Bayesian Non-Parametric Event Reconstruction in Super-Kamiokande

- Phystat-ν 31st June 2016

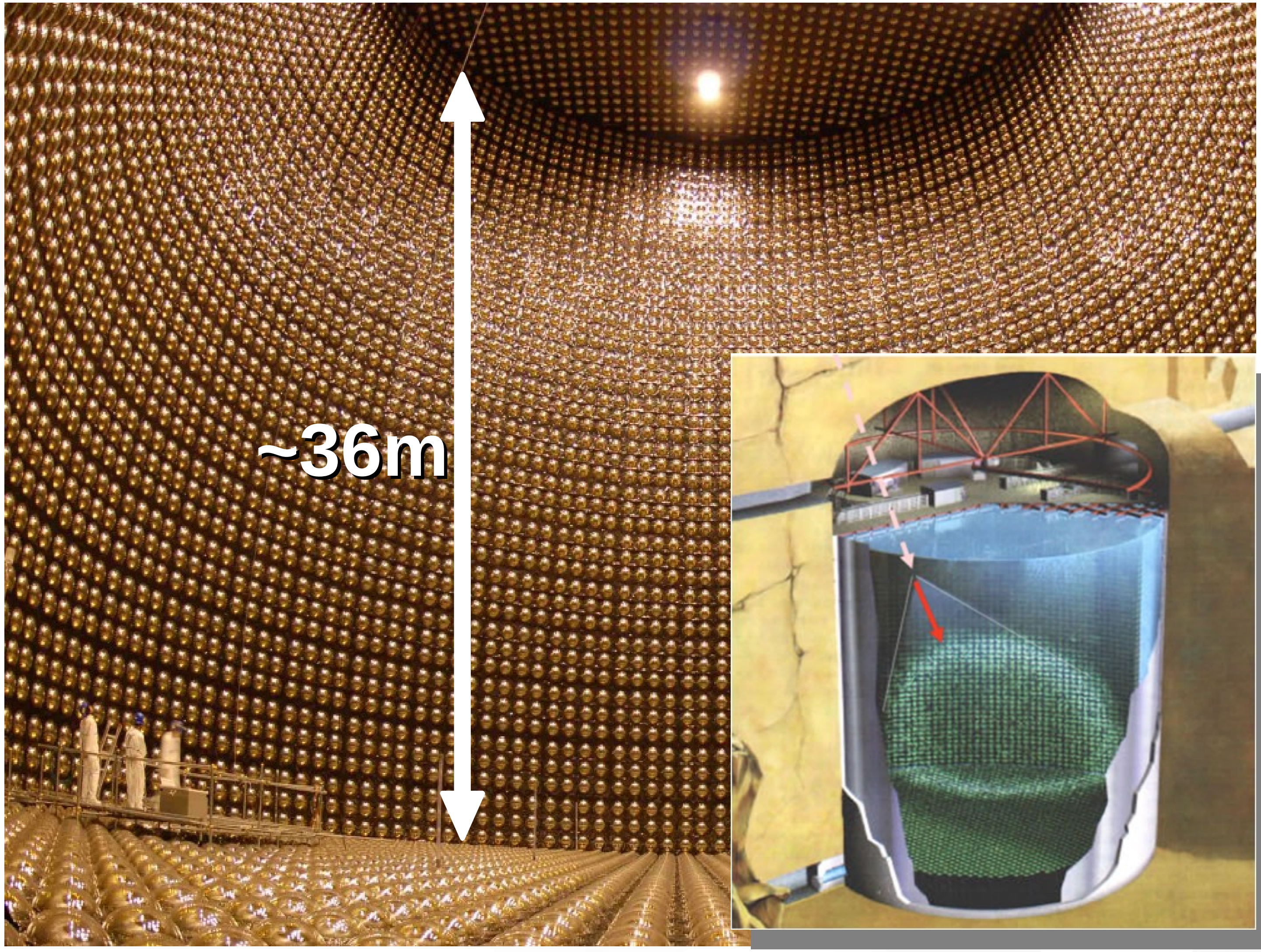
- **Richard Calland** – Kavli IPMU (UTokyo)

richard.calland@ipmu.jp

Introduction

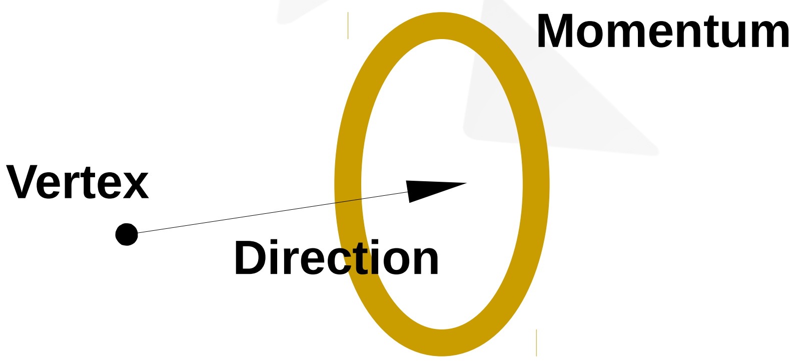
- **Super-Kamiokande** (SK) is the worlds largest **water Cherenkov** detector
- It detects **neutrinos** from space, the sun, and man-made sources
- Charged particles produce **cones** of light inside the detector via Cherenkov radiation

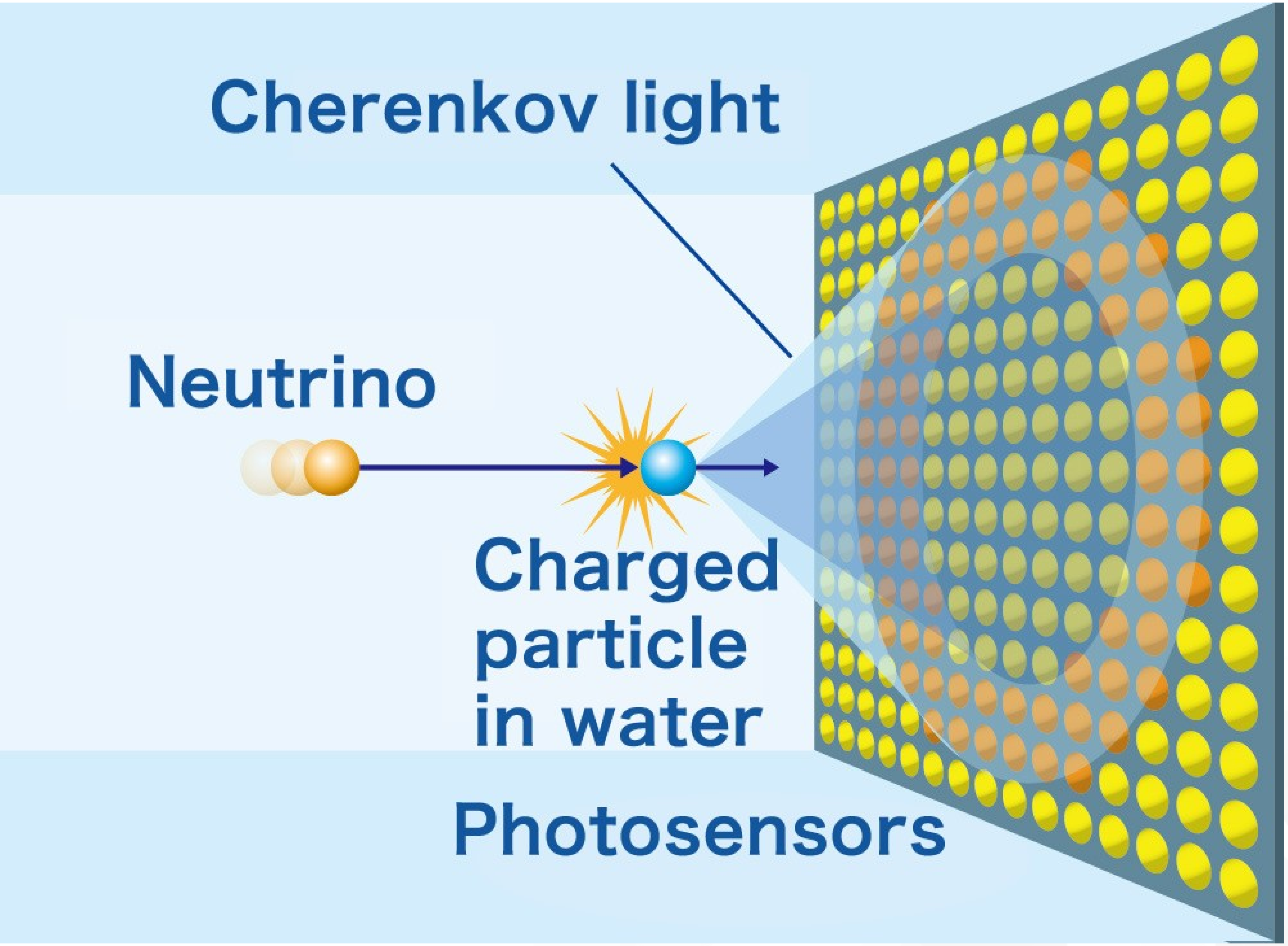
Photomultiplier Tube



Event Reconstruction

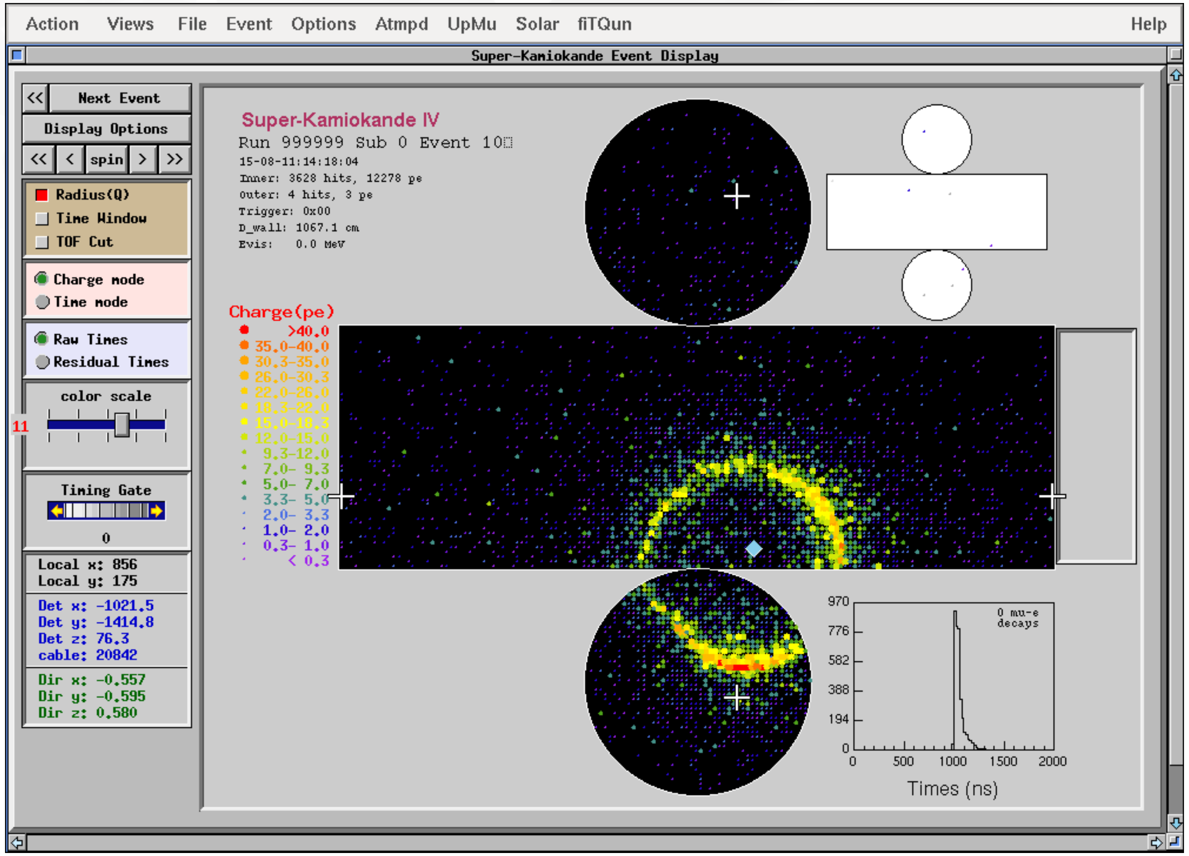
- In order to study neutrinos, we need to reconstruct their variables from data
- The photosensors collect **charge** and **time** information
- We want to turn this raw data into **parameters describing the particle**



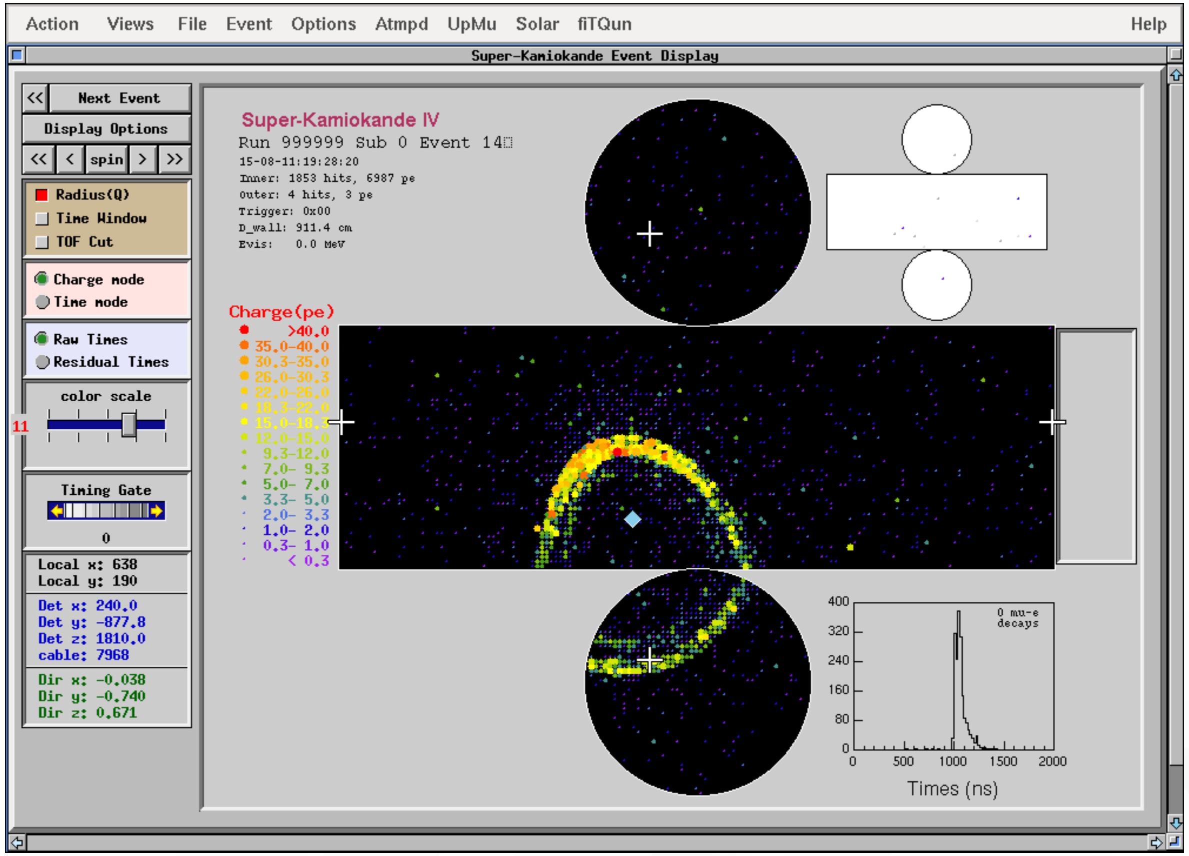


Particle Identification

electron

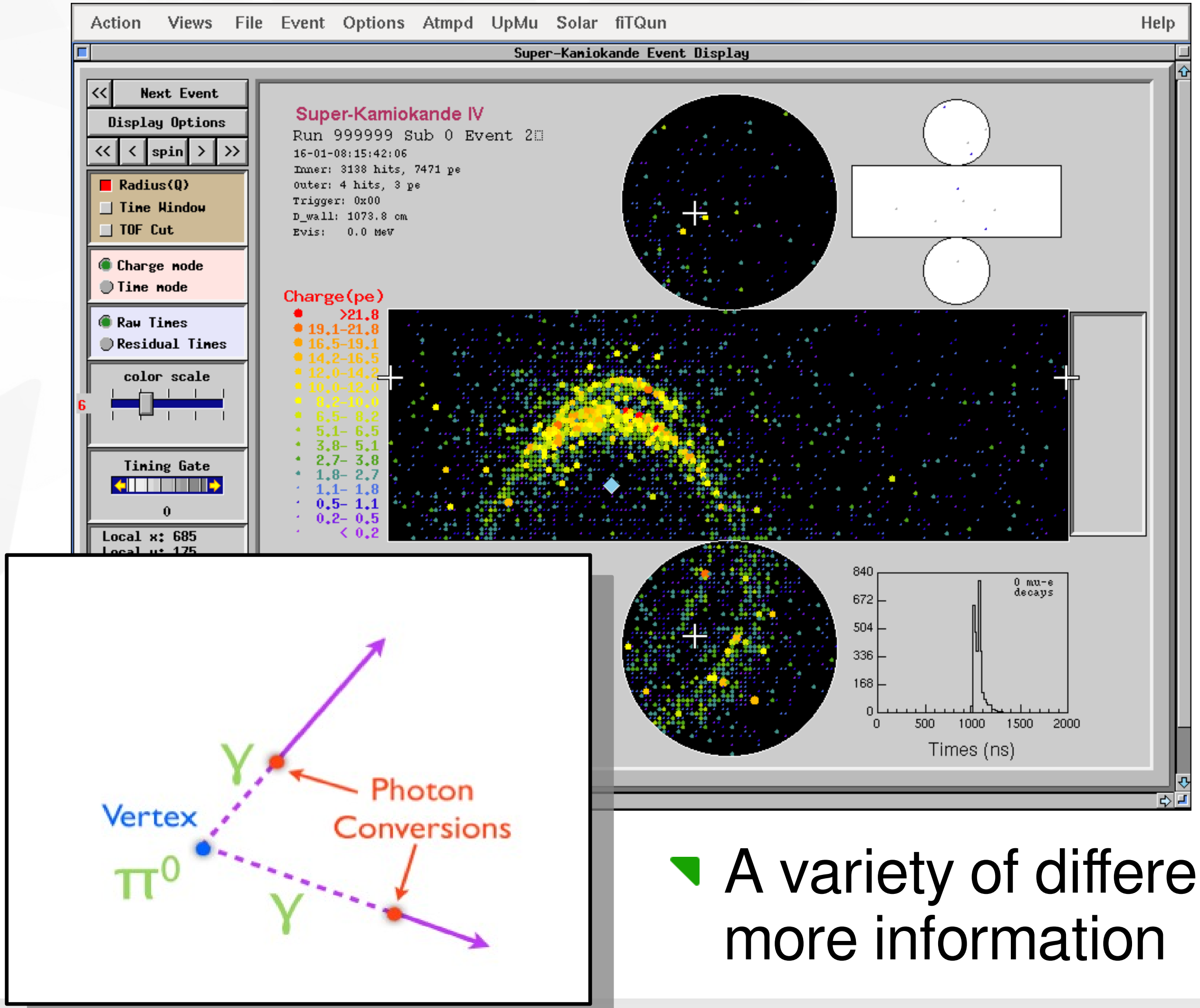


muon

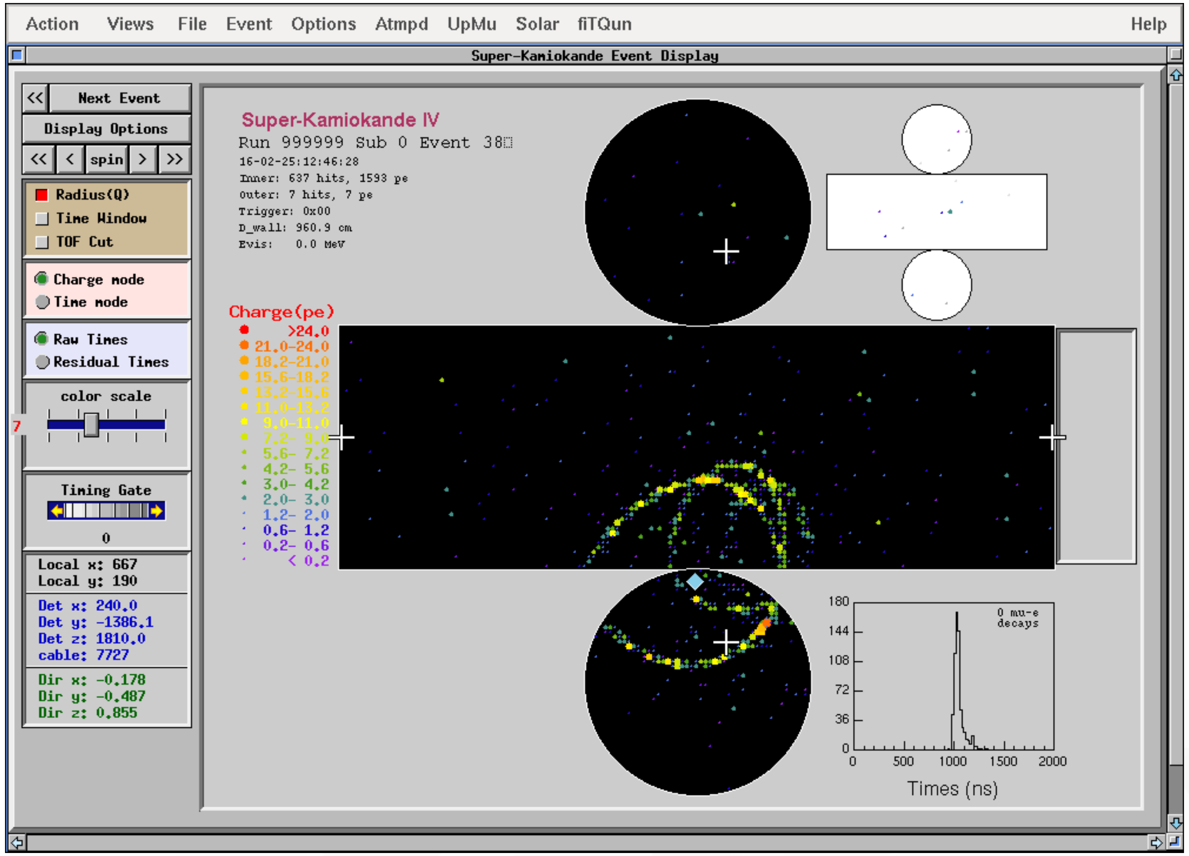


- Different charged particles produce different patterns of light
- We can use this difference to infer the type of particle

pi0

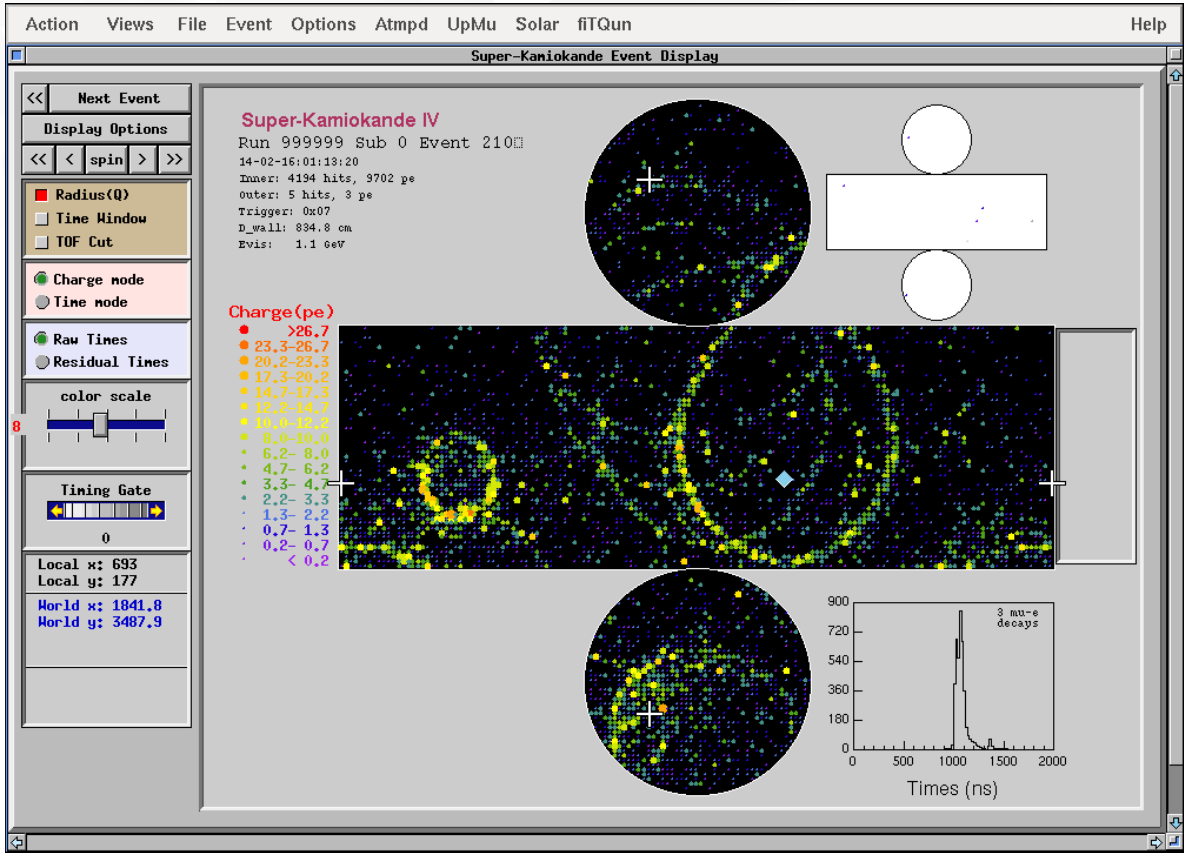


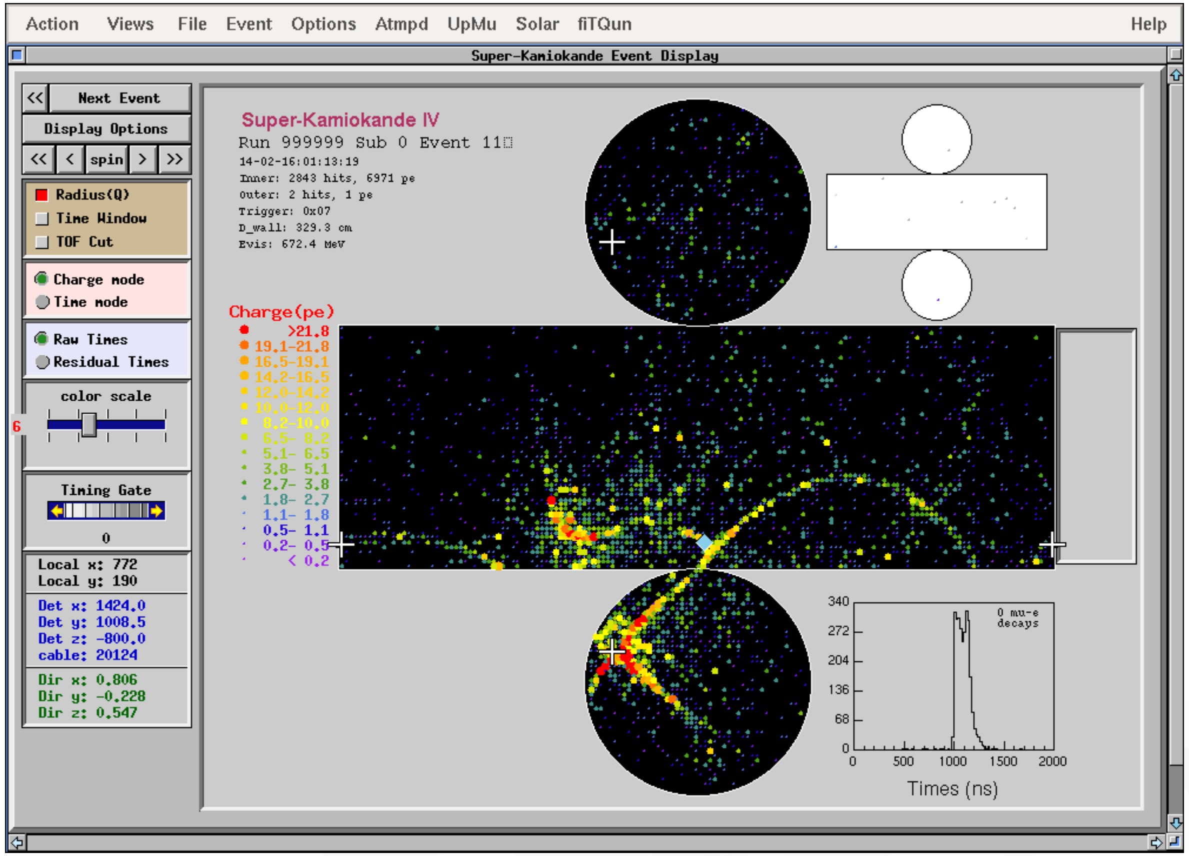
proton



- A variety of different physics processes can happen that give us more information

Multi-Ring Event Examples





- Sometimes too much information...

Joint Likelihood

- SK collects **charge** and **time** information for every PMT
- Can construct a **joint likelihood** distribution:

data

$$L(\mathbf{x}) = \prod_{j}^{\text{unhit}} P_j(\text{unhit}|\mathbf{x}) \prod_{i}^{\text{hit}} \{1 - P_i(\text{unhit}|\mathbf{x})\} f_q(q_i|\mathbf{x}) f_t(t_i|\mathbf{x})$$

Probability that PMT was <u>not</u> hit by photon

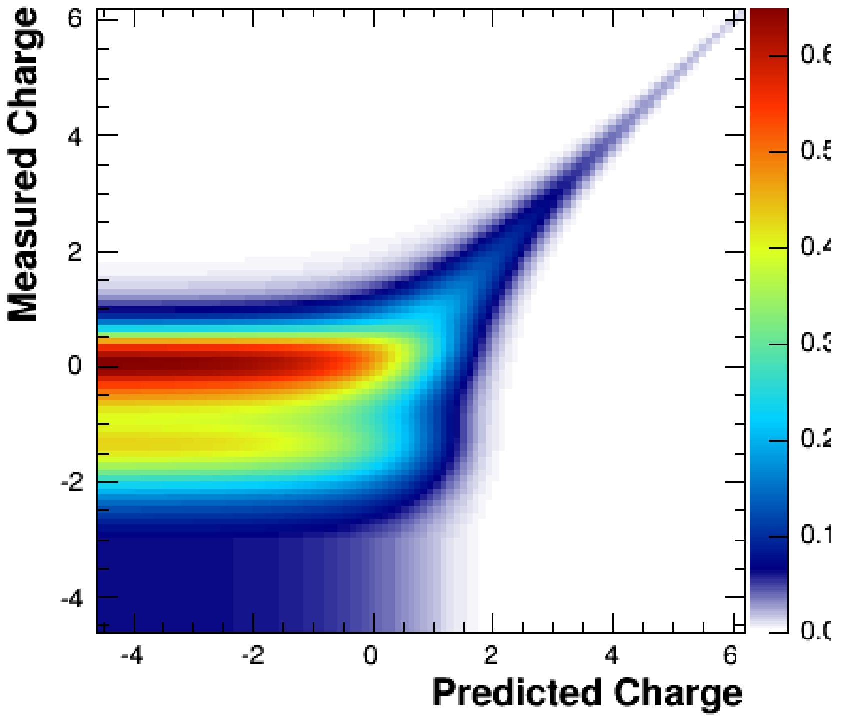
Probability that PMT was hit

Charge & time PDFs

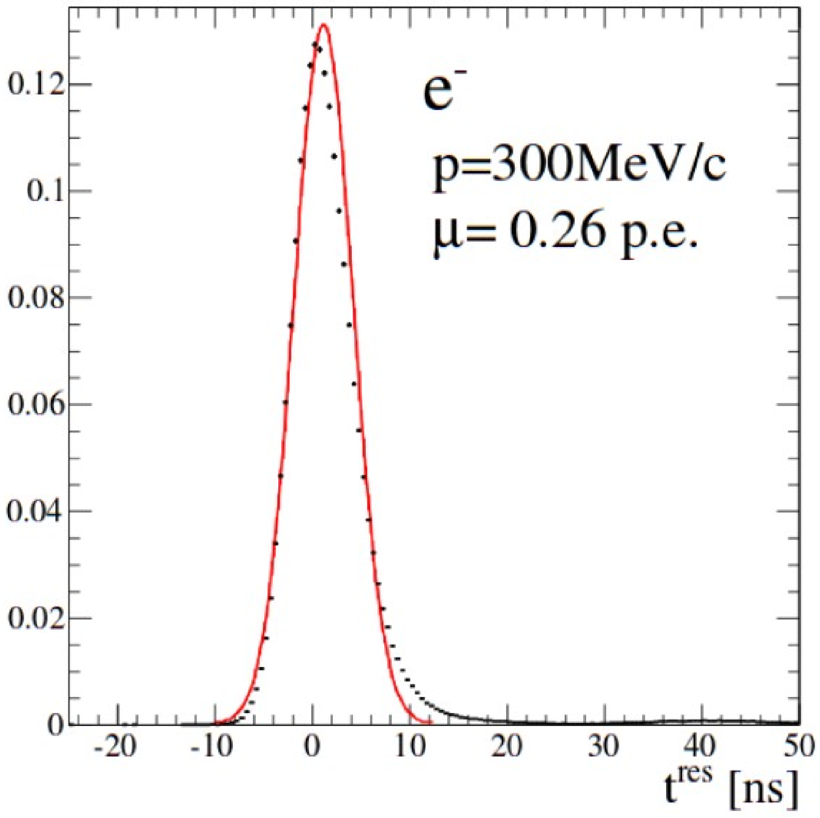
We build the PDFs empirically from Monte Carlo simulations

x is the ring parameterization:
- Vertex [x,y,z,t]
- Direction [θ,ϕ]
- Momentum [p]

Charge PDF

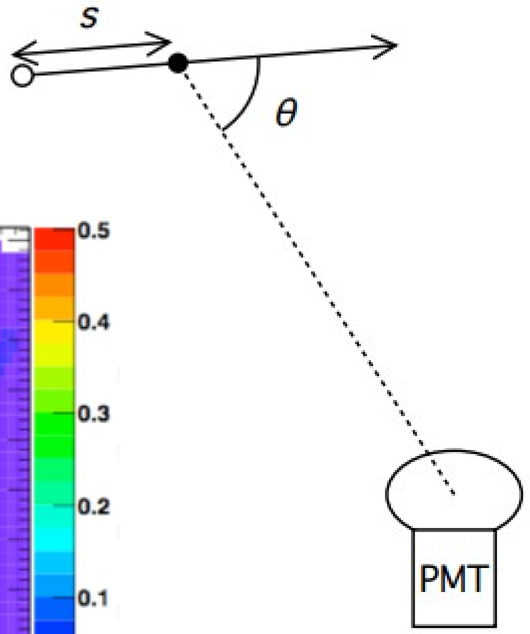


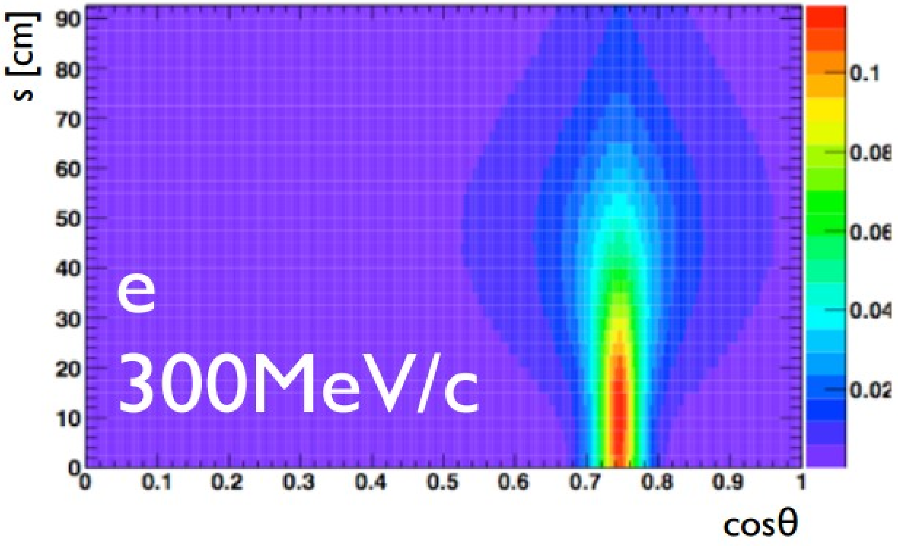
Time PDF

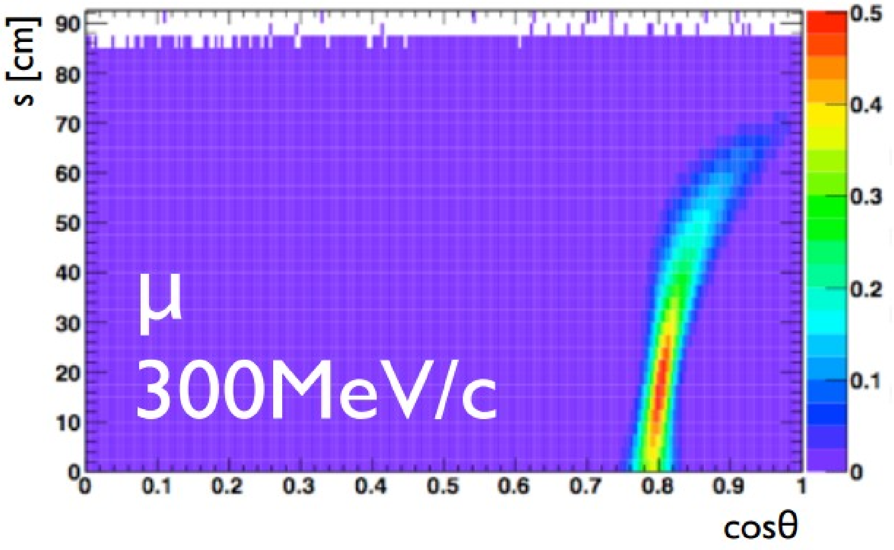


PDF Construction

- Charge and time are predicted given the parameter hypothesis
- Parameterize particle-PMT response (vertex, PID etc)
- Encode MC simulations into lookup tables according to this
- Emission profile of cherenkov light differs for each PID

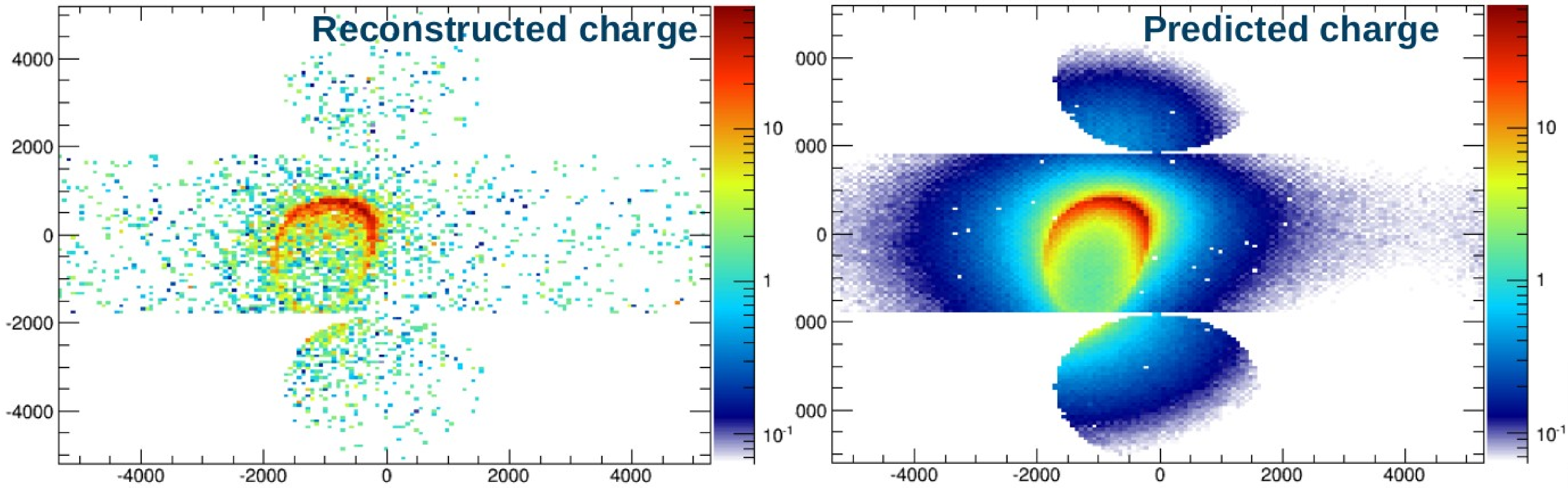






Data vs prediction

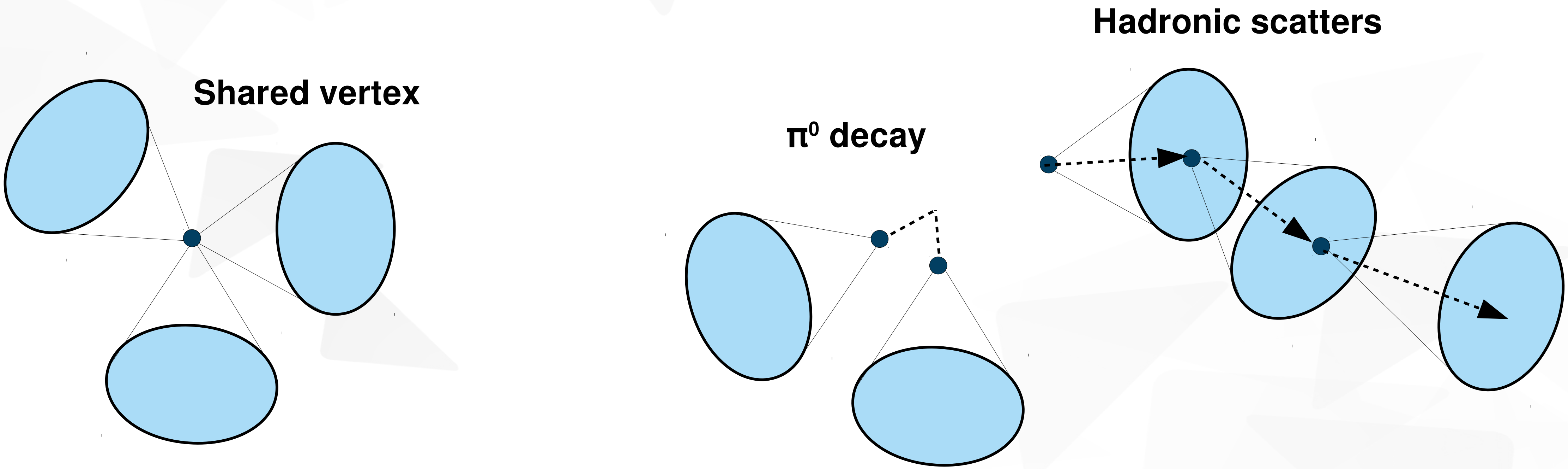
- Comparison of charge prediction at true parameter values



- Now we have a likelihood function that models a cherenkov ring
- Can optimize this function to reconstruct the event
- But what about events with more than one ring?
- How do you solve a problem where ***"the number of things you don't know is one of the things you don't know"***

- You could perform a piecewise optimization, running a series of fits adding a new ring each time, and compare results with some statistic to select your favourite
- With optimization routines, this approach doesn’t scale well
- Prone to getting trapped in local minima
- Likelihood ratios are often misleading; AIC? BIC?

- One semi-solution is to assume all rings share a vertex, to reduce the dimensionality
- However, multi-ring events do not necessarily have the same vertex



Bayesian Model Comparison

- The Posterior probability derived from Bayes Theorem is normalized to 1
- To compare different models, can look at the normalization term of each
- The **evidence** or **marginal likelihood** involves the integration of the likelihood over the *range* of the prior
  - Posterior is integration of the *product* of likelihood and prior ← ***MCMC only does this***
- However, this is often computationally intractable

$$P(H|D) = \frac{P(D|H)\,P(H)}{P(D)}$$

$$\text{Posterior} = \frac{\text{likelihood} * \text{prior}}{\text{evidence}}$$

Enter Bayesian non-parametrics

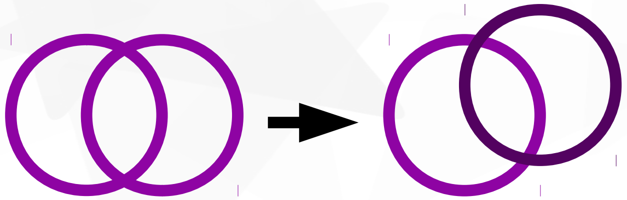
- Model the data with a **mixture model:** $p(x) = \sum w_i \, f(x|\Theta)$
  - Each ring is a component of the mixture (Dirichlet Prior)
  - No single-vertex assumption
- Use a **Reversible Jump Markov chain Monte Carlo** to sample the Posterior probability of the model*
  - Add/remove rings as discrete state jumps in the MCMC, to shift between models
- This performs not only **parameter inference**, but also **model selection**
  - The Bayesian **evidence** ratio is preserved in the MCMC chain
- Number of rings is now a parameter that we can measure
- *"Non-parametric"* means number of parameters is not fixed

****Reversible jump Markov chain Monte Carlo computation and Bayesian model determination*** **[Green 1995]**

RJMCMC move types

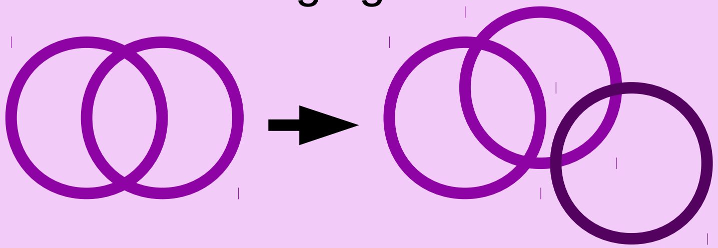
Dimension preserving

This is the typical MCMC algorithm

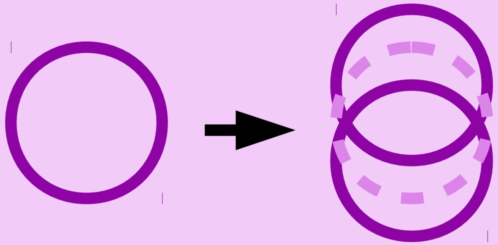


Alter a ring (or N rings)

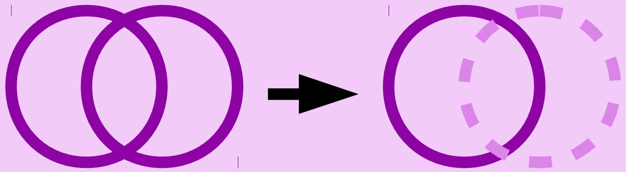
Dimension changing



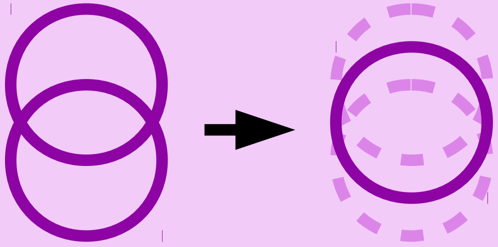
Add ring



Split ring



Remove ring

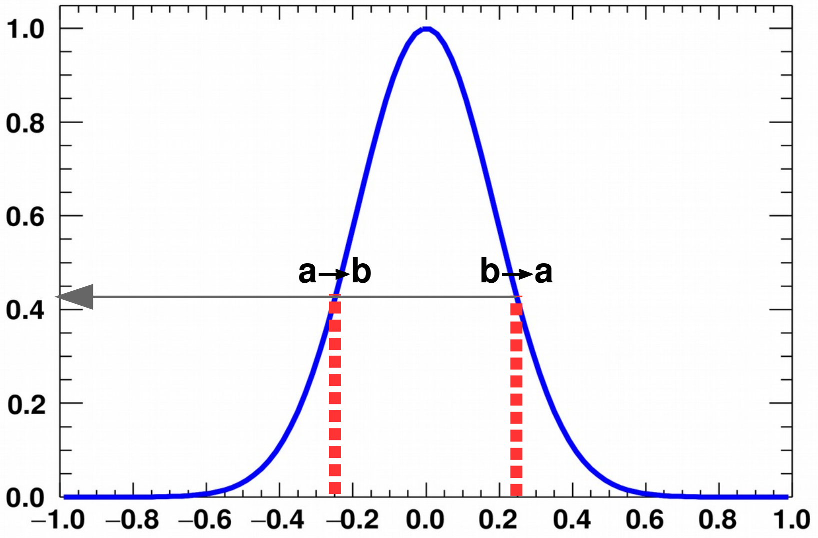


Merge rings

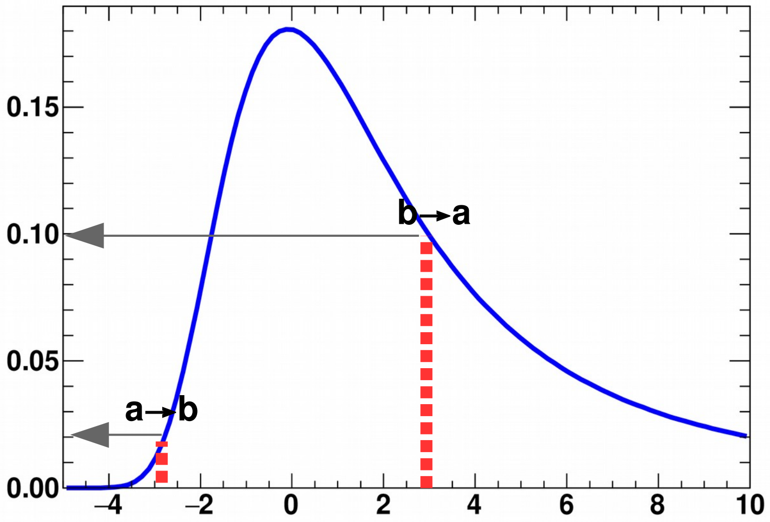
Acceptance Probability

Metropolis-Hastings:

$$a = \min\left\{1, \frac{\text{Posterior}_{\text{new}} \quad \text{Proposal}_{\text{new}}}{\text{Posterior}_{\text{old}} \quad \text{Proposal}_{\text{old}}}\right\}$$



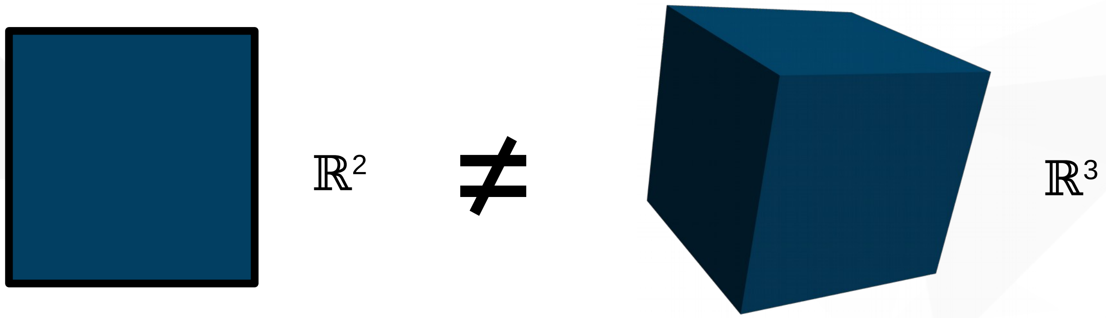
Symmetric Proposal function



Non-Symmetric Proposal function

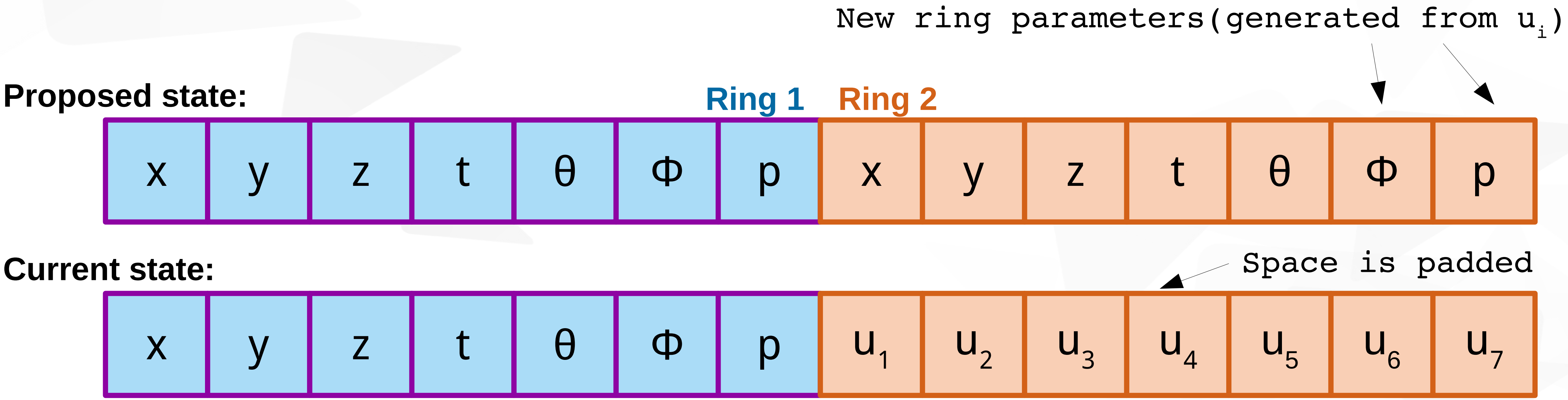
Dimensional Disaster

- But if we are comparing two states of different dimension, how can we ensure to maintain detailed balance?
- The dimensions of the proposal functions are different, and therefore the densities will not be normalized relative to each other
- Chain will bias towards a smaller number of rings



Dimension Matching

- Additional terms to enforce detailed balance with reversible jump moves
- Where $u_i \sim g()$ are the random variables used to transform from one state to the other
- *Dimension matching* with *auxiliary variables* is crucial to ensuring detailed balance of the RJMCMC



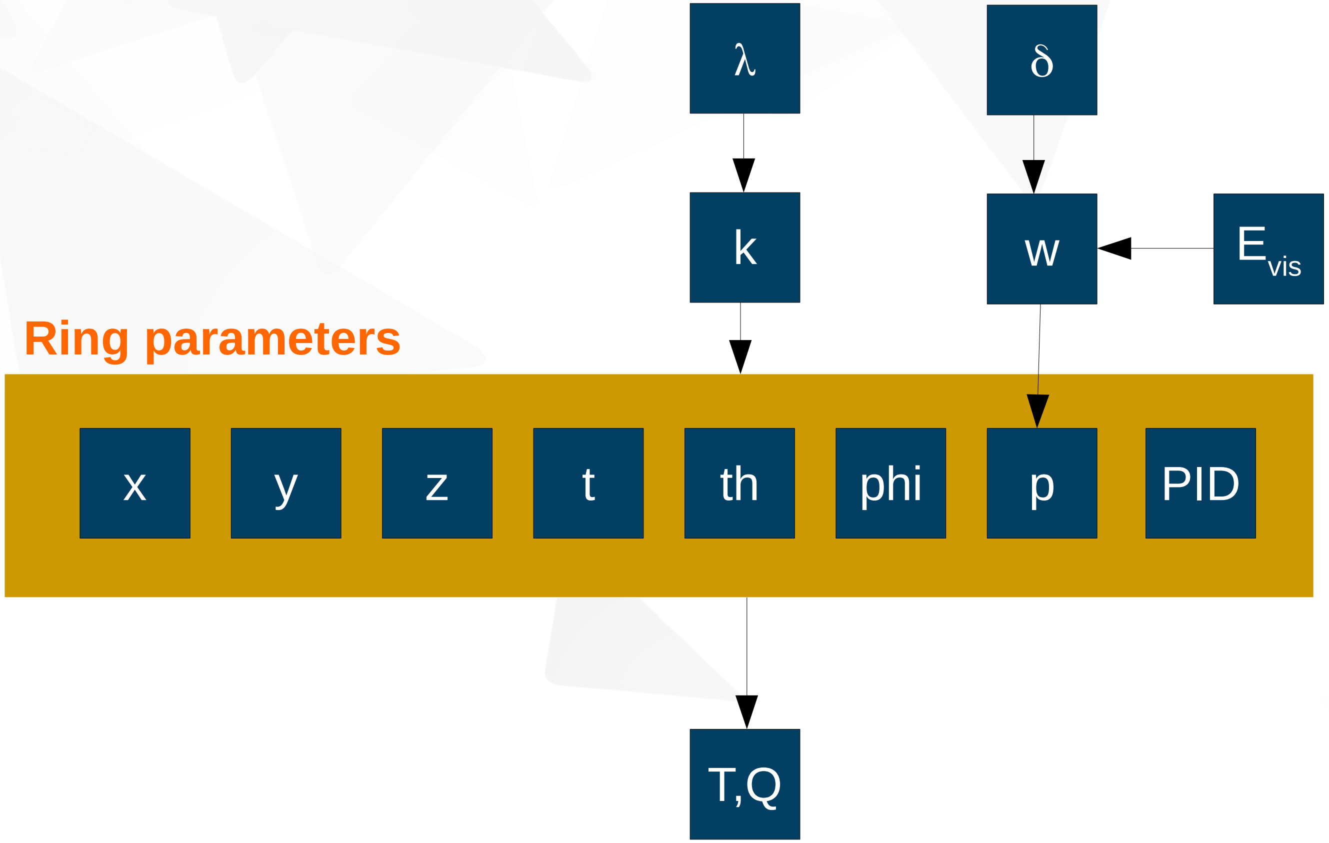
Acceptance Probability for RJ Move

Reversible Jump:

$$a = \min\left\{1,\ \frac{\text{Posterior}_{\text{new}} \quad \text{Proposal}_{\text{new}}}{\text{Posterior}_{\text{old}} \quad \text{Proposal}_{\text{old}}}\ g(u_i)^{-1} \left| \frac{\partial\theta_{\text{new}}}{\partial\theta_{\text{old}}, u_i} \right| \right\}$$

- New term added to maintain detailed balance across dimension-changing moves
- Generate **random variables** to construct the new state, then use these **densities** to pad the dimension of the ratio
- Jacobian determinant term to account for variable transformations
  - Jump moves can be designed such that this term is 1

Hierarchical Model Definition

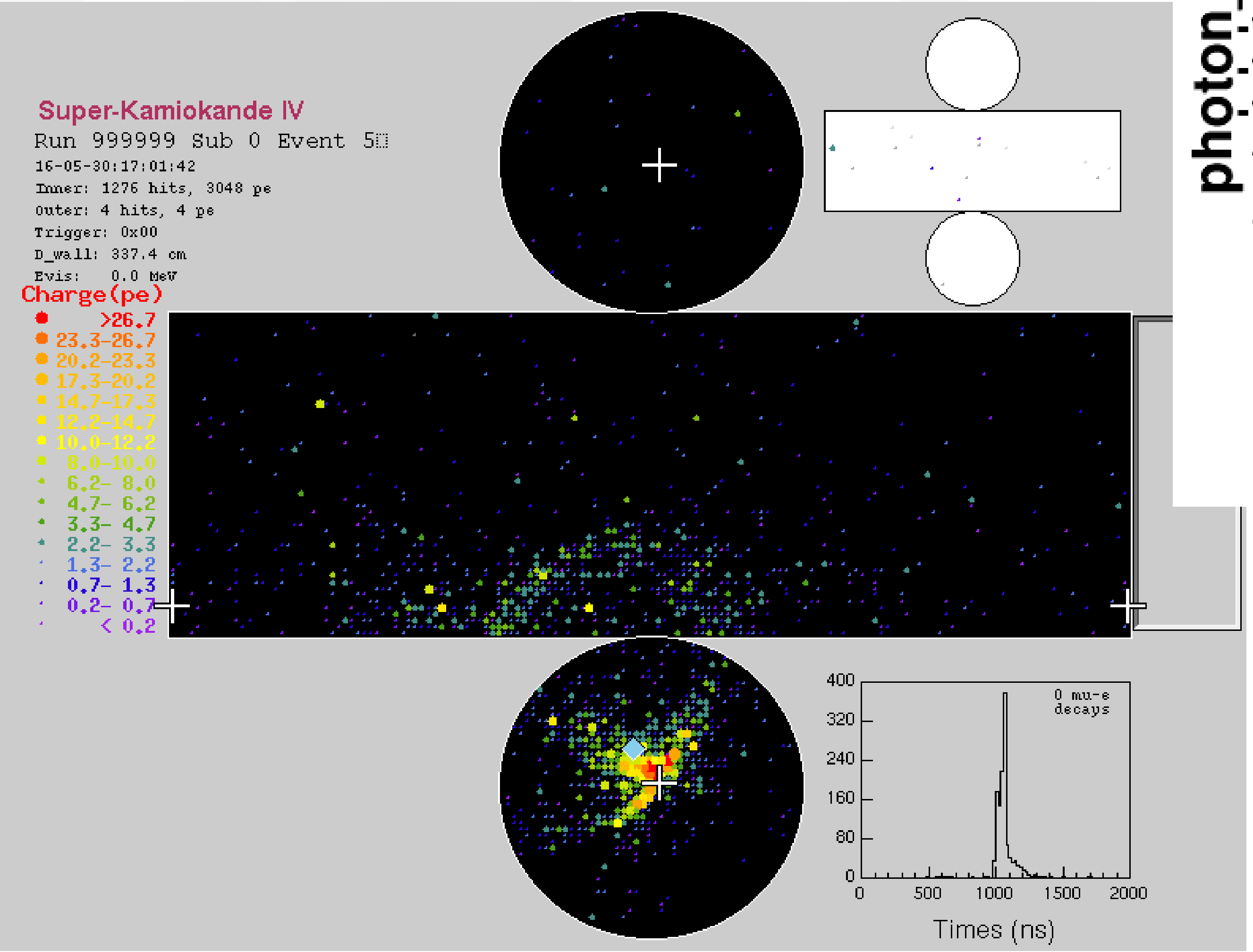


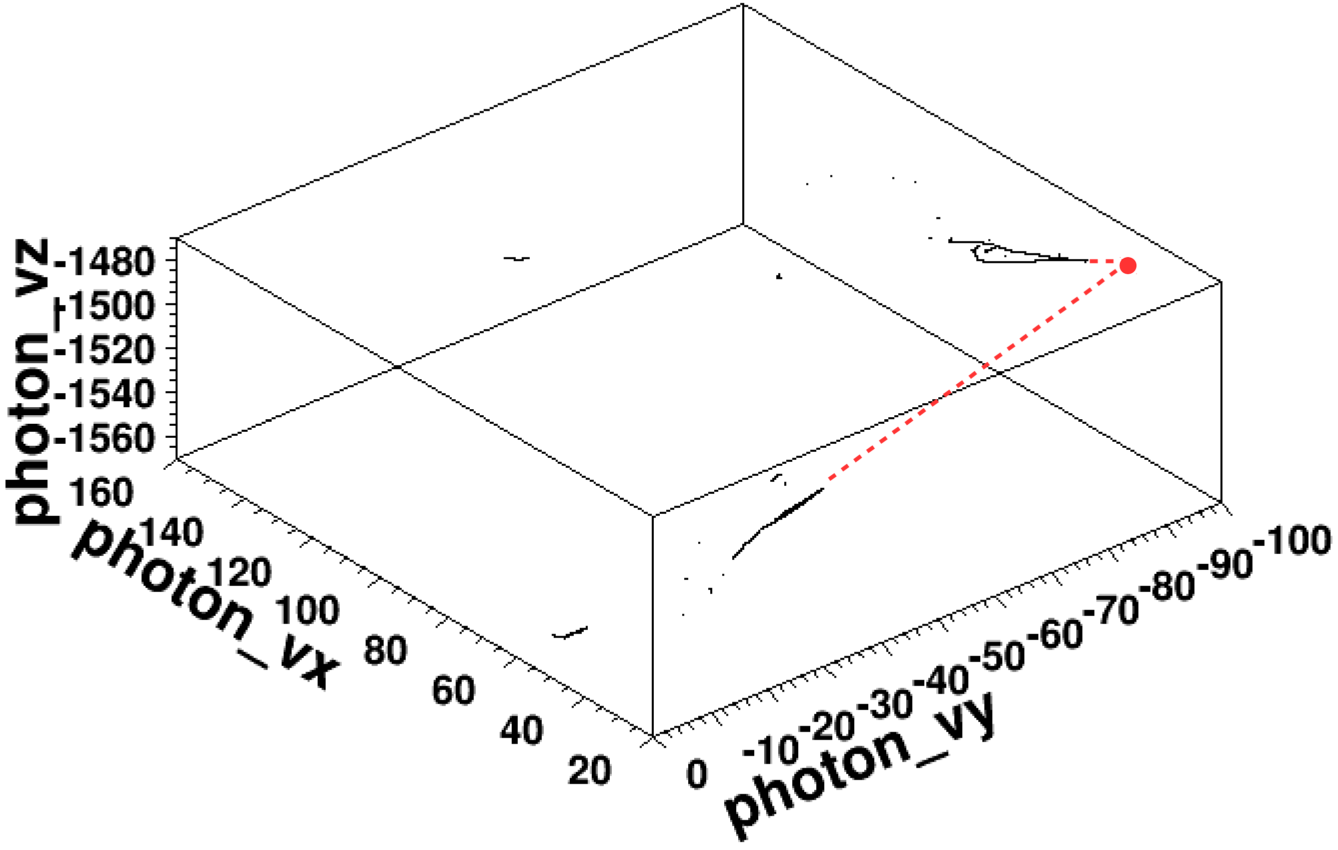
- **k** components: U() or Poisson(λ)
- **w** weights: Dirichlet(δ)
- All others have uninformative priors

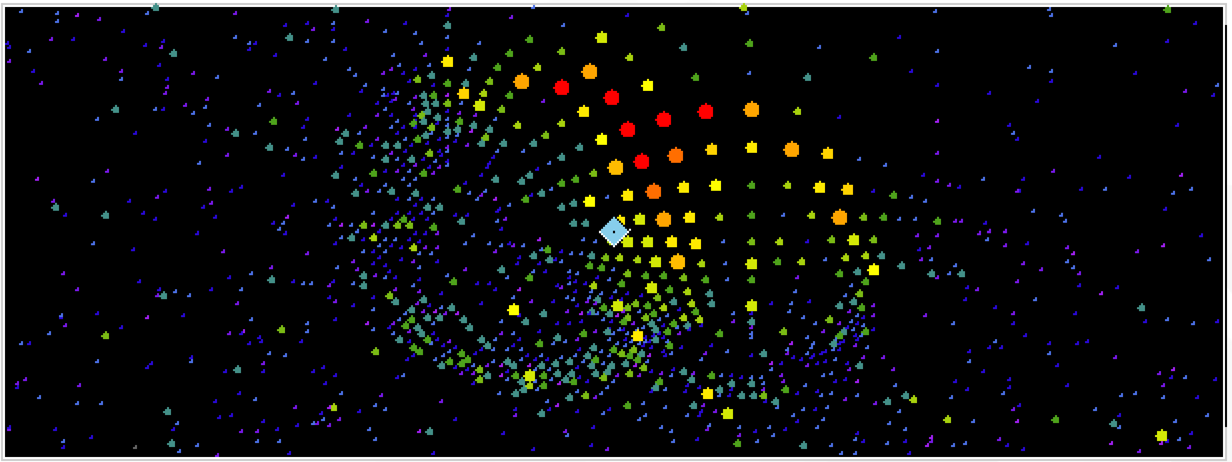
***Dirichlet process (backup) allows the momenta for all rings to be redistributed in one move, whilst preserving the total visible energy.**
i.e. respects correlations between ring momenta

Example Events

π^0 MC Event



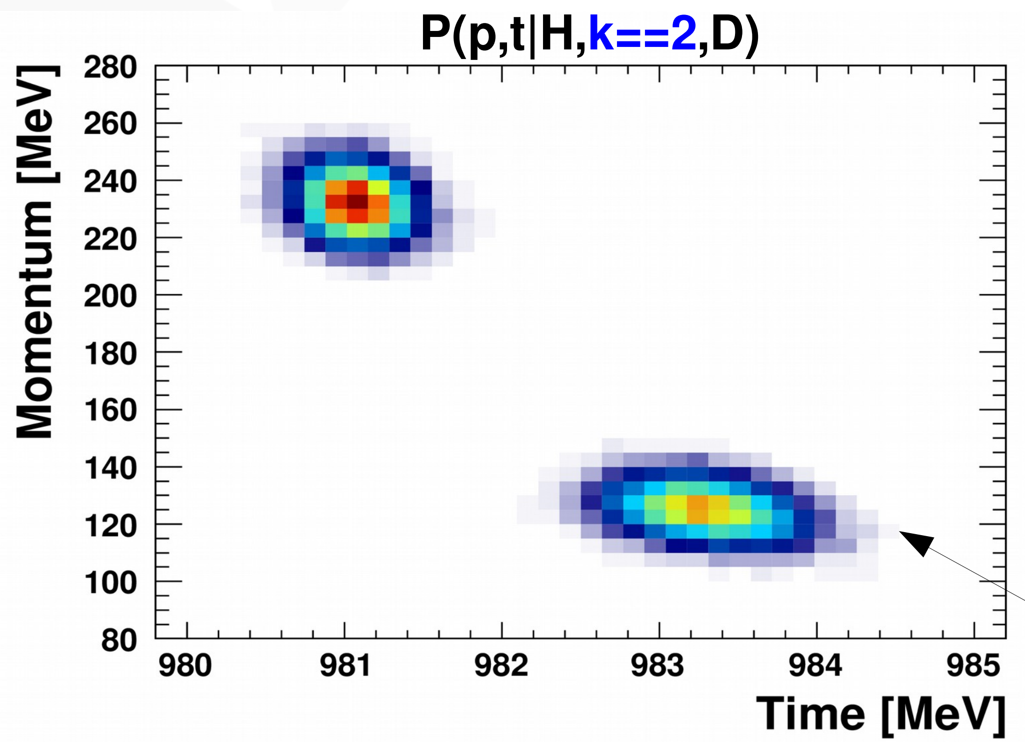


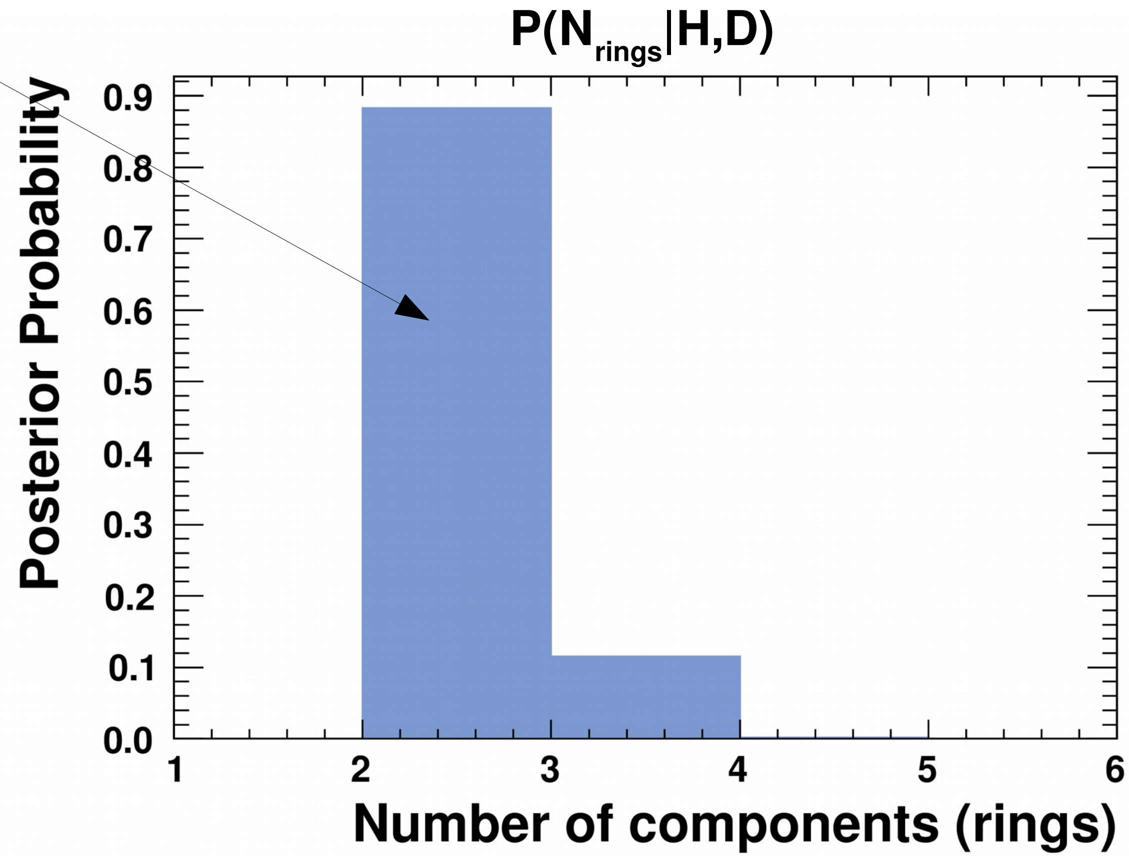


*theta-phi view

Processing of the chain

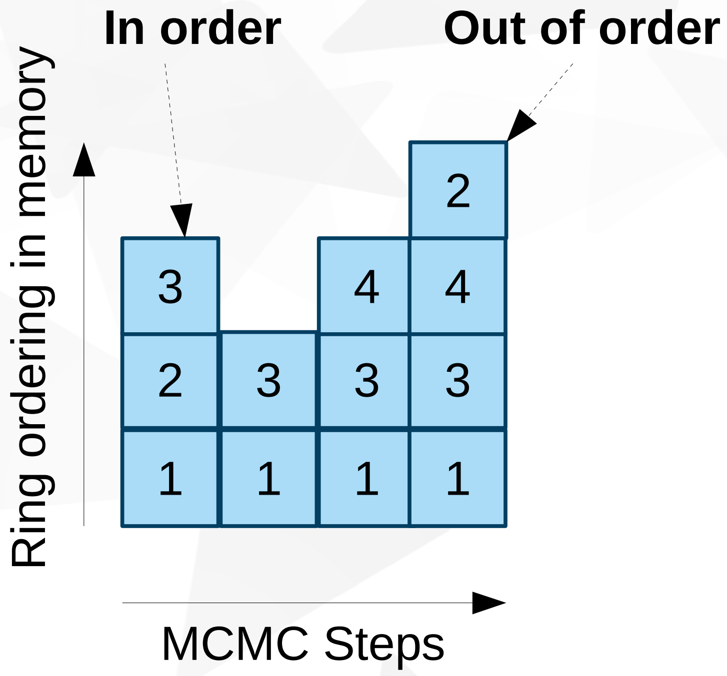
- Condition on samples that have **2 components**
- Group samples into rings
- Perform usual MCMC analysis methods

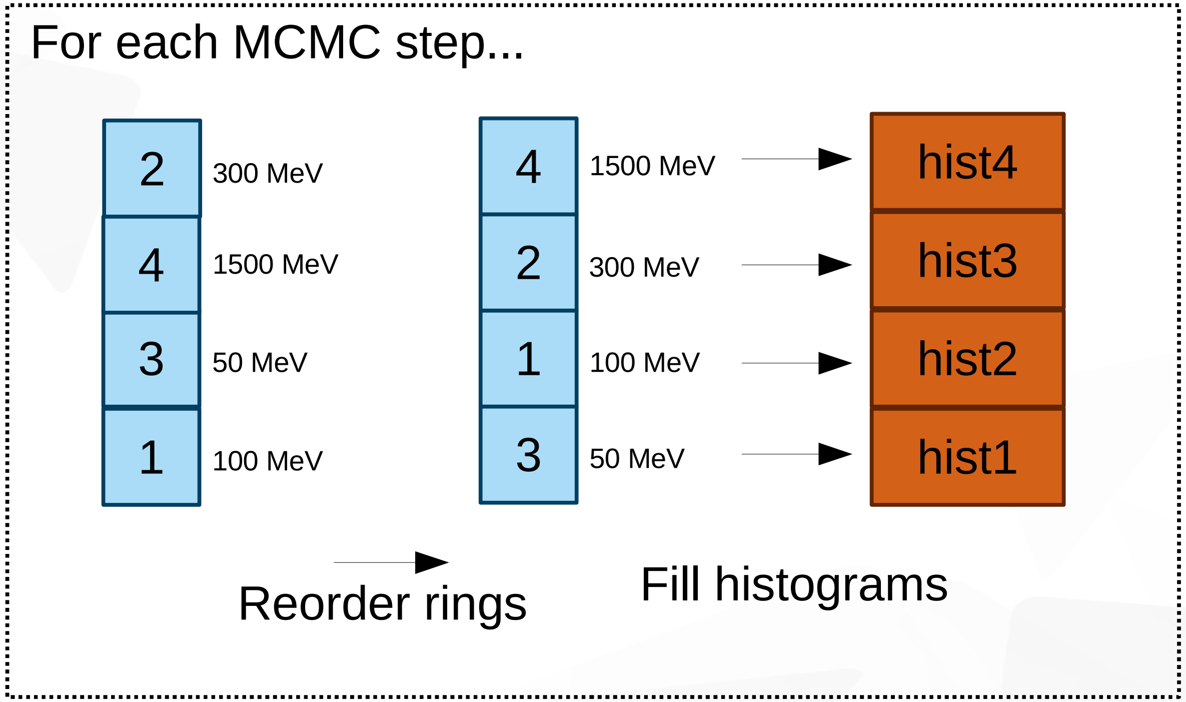




Lower momentum photon has larger time uncertainty

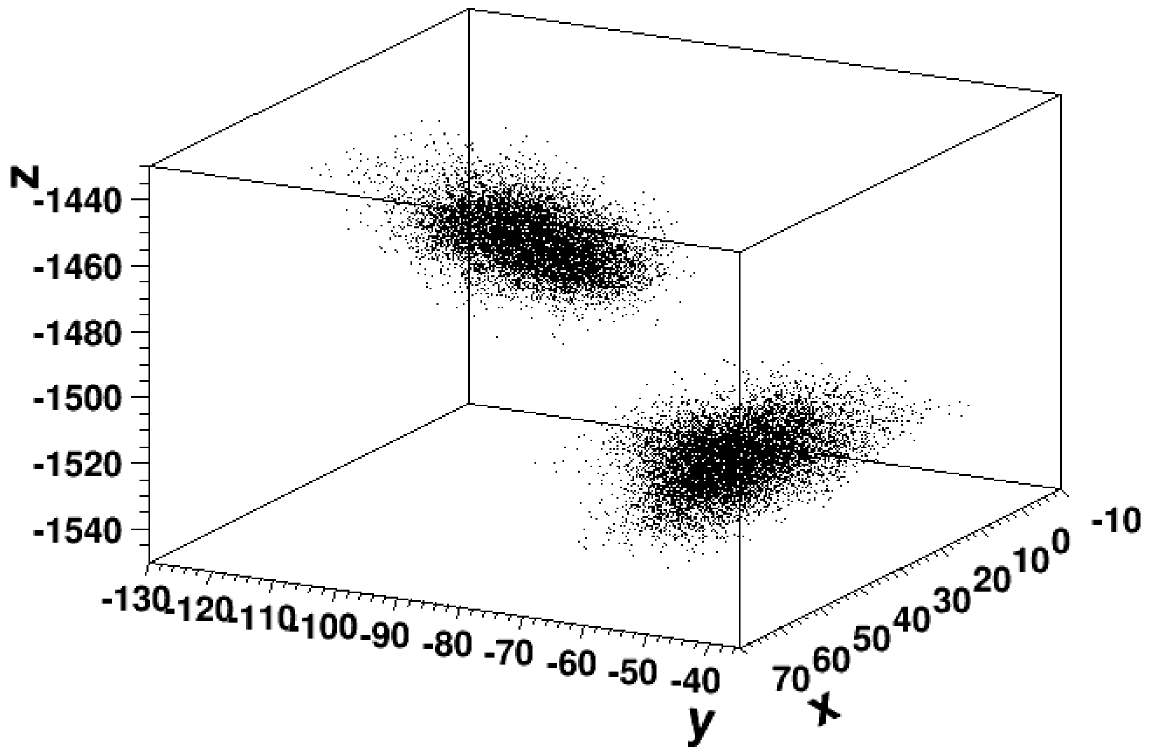
Label Switching Problem

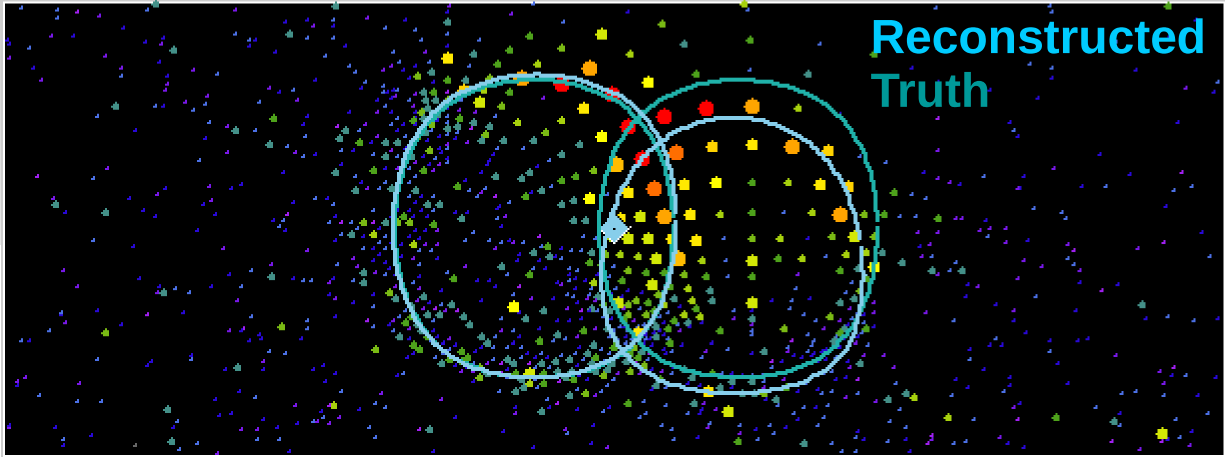




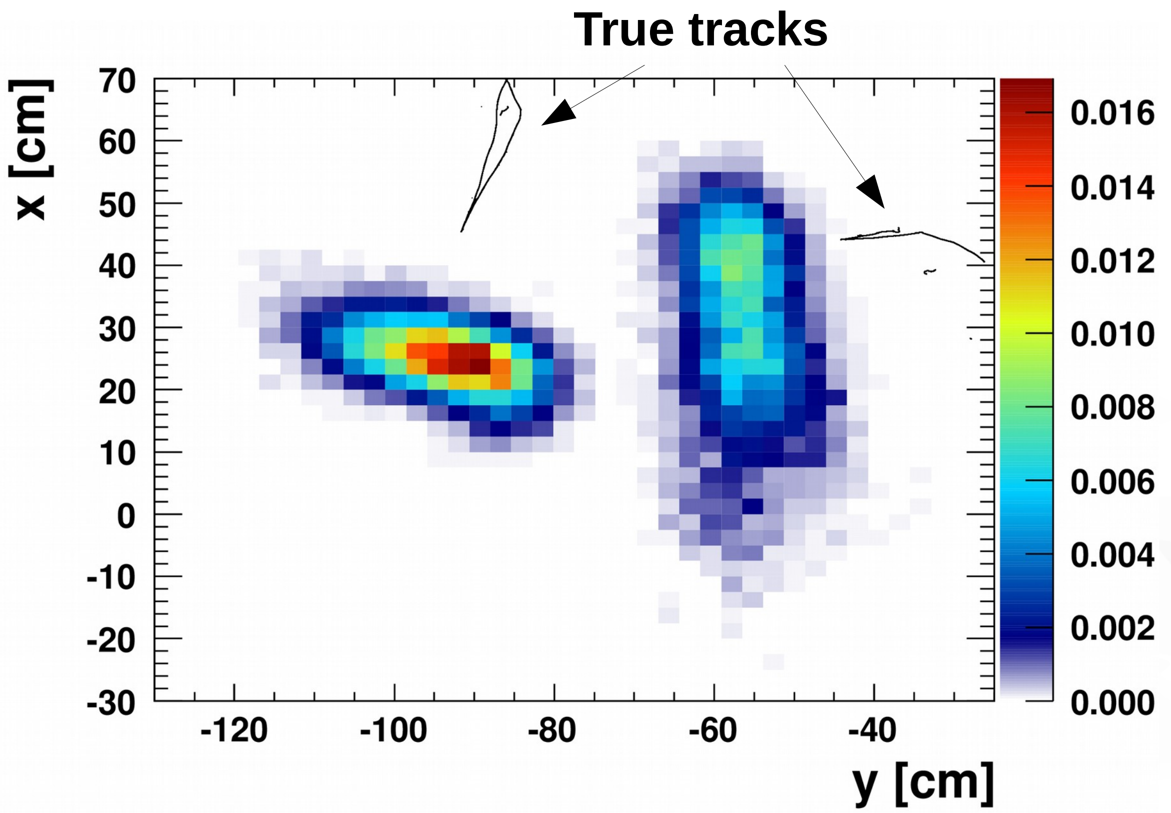
- Each MCMC step contains a list of rings
- Need to find MAP estimate to get most likely ring parameters
- Not necessarily in the same order each time
- Reorder rings in each MCMC step, based on some criteria
- Order by increasing ring momenta works fairly well

Vertex Information

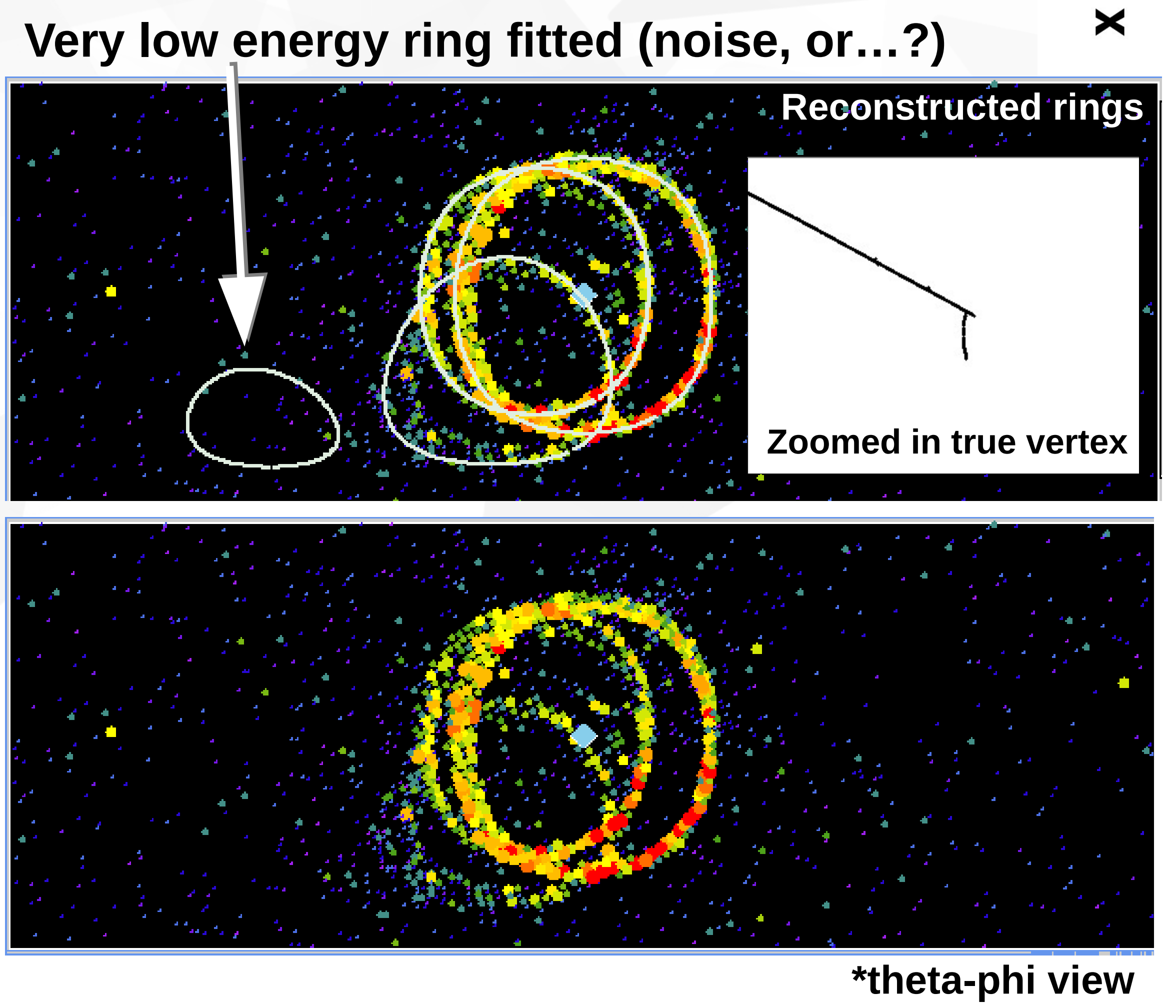


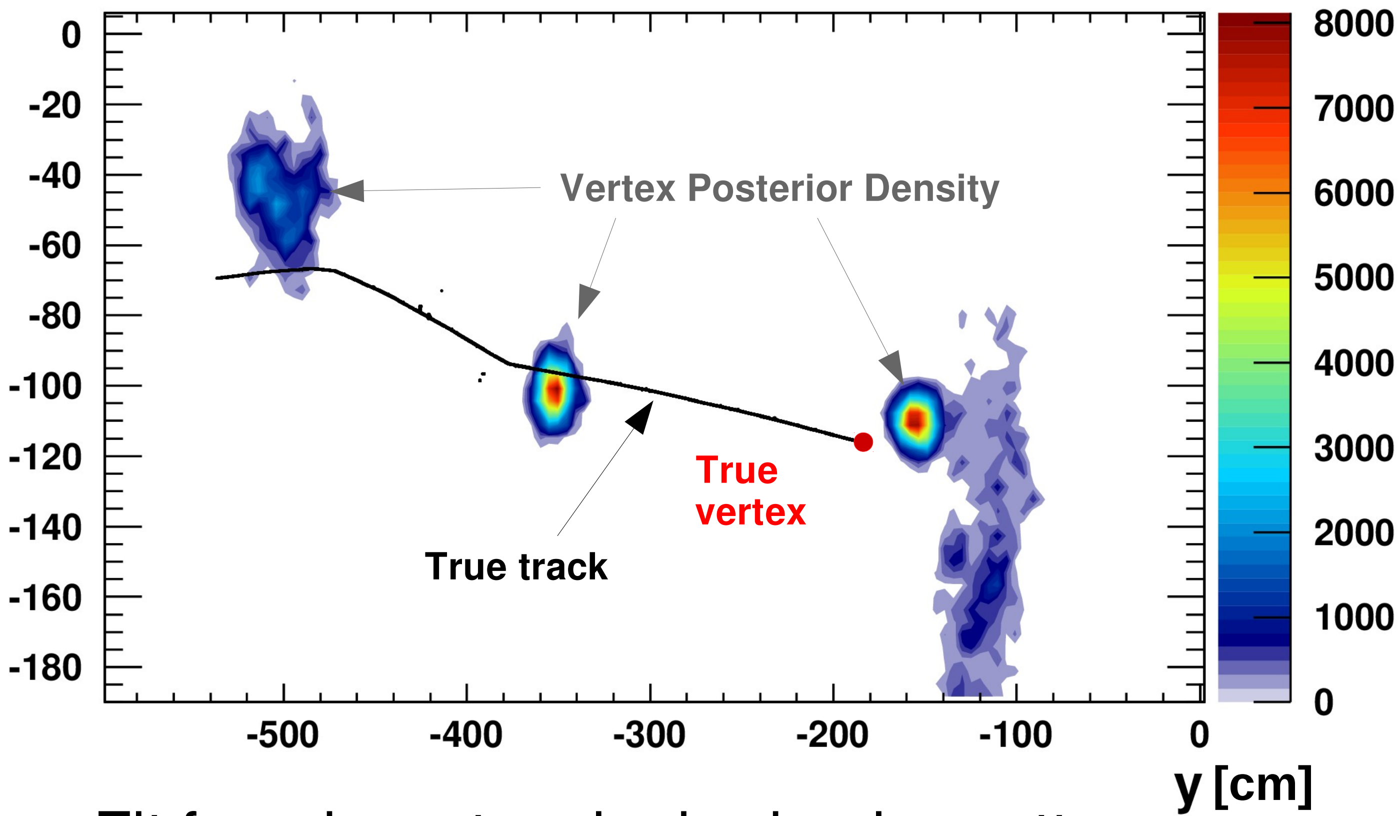


*theta-phi view



π⁺ MC Event

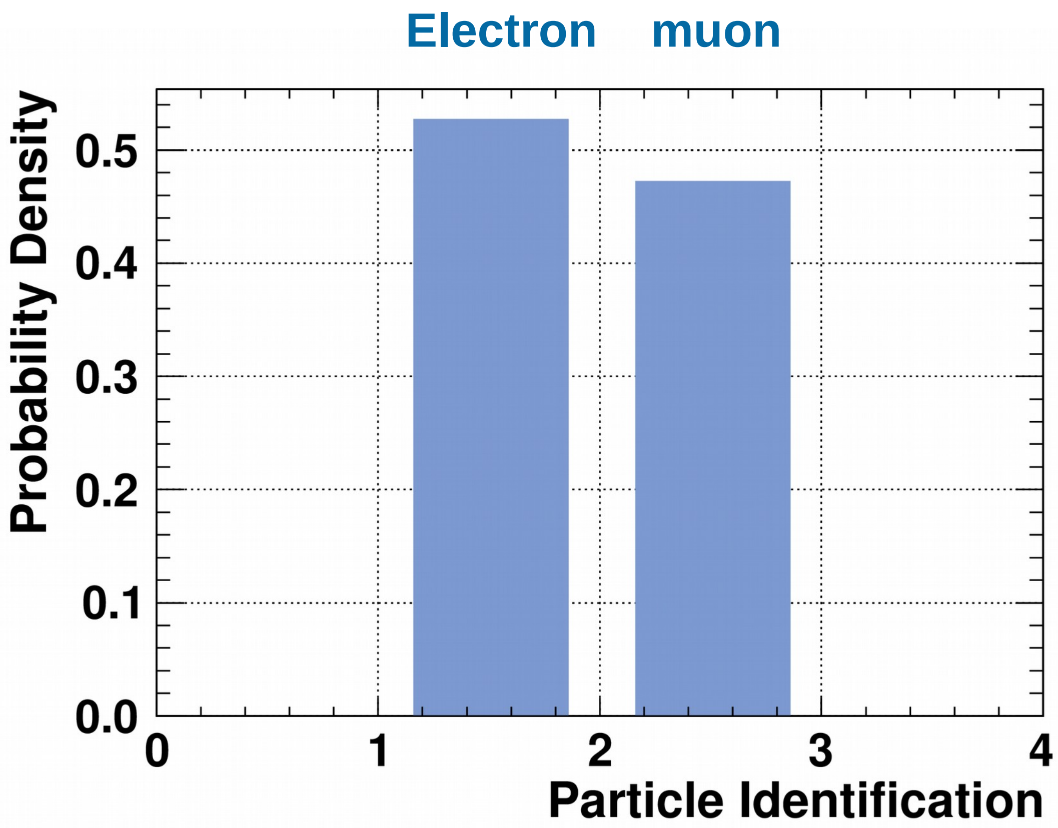




- Fit found most major hadronic scatters
- Also found very low energy ring
- Need to extract more truth info from simulation

Summary

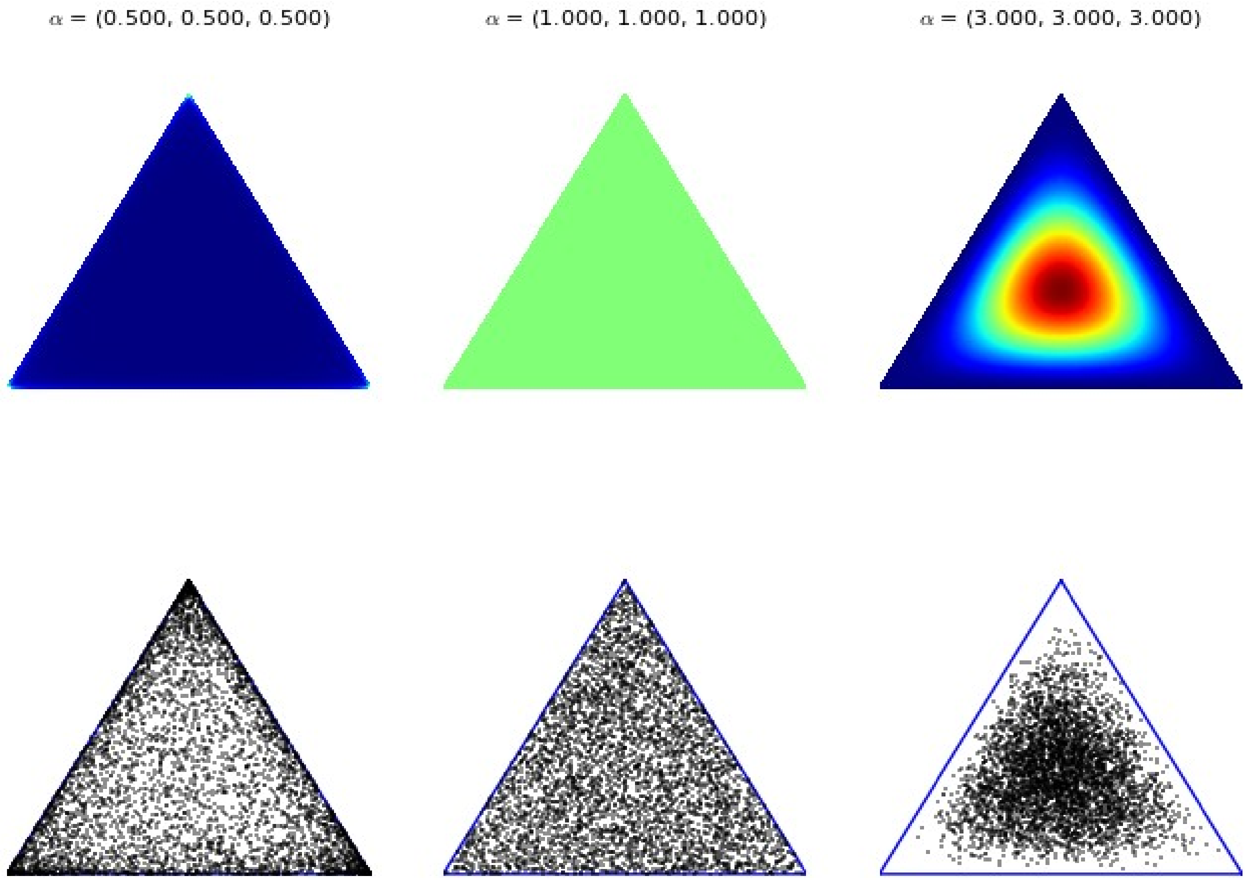
- Developing non-parametric Bayesian statistical tools to make better use of our data
- Extract Posterior probabilities about the parameters **(e.g. particle type probability)**
- Scales well for the future
  - Bigger detectors $\rightarrow$ more rings
  - Intense ν beams $\rightarrow$ event pileup
  - MCMC scales linearly with parameters
- Reconstructing complicated events will give us better sensitivity to real **physics processes**

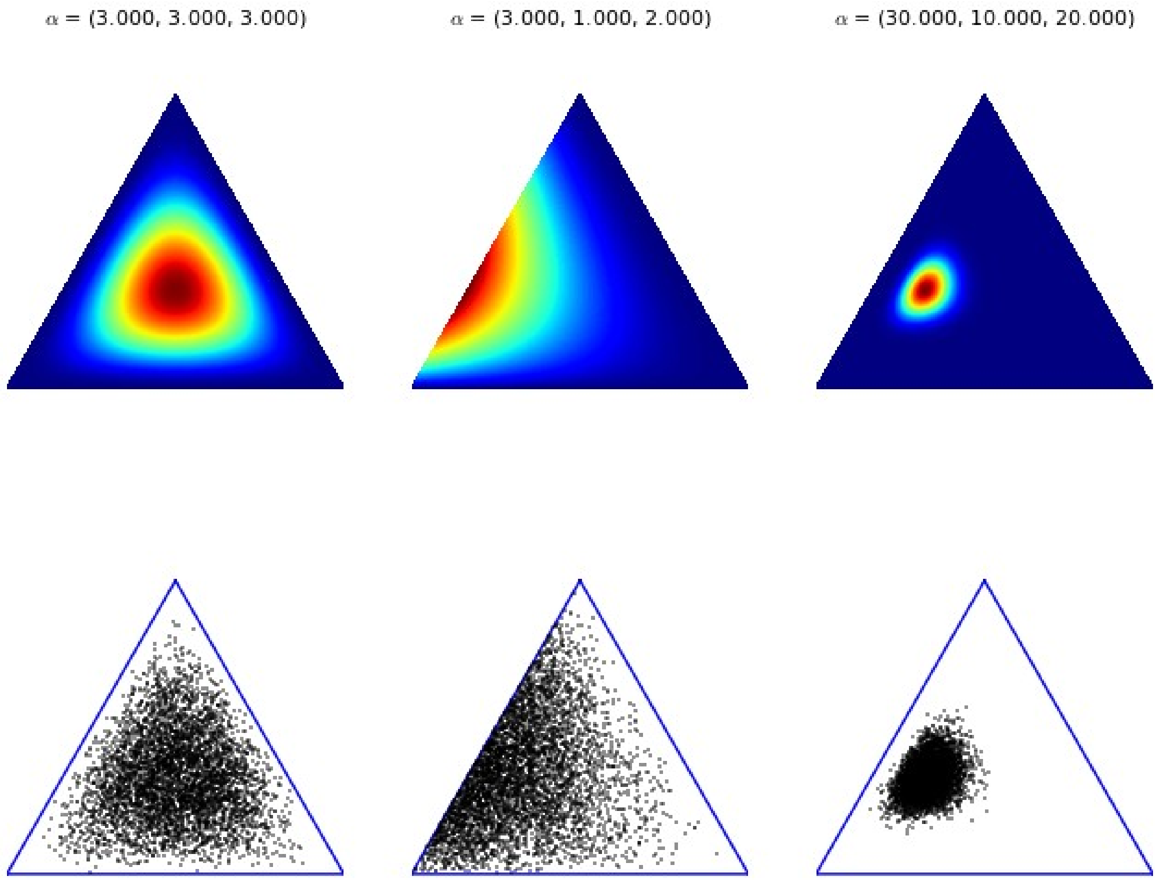


Backup

Dirichlet distribution

- The Dirichlet distribution extends the 2D beta distribution to ND
  - In the limit of infinite outcomes, it becomes the Dirichlet process
- The Dirichlet is the conjugate prior of the multinomial distribution
  - e.g. coin toss $\rightarrow$ die roll
- Indeed, the Dirichlet distribution becomes a beta when setting the dimensions to 2


α = (0.500, 0.500, 0.500)
α = (1.000, 1.000, 1.000)
α = (3.000, 3.000, 3.000)


α = (3.000, 3.000, 3.000)
α = (3.000, 1.000, 2.000)
α = (30.000, 10.000, 20.000)

Time likelihood in more detail

- Time likelihood is calculated roughly as follows:
  - Phit probability
  - Punit probability

  Poisson statistics

  - Direct light time PDF
  - Indirect light time PDF

  Empirical
- **Direct light PDF** is constructed by making **predicted charge vs residual time** plot for slices of momentum
  - Residual time is fit with Gaussian and then fit results are combined into a polynomial which gets evaluated for each PMT
- **Indirect light PDF** is much simpler, see next slide

Residual time

Vertex pos and dir

$$t_i^{\mathrm{res}} \equiv t_i - t - s_{\mathrm{mid}}/c - |\boldsymbol{R}^i_{\mathrm{PMT}} - \boldsymbol{x} - s_{\mathrm{mid}}\boldsymbol{d}|/c_n$$

Assumes all light is emitted from mid-track

Raw PMT hit time

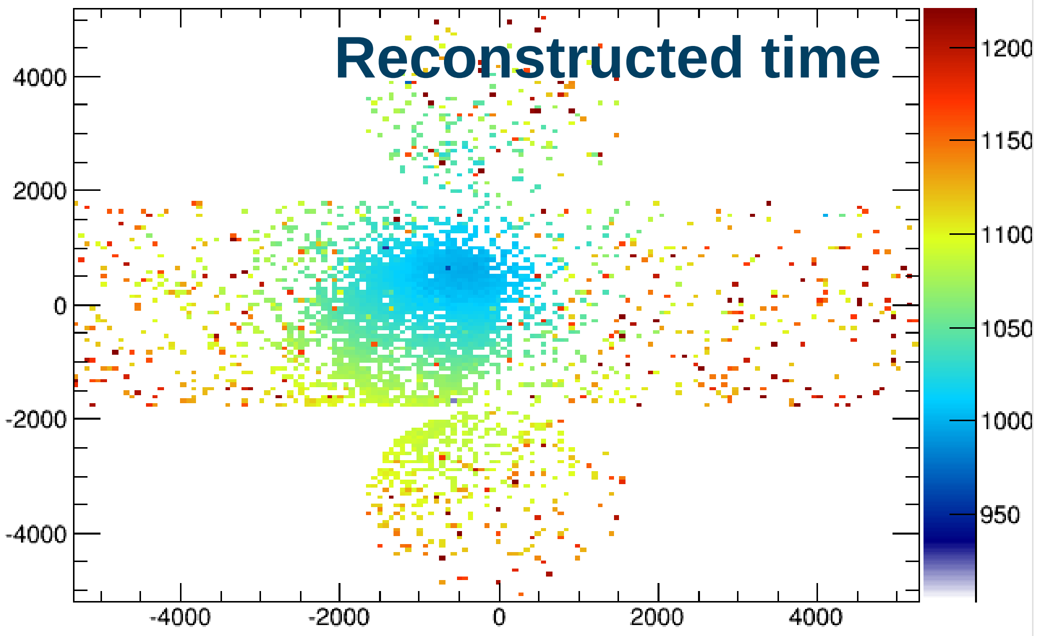
Vertex time

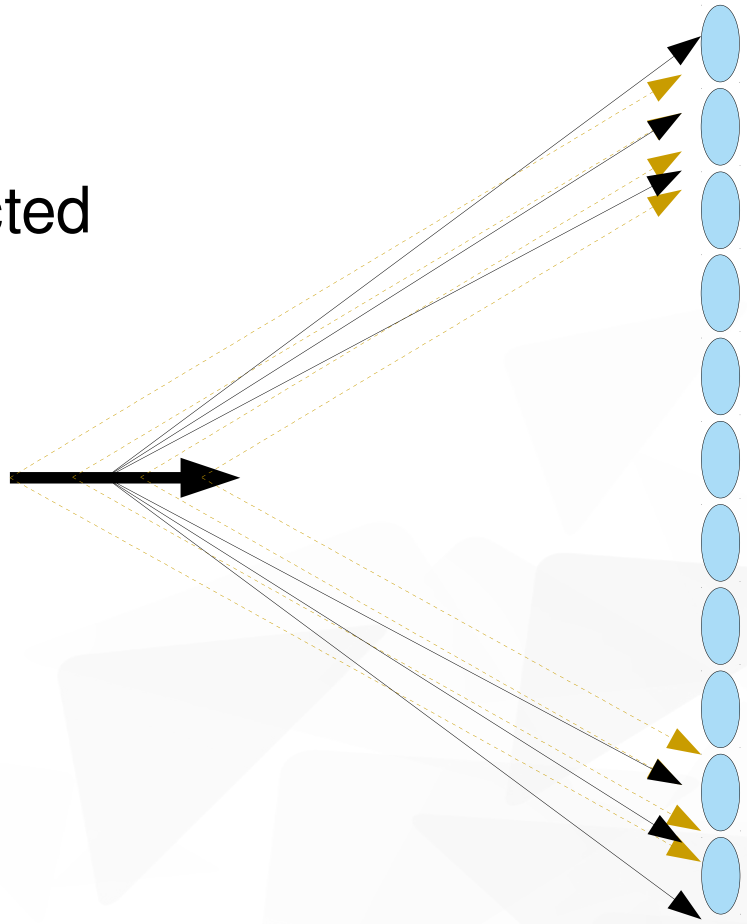
PMT pos

Distance to track mid-point

Time prediction

- To make the "predicted time", assume that all hits come from the middle of the track
- Calculate the time of flight from mid-track to PMT
- Create **residual** by subtracting TOF from reconstructed time

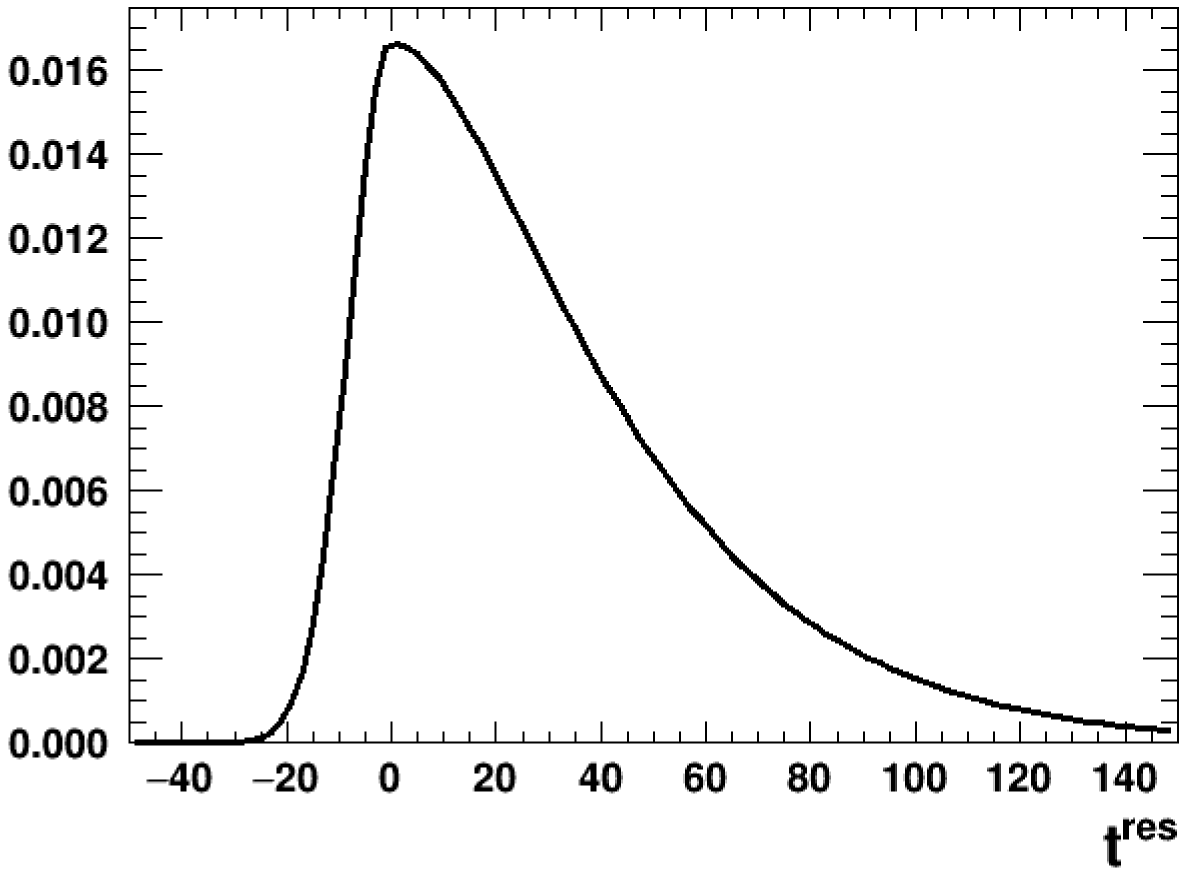




Indirect light PDF

- Indirect light PDF follows the functional form below plot
- Parameter values are fixed at:

  $\tau = t^{\mathrm{res}} - 25\mathrm{ns},\ \sigma = 8\mathrm{ns}\ \mathrm{and}\ \gamma = 25\mathrm{ns}$
- There is **no** dependence on predicted charge or momentum
  - All PMTs for all events use this distribution



$$f_t^{\mathrm{sct}}(t^{\mathrm{res}}) = 1/\left(\sqrt{\frac{\pi}{2}}\sigma + 2\gamma\right) \times \begin{cases} \exp(-\tau^2/2\sigma^2) & (\tau < 0) \\ (\tau/\gamma + 1)\exp(-\tau/\gamma) & (\tau > 0) \end{cases}$$