Uniform Beam Simulation Study with g4numi (Details on Simulation : What We Changed)

Sudeshna Ganguly

03/11/2021

Set BEAM_POSITION_X to be a Uniform Distribution

In g4numi simulation source code

- Aim: Perform Beam position X-scan analysis
  - Select any given Gaussian on the beam position x Uniform distribution
  - For each Gaussian distribution plot muon flux on muon monitors
  - For each Gaussian distribution plot muon monitor centroid
  - Perform similar Y-scan analysis next
  - We can create large statistics with uniform beam x and y and reproduce target scan study with simulation

Create Uniform beam x distribution

Code added:

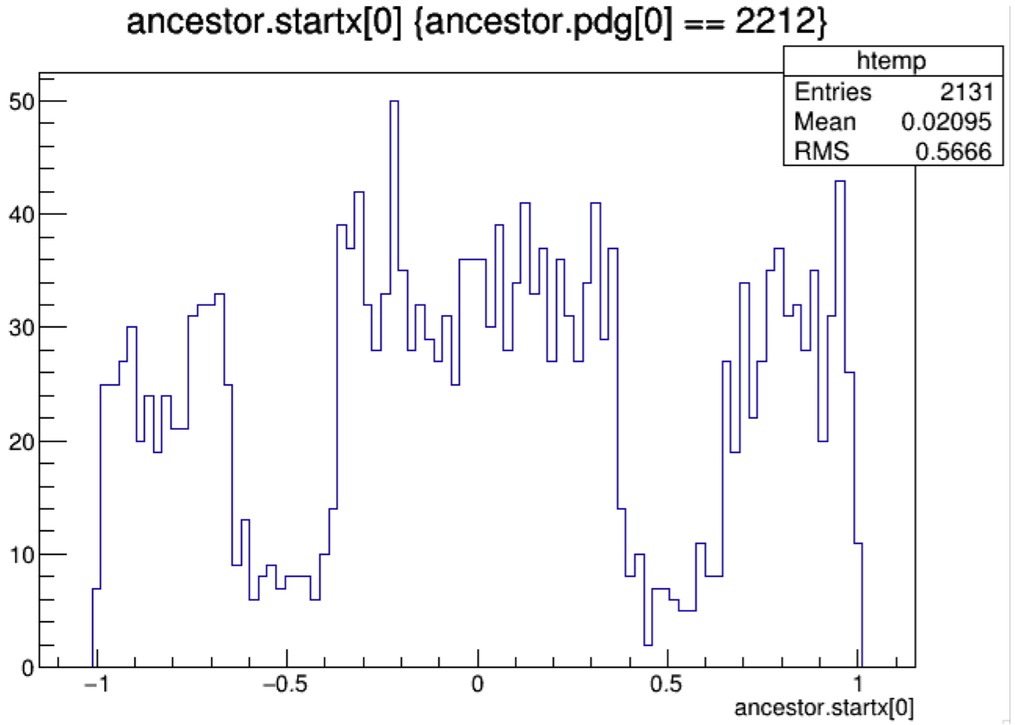
- In /nashome/s/sganguly/clean_backup_g4numi/g4numi/src/NumiPrimaryGeneratorAction.cc : add uniform distribution for beam position x instead of the default Gaussian distribution
- This src code change will change the proton beam x distribution in the output root file

```
if (outR > 0)
{
  G4double r0  = G4UniformRand()*(outR - inR) + inR;
  G4double phi0 = G4UniformRand()*2*M_PI;

  x0 = cos(phi0) * r0;
  y0 = sin(phi0) * r0;
}
else
{
   //
   //I changed this.  It doesn't make sense to ever have x=y=0 -Laura
   //
   //x0 = 0;
   //y0 = 0;

   G4double sigmax=fND->beamSigmaX;
   G4double sigmay=fND->beamSigmaY;

   //x0 = G4RandGauss::shoot(fND->beamPosition[0],sigmax);
   x0 = 8*(G4UniformRand() - 0.5)*mm;
   y0 = G4RandGauss::shoot(fND->beamPosition[1],sigmay);
}
```

```
        this->Geantino(anEvent);
  }
  else
  {
     // If nothing else is set, use a basic proton beam
     G4double x0;
     G4double y0;
     G4double z0;
     G4double sigmax=fND->beamSigmaX;
     G4double sigmay=fND->beamSigmaY;

     //x0 = G4RandGauss::shoot(fND->beamPosition[0],sigmax);
     x0 = 8*(G4UniformRand() - 0.5)*mm;
     y0 = G4RandGauss::shoot(fND->beamPosition[1],sigmay);
     z0 = fND->beamPosition[2];

     fProtonOrigin    = G4ThreeVector(x0, y0, z0);
     fProtonMomentum = G4ThreeVector(0, 0, fND->protonMomentum);
     fProtonIntVertex = G4ThreeVector(-9999.,-9999.,-9999.);
     fWeight=1.; //for primary protons set weight and tgen
     fTgen=0;

     fParticleGun->SetParticlePosition(G4ThreeVector(x0,y0,z0));
     fParticleGun->GeneratePrimaryVertex(anEvent);
  }

  fCurrentPrimaryNo++;
}
```

Create Uniform beam x distribution

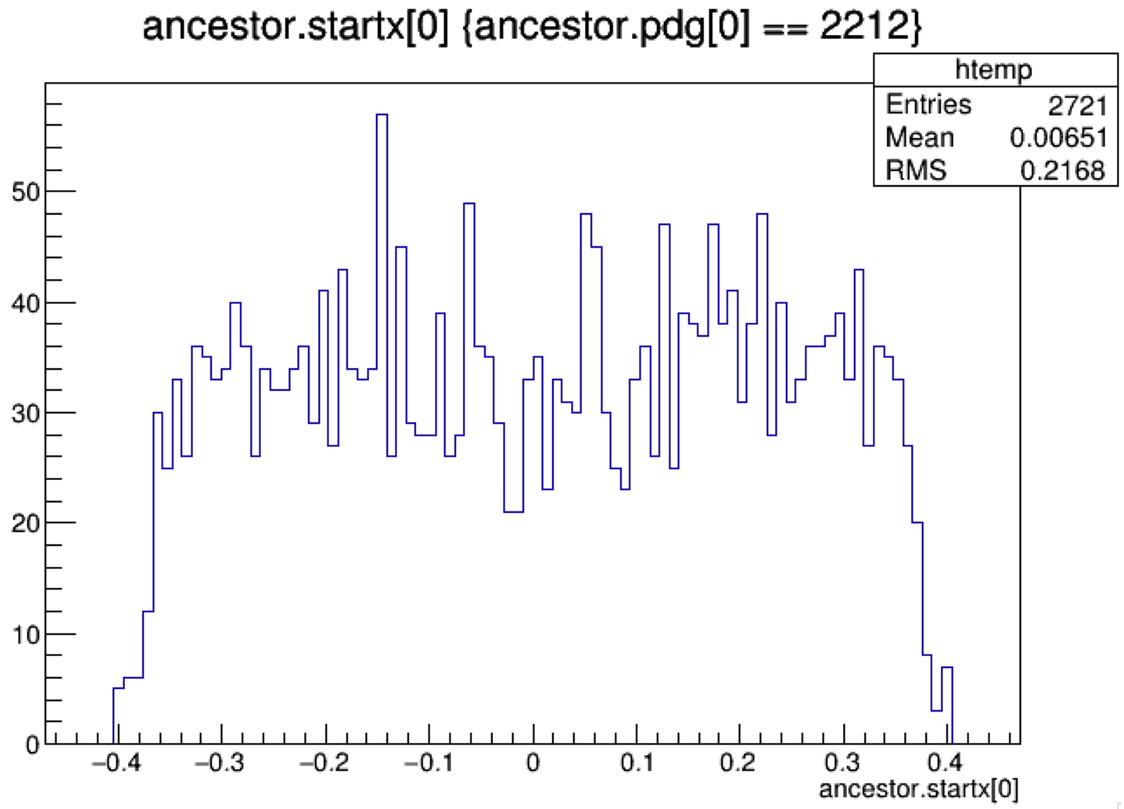
- Choosing x0 = 20*(G4UniformRand() - 0.5)*mm; (distribution b/w +/- 10 mm)
- Width of target = 9 mm, following plot shows target interaction b/w +/- 4.5 mm
- After that shows interaction with baffle

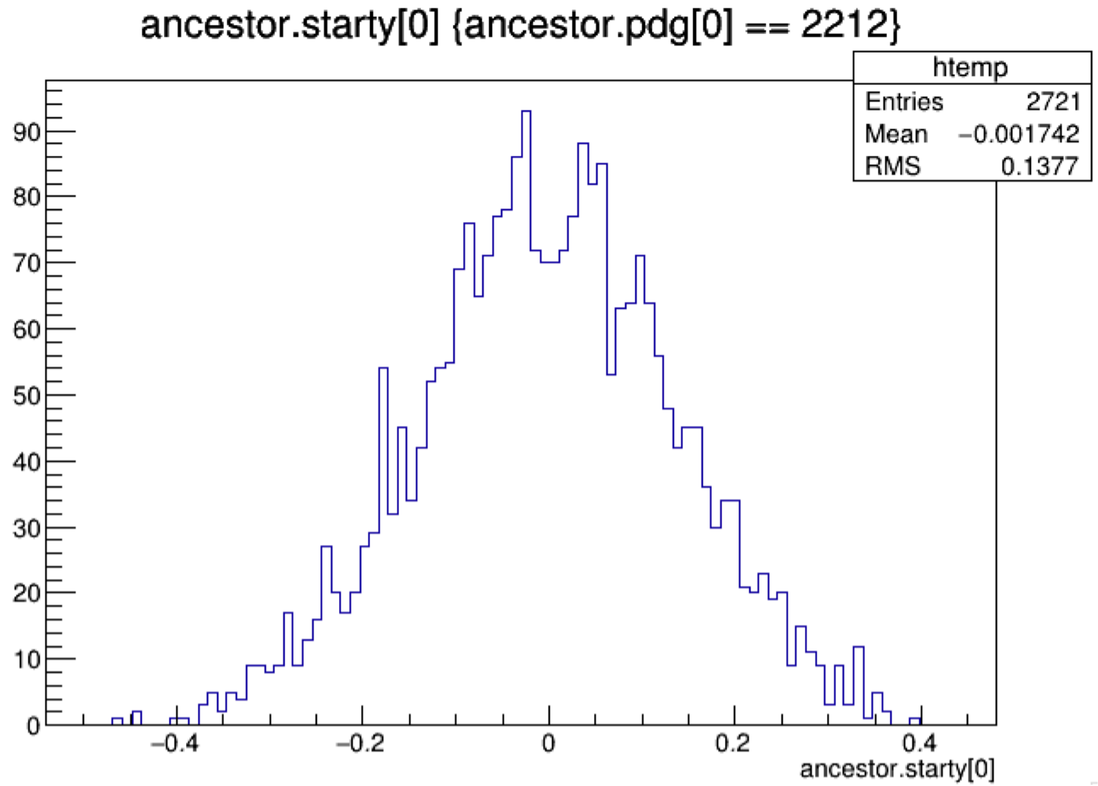


Create Uniform beam x distribution

- To do scan within target we can choose x0 = 8*(G4UniformRand() - 0.5)*mm; (distribution b/w +/- 4 mm)

- We have only changed x-distribution for now, y remains the same (can do another study in y later)



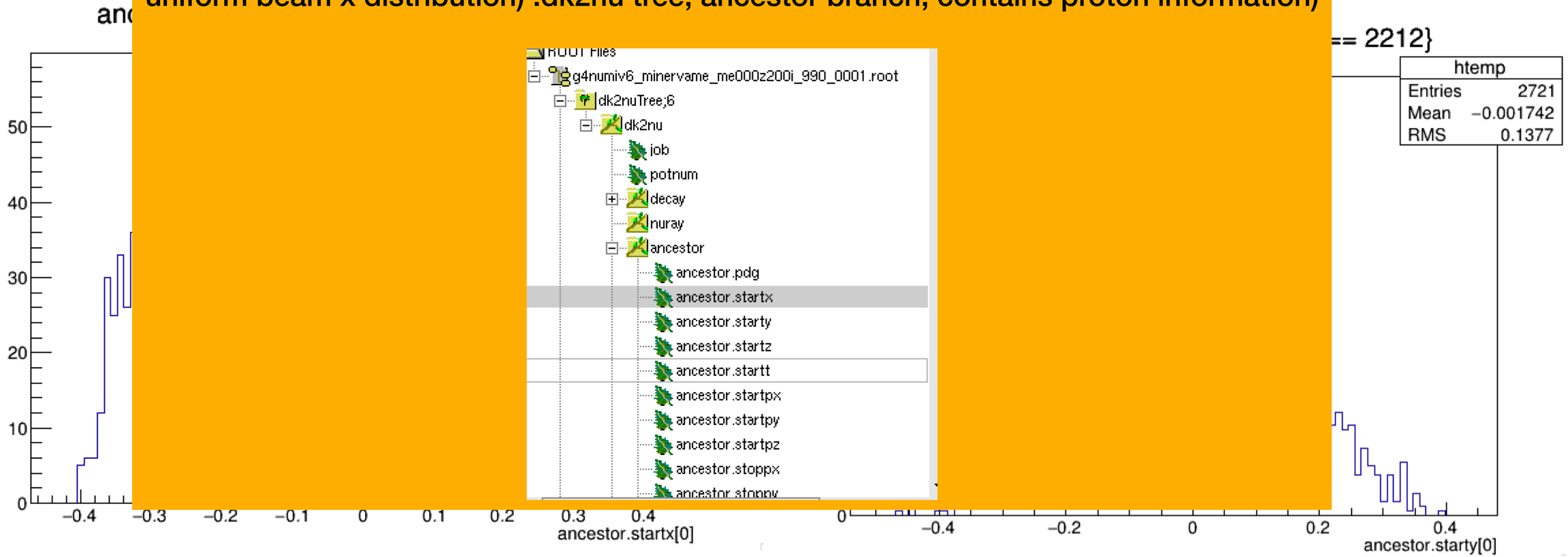


Create Uniform beam x distribution

- To do scan within target we can choose x0 = 8*(G4UniformRand() - 0.5)*mm; (distribution b/w +/- 4 mm)

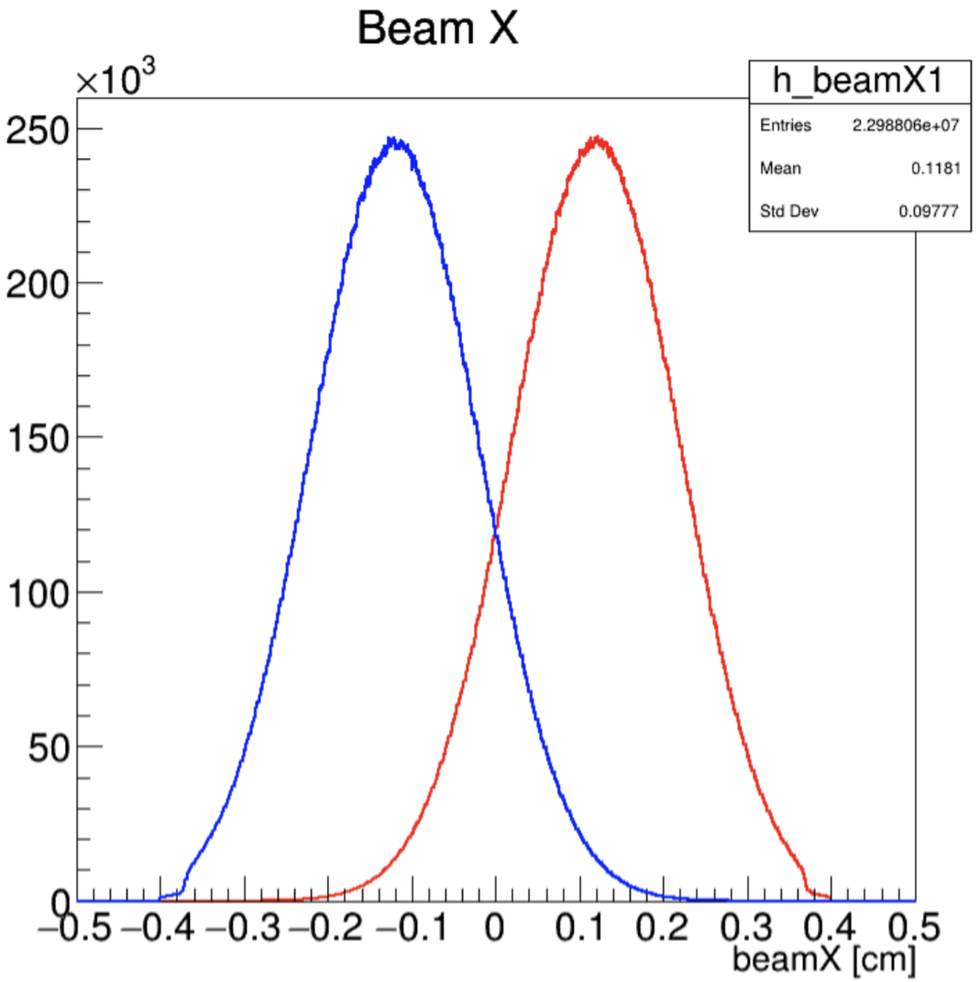
- We have only changed x-distribution for now, y remains the same (can do anther study in y later)

These plots are from g4numiv6_minervame_me000z200i_*.root files (containing uniform beam x distribution) :dk2nu tree, ancestor branch, contains proton information)

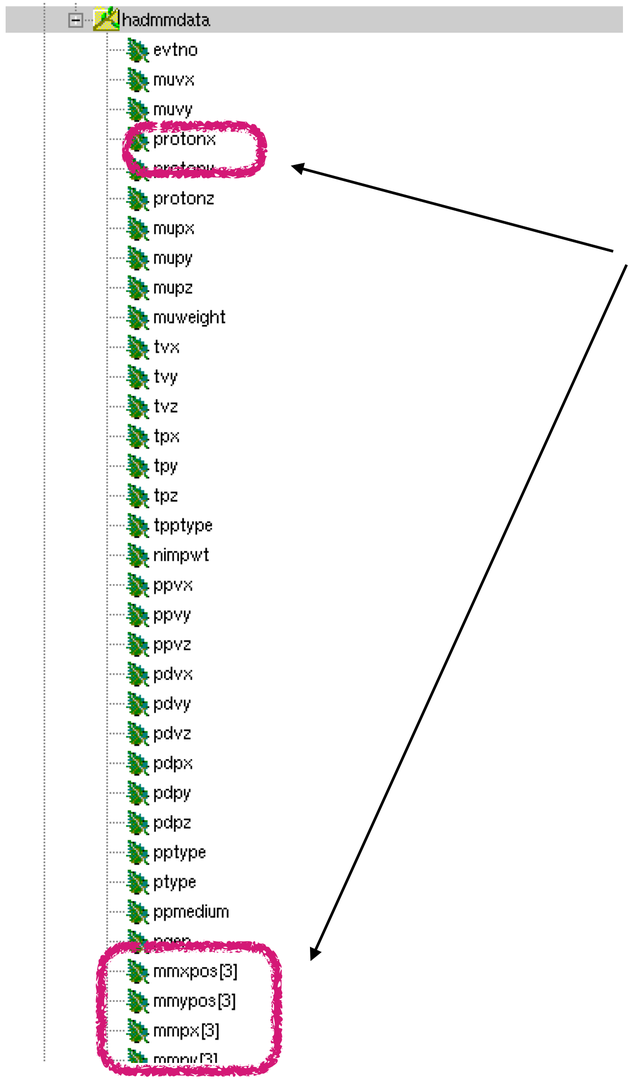


- At this stage, we can create uniform beam X distribution
- Next need to select a given Gaussian on the beam position x Uniform distribution (can select many): calculate weights along beam x according to defined gaussian
- For example:

From Athula



- Next want to find correlation on Muon Monitor centroid for each of the gaussian distribution - needs to look at Muon Monitor ntuples : hadmmtuple_t.cc



- In order to find correlations b/w proton beam x and muon monitor information, the **protonx** branch needs to be added to this root branch where the beam x is a Uniform distribution

- In other words, the code changes we made in previous slides, only changes proton beam x in the dk2nu ntuple

- To apply the same changes in MM ntuple, requires a lot of code change

- Athula, I and Yiding did a marathon zoom coding/brainstorming session

Code changes to add Uniform protonx distribution in MM ntuple (Writing it down here for future reference)

- In /nashome/s/sganguly/clean_backup_g4numi/g4numi/dk2nu_local/src/nu2mubatch, add proton beam position information into the tree

```
UShort_t  evtno;
Float_t  muvx,  muvy,  muvz,  protonx,  protony,  protonz;
Float_t  mupx,  mupy,  mupz;
```

```
muonTree->Branch("evtno",&evtno,"evtno/s");
muonTree->Branch("protonx",  &protonx,  "protonx/F");
muonTree->Branch("protony",  &protony,  "protony/F");
muonTree->Branch("protonz",  &protonz,  "protonz/F");
```

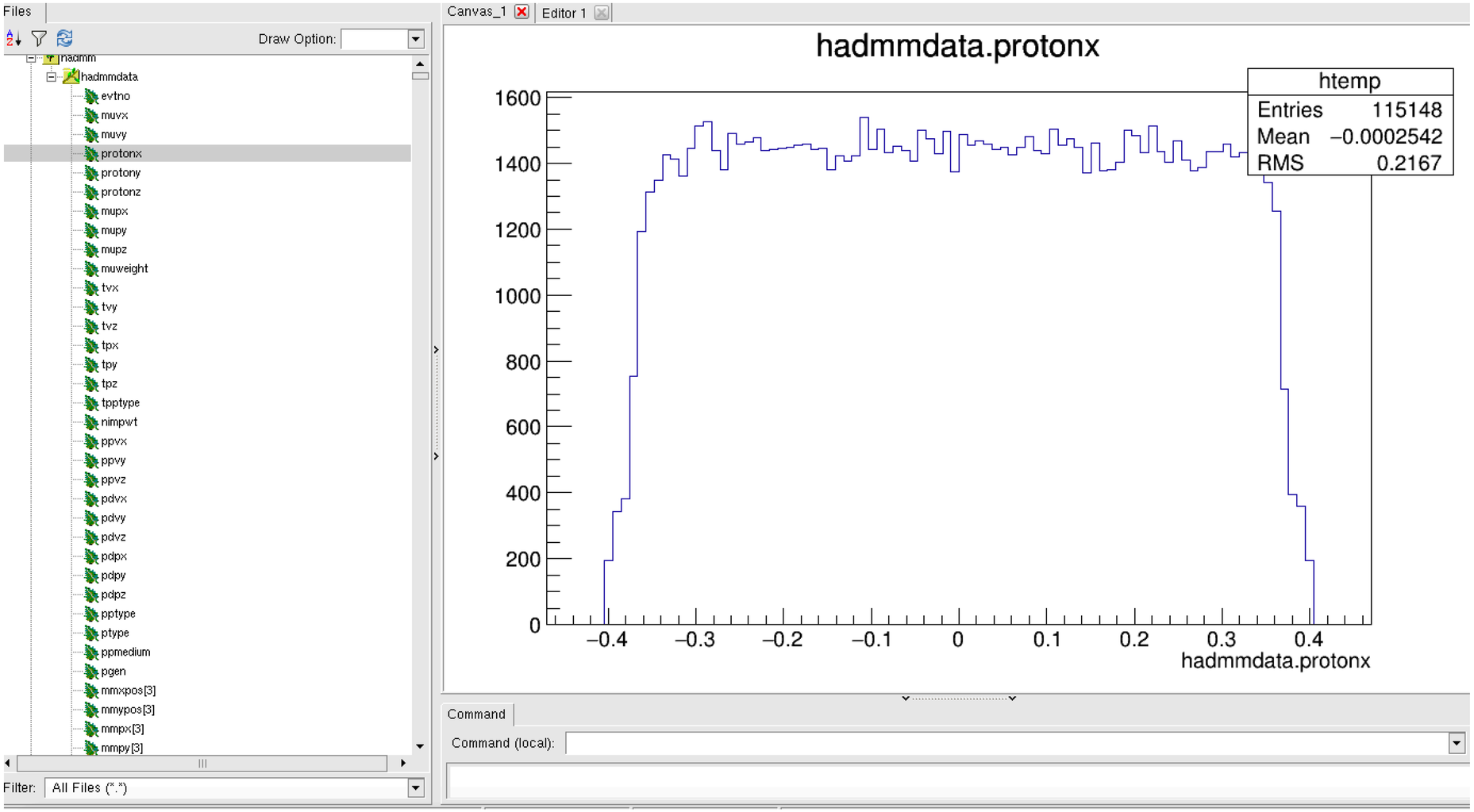
```
if(goodDecay) {

  //double temp_mupx = (Float_t)(dk2nu->decay.pdpx - dk2nu->nuray[0].px);
  //double temp_mupy = (Float_t)(dk2nu->decay.pdpy - dk2nu->nuray[0].py);
  //double temp_mupz = (Float_t)(dk2nu->decay.pdpz - dk2nu->nuray[0].pz);
  //double tot_momentum = sqrt(temp_mupx*temp_mupx+temp_mupy*temp_mupy+temp_mupz*temp_mupz);
        //std::cout<<" Total muon Momentum "<<tot_momentum<<std::endl;
        //if(tot_momentum<4.0)continue;
        evtno = dk2nu->potnum;
        protonx = dk2nu->ancestor[0].startx;
        protony = dk2nu->ancestor[0].starty;
        protonz = dk2nu->ancestor[0].startz;
```

Code changes to add Uniform protonx distribution in MM ntuple (Writing it down here for future reference)

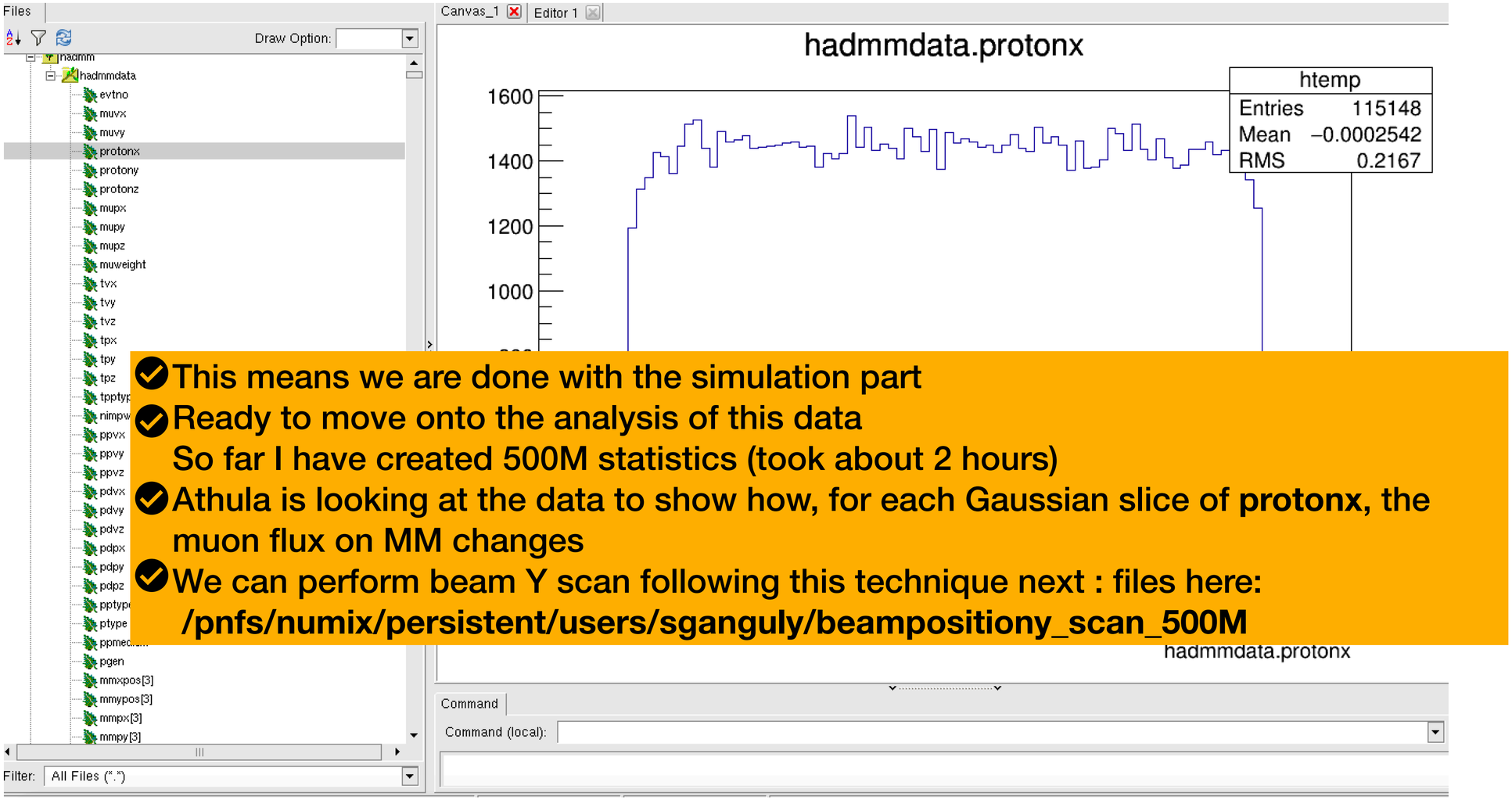
- Then make more code changes (could be pushed to the repository):
- Do a “grep” to see which pieces of code went through a change:

```
<numixgpvm01.fnal.gov> grep "protonx" *.cc
hadmmtuple_t.cc:  protonx(-99999.),
hadmmtuple_t.cc:  protonx = -99999.;
NtpMuon.cc:   protonx(-999.0),
NtpMuon.cc:  protonxF(-999.0),
NtpMuon.cc:   SetBranch(&protonxF, "protonx");
NtpMuon.cc:   protonx = (Double_t)protonxF;
NtpMuon.cc:   protonxF = -999.0;
NumiAnalysis.cc:     g4hmmdata->protonx = NPGA->Getprotonx();
NumiPrimaryGeneratorAction.cc:    fprotonx = fMuon->protonx;

<numixgpvm01.fnal.gov> grep "protonx" ../include/*.hh
../include/hadmmtuple_t.hh:  Float_t protonx;
../include/NtpMuon.hh:  Double_t protonx;
../include/NtpMuon.hh:  Float_t protonxF;
../include/NumiPrimaryGeneratorAction.hh:   G4double Getprotonx()                {return fprotonx; }
../include/NumiPrimaryGeneratorAction.hh:  G4double fprotonx;
```

After the code change look at MM ntuple



After the code change look at MM ntuple



- This means we are done with the simulation part
- Ready to move onto the analysis of this data
  So far I have created 500M statistics (took about 2 hours)
- Athula is looking at the data to show how, for each Gaussian slice of **protonx**, the muon flux on MM changes
- We can perform beam Y scan following this technique next : files here:
  /pnfs/numix/persistent/users/sganguly/beampositiony_scan_500M