ParticleBomb: a model independent event generator

Andy Furmanski

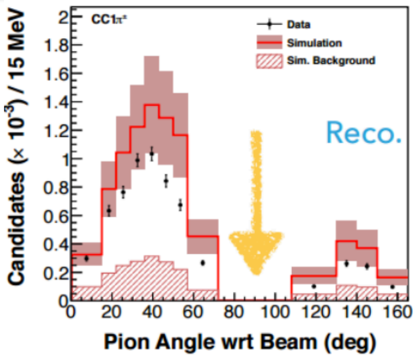
Sim meeting Feb 21 2017

MANCHESTER
1824
The University of Manchester

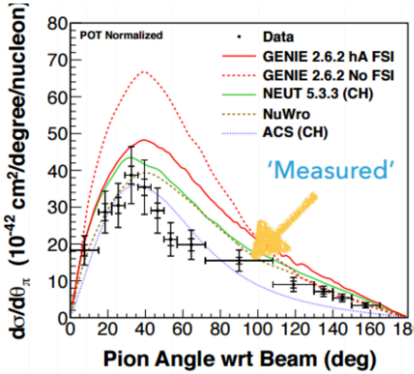
Model dependency

- A common discussion within the neutrino xsec community:
  - How model-dependent is this measurement?
  - Oh, it's completely model-independent
  - So how do you estimate the efficiency?
  - Oh, we use MC with the default GENIE model
  - so your data depends on the GENIE model…

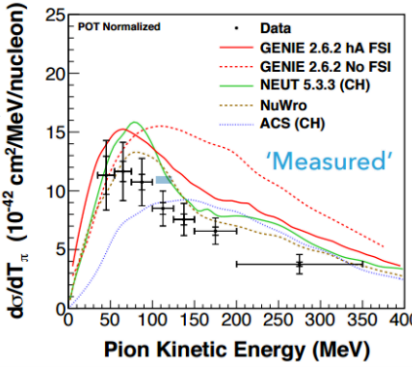
- One problem is that GENIE only produces events in parts of phase space that GENIE wants to
  - Different models can cover different phase-spaces
  - What if nature fills some part of phase space that no generator does?



Reco.



'Measured'



'Measured'

But what can we do about it?

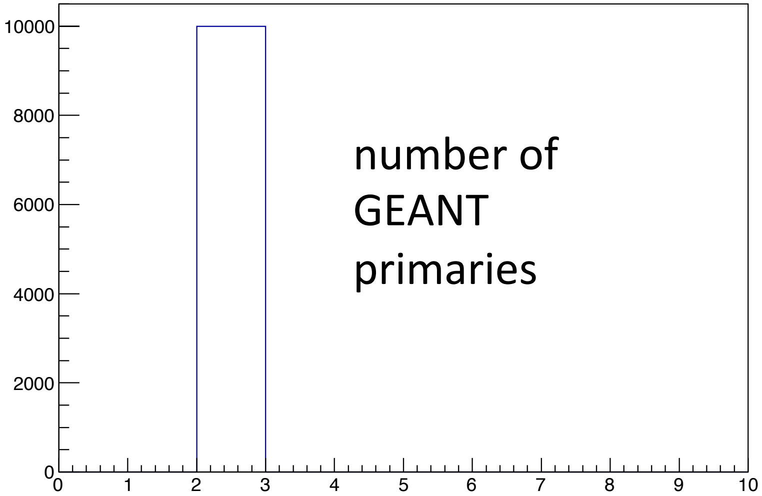
- Instead of relying on GENIE's predictions, just put events everywhere in the phase space you care about!
- Use a particle bomb
  - Like a particle gun, but with more than one particle…
- SingleGen currently can generate multiple particles, but their locations are either:
  - all random within an event
  - fixed event-to-event

Task #1

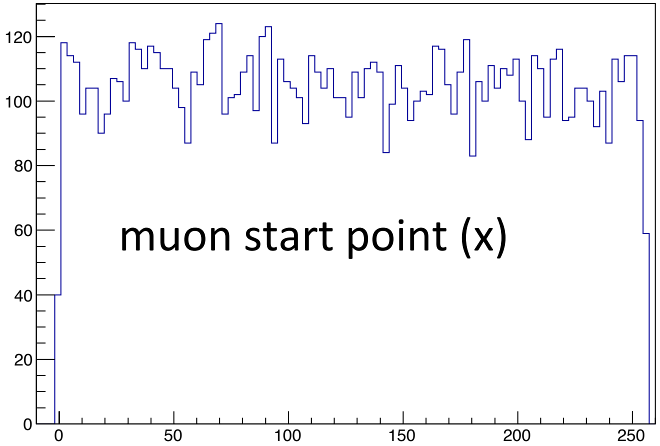
- **Modify SingleGen** to be able to generate multiple particles all in a single, random location
- new fcl parameter – SingleVertex: true
- If set to true, it selects a location first, then picks particle kinematics for each particle individually
- I believe this is a non-breaking change

Did I do it right?

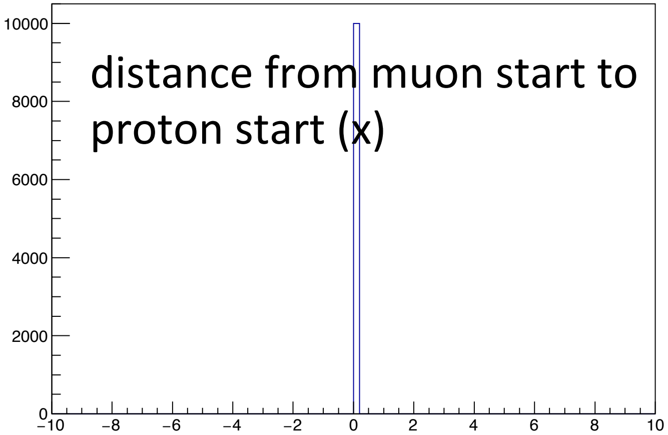
- Generated 10k mu+p events in uBooNE



number of GEANT primaries



muon start point (x)



distance from muon start to proton start (x)

- Plots look the same for y and z positions
- I seem to be generating pairs of particles in a single, random location ✓

Awesome, sounds perfect!

- **It's not**.

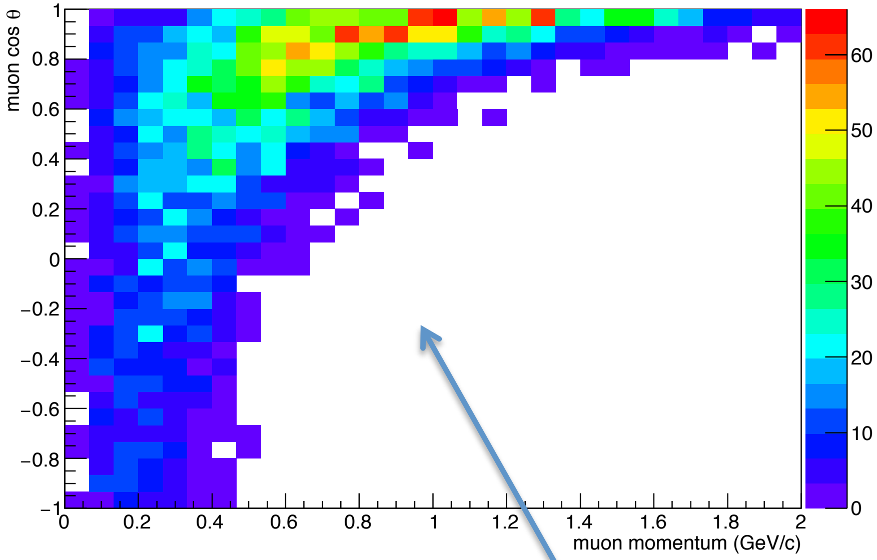
- **Statistics** will become your enemy. Quickly
- You have to **be** a bit **smart** about it.
  - This is up to each analyser to do
  - But, for example, how much do you care about the proton efficiency for a 4GeV/c proton?
    - So maybe you don't need so many events produced over there?

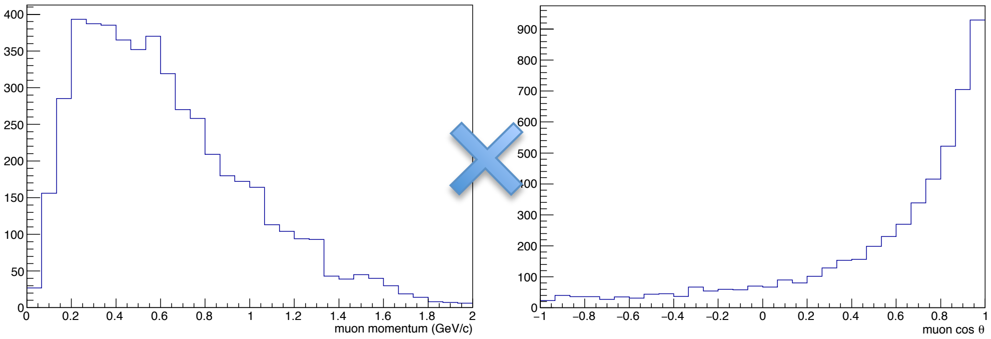
- Useful idea:
  - Generate events according to input histogram distributions
  - Decouple angle, momentum, multiplicity, etc
  - But stick to "realistic" areas of phase space, and weight according to how much you care

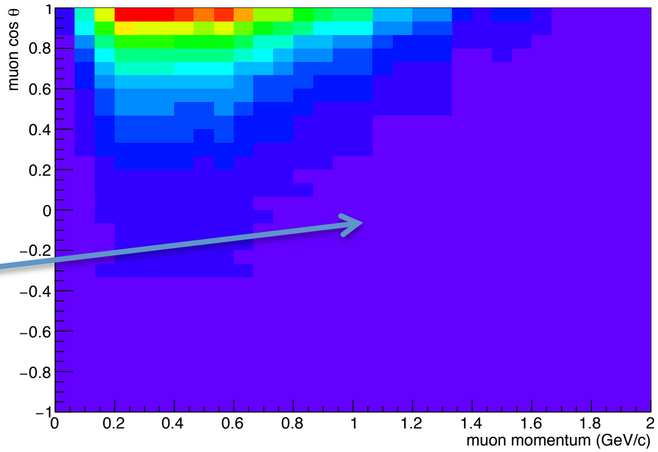
I need an example!

- GENIE couples muon angle and momentum implicitly:

- But if we take away the correlation:







We've filled in this area whilst still retaining the general shape of the distribution

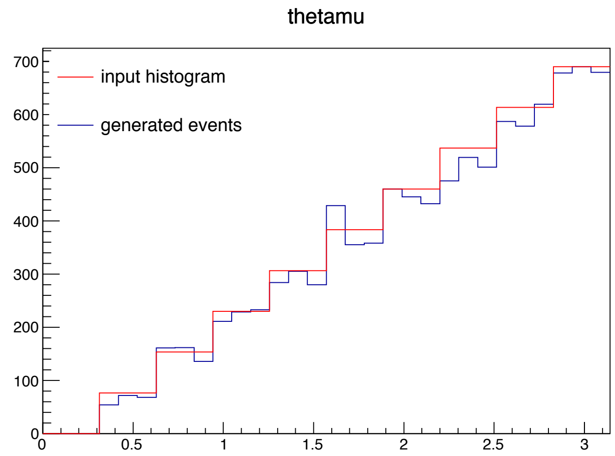
Task #2

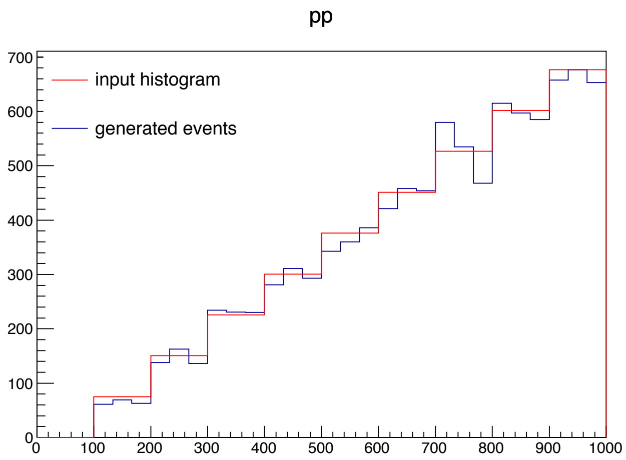
- **Modify SingleGen** to accept input histograms for particle momenta/angles
- Simply set Pdist and AngleDist to 2
- fcl parameters:
  - HistogramFile: “filename.root”
  - Phist: [“h_muonmom”, “h_protonmom”]
  - ThetaXzYzHist: [“h_muonangle”,”h_protonangle”]
- Just need a file with histograms in
- Can re-use histograms for multiple particles

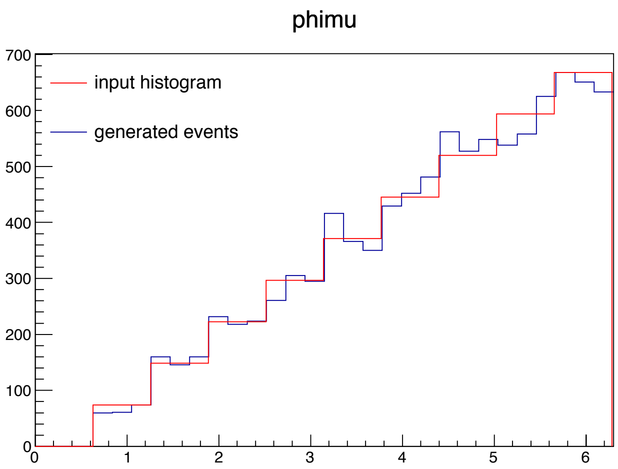
Note – the angular histogram is a 2D histogram in thetaxz ,thetayz

And did I do this one right?

- 10k muon+proton events generated
- Simple toy input histograms – SetBinContent(i, i)
  - i.e. linearly increasing
- Generated events match input histograms ✓







Wait, I want to vary my multiplicity!

- I didn't implement a multiplicity variation
  - Mainly because it's slightly more complex
  - Also because of the statistics problem
    - Phase space just grows and grows
- My (Wes's) solution:
  - Use a histogram with a large bin at 0
  - 0 momentum particles won't get tracked by G4
  - Therefore the multiplicity will be less at random
- Alternative solution
  - Think about the exact topologies you care about, and produce separate samples

Conclusions

- I have **changes available for larsim**, that enable 2 things:
  - **Producing multiple particles** in a single, random location
  - **Selecting momenta and angles from input histograms**
- I believe this is all non-breaking
- This is potentially useful, but analysers still need to **think carefully about the statistics available**
  - Most useful when **restricting yourself to a specific topology**
- Code is ready to merge