fiTQun MCMC-based fitter

- Richard Calland – Kavli IPMU (UTokyo)

- fiTQun! Workshop @TRIUMF - 20th August 2015

Disclaimer

- I haven't really made any real progress recently
- I will be presenting a recap of the idea, along with some new ideas and studies
- I plan to put some more momentum behind this now that the GPU work has reached a milestone

Issues with minimization

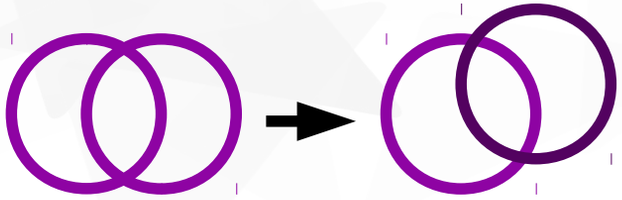
- Currently, fiTQun multi-ring fitter will minimize different ring configurations and make decisions based on the maximum likelihood estimation (MLE)
- This is flawed; comparing likelihoods of different dimensionality
- Instead of minimizing, why not try sampling the posterior probability
  - The *maximum a posteriori* (MAP) estimate will provide the most likely configuration
- How to sample from a likelihood function where the number of parameters change?

Reversible Jump MCMC

- Reversible jump Markov chain Monte Carlo (RJMCMC) was proposed by Green (1995) as an extension to MCMC algorithms
- Allows the design of dimension-changing “jumps” to move between N discrete component models
  - Parameter inference **and** model comparison is done within one MCMC
- Critically, to preserve the detailed balance of the MCMC, *dimension matching* terms are included in the acceptance probability
  - Detailed balance means $P(a \rightarrow b) = P(b \rightarrow a)$

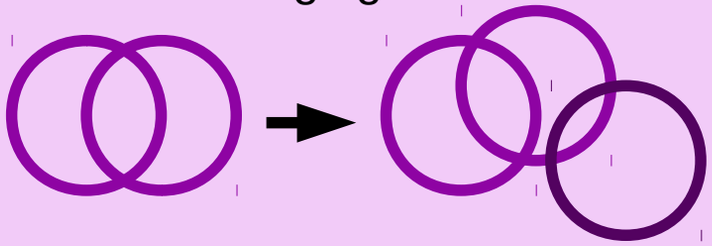
RJMCMC move types

Dimension preserving

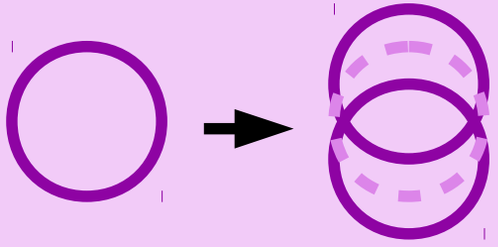


Alter a ring (or N rings)

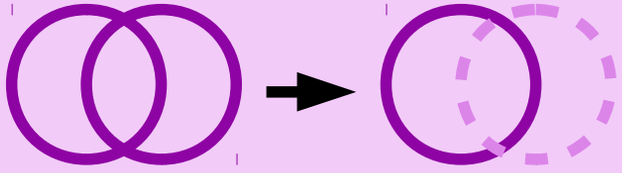
Dimension changing



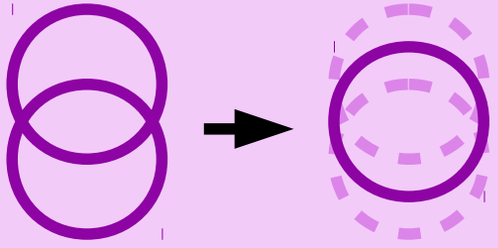
Add ring



Split ring



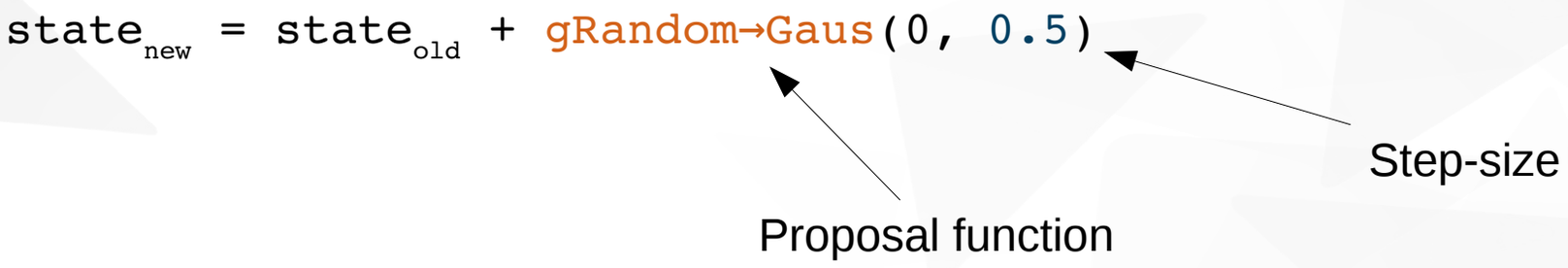
Remove ring



Merge rings

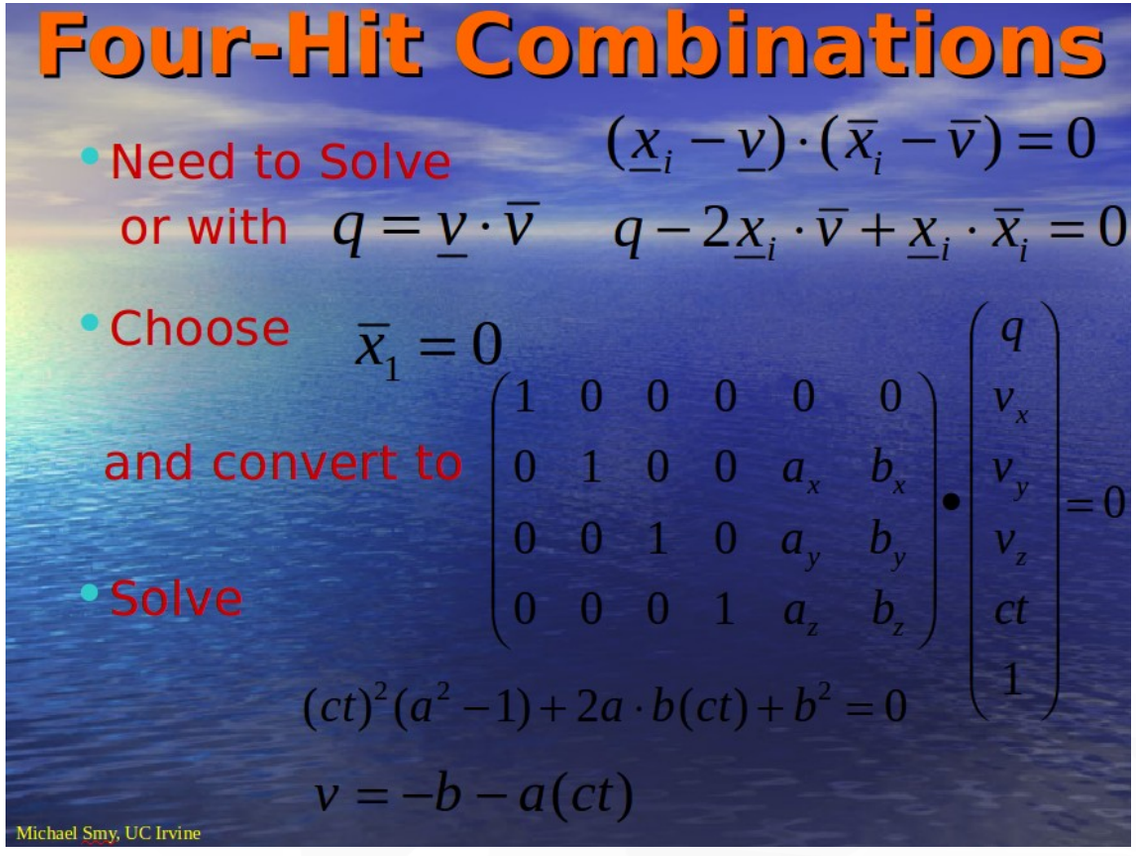
Smarter proposal algorithms

- MCMC proposes new states using a proposal function (e.g. a Gaussian)
- Proposal function is independent of posterior probability
- However, convergence is highly dependent on the proposal function
- Use a data-driven proposal function to propose likely new rings when performing the “add ring” move



Proposal algorithm

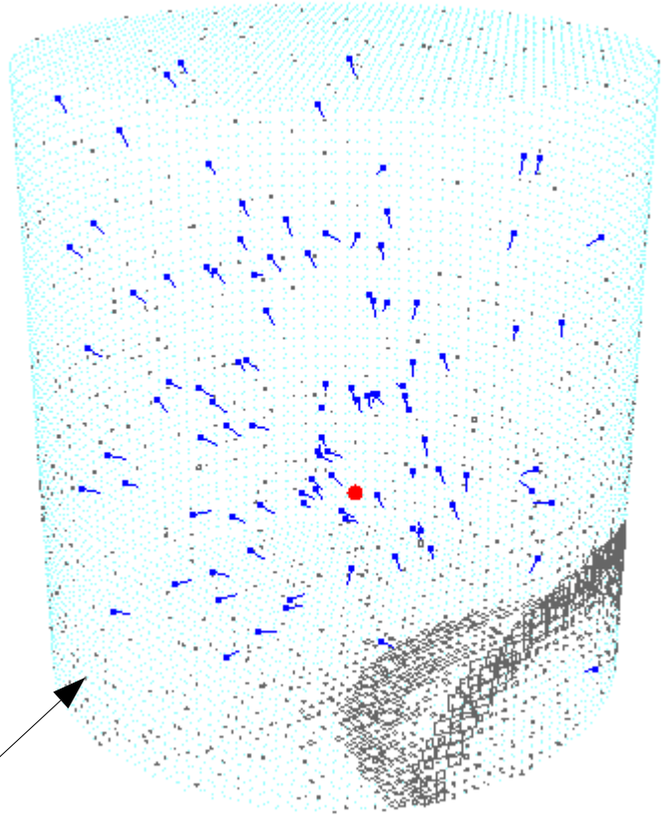
- Borrowed method from SK's bonsai fitter
  - Analytical solution for a vertex [x,y,z,t] given a 4 hit combination
- Generate many 4-hit combinations
  - Select 3 hits within a small time window from the first selected hit
- Resulting distribution of vertices can be used as a proposal distribution
- Use vector from vertex to centre of mass of 4 hits to calculate a direction [θ,φ]
- Can generate a 6D proposal distribution
- No constraint on cherenkov angle!



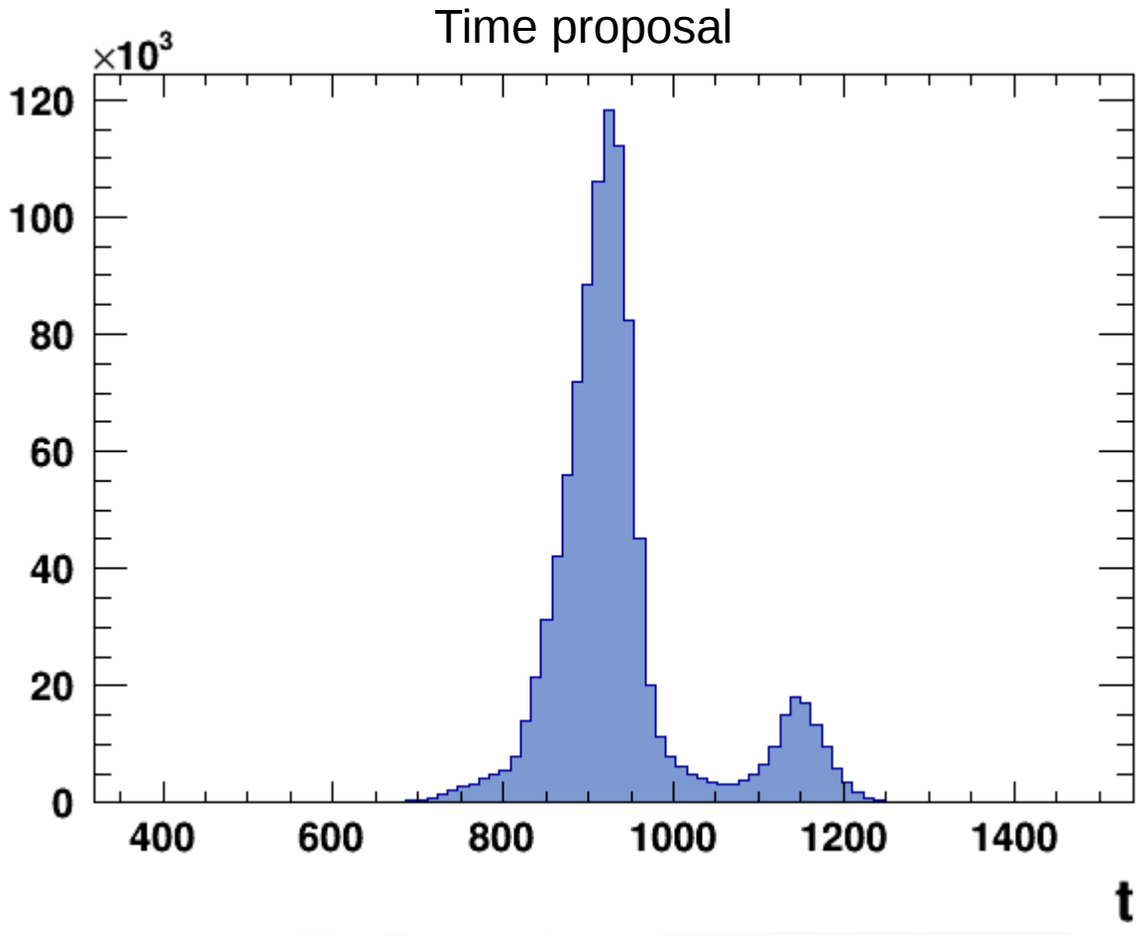
SK Collaboration Meeting June 2003

Muon with decay electron

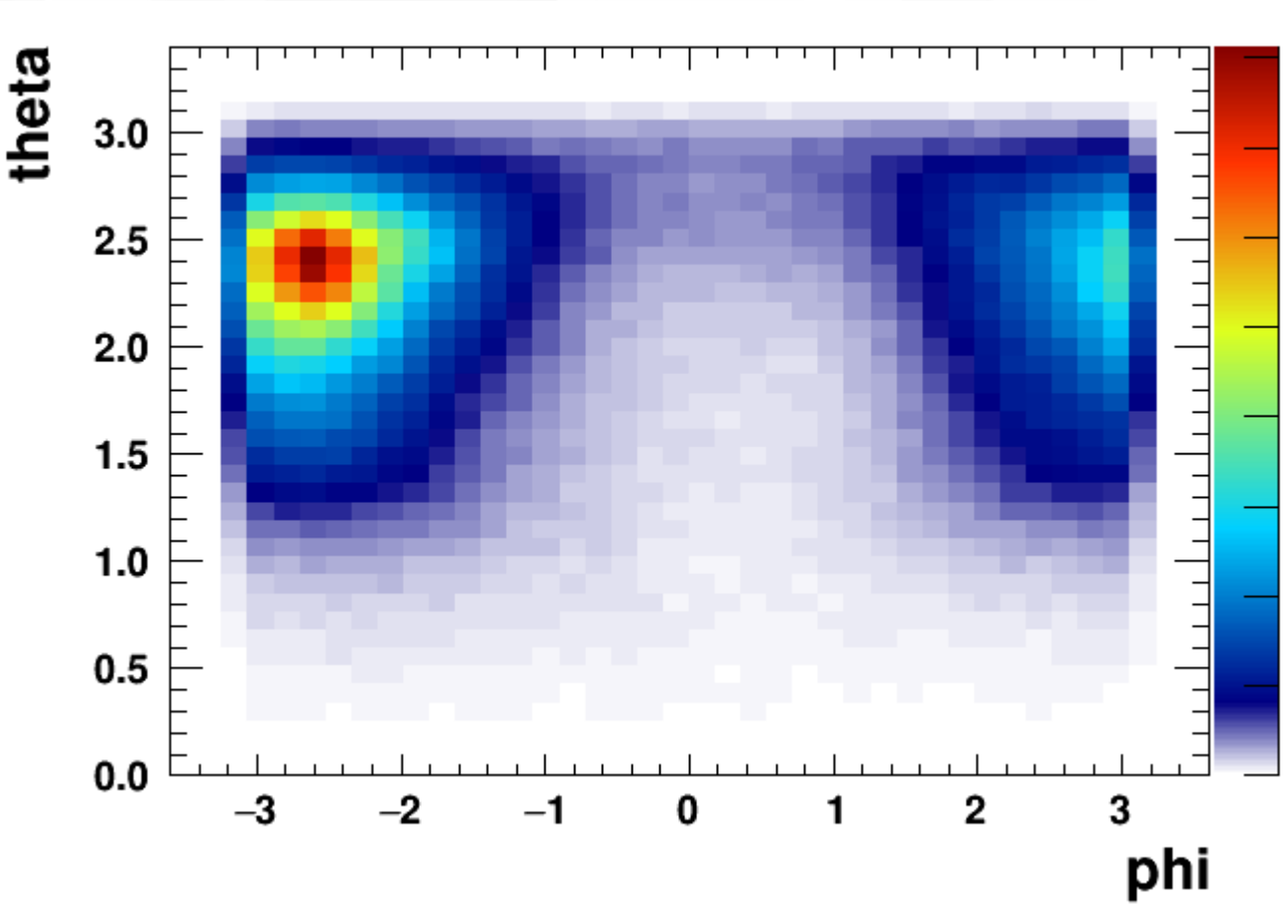
Blue arrows are samples from the proposal distribution

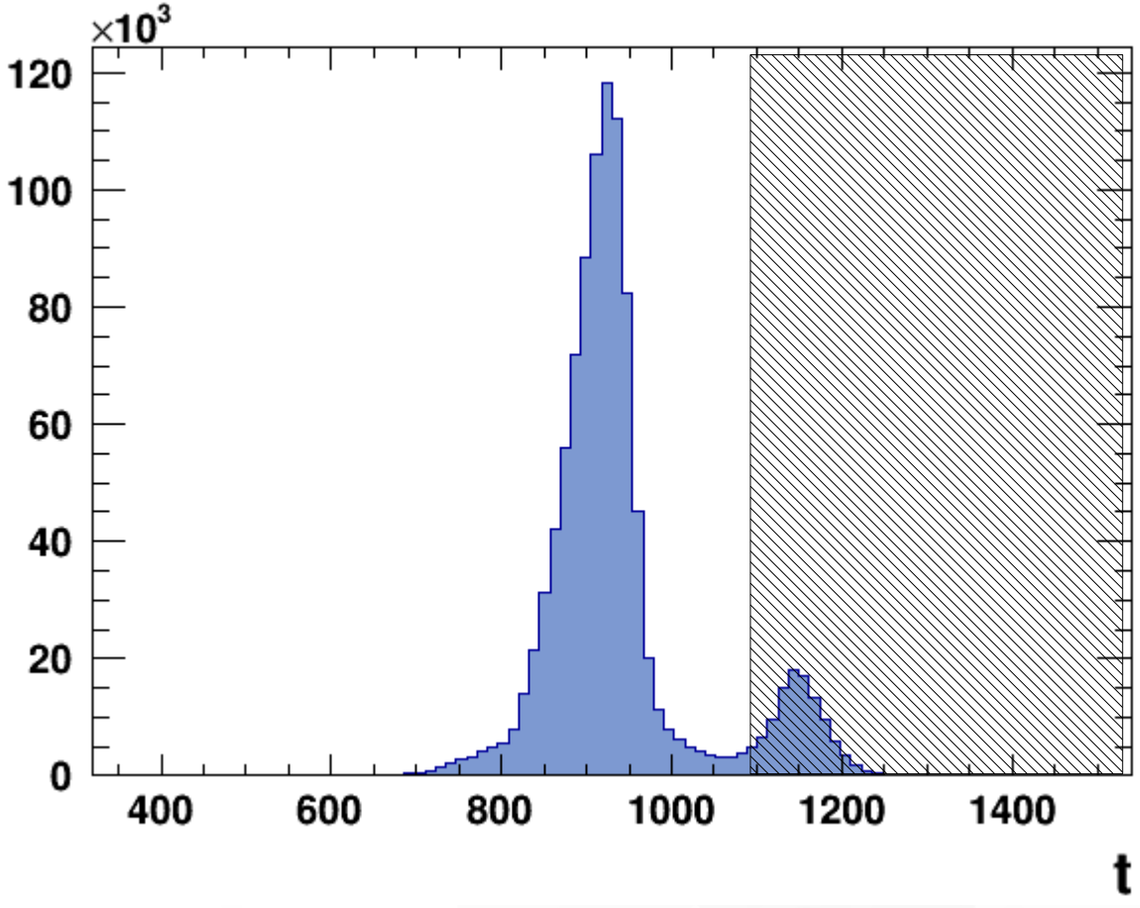


Faint decay-e ring

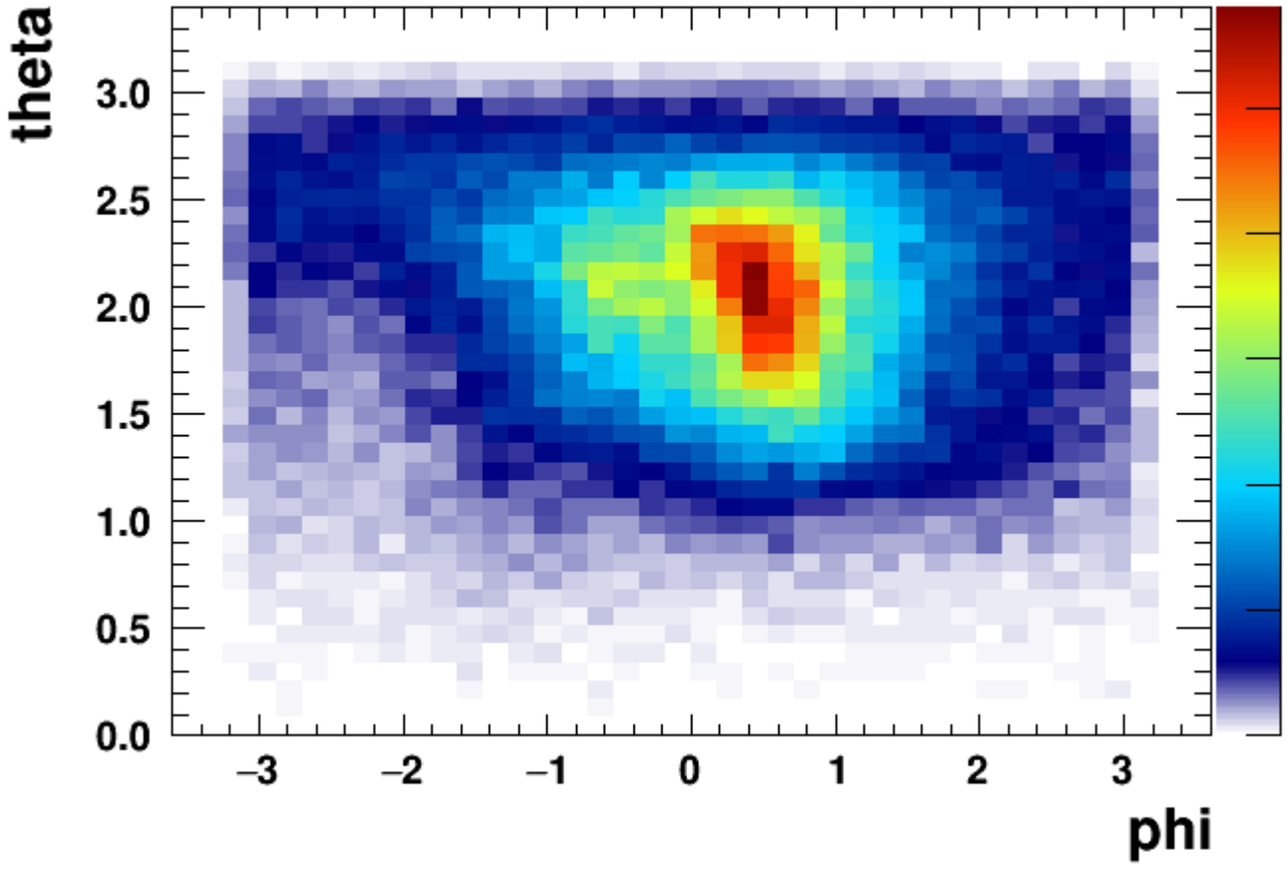


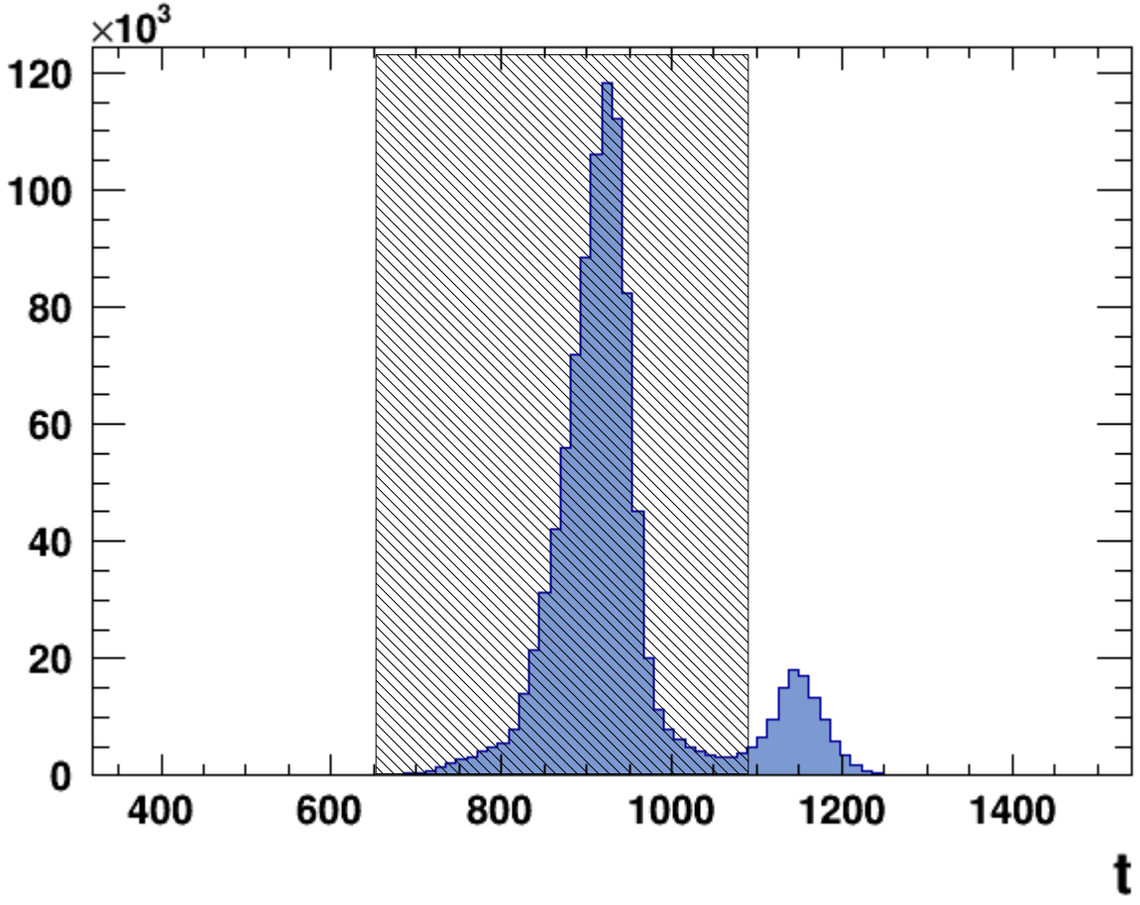
Direction Distribution



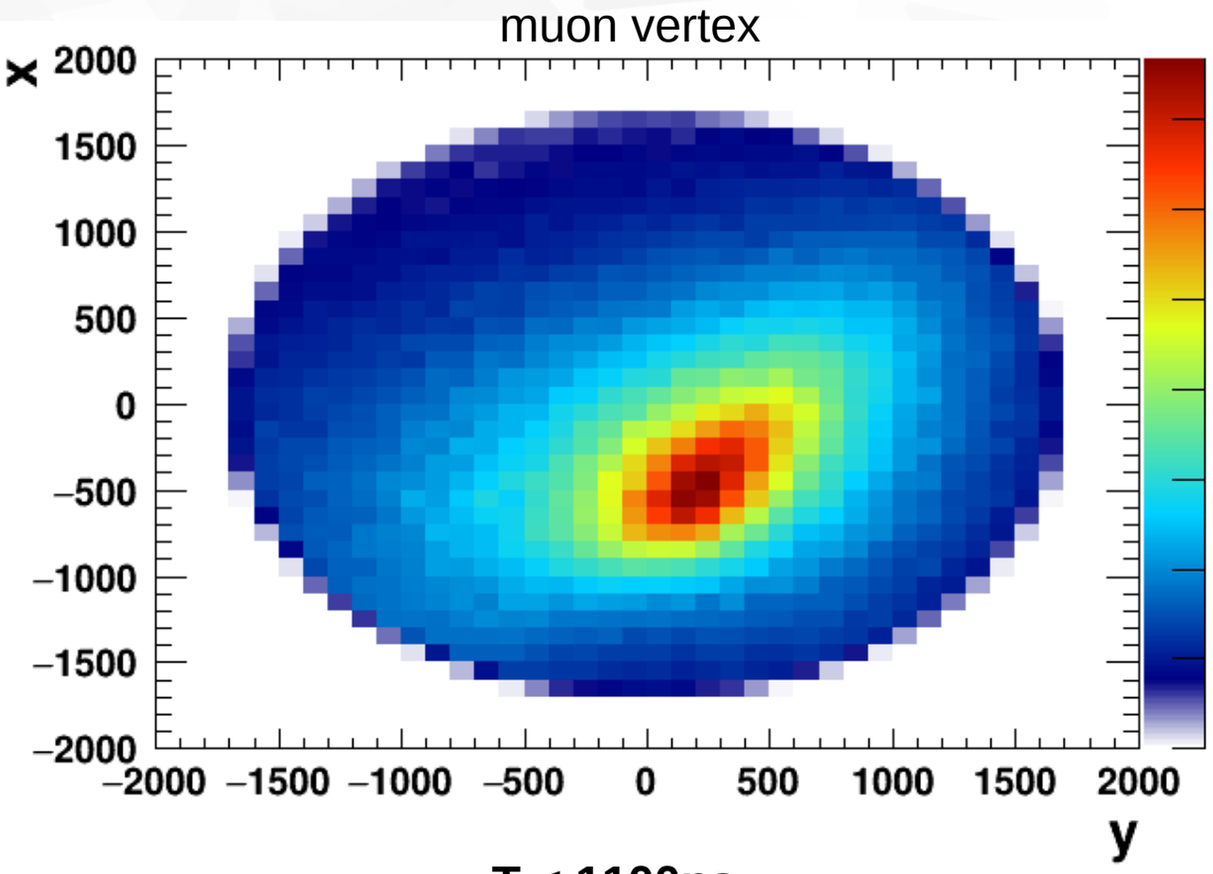


Direction Distribution

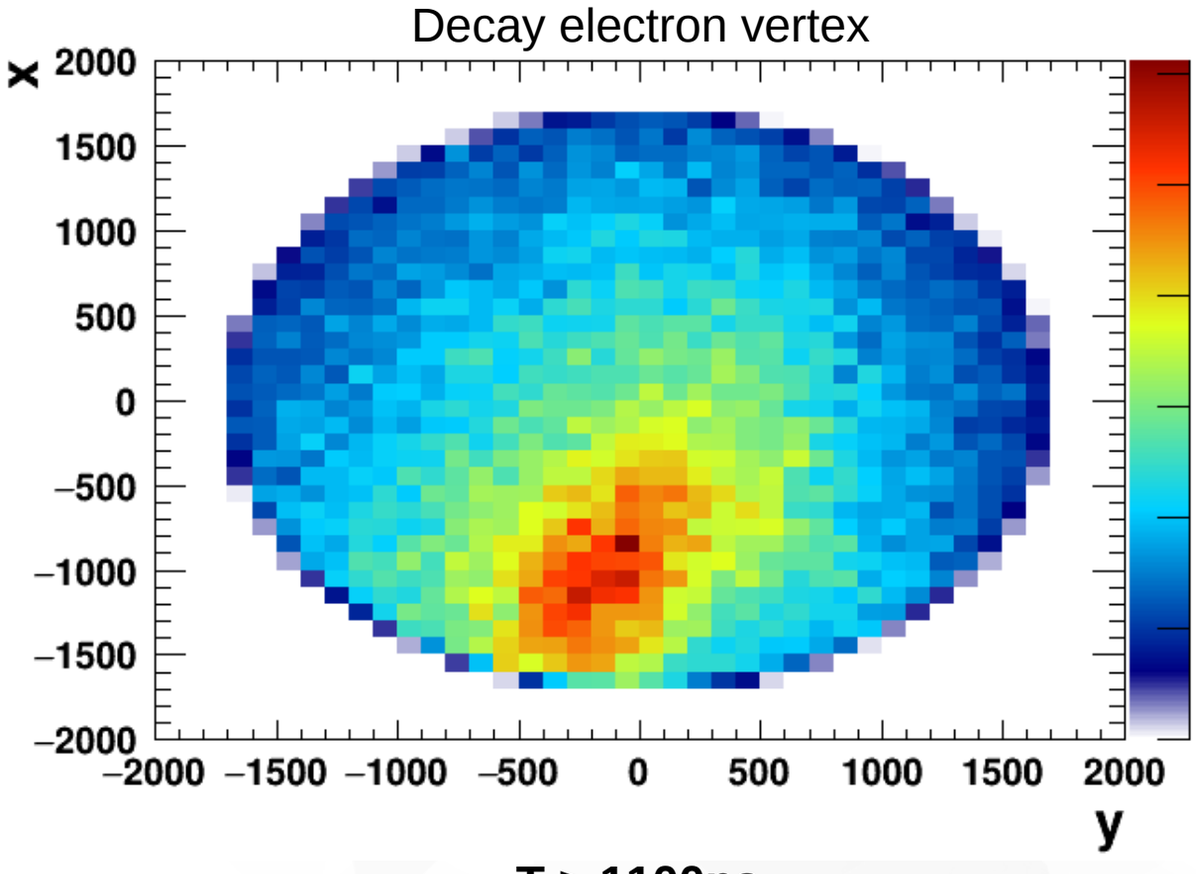




Vertex [x,y]

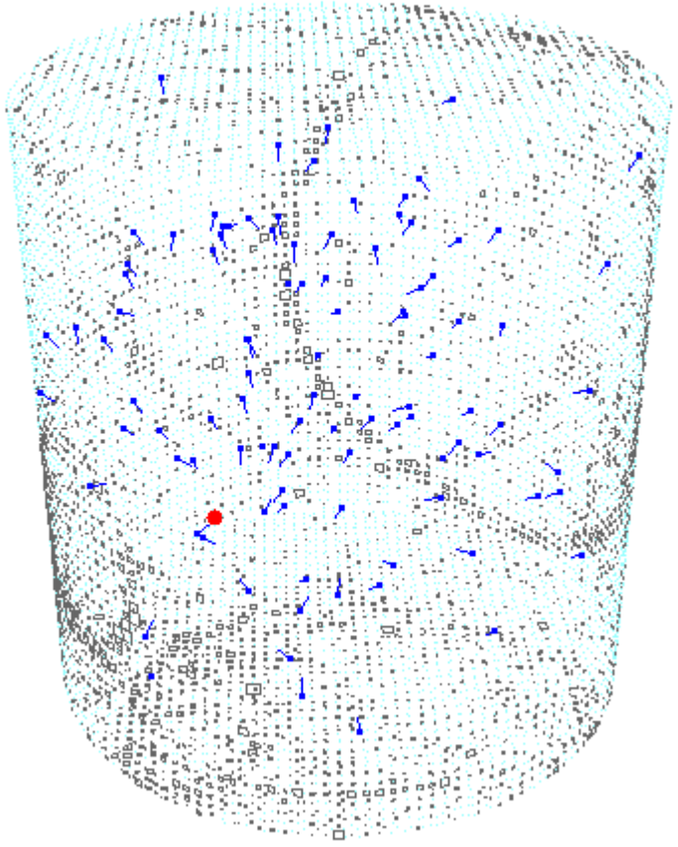


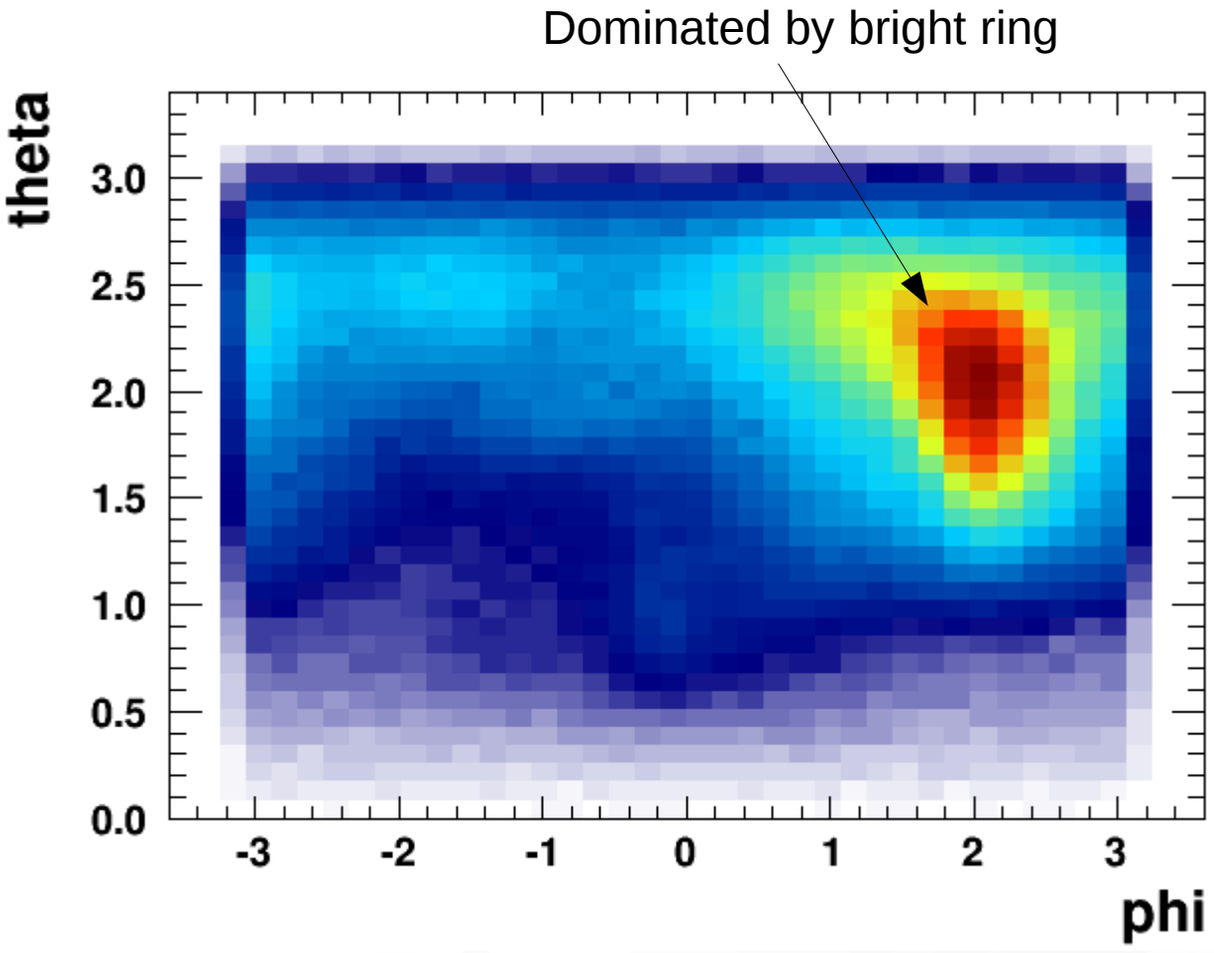
T < 1100ns



T > 1100ns

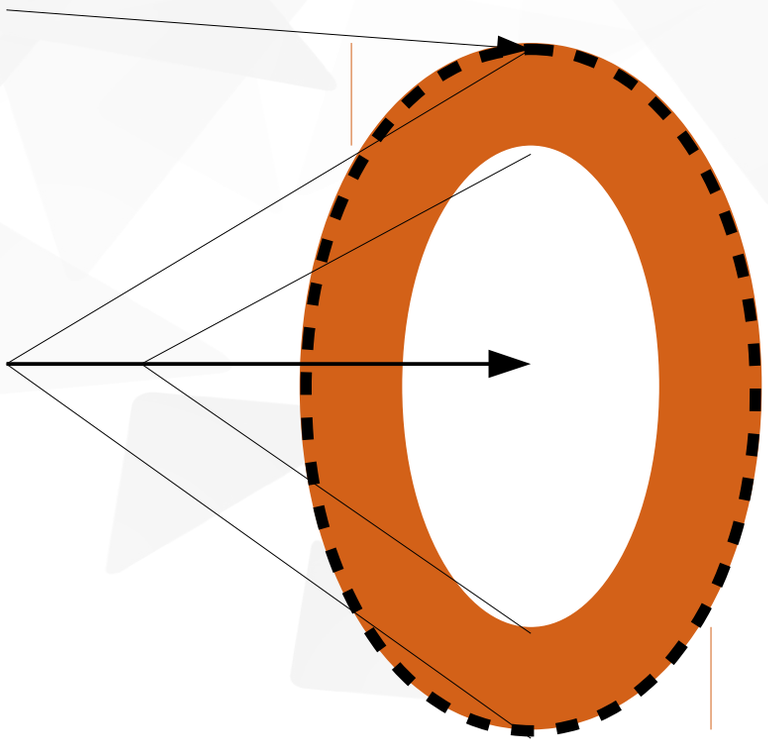
Multi-ring event





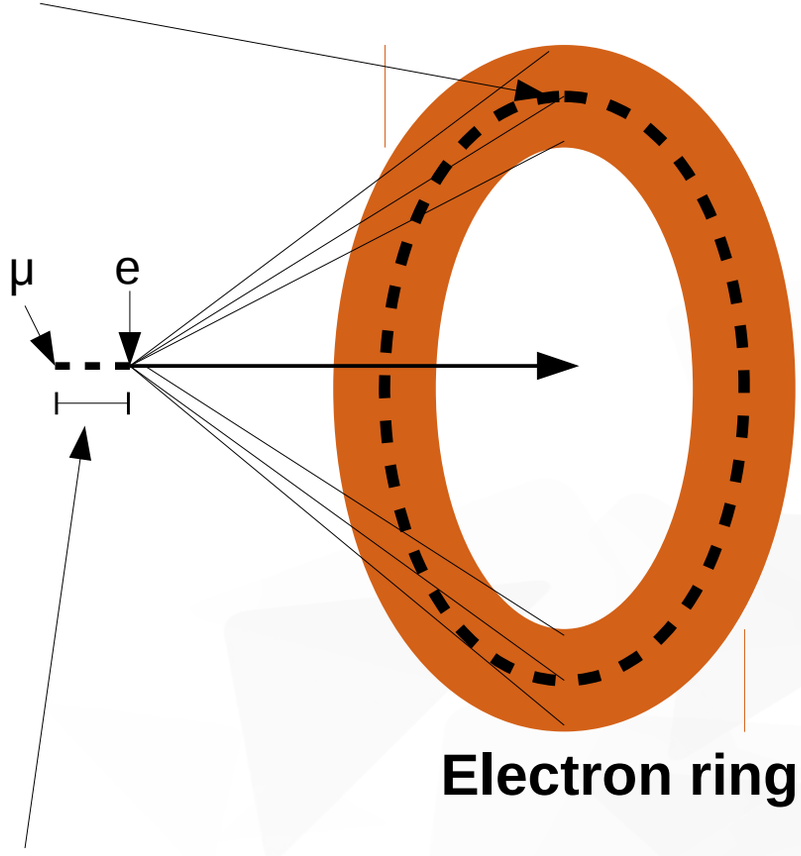
Stepping between PID states

Light on the outer edge of ring
is associated with the muon vertex



Muon ring

Light at the centre of the ring is
Associated with the electron vertex

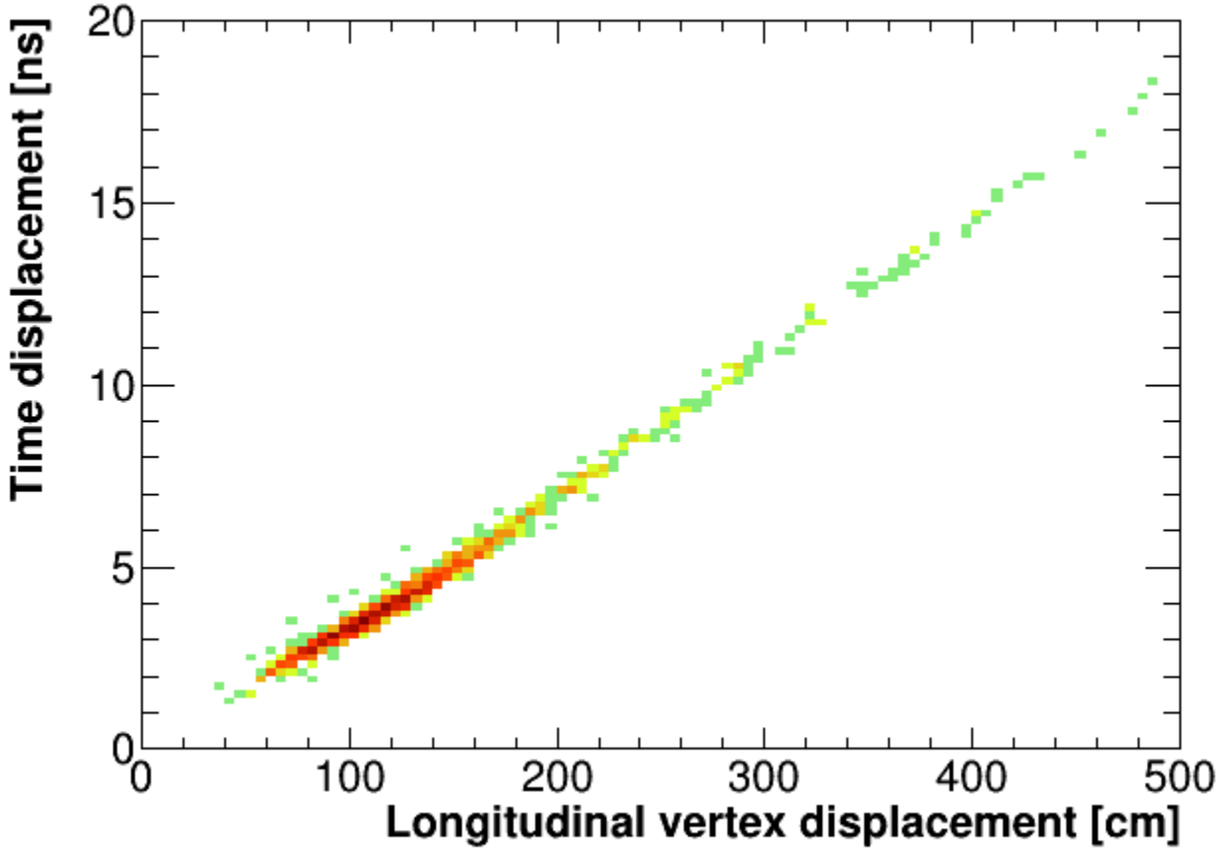


Electron ring

Muon hypothesis vertex is “upstream” from the electron hypothesis

PID vertex translation

- Need to account for the shift in hypothesis vertex when changing PID for a ring
- Fit particle gun MC with e-like and mu-like hypothesis
- Plot difference between vertex for each hypothesis
- When running MCMC, sample from this distribution to calculate the jump
- This needs to be produced for all PID combinations

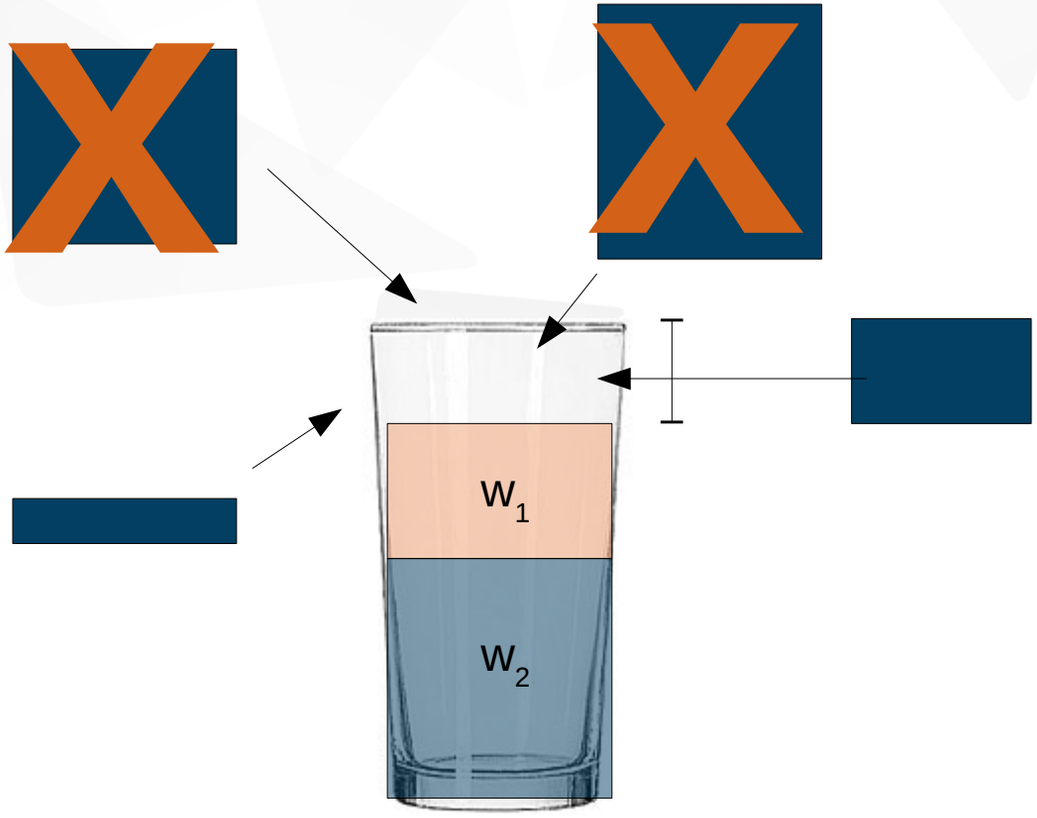


Momentum

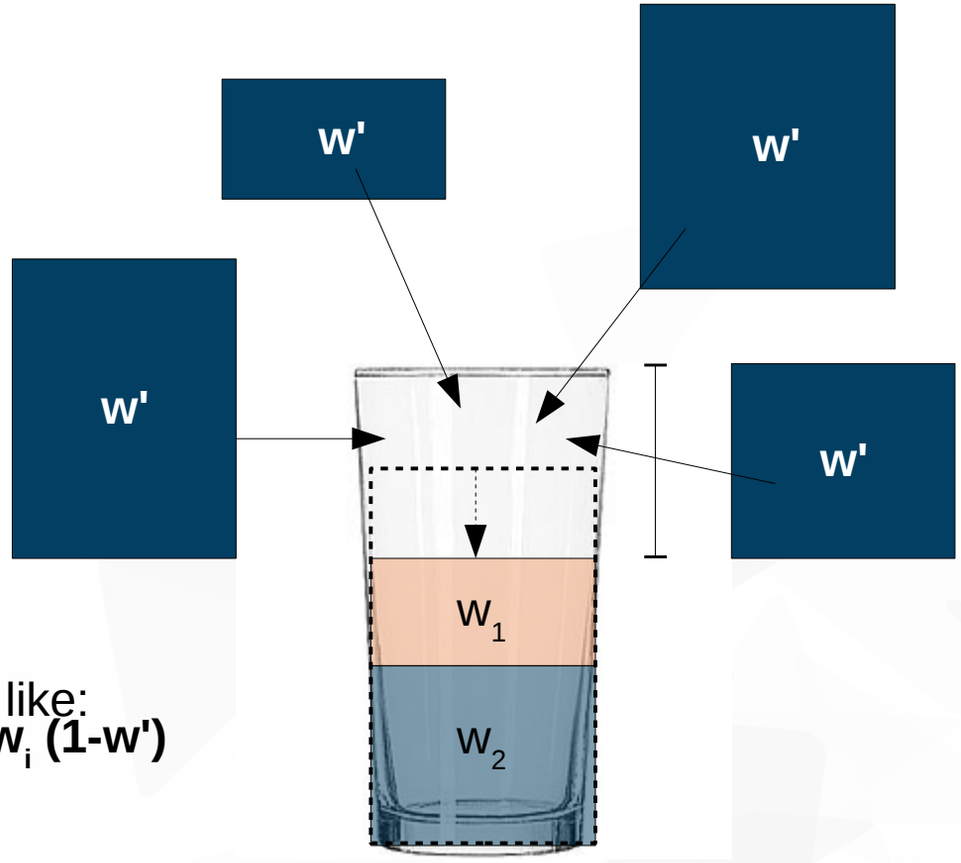
- So far the algorithm seems to be struggling with rings of different momentum
- Currently, there is no intelligent way to pick momentum of a new ring
- Algorithm is getting stuck in minima of intermediate states; need to improve mixing between rings of various momentum
- A solution seems to be to put a constraint on the total charge, and rescale when adding/removing rings
- Use a Dirichlet process as a prior on the momenta of the rings

Imagine if the deposited charge from a ring filled a glass...

Partition the charge inside a PMT to the current ring hypotheses. As more ring hypotheses are added, the remaining unclaimed charge can only be claimed by a new ring hypothesis that doesn't exceed the remaining charge. If instead, the total “weights” of all the current + proposed rings must total 1.0, then this allows more mobility in adding and removing intermediate rings.



No rescaling



Scale like:
$\mathbf{w_i = w_i\,(1\text{-}w')}$

Rescale so all weights sum to 1

Dirichlet process

- The Dirichlet process is a conjugate prior to the multinomial distribution
- i.e. Dirichlet prior X multinomial likelihood = Dirichlet posterior
- Dirichlet is a prior over discrete outcomes
  - e.g. rolling a dice has 6 outcomes
  - An unbiased dice has 1/6 chance of landing on a particular number
  - Dir(1/6, 1/6, 1/6, 1/6, 1/6, 1/6)
  - Draws from this Dirichlet distribution will produce 6 probability distributions for each side of the dice
- This can be used as an uninformative prior on N ring momentas
- **Disclaimer #2:** I don't fully understand this application yet, however a lot of literature on N-mixture model fitting talks about the necessity of using a Dirichlet prior in this way

Conclusions

- The MCMC-based fitter development has stagnated recently
- Plan to rewrite the current clunky code; now that I understand how to do the dimension matching properly, I can make a better framework to support development
- Plan to implement these features:
  - Rescaling of ring momentum when adding/subtracting new rings to improve fluidity in moving between ring hypotheses
  - Study/add Dirichlet prior
  - New Bonsai-based ring proposal distribution
- Hope to dedicate more time to developing this fitter soon!