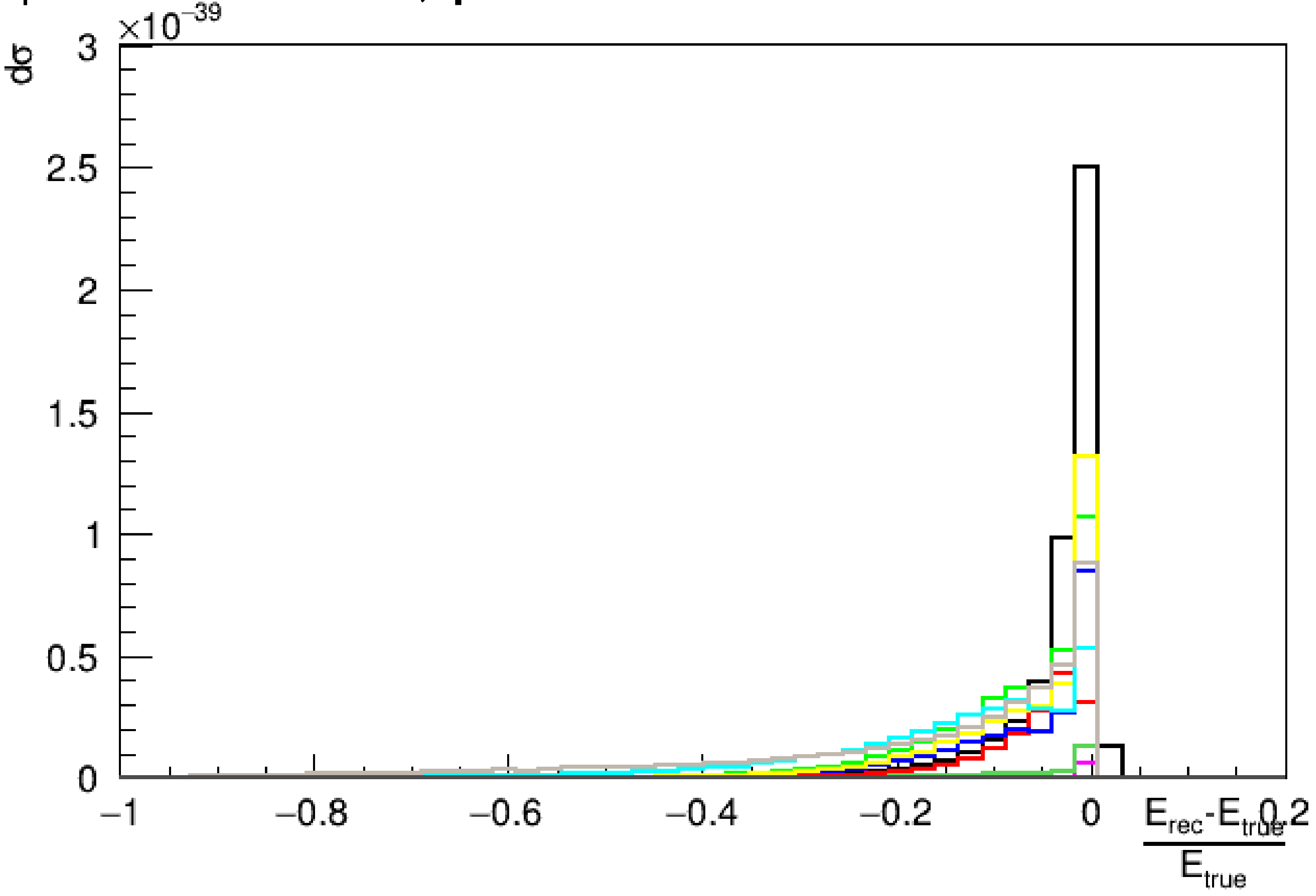


$$E_\nu^{\text{calo}} = E_l + \left(\sum_{i=\pi^0,\gamma,\pi^\pm} E^i \right) + \left(\sum_{i=p,n} T^i \right)$$

Clarence Wret

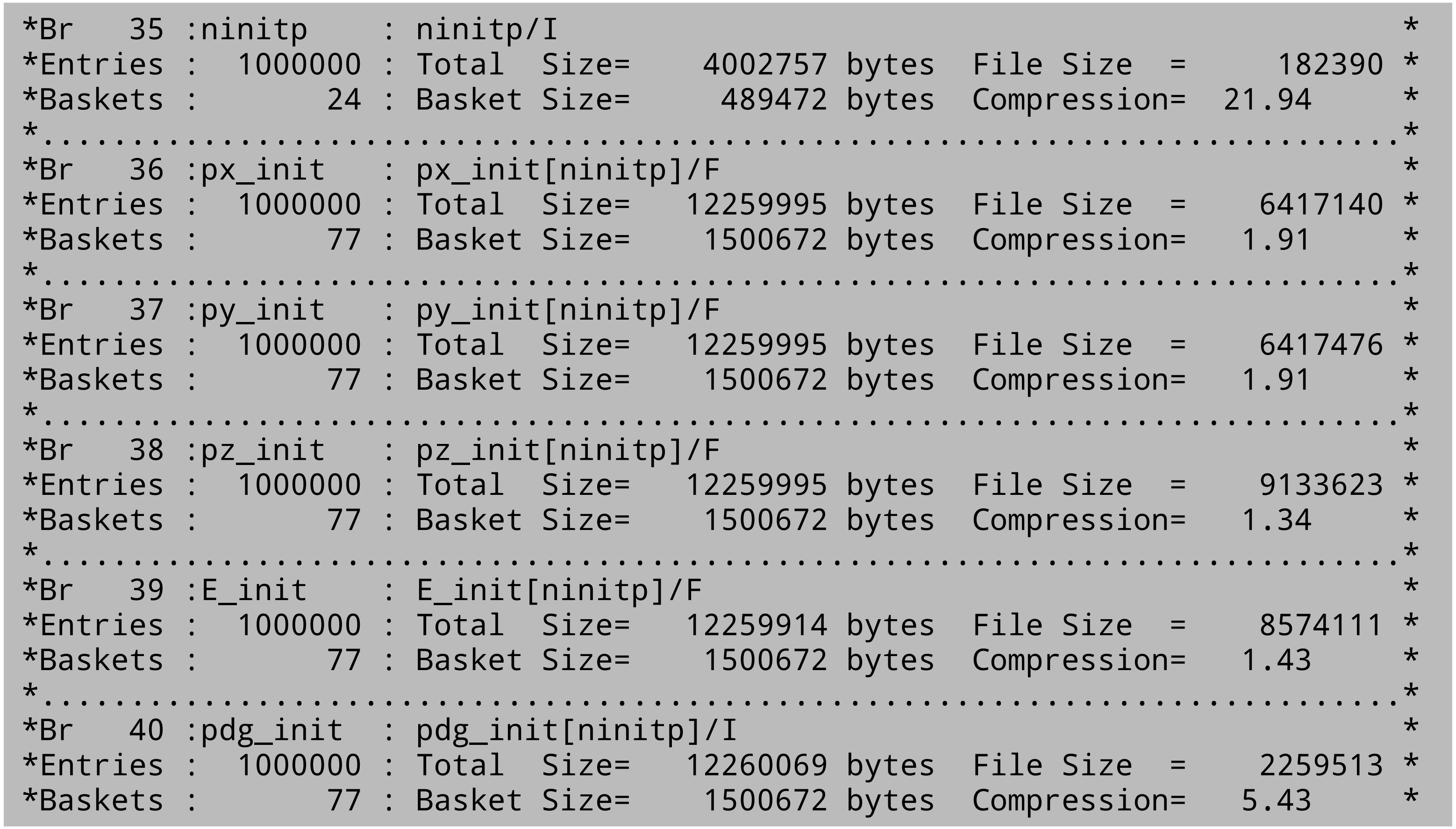
Energy estimation for CC-inc

- Calorimetric reconstruction in DUNE
- Put in some reasonable thresholds, e.g. $p_\pi > 0.2$ GeV/c, $p_p > 0.4$ GeV/c, $p_n > 0.5$ GeV/c



Initial state information

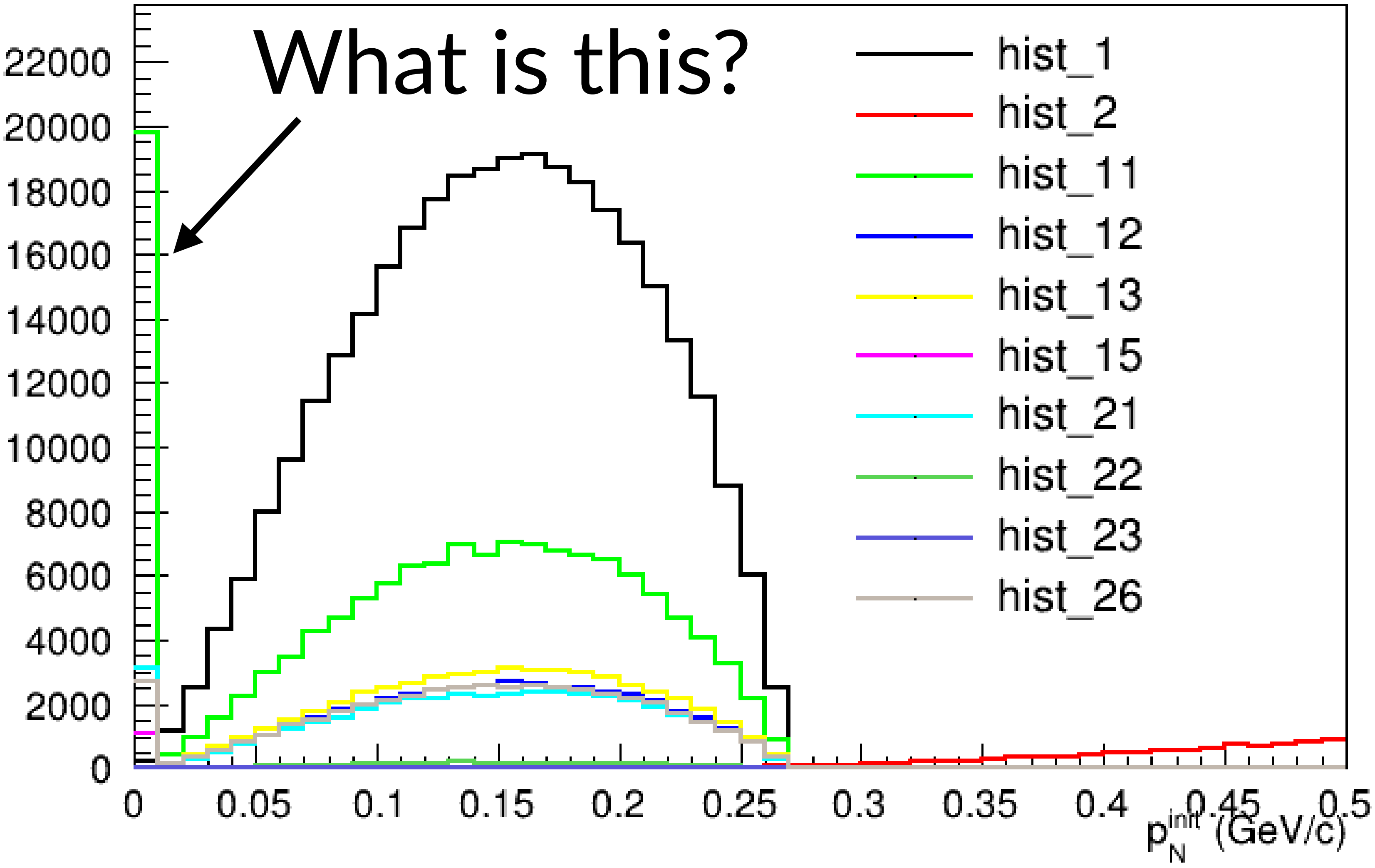
- What about the initial state?
- Not **directly** measurable by experiment, but can still be interesting to better understand the model
- NUISANCE also saves the initial state particle stack



```
*Br   35 :ninitp    : ninitp/I                                               *
*Entries :  1000000 : Total  Size=    4002757 bytes  File Size  =     182390 *
*Baskets :       24 : Basket Size=     489472 bytes  Compression=  21.94     *
*............................................................................*
*Br   36 :px_init   : px_init[ninitp]/F                                      *
*Entries :  1000000 : Total  Size=   12259995 bytes  File Size  =    6417140 *
*Baskets :       77 : Basket Size=    1500672 bytes  Compression=   1.91     *
*............................................................................*
*Br   37 :py_init   : py_init[ninitp]/F                                      *
*Entries :  1000000 : Total  Size=   12259995 bytes  File Size  =    6417476 *
*Baskets :       77 : Basket Size=    1500672 bytes  Compression=   1.91     *
*............................................................................*
*Br   38 :pz_init   : pz_init[ninitp]/F                                      *
*Entries :  1000000 : Total  Size=   12259995 bytes  File Size  =    9133623 *
*Baskets :       77 : Basket Size=    1500672 bytes  Compression=   1.34     *
*............................................................................*
*Br   39 :E_init    : E_init[ninitp]/F                                       *
*Entries :  1000000 : Total  Size=   12259914 bytes  File Size  =    8574111 *
*Baskets :       77 : Basket Size=    1500672 bytes  Compression=   1.43     *
*............................................................................*
*Br   40 :pdg_init  : pdg_init[ninitp]/I                                     *
*Entries :  1000000 : Total  Size=   12260069 bytes  File Size  =    2259513 *
*Baskets :       77 : Basket Size=    1500672 bytes  Compression=   5.43     *
```

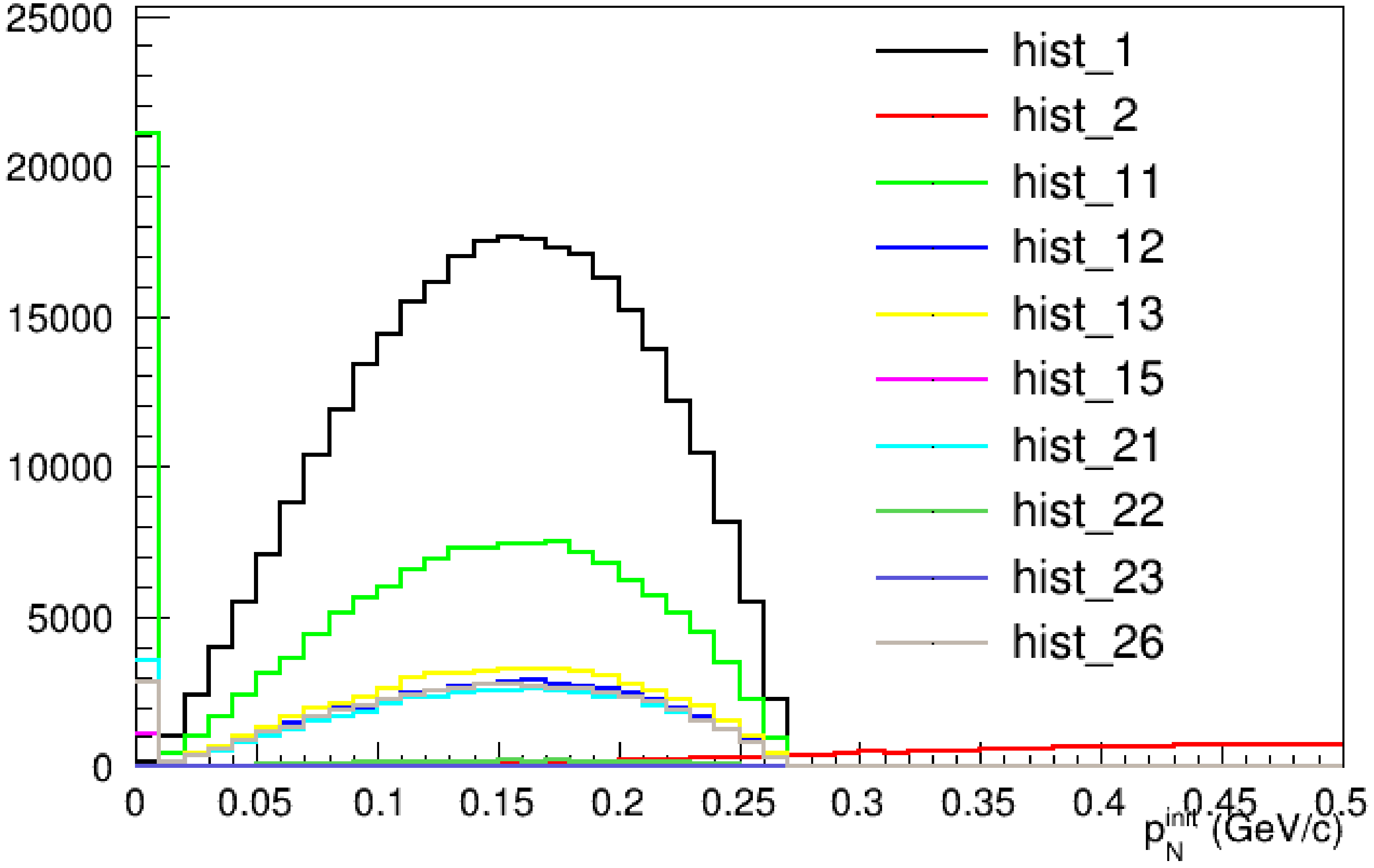
Initial state information

- GENIE CRPA



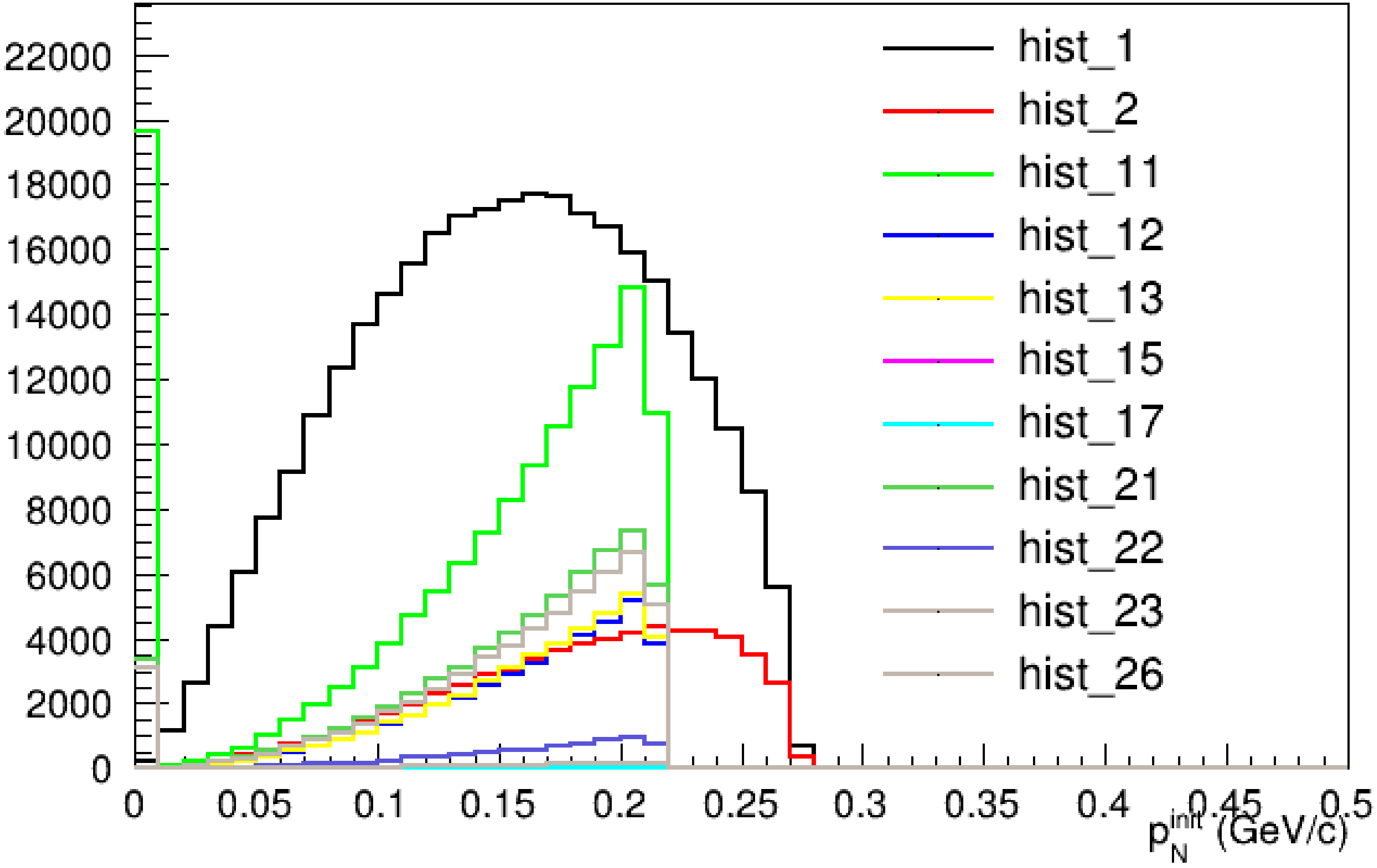
Initial state information

- GENIE G18 10a, very similar to CRPA
- Same initial state model



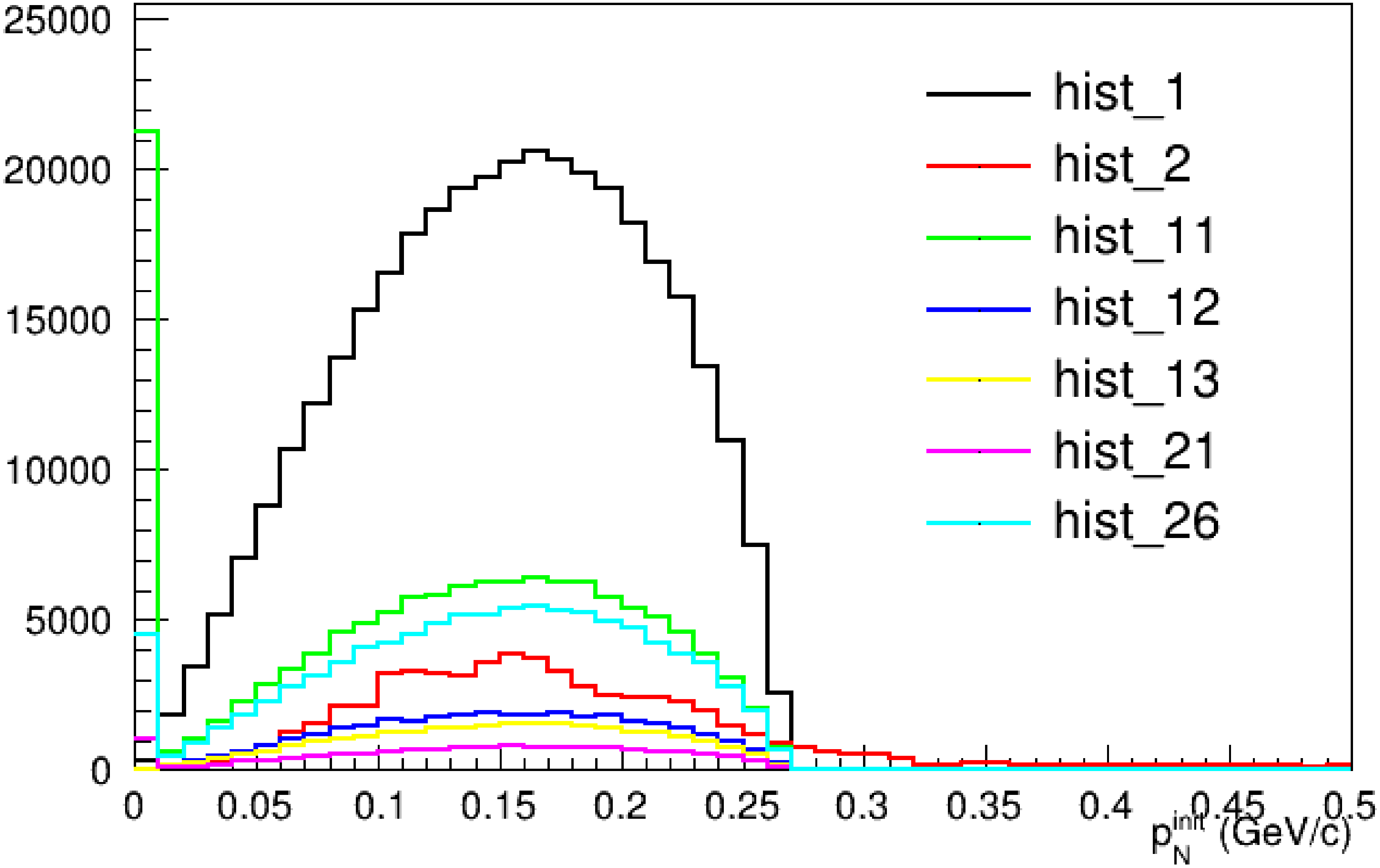
Initial state information

- NEUT clearly uses different initial state models for different models
- Which initial state is used for which model?



Initial state information

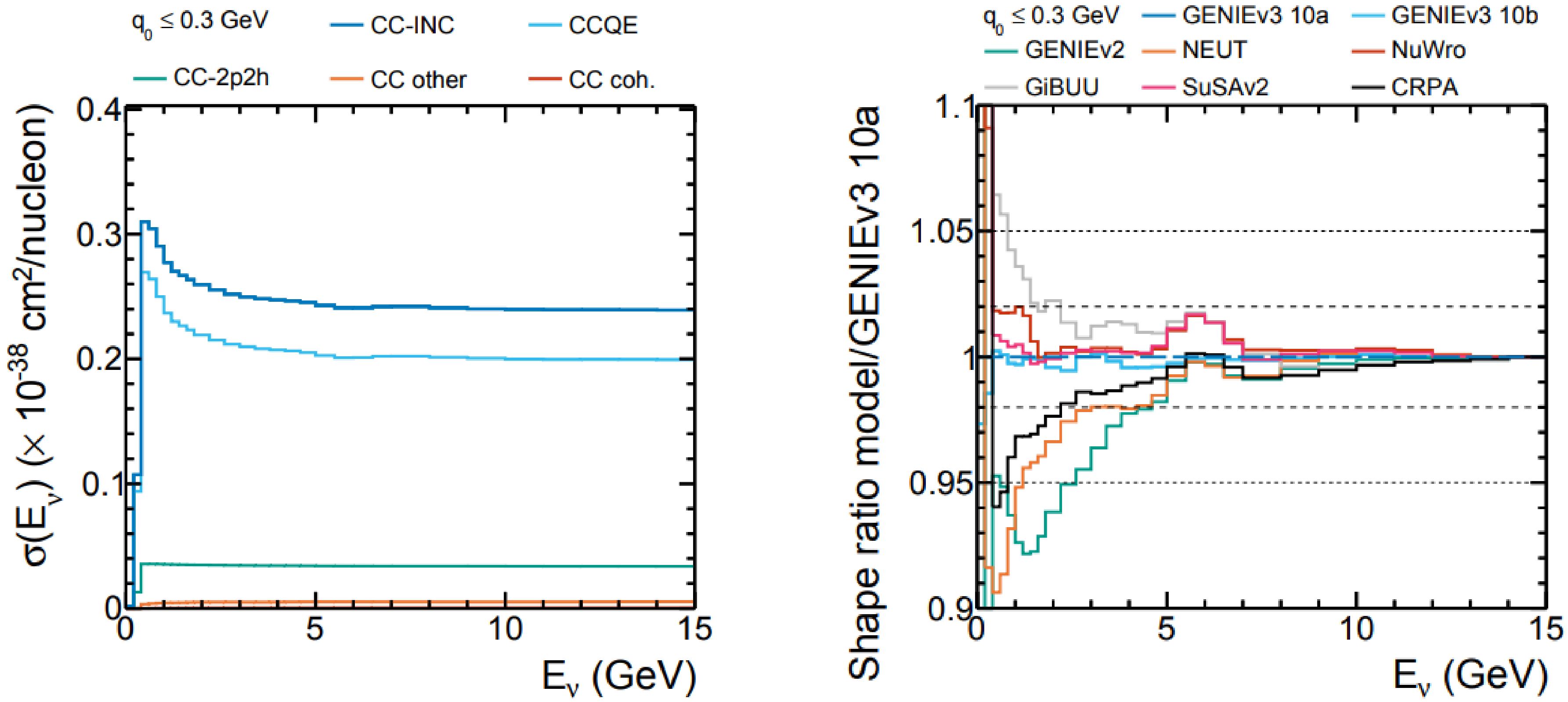
- NuWro interesting tail in 2p2h distribution



So what can you do with this new found skill?

NUISANCE examples

- Studied the model-dependence in using low-nu method to understand the neutrino flux



(a) GENIEv3 10a

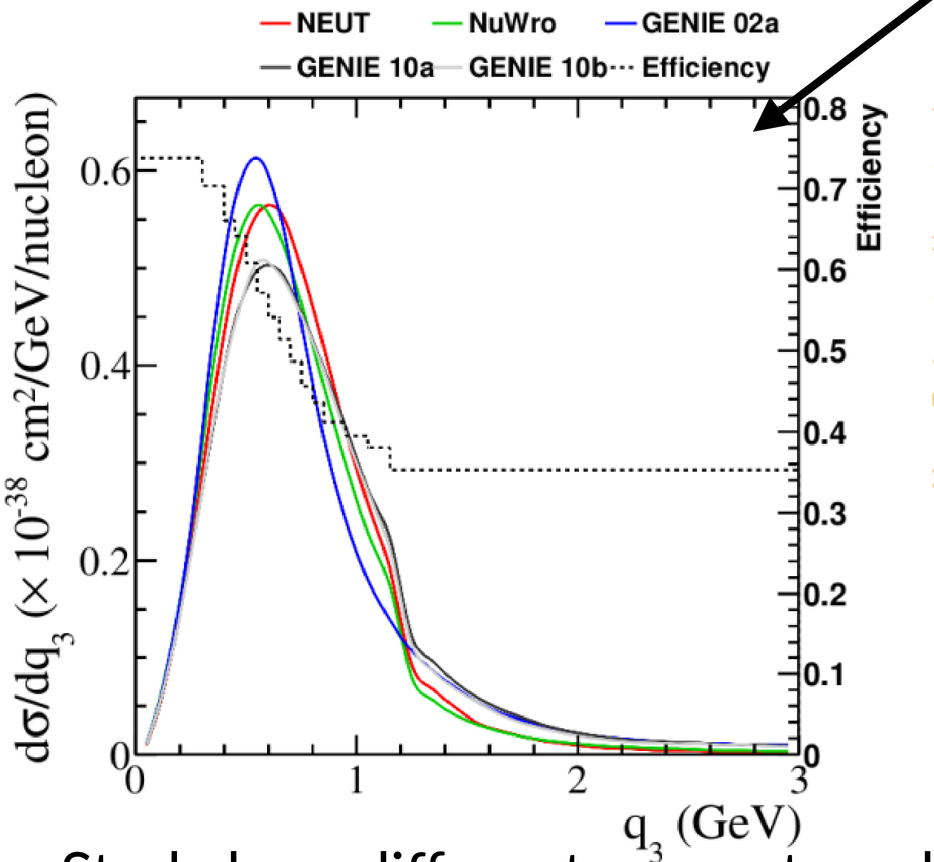
(b) ν_μ–^{40}Ar, $q_0 \leq 0.3$ GeV

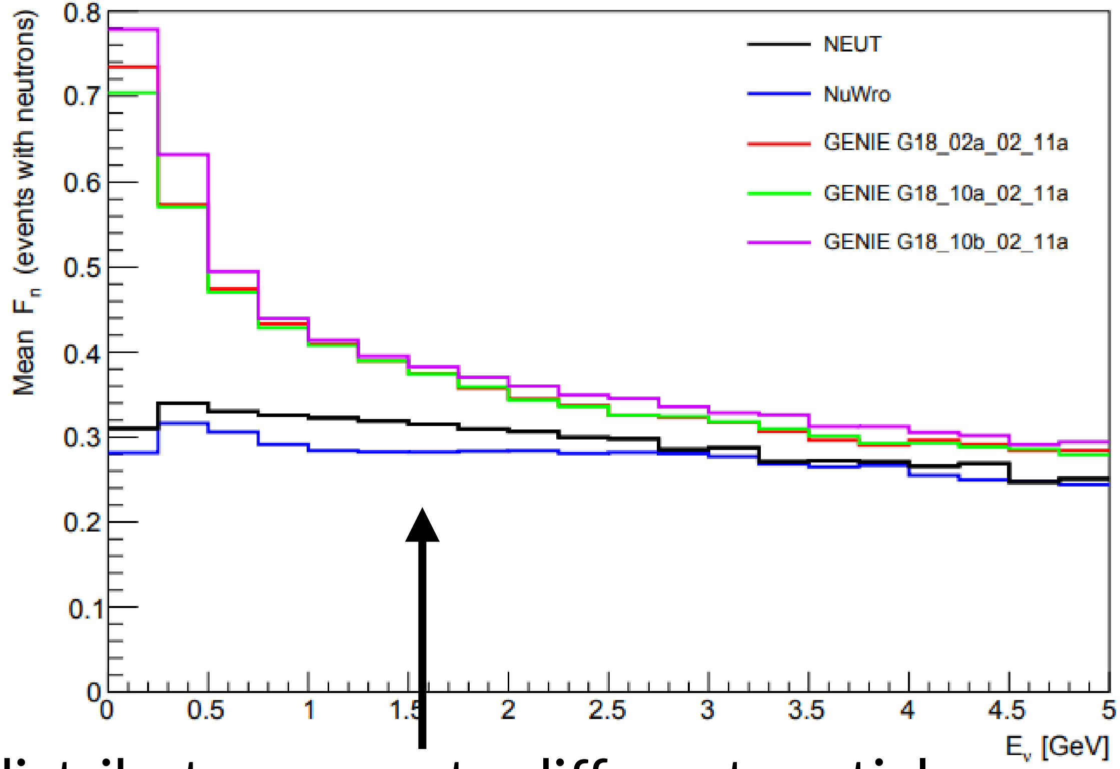
- Concluded that below 5 GeV, utility of low-nu is limited because we don't understand the physics well enough

Eur. Phys. J. C 82, 808 (2022).

NUISANCE examples

- Study model-dependence in regions of rapidly changing efficiencies for a wide range of experiments and generators
  - Dan mentioned this in his talk



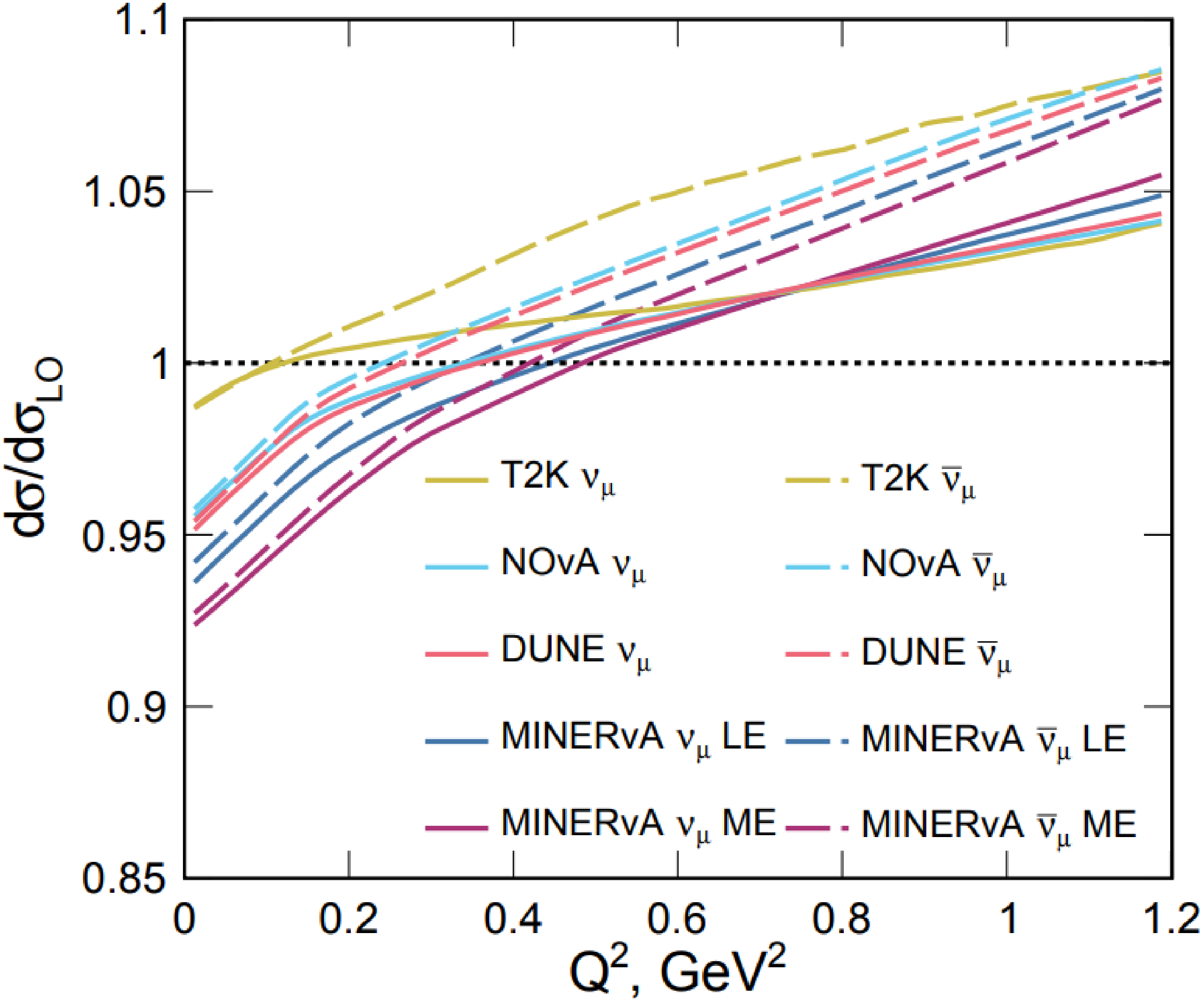


- Study how different generators distribute energy to different particles, here the mean fraction of energy given to neutrons
  - mentioned by Stephen yesterday

Phys. Rev. D 105, 092004

NUISANCE examples

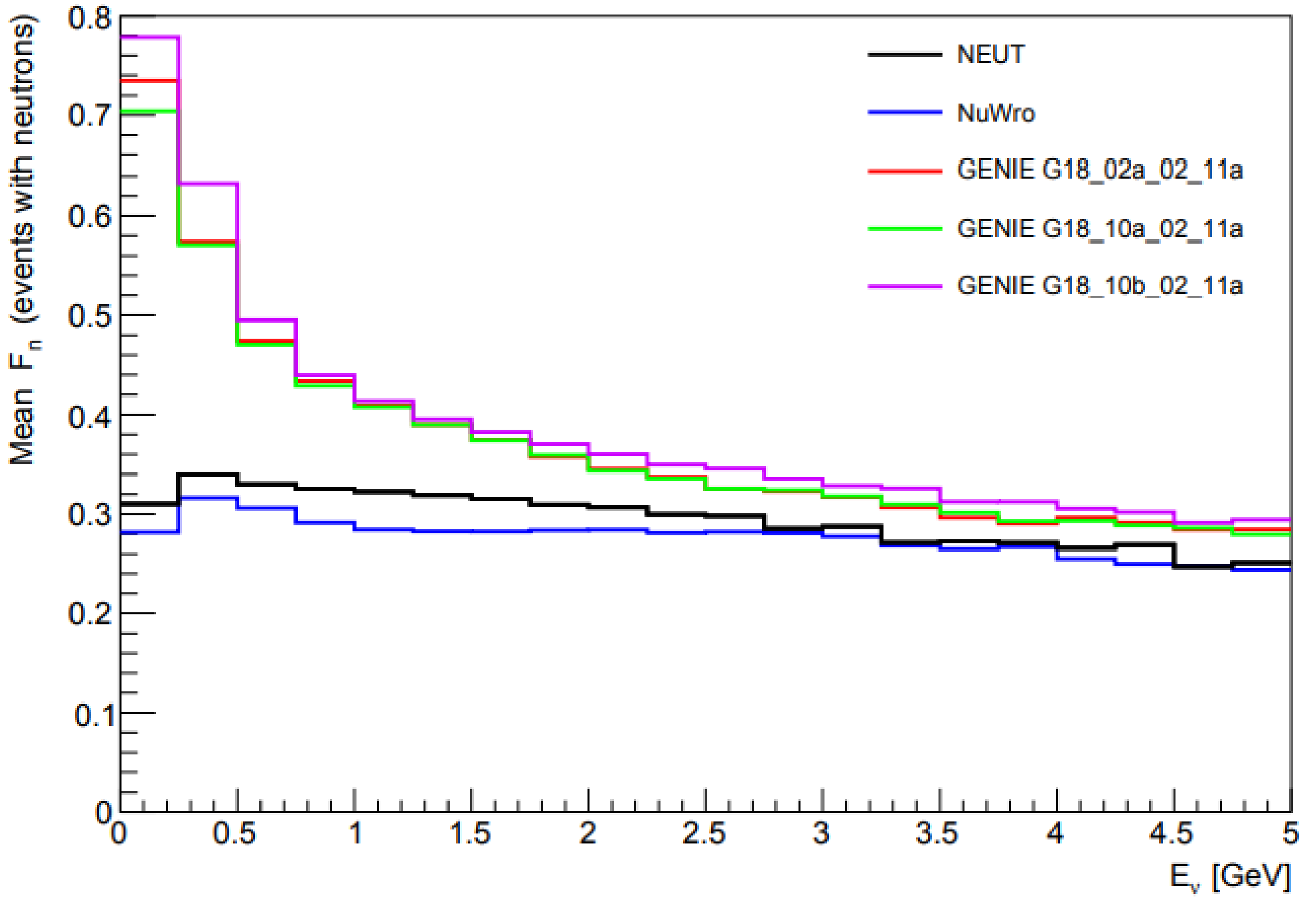
- Study impact of radiative corrections on different neutrino experiments



Phys.Rev.D 106 (2022) 9, 093006

NUISANCE examples

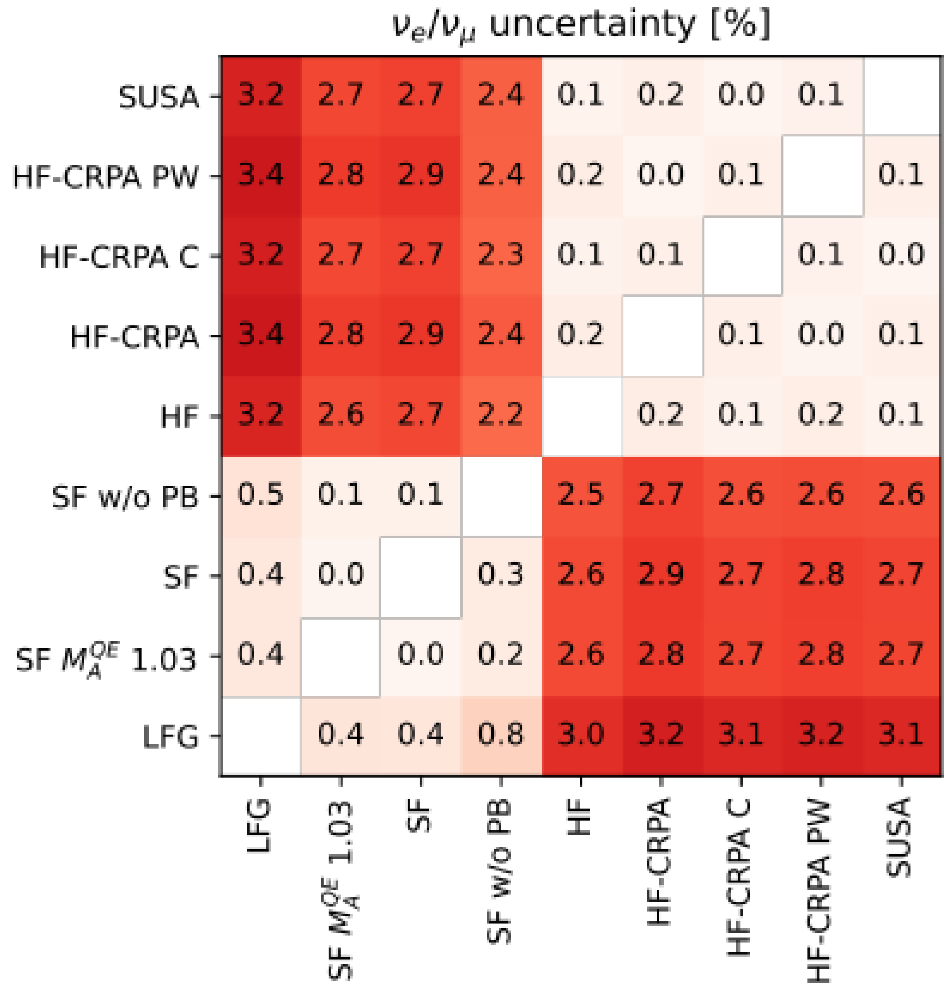
- Design new uncertainties in your oscillation, cross section, or BSM analysis inspired by model differences
- e.g. if you're dependent on measuring neutrons you should really study the generator differences (or specific data) carefully!



Phys. Rev. D 105, 092004

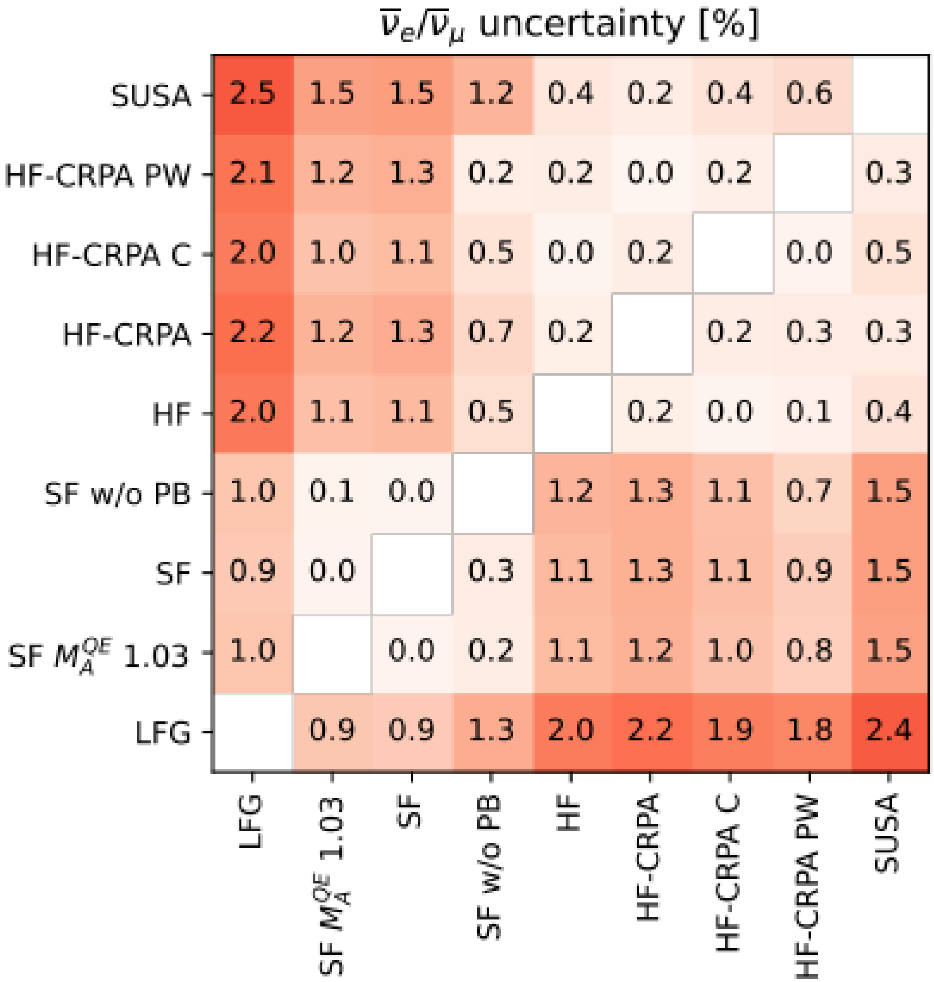
NUISANCE examples

- Study the impact of new models on ν_e/ν_μ uncertainties, critical for CP violation searches



ν_e/ν_μ uncertainty [%]

	LFG	SF M_A^{QE} 1.03	SF	SF w/o PB	HF	HF-CRPA	HF-CRPA C	HF-CRPA PW	SUSA
SUSA	3.2	2.7	2.7	2.4	0.1	0.2	0.0	0.1	
HF-CRPA PW	3.4	2.8	2.9	2.4	0.2	0.0	0.1		0.1
HF-CRPA C	3.2	2.7	2.7	2.3	0.1	0.1		0.1	0.0
HF-CRPA	3.4	2.8	2.9	2.4	0.2		0.1	0.0	0.1
HF	3.2	2.6	2.7	2.2		0.2	0.1	0.2	0.1
SF w/o PB	0.5	0.1	0.1		2.5	2.7	2.6	2.6	2.6
SF	0.4	0.0		0.3	2.6	2.9	2.7	2.8	2.7
SF M_A^{QE} 1.03	0.4		0.0	0.2	2.6	2.8	2.7	2.8	2.7
LFG		0.4	0.4	0.8	3.0	3.2	3.1	3.2	3.1



$\bar{\nu}_e/\bar{\nu}_\mu$ uncertainty [%]

	LFG	SF M_A^{QE} 1.03	SF	SF w/o PB	HF	HF-CRPA	HF-CRPA C	HF-CRPA PW	SUSA
SUSA	2.5	1.5	1.5	1.2	0.4	0.2	0.4	0.6	
HF-CRPA PW	2.1	1.2	1.3	0.2	0.2	0.0	0.2		0.3
HF-CRPA C	2.0	1.0	1.1	0.5	0.0	0.2		0.0	0.5
HF-CRPA	2.2	1.2	1.3	0.7	0.2		0.2	0.3	0.3
HF	2.0	1.1	1.1	0.5		0.2	0.0	0.1	0.4
SF w/o PB	1.0	0.1	0.0		1.2	1.3	1.1	0.7	1.5
SF	0.9	0.0		0.3	1.1	1.3	1.1	0.9	1.5
SF M_A^{QE} 1.03	1.0		0.0	0.2	1.1	1.2	1.0	0.8	1.5
LFG		0.9	0.9	1.3	2.0	2.2	1.9	1.8	2.4

- Led to new uncertainties in T2K’s oscillation analysis

Phys.Rev.D 108 (2023) L031301

Preparing for NuInt

- Now when you see new results from experiment N in variable X, you can do a multi-generator comparison!
  - (Provided we have the flat tree generated)
  - (And they provide a sufficient signal definition)
- You can start exploring what measurements you can make in your experiment to expose regions where generator differences are the largest
  - Guide the generator community by measurements

- Do you want even more?
  - NUISANCE page for tutorials: https://nuisance.hepforge.org/tutorials/general.html

Extra

- What about adding a variable to the flat-tree?
- In NUISANCE, the flat tree is just another “experiment” class
- Class members are set and written to file for each event, with default values
- Does require a recompilation
- We’ll mention this briefly in tomorrow’s tutorials