DUNE-France workshop #2 *November 16, 2023*

Neutrino oscillation parameter inference with MaCh3 in DUNE

Leïla Haegel / *IP2I Lyon*

Neutrino oscillations softwares in DUNE

- **Publications with neutrino oscillation parameters estimations:**
  - Long-baseline neutrino oscillation physics potential of the DUNE experiment arxiv:2006.16043 (overlap with DUNE TDR Vol II)
  - Low exposure long-baseline neutrino oscillation sensitivity of the DUNE experiment arxiv:2109.01304

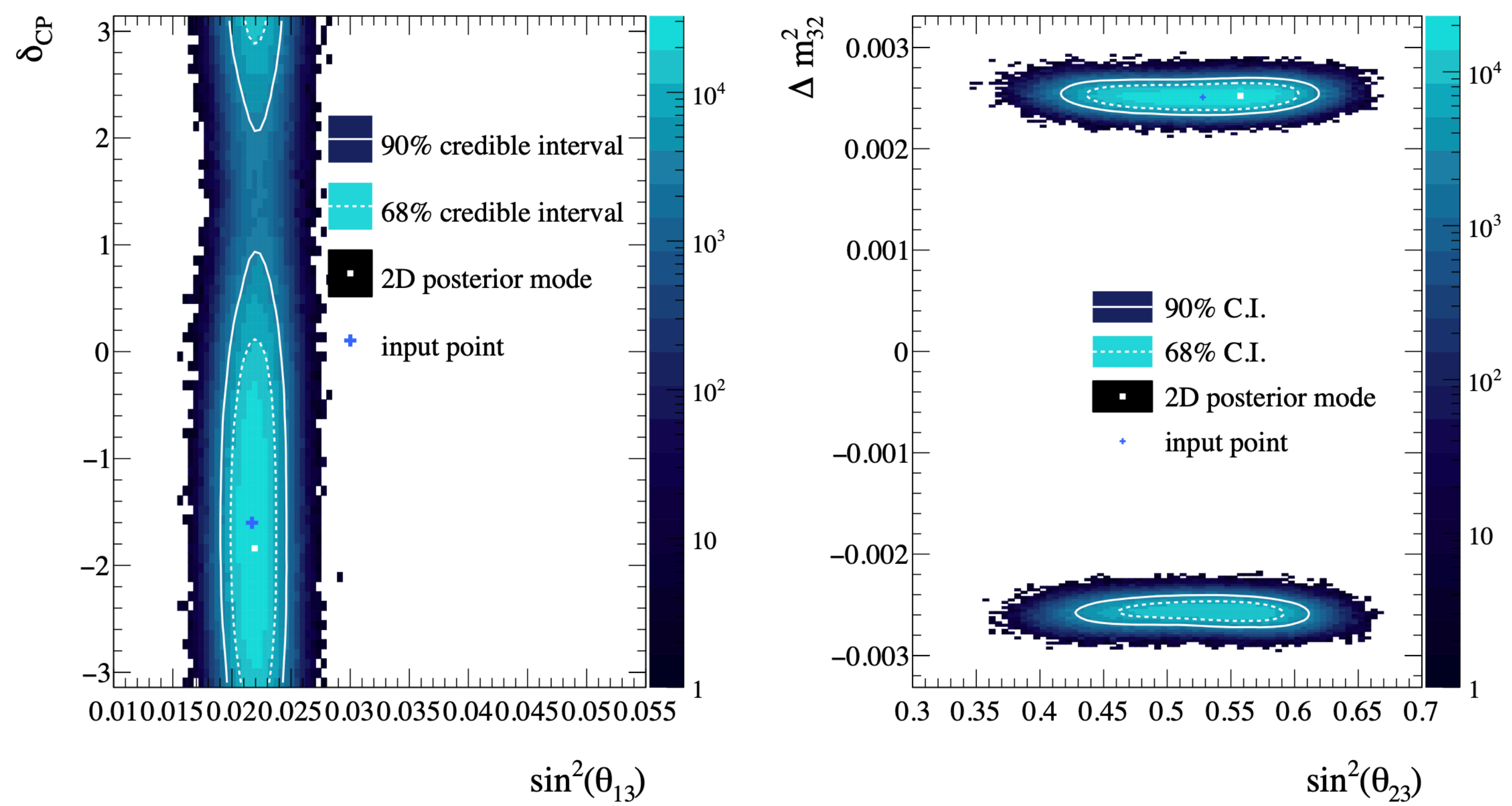
- **Based on the CAFana framework**
  - Frequentist framework based on MINUIT (ROOT)
  - Developed for NOvA, ported over to DUNE
  - Not used anymore since end of 2022

- **Current oscillation parameter estimation framework: MaCh3**
  - Bayesian framework based on Markov Chain Monte-Carlo sampling
  - Developed for T2K, ported over to DUNE
  - Only DUNE software since 2023

MaCh3

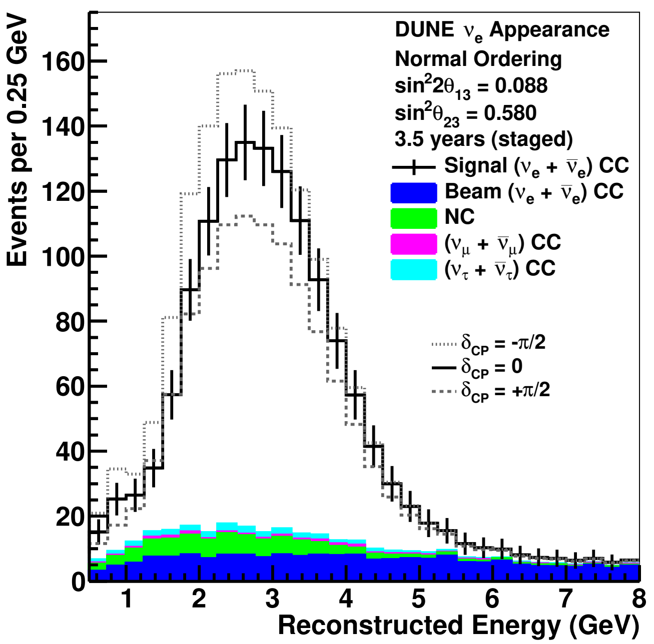
- **<u>Ma</u>rkov <u>Ch</u>ain <u>3</u>-ν oscillation parameters estimation framework:**
  - Performs Bayesian inference: output is posterior probability distributions
  - Uses Markov Chain Monte-Carlo to sample the probability distributions
  - Reported intervals are credible intervals (i.e. 90% region of higher posterior probability)
  - Not made for point estimates

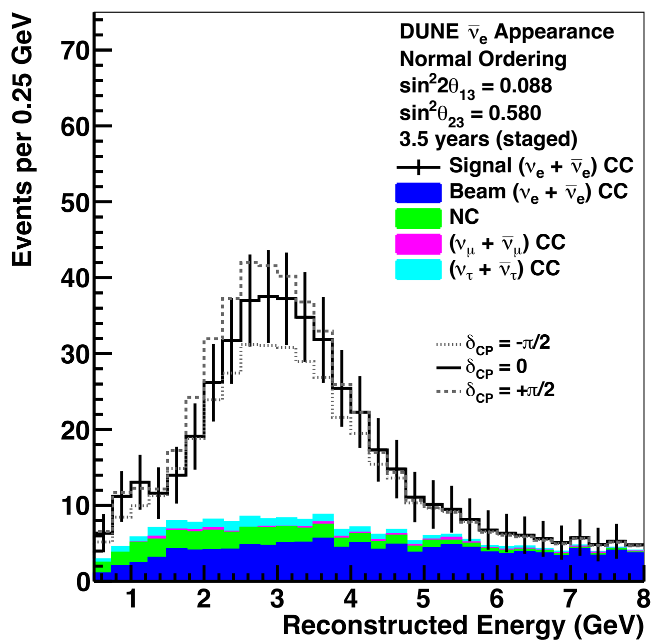
<u>T2K 2017 (my PhD thesis)</u>



Details on the statistical method: see yesterday's masterclass
<u>"Bayesian statistics, inference and sampling methods for neutrino physics"</u>

Steps by steps

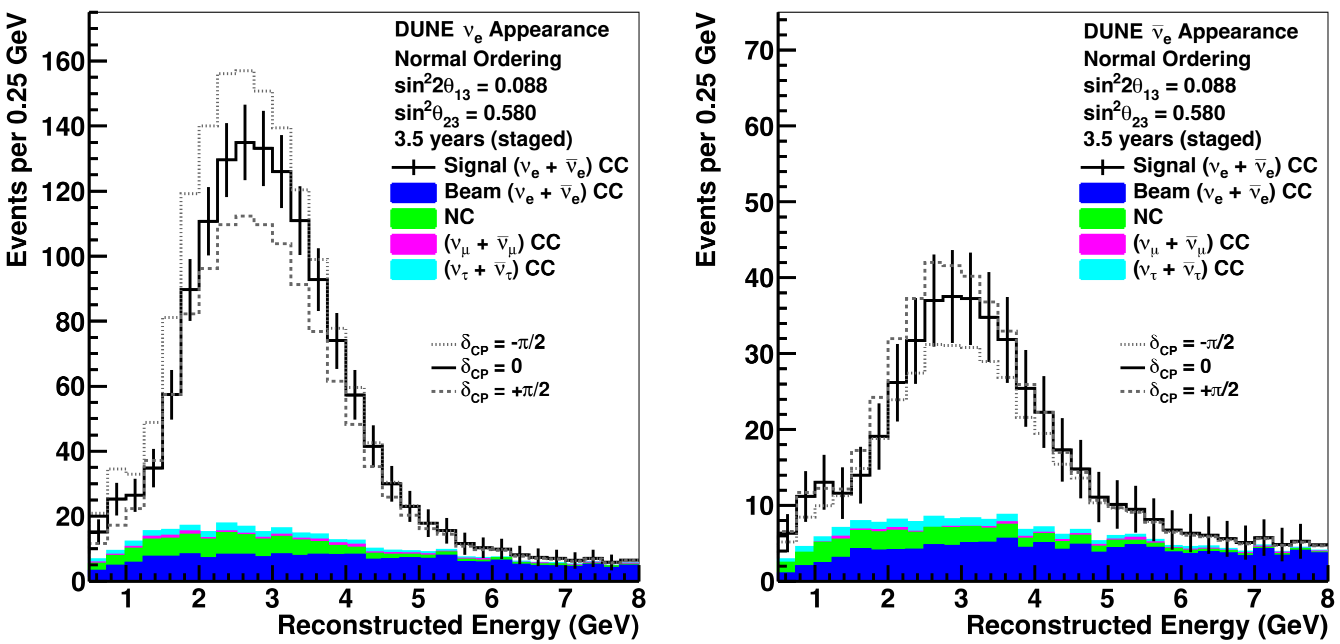




- Select a sample of data
- Get the first iteration of a Monte-Carlo sample
- Compute the posterior probability:

$$P(\vec{\vartheta}_{s=0} | D) = P(D | \vec{\vartheta}_{s=0})\ P(\vec{\vartheta}_{s=0})$$

Steps by steps



- Select a sample of data
- Get the first iteration of a Monte-Carlo sample
- Compute the posterior probability:

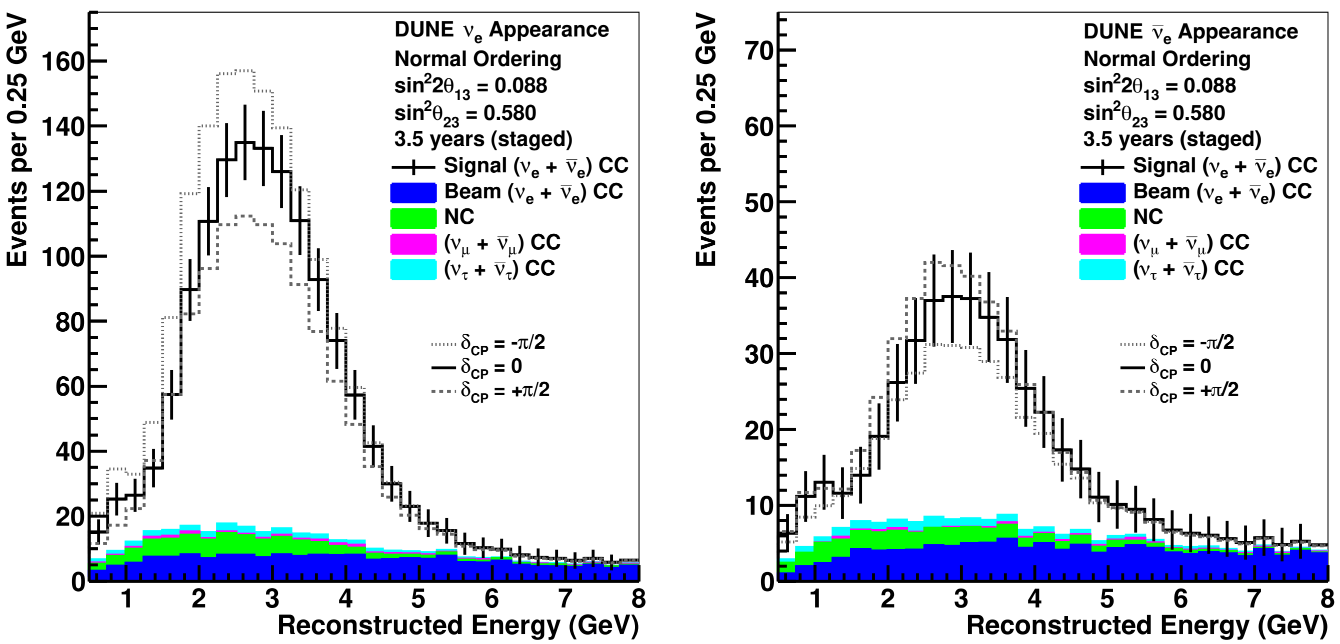
$$P(\vec{\vartheta}_{s=0} | D) = P(D | \vec{\vartheta}_{s=0})\; P(\vec{\vartheta}_{s=0})$$

posterior probability

$\vec{\vartheta}$ *: all the parameters varied during the inference includes oscillation parameters, but also systematics parameters*

s=0: the 0th step

Steps by steps



- Select a sample of data
- Get the first iteration of a Monte-Carlo sample
- Compute the posterior probability:

$$P(\vec{\vartheta}_{s=0} | D) = \boxed{P(D | \vec{\vartheta}_{s=0})}\, P(\vec{\vartheta}_{s=0})$$

likelihood

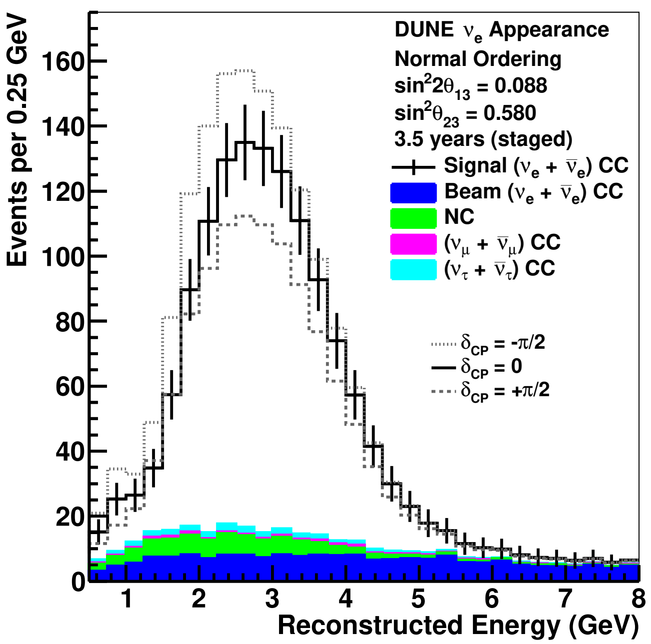
$$P(D | \vec{\vartheta}_{s=0}) = \sum_{i}^{B\ bins} N_i^{MC} - N_i^{D} + N_i^{D}\ ln \frac{N_i^D}{N_i^{MC}}$$

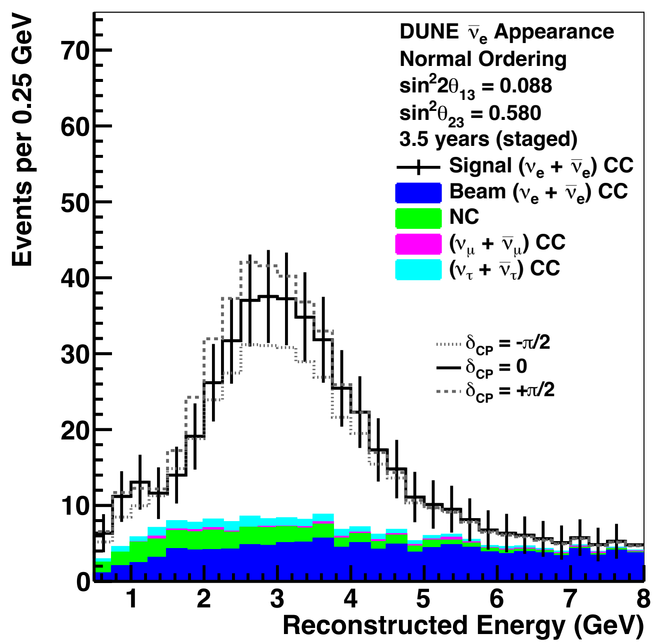
$$+ \frac{1}{2} \sum_{j}^{o,b,i,d} \sum_{k}^{o,b,i,d} \Delta(o,b,i,d)_j\ (V^{-1}_{o,b,i,d})_{j,k}\ \Delta(o,b,i,d)_k$$

Binned Poisson log-likelihood ratio:

- N_i^{MC} *: number of MC events in the bin i*
- N_i^{D} *: number of data events in the bin i*

Steps by steps





- Select a sample of data
- Get the first iteration of a Monte-Carlo sample
- Compute the posterior probability:

$$P(\vec{\vartheta}_{s=0} | D) = P(D | \vec{\vartheta}_{s=0})\, P(\vec{\vartheta}_{s=0})$$

likelihood

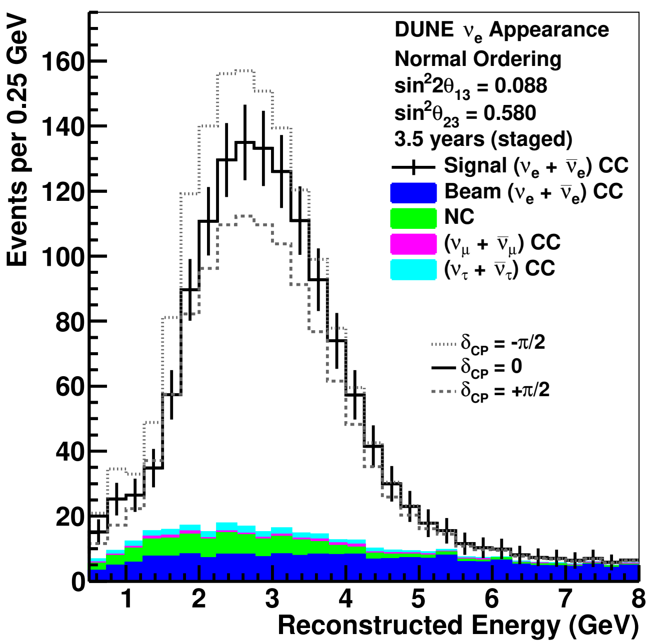
$$P(D | \vec{\vartheta}_{s=0}) = \sum_{i}^{B\ bins} N_i^{MC} - N_i^D + N_i^D \ ln \frac{N_i^D}{N_i^{MC}}$$

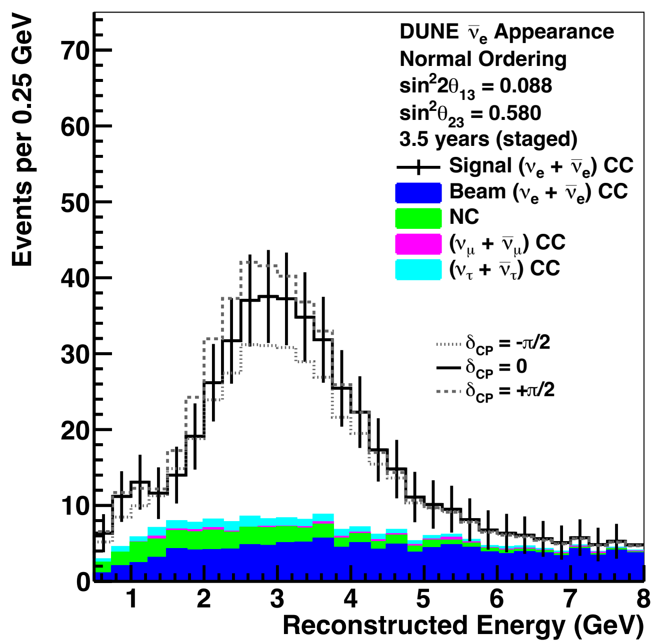
$$+ \frac{1}{2} \sum_{j}^{o,b,i,d} \sum_{k}^{o,b,i,d} \Delta(o,b,i,d)_j \, (V^{-1}_{o,b,i,d})_{j,k} \, \Delta(o,b,i,d)_k$$

Penalty term on the systematics parameters:

- *$V_{o,b,c,d}$: covariance matrix for oscillation (o), beam (b), cross-section (c) and detector (d) parameters*
- *$\Delta(o,b,i,d)_j$: difference between the parameters at this step and the previous one*

Steps by steps





- Select a sample of data
- Get the first iteration of a Monte-Carlo sample
- Compute the posterior probability:

$$P(\vec{\vartheta}_{s=0}|D) = P(D|\vec{\vartheta}_{s=0})\,P(\vec{\vartheta}_{s=0})$$

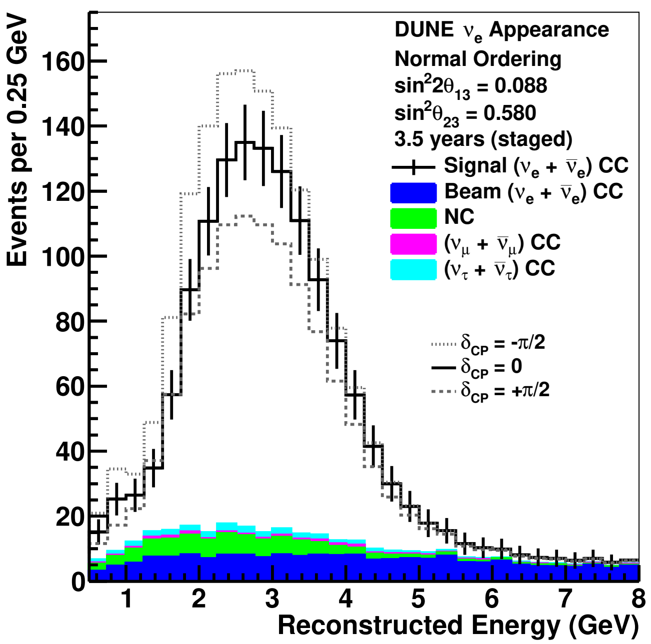
prior probability

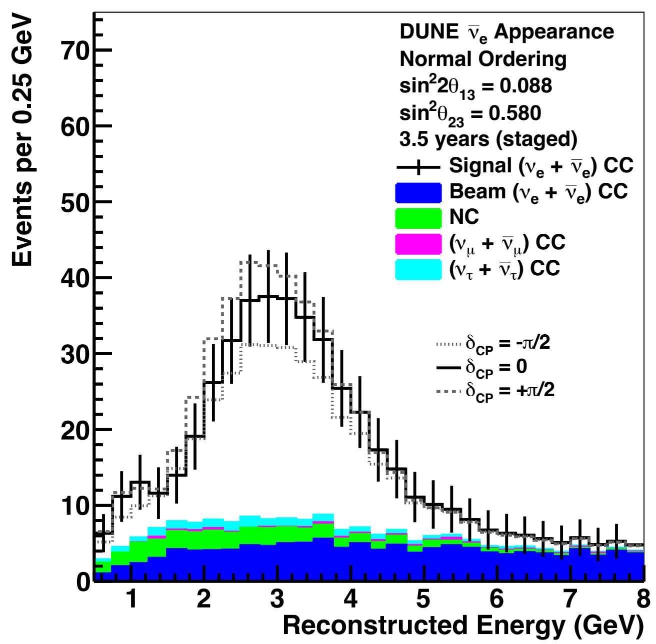
$$P(\vec{\vartheta}_{s=0}) = P(\vec{\vartheta}) = \sqcap$$

Uniform prior probability for oscillation parameters to determine

$$= \mathcal{N}(\mu_{o,b,c,d}, \sigma_{o,b,c,d})$$

Steps by steps





- Select a sample of data
- Get the first iteration of a Monte-Carlo sample
- Compute the posterior probability:

$$P(\vec{\vartheta}_{s=0}\,|\,D) = P(D\,|\,\vec{\vartheta}_{s=0})\,P(\vec{\vartheta}_{s=0})$$

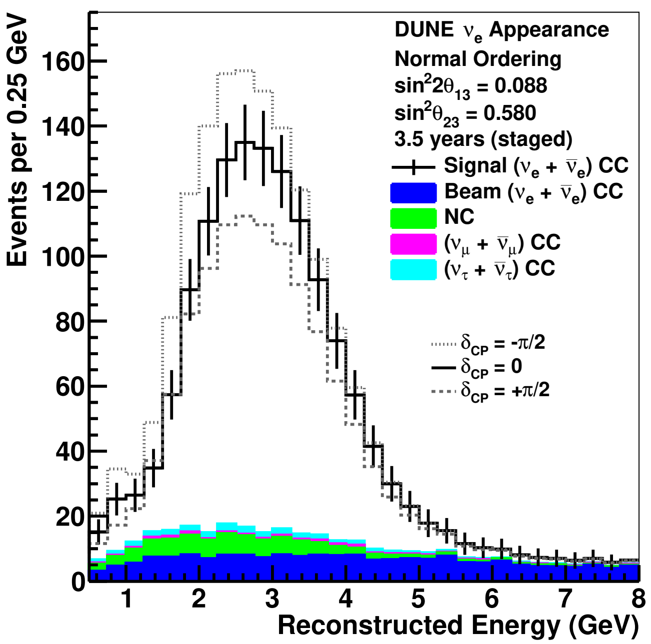
prior probability

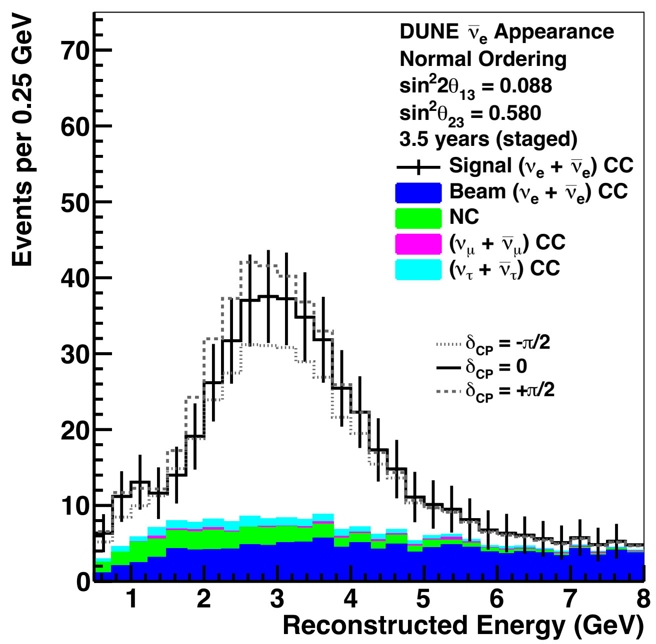
$$P(\vec{\vartheta}_{s=0}) = P(\vec{\vartheta}) = \sqcap$$

$$= \mathcal{N}(\mu_{o,b,c,d}, \sigma_{o,b,c,d})$$

→ *Gaussian prior probability for systematic parameters with prior knowledge*

Steps by steps

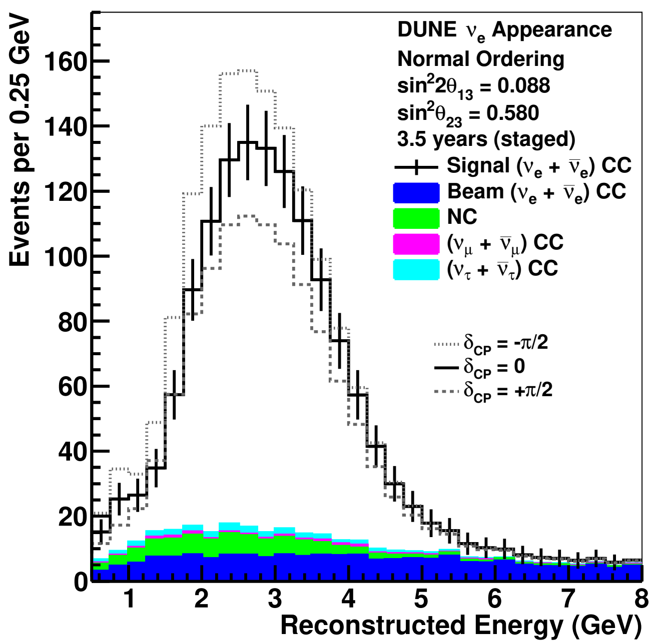


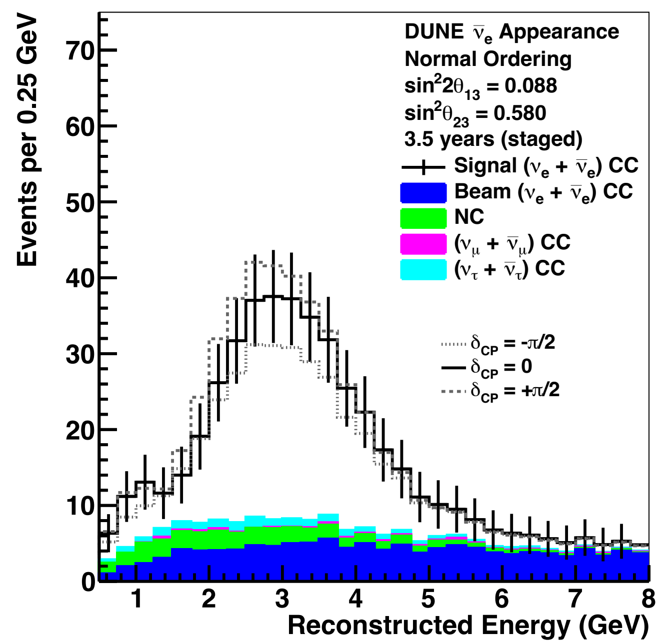


- Select a sample of data
- Get the first iteration of a Monte-Carlo sample
- Compute the posterior probability
- Save the step $s=0$
- Go to the next step $s=1$

Throw a new value of the parameters $\vec{\vartheta}_{s=1}$
using the « jump function » (or « proposal function »)
= multivariate normal distribution $\mathcal{N}(\vec{\mu} = \vec{\vartheta}_{s=0}, \vec{\sigma})$

Steps by steps



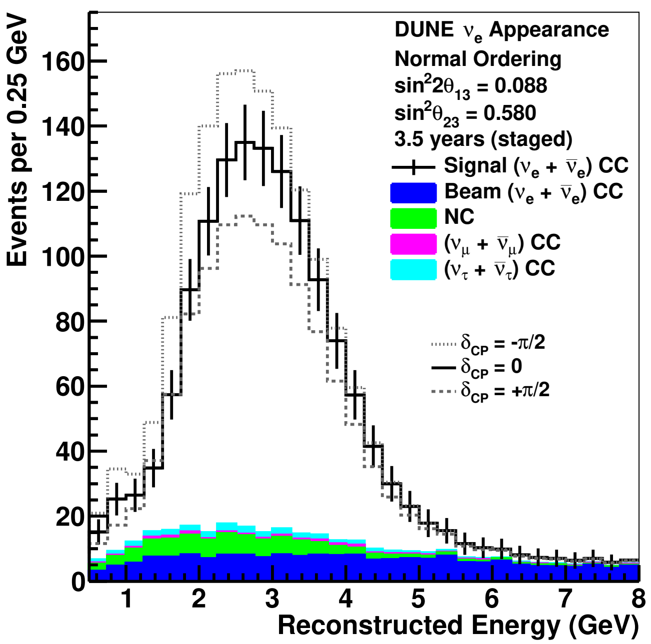


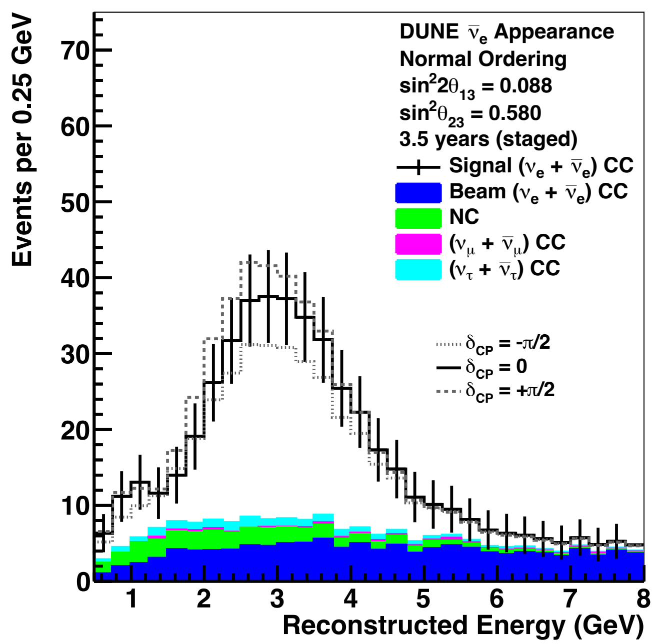
- Select a sample of data
- Get the first iteration of a Monte-Carlo sample
- Compute the posterior probability
- Save the step $s=0$
- Go to the next step $s=1$

Throw a new value of the parameters $\vec{\vartheta}_{s=1}$
using the « jump function » $J(\vec{\vartheta}_1 \mid \vec{\vartheta}_0)$
$=$ *multivariate normal distribution* $\mathcal{N}(\vec{\mu} = \vec{\vartheta}_{s=0}, \vec{\sigma})$

Step size:
- *oscillation parameters :*
  standard deviation × scale factor
- *systematics parameters :*
  prior uncertainty × scale factor

Steps by steps





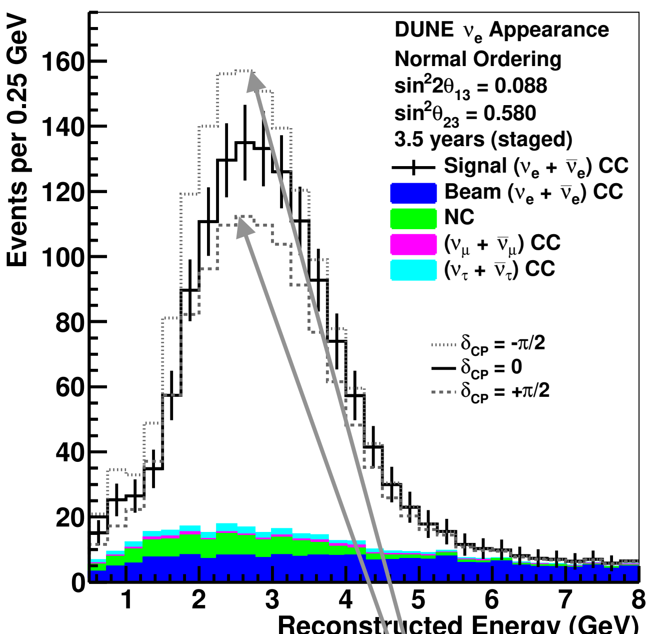
- Select a sample of data
- Get the first iteration of a Monte-Carlo sample
- Compute the posterior probability
- Save the step $s=0$
- Go to the next step $s=1$

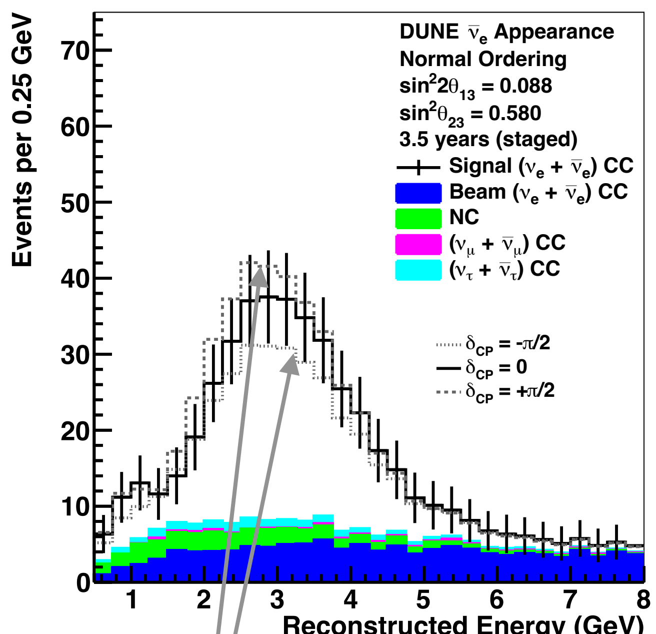
Throw a new value of the parameters $\vec{\vartheta}_{s=1}$
using the « jump function » $J(\vec{\vartheta}_1 \mid \vec{\vartheta}_0)$
$=$ *multivariate normal distribution* $\mathcal{N}(\vec{\mu} = \vec{\vartheta}_{s=0}, \vec{\sigma})$

Step size:
- *oscillation parameters :*
  standard deviation × scale factor
- *systematics parameters :*
  prior uncertainty × scale factor

Heuristic determination of the scale factor,
impacts a lot the speed of the inference process
=> work ongoing in MaCh3

Steps by steps

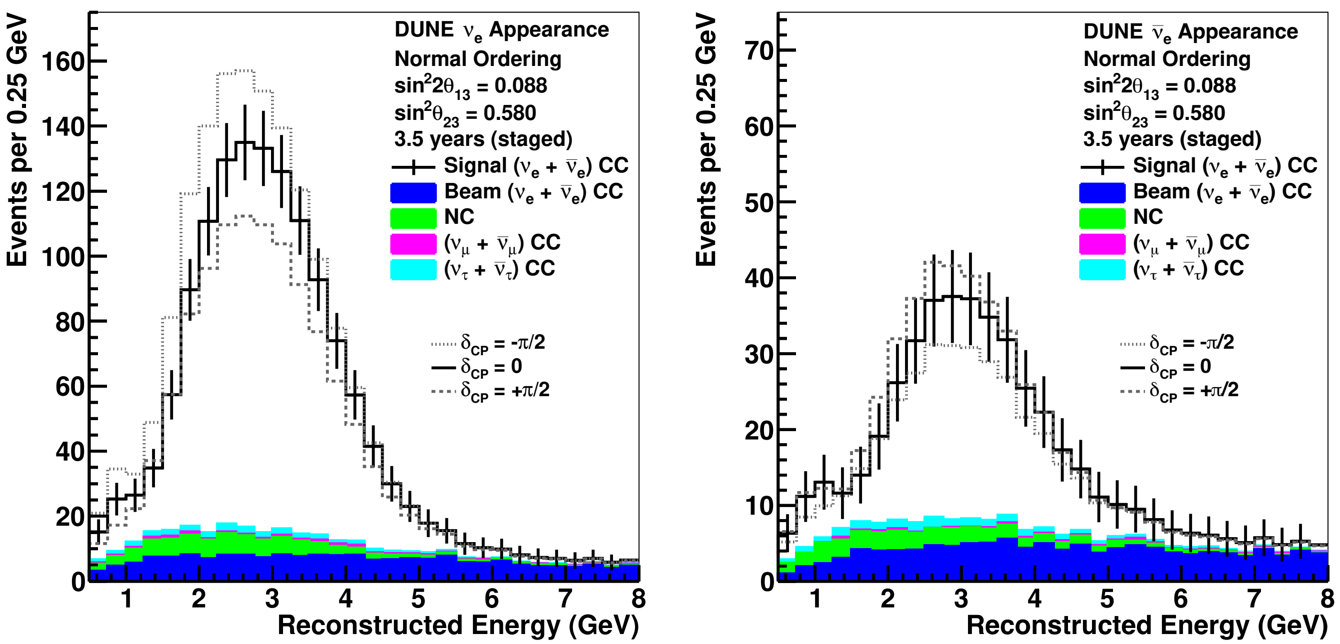




- Select a sample of data
- Get the first iteration of a Monte-Carlo sample
- Compute the posterior probability
- Save the step $s=0$
- Go to the next step $s=1$
- Recompute MC histogram

Simple case with 1 parameter:
- *throw a new value of* δ_{CP}
- *compute the new value of the oscillation probability*
- *apply it to the MC histogram*

Steps by steps



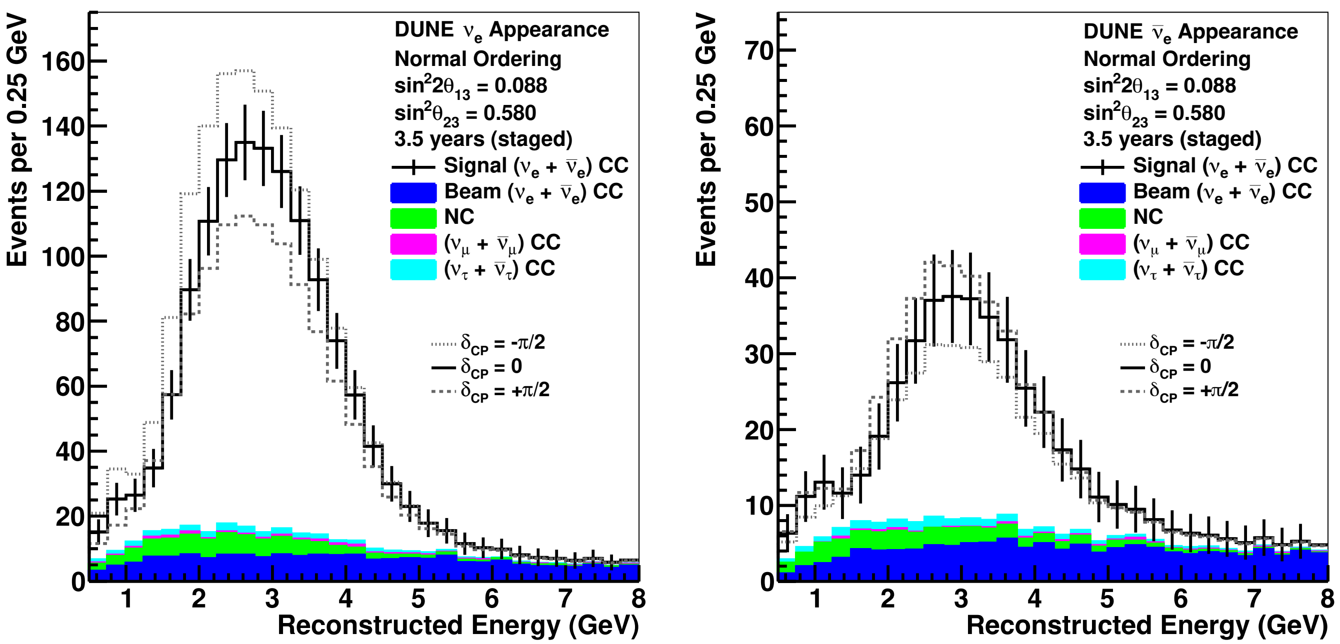
- Select a sample of data
- Get the first iteration of a Monte-Carlo sample
- Compute the posterior probability
- Save the step $s=0$
- Go to the next step $s=1$
- Recompute MC histogram

Realistic case:
throw all parameters at every step
i.e. both oscillation and systematics

Effect of modifying the systematics parameters:
- *for each bin, apply a new weight*
  => modify the number of event according to the new parameter value
- *the weight is computed with cubic spline functions (i.e. piecewise polynomial functions with continuity conditions)*

Heuristic determination of the scale factor, impacts a lot the speed of the inference process
=> work ongoing in MaCh3

Steps by steps



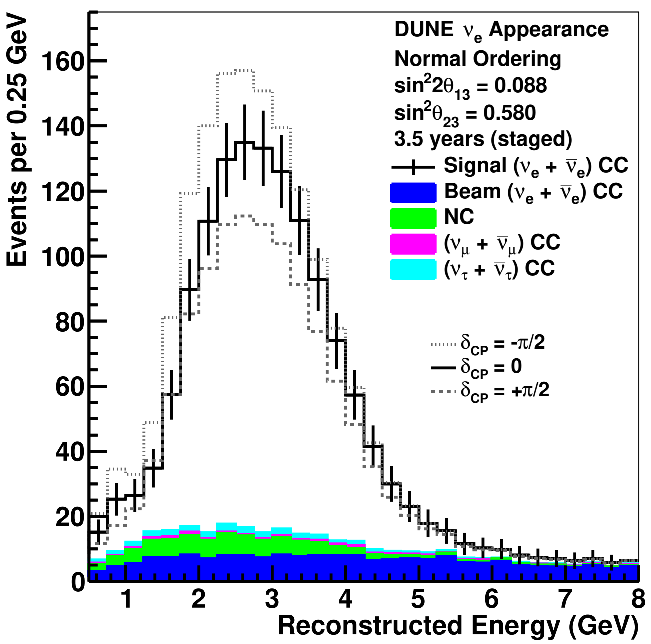
- Select a sample of data
- Get the first iteration of a Monte-Carlo sample
- Compute the posterior probability
- Save the step $s=0$
- Go to the next step $s=1$
- Recompute MC histogram

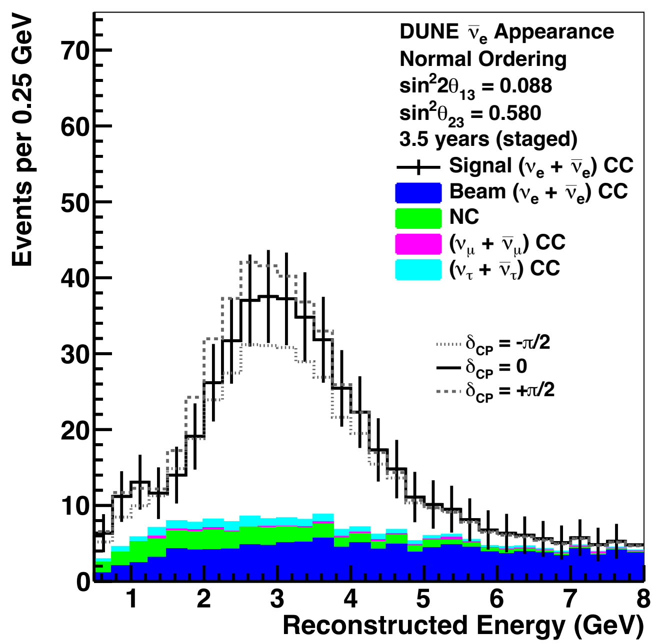
Realistic case:
throw all parameters at every step
i.e. both oscillation and systematics

Effect of modifying the systematics parameters:
- *for each bin, apply a new weight*
  => modify the number of event according to the new parameter value
- *the weight is computed with cubic spline functions (i.e. piecewise polynomial functions with continuity conditions)*

Splines are computed with the nusystematics / xsectools packages -> validation status?

Steps by steps

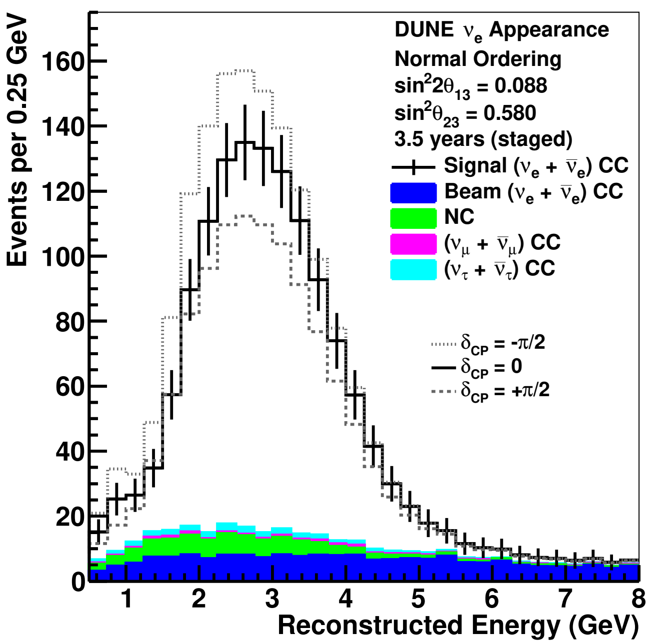


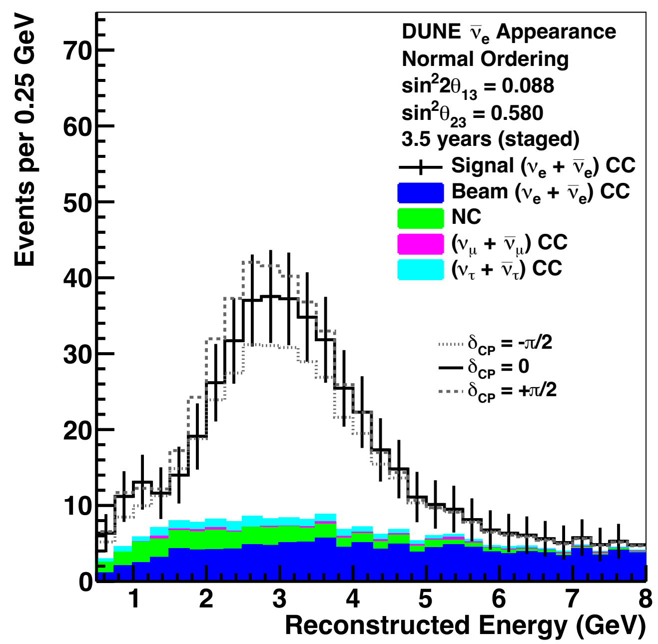


- Select a sample of data
- Get the first iteration of a Monte-Carlo sample
- Compute the posterior probability
- Save the step $s=0$
- Go to the next step $s=1$
- Recompute MC histogram
- Recompute the posterior probability

$$P(\overrightarrow{\vartheta}_{s=1} | D) = P(D | \overrightarrow{\vartheta}_{s=1})\ P(\overrightarrow{\vartheta}_{s=1})$$

Steps by steps





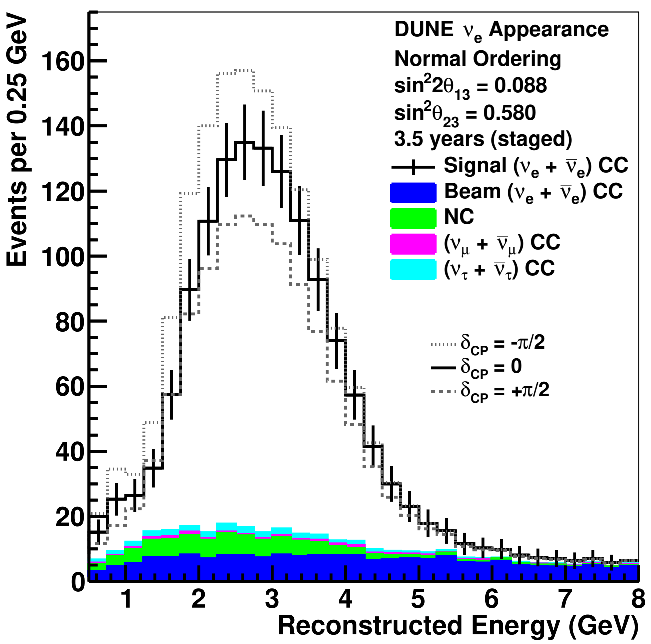
- Select a sample of data
- Get the first iteration of a Monte-Carlo sample
- Compute the posterior probability
- Save the step $s=0$
- Go to the next step $s=1$
- Recompute MC histogram
- Recompute the posterior probability
- Apply the Metropolis-Hasting theorem

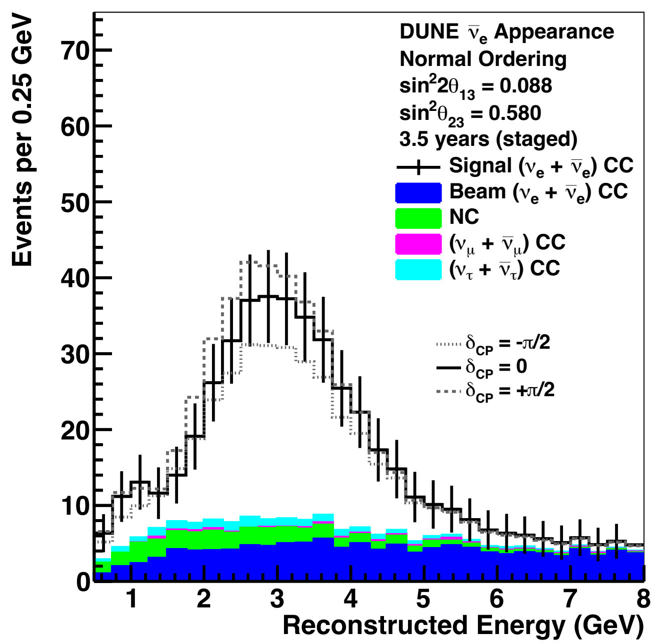
Compute the Metropolis-Hastings ratio

$$r = \frac{P(\overrightarrow{\vartheta}_{s=1}|D)\ J(\overrightarrow{\vartheta}_1|\overrightarrow{\vartheta}_0)}{P(\overrightarrow{\vartheta}_{s=0}|D)\ J(\overrightarrow{\vartheta}_0|\overrightarrow{\vartheta}_1)}$$

If $r \geq 1$: accept the step
If $r < 1$: throw a random number $X \in [0,1]$
and accept the step if $r > X$

Steps by steps

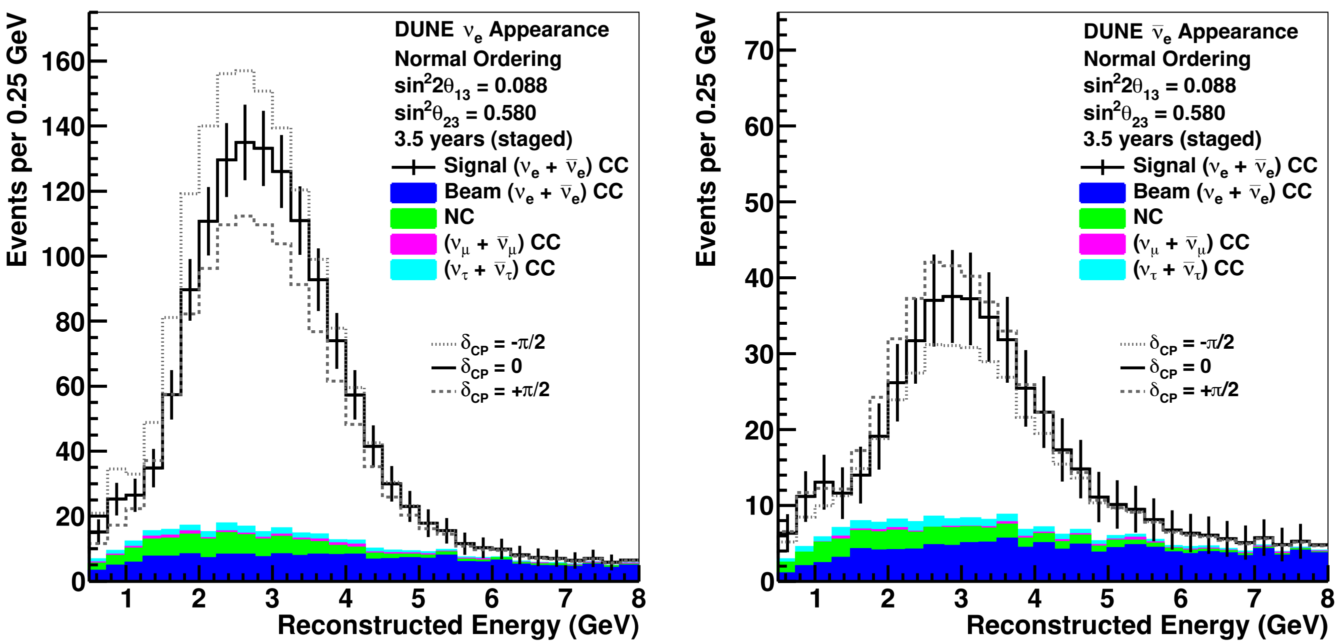




- Select a sample of data
- Get the first iteration of a Monte-Carlo sample
- Compute the posterior probability
- Save the step $s=0$
- Go to the next step $s=1$
- Recompute MC histogram
- Recompute the posterior probability
- Apply the Metropolis-Hasting theorem
- Continue until reaching ~10^8 steps

Parallelisation is mandatory!
- *Can run with CPU and GPU*
- *GPU only available on selected clusters (Wilson cluster @FNAL, others?)*
- *Take several days*

Steps by steps

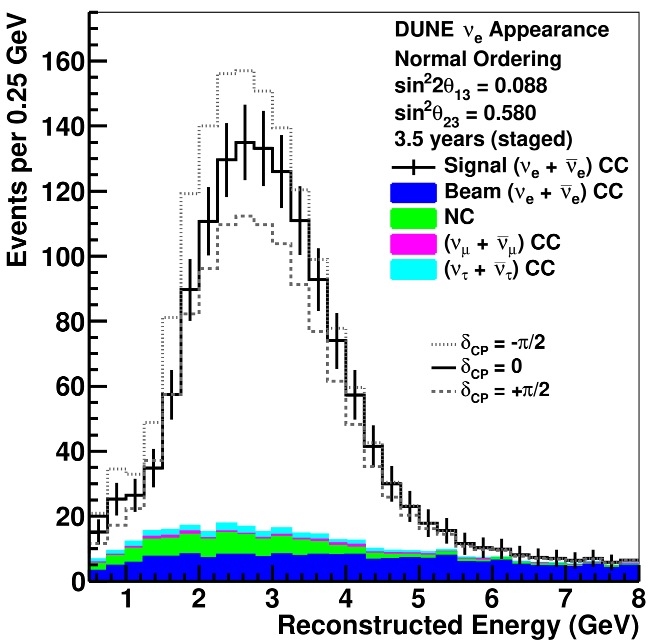


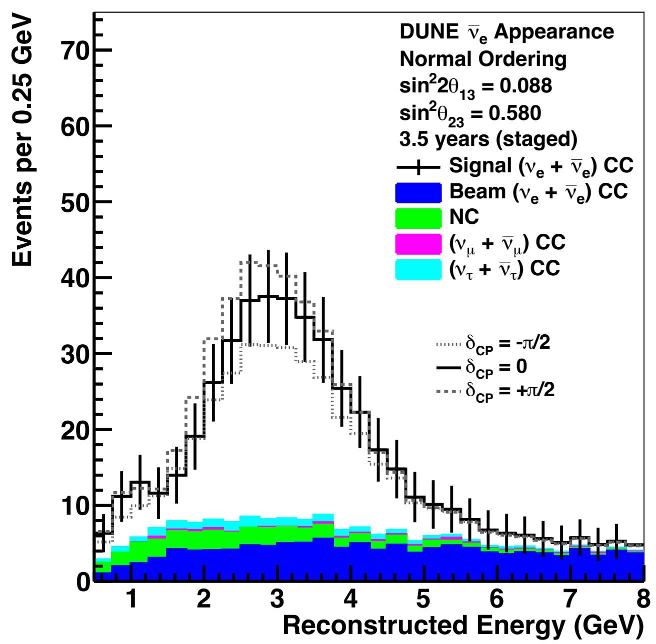
- Select a sample of data
- Get the first iteration of a Monte-Carlo sample
- Compute the posterior probability
- Save the step $s=0$
- Go to the next step $s=1$
- Recompute MC histogram
- Recompute the posterior probability
- Apply the Metropolis-Hasting theorem
- Continue until reaching ~10^8 steps

Parallelisation is mandatory!
- *Can run with CPU and GPU*
- *GPU only available on selected clusters (Wilson cluster @FNAL, others?)*
- *Take several days*

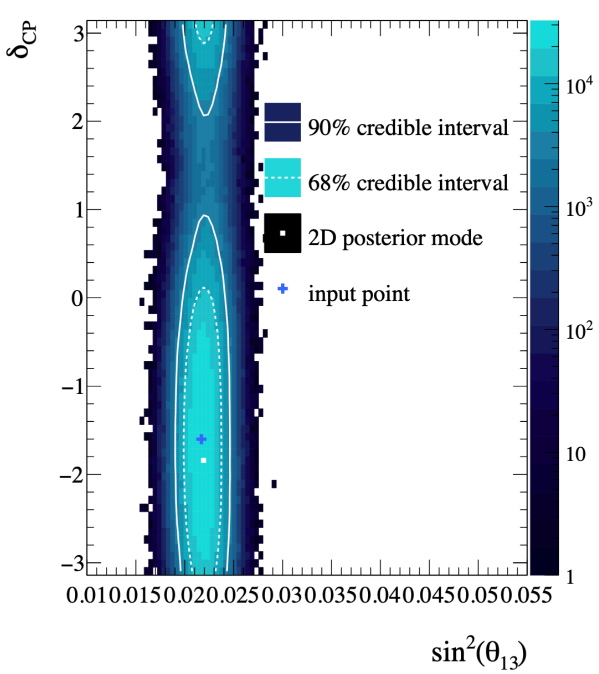
```
Important bottleneck! Can the process be faster?
=> identify the longest process and see which can be shorten
=> test other sampling algorithms
```

Steps by steps

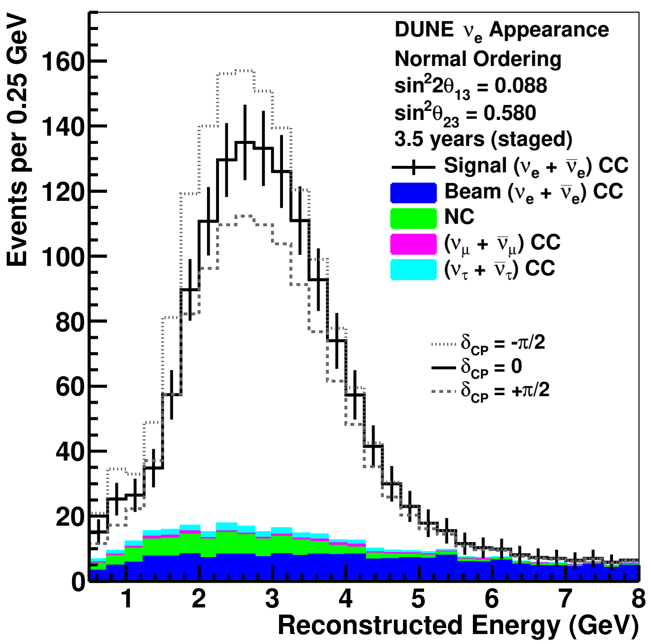


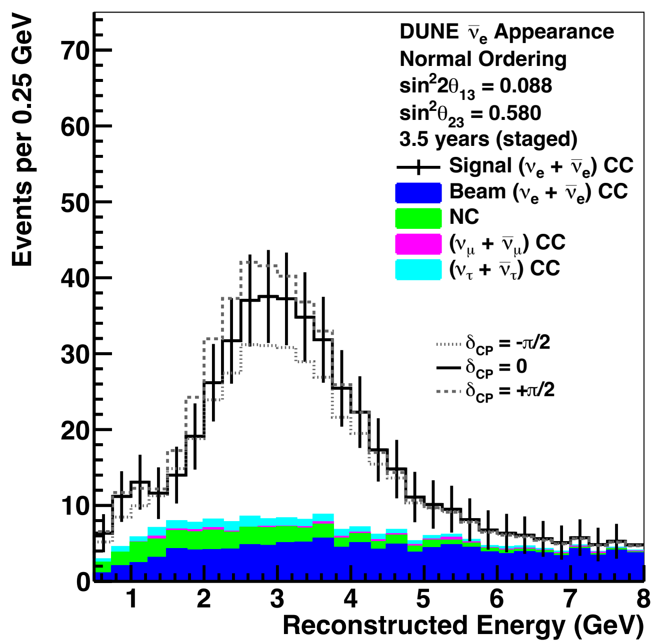


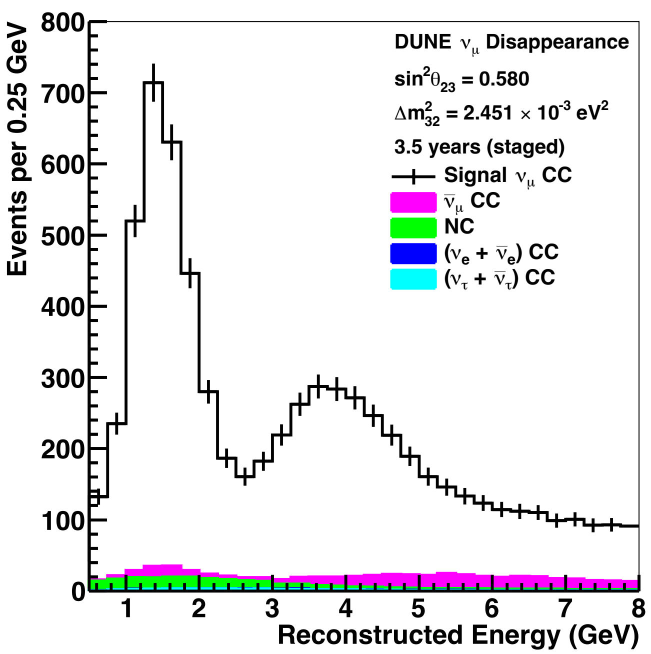
- Select a sample of data
- Get the first iteration of a Monte-Carlo sample
- Compute the posterior probability
- Save the step $s=0$
- Go to the next step $s=1$
- Recompute MC histogram
- Recompute the posterior probability
- Apply the Metropolis-Hasting theorem
- Continue until reaching ~10^8 steps
- The steps are samples from the posterior probability => fill a histogram and extract credible intervals

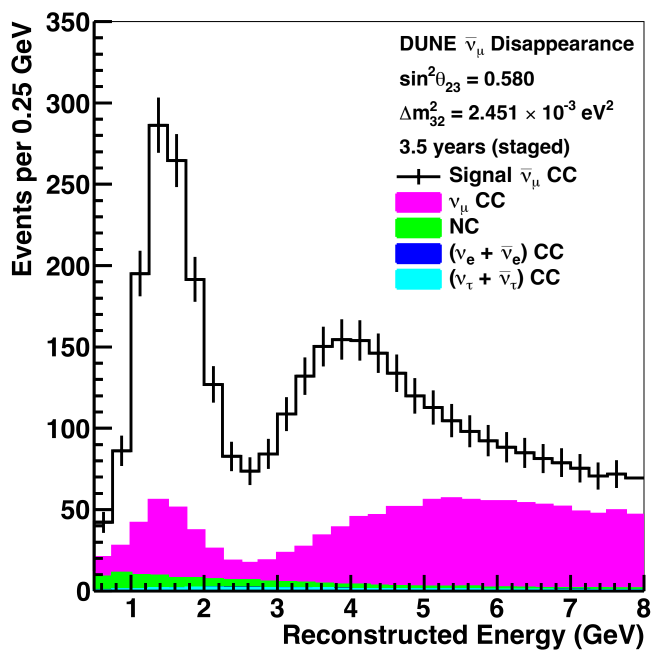


What else can be done?



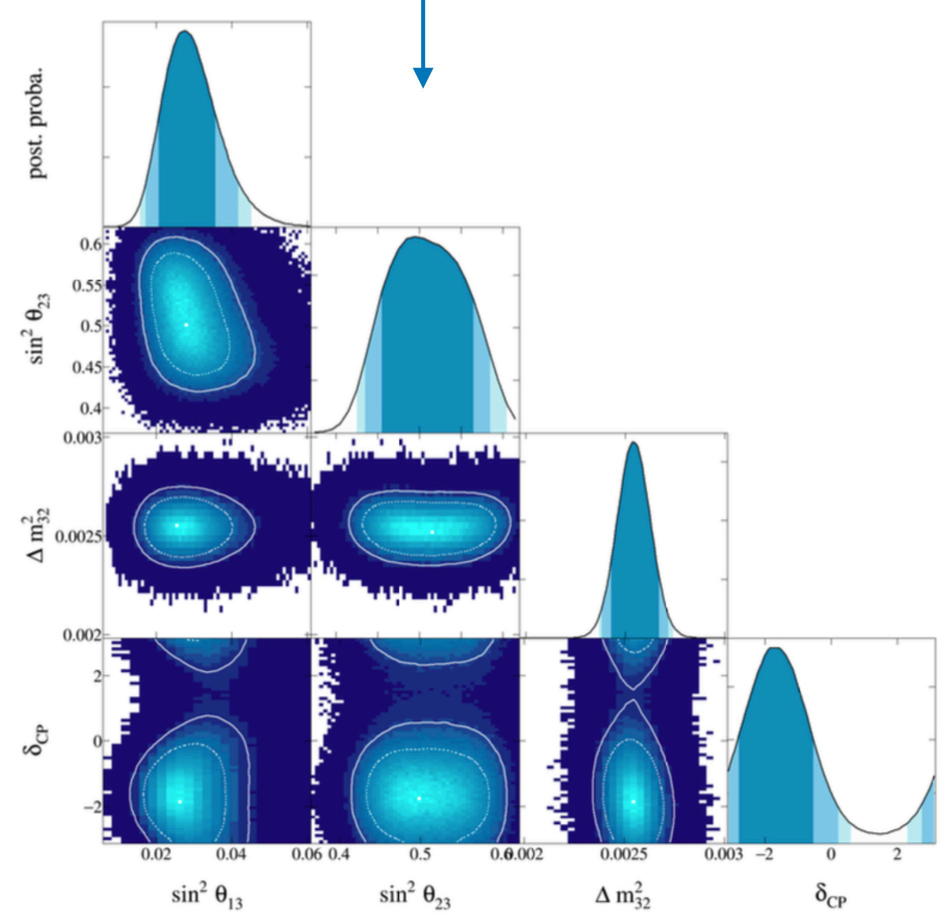




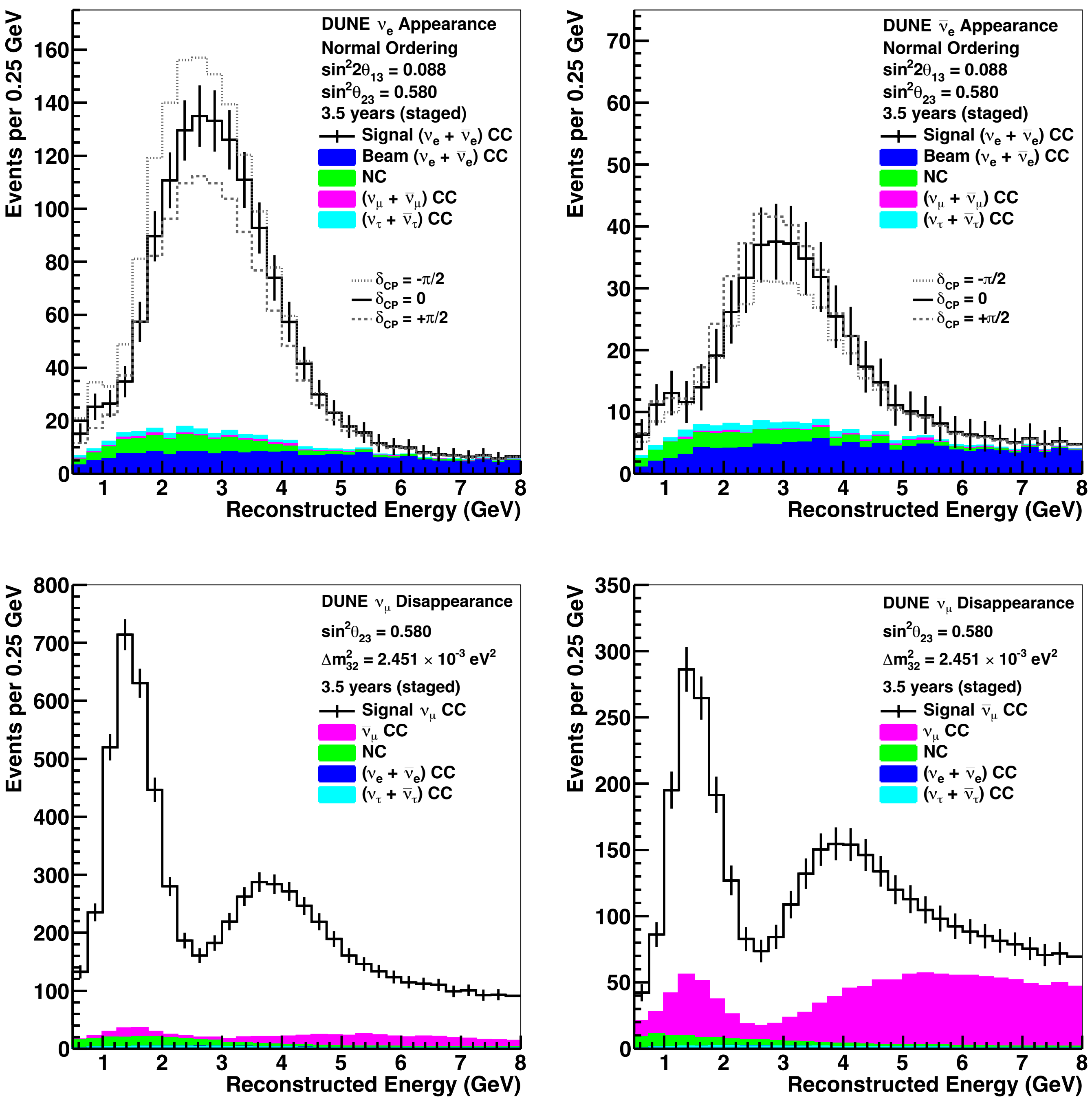


- Joint inference of $\overset{(-)}{\nu_e}$ appearance and $\overset{(-)}{\nu_\mu}$ disappearance

All 4 oscillation parameters are inferred at the same time

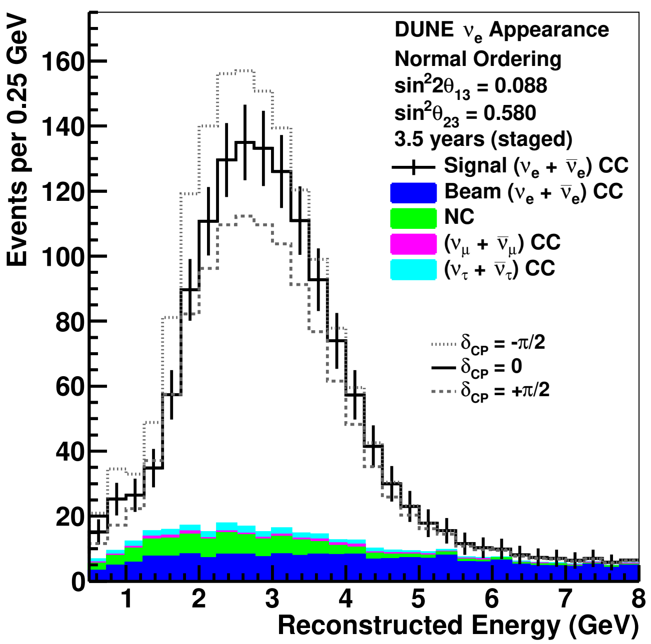


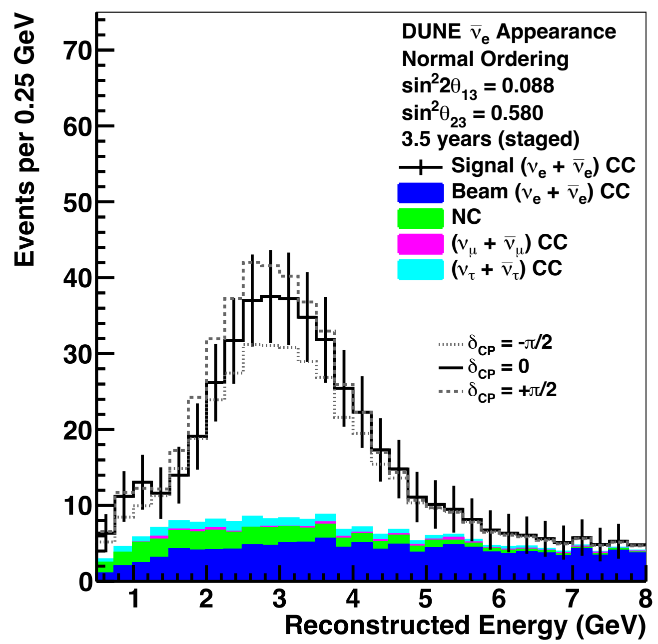
What else can be done?

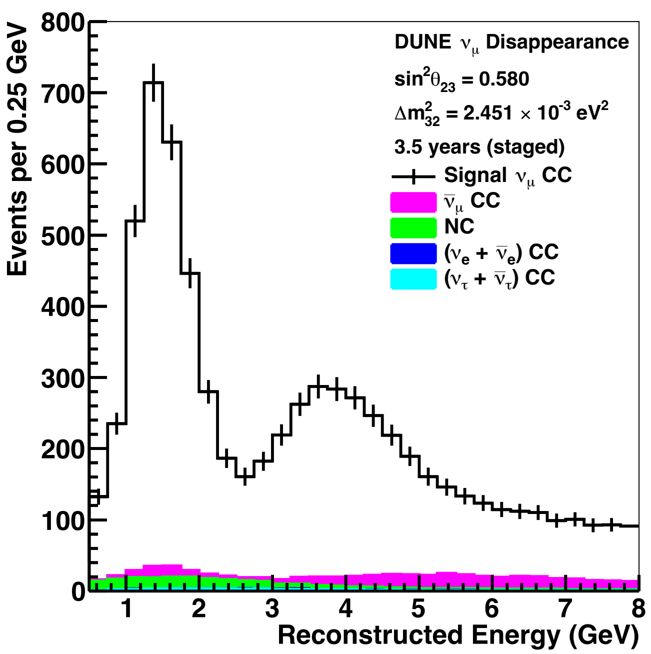


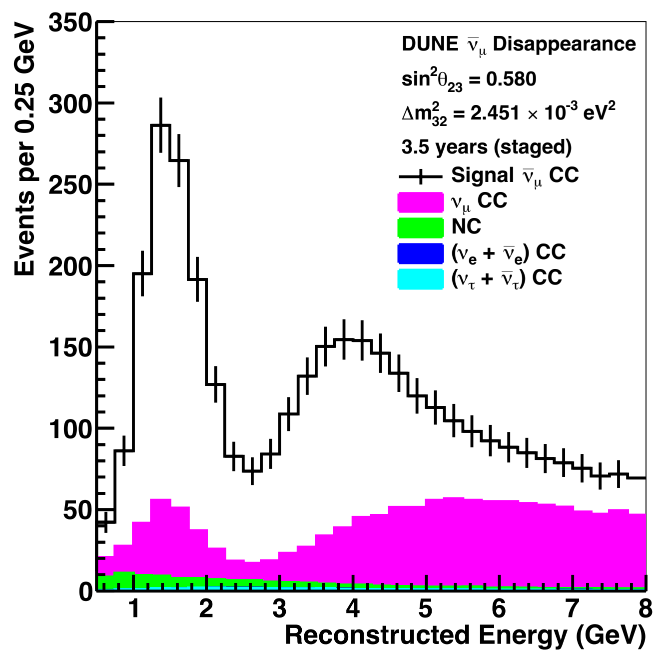
- Inclusion of near detector samples
- Work ongoing (Imperial London group)
- The near and far detectors samples are analysed at the same time
- Automatic constraint on flux and cross-section parameters
- No need to perform near to far extrapolation (assuming Gaussian errors)

What else can be done?





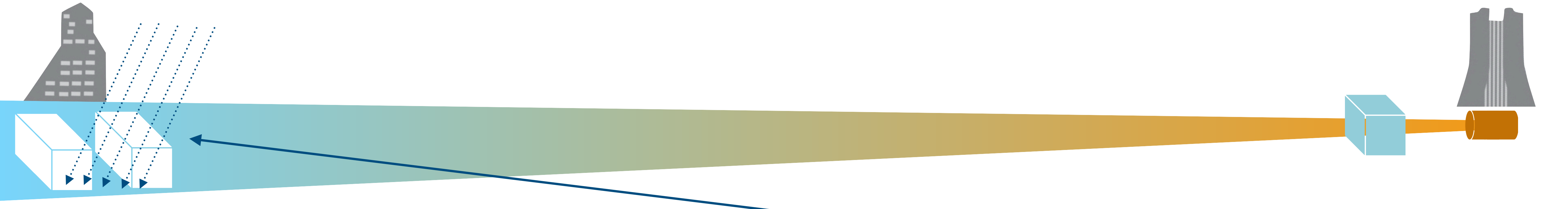




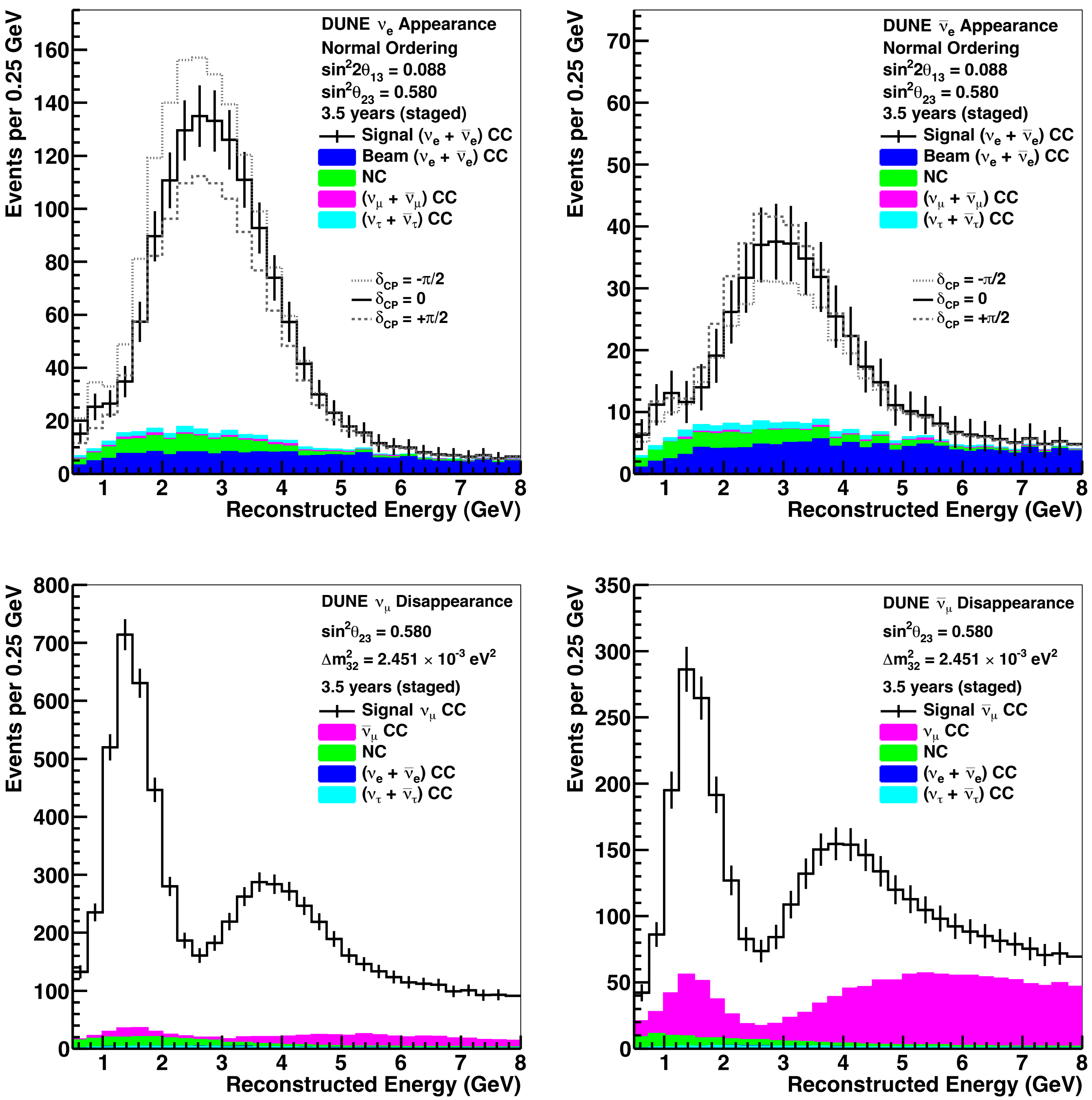
- A lot of performance study to do!
  - A more realistic description of the interaction parameters
  - More fined-grained samples (not only CC vs NC)
  - Study of adding information to the data samples (e.g. lepton angle?)
  - What are the correlations between the oscillation and systematics parameters?
  - What are the main detector systematics and can we think about it during the production phase?
  - Is the Earth matter model realist enough, does it have an impact?

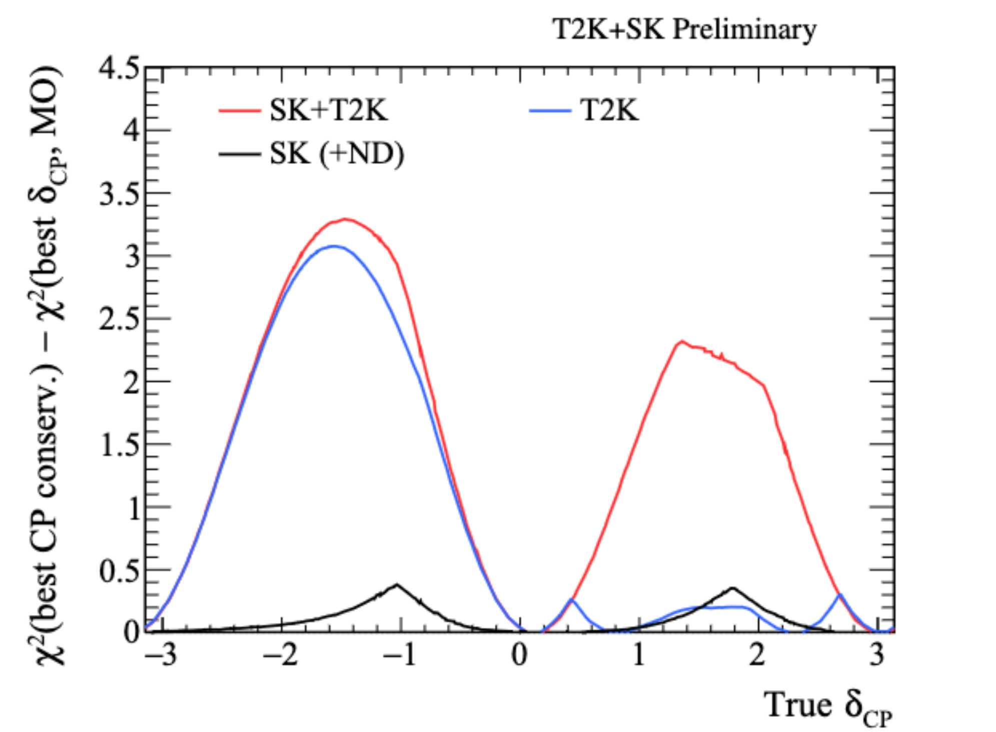
Interface with reconstruction & NIUWG groups

What else can be done?



- Beam + atmospheric fit
- How much more information is brought by adding atmospheric in the fit?
- Has not been studied for DUNE afaik
- Can see the impact on the T2K + SK fit





(a) Sensitivity as a function of true δ_{CP} and assuming unknown MO

What else can be done?

- **MaCh3 -> MaCh3+**
  - At the oscillation probability computation level:
    - Add cases with > 3ν (sterile neutrinos, non-unitarity)
    - Sample directly the mixing matrix elements $U_{\alpha i}$ instead of the rotation angles and mass splittings?

- **Multi-experiment inference**
  - Prepare the inclusion of samples from different experiment
  - Different flux and cross-section generators => how to find a common format?
  - Solar + atmo + beam fit?

- **A frequentist framework?**
  - Comparisons, cross-checks, validation of results…