UNIVERSITÄT HEIDELBERG
ZUKUNFT
SEIT 1386

IRN Terascale, 26.10.2023

Precision-Machine Learning for the Matrix Element Method

T. Heimel, **N. Huetsch**, R. Winterhalder, T. Plehn, A. Butter
arXiv: 2310.07752

SPONSORED BY THE
Federal Ministry
of Education
and Research

Building on: A. Butter, T. Heimel, T. Martini, S. Peitzsch, T. Plehn: arXiv:2210.00019

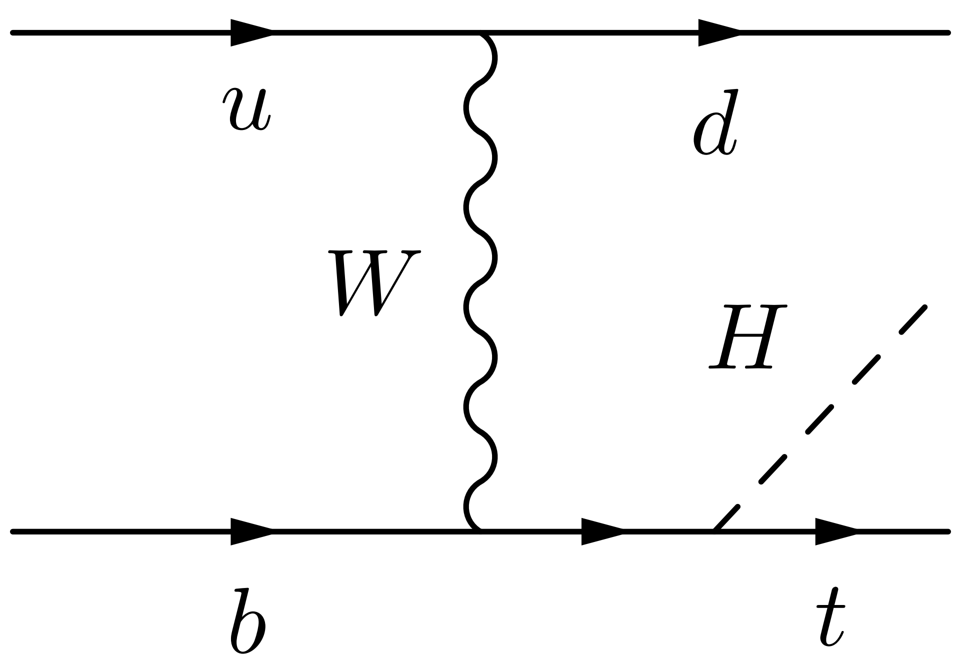
The physics problem

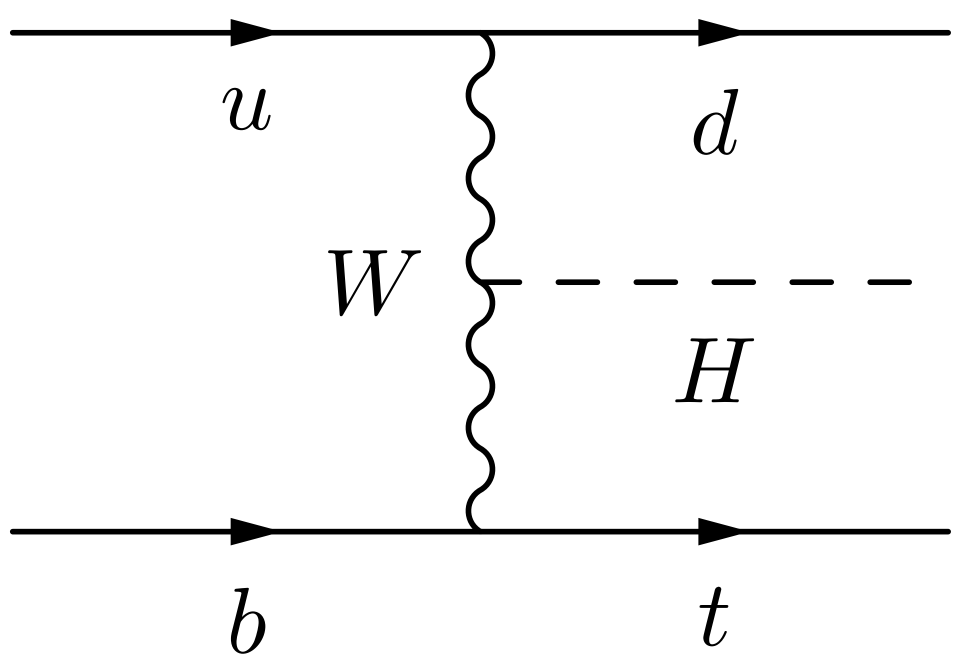
The problem: Measuring a CP-phase in the top Yukawa coupling

$$\mathscr{L}_{t\bar{t}H} = -\frac{y_t}{\sqrt{2}}\left[\cos\alpha\,\bar{t}t + \frac{2}{3}i\sin\alpha\,\bar{t}\gamma_5 t\right]H$$

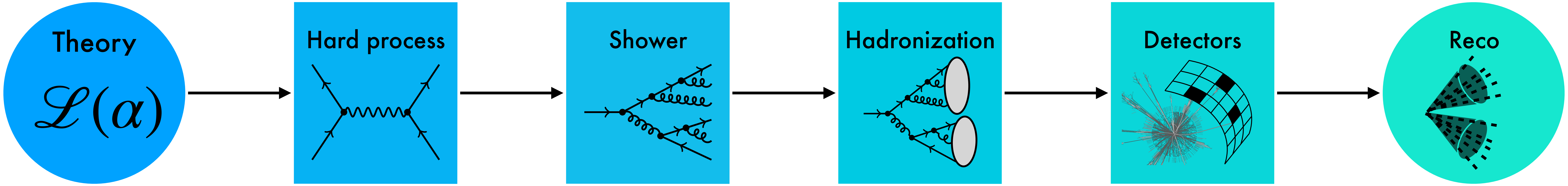
The process: Associated single-top and Higgs production

$$pp \to tHj \to (bjj)\,(\gamma\gamma)\,j \;+\; \text{ISR Jets}$$



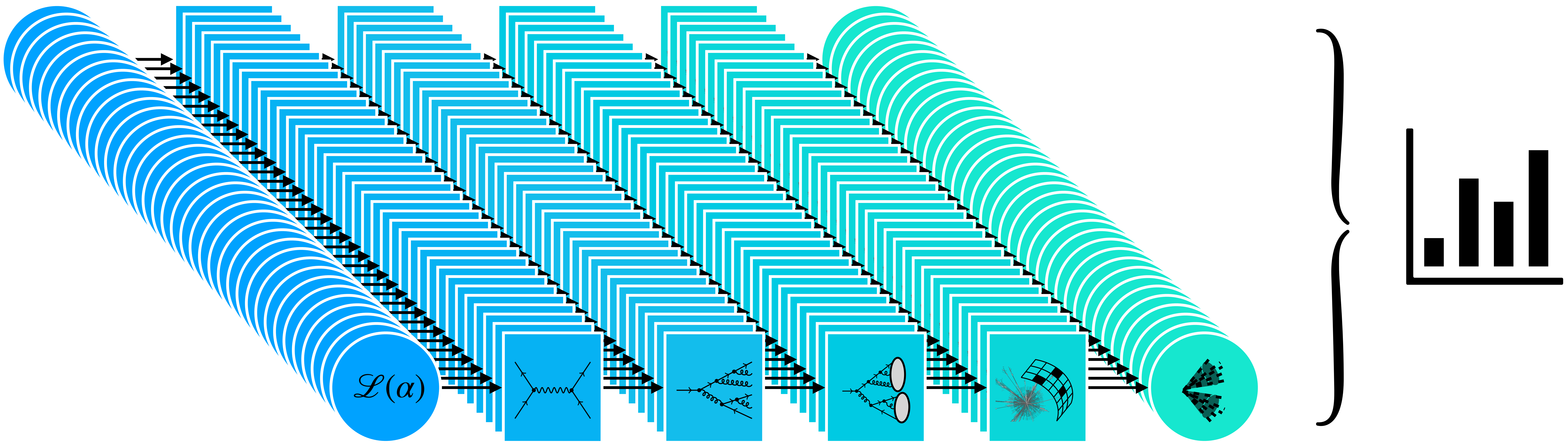


From Theory to Experiment in LHC Physics



Each event undergoes reconstruction

From Theory to Experiment in LHC Physics



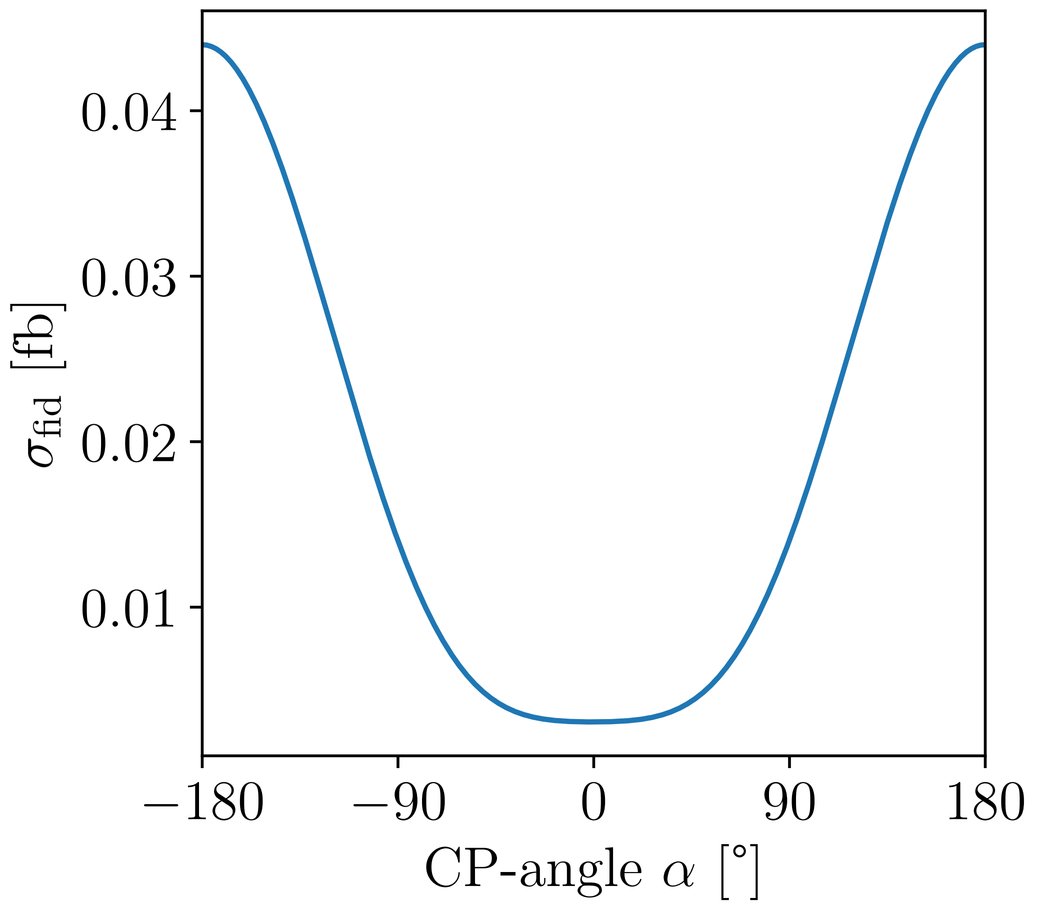
Event samples are combined into observable histogram

Why is this a problem here?

Cross-section:

- very small

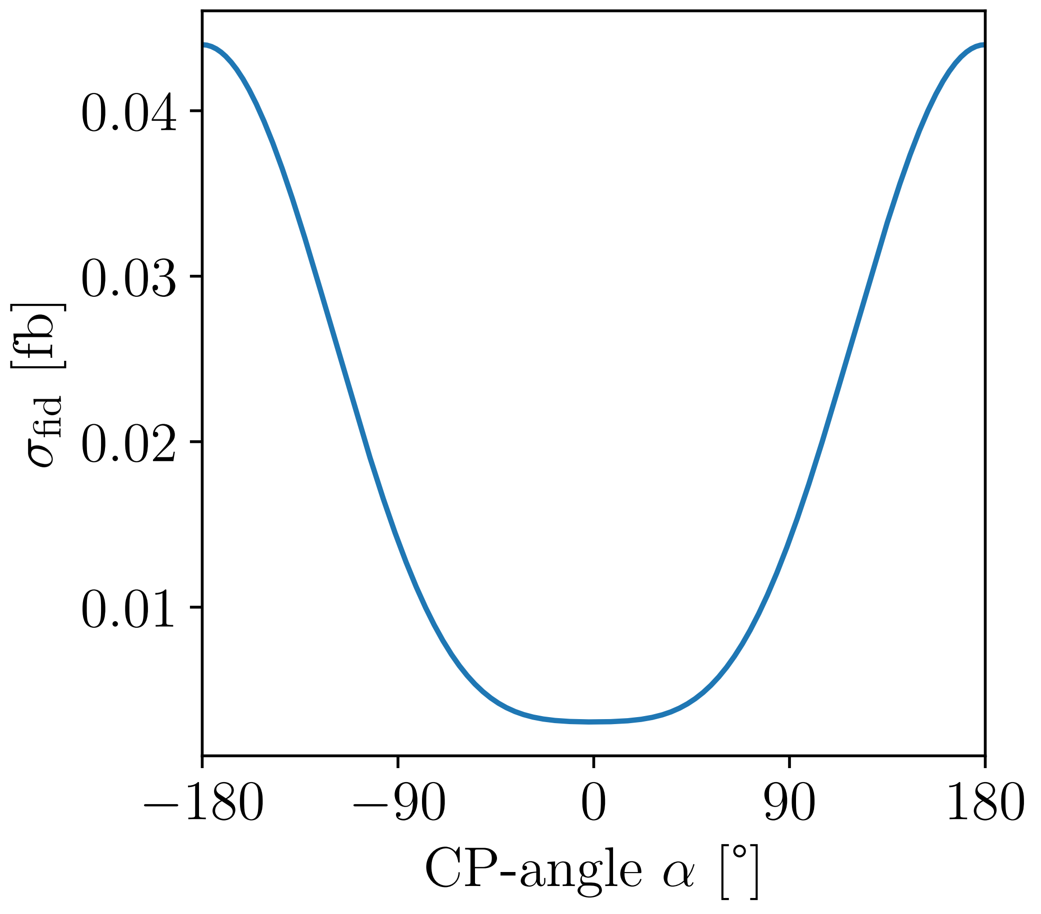
- insensitive to variations in α



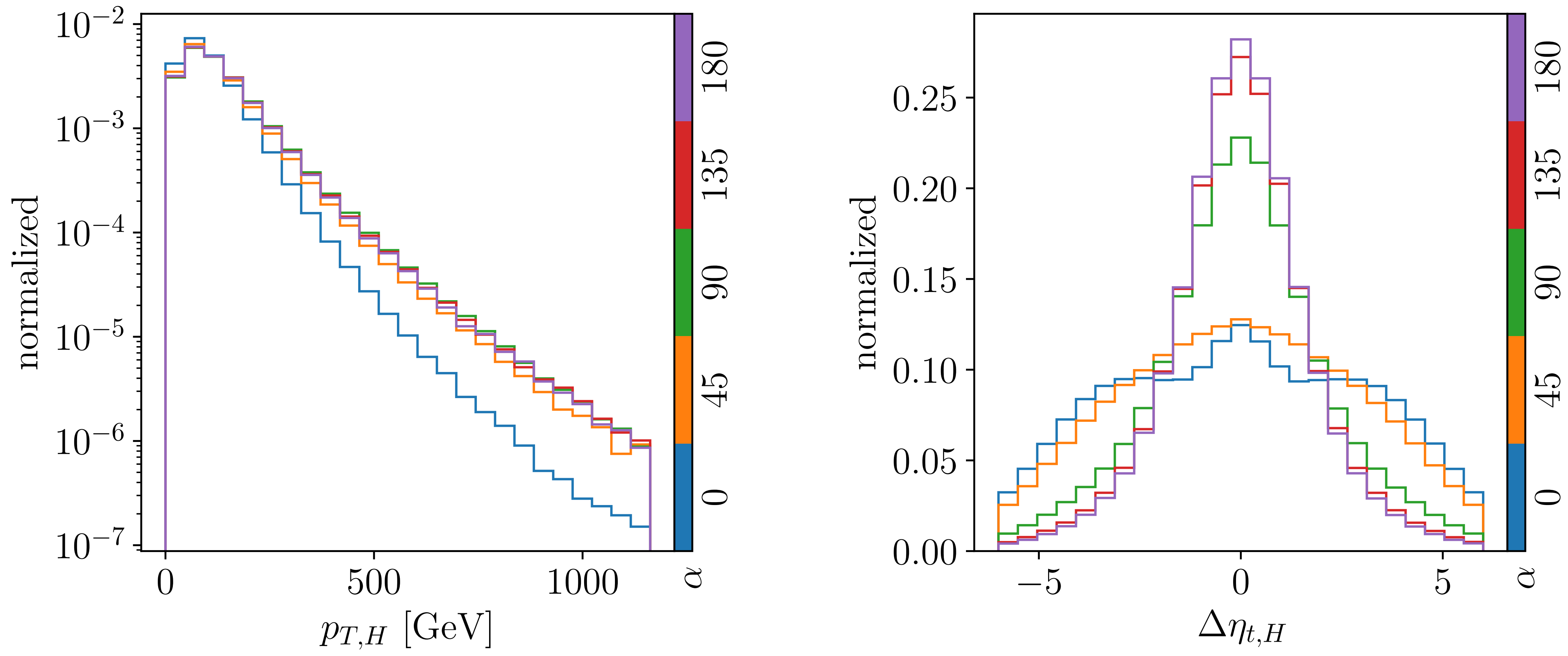
Why is this a problem here?

Cross-section:

- very small
- insensitive to variations in α



Hard-scattering $p(x_{hard} | \alpha)$ kinematics are sensitive

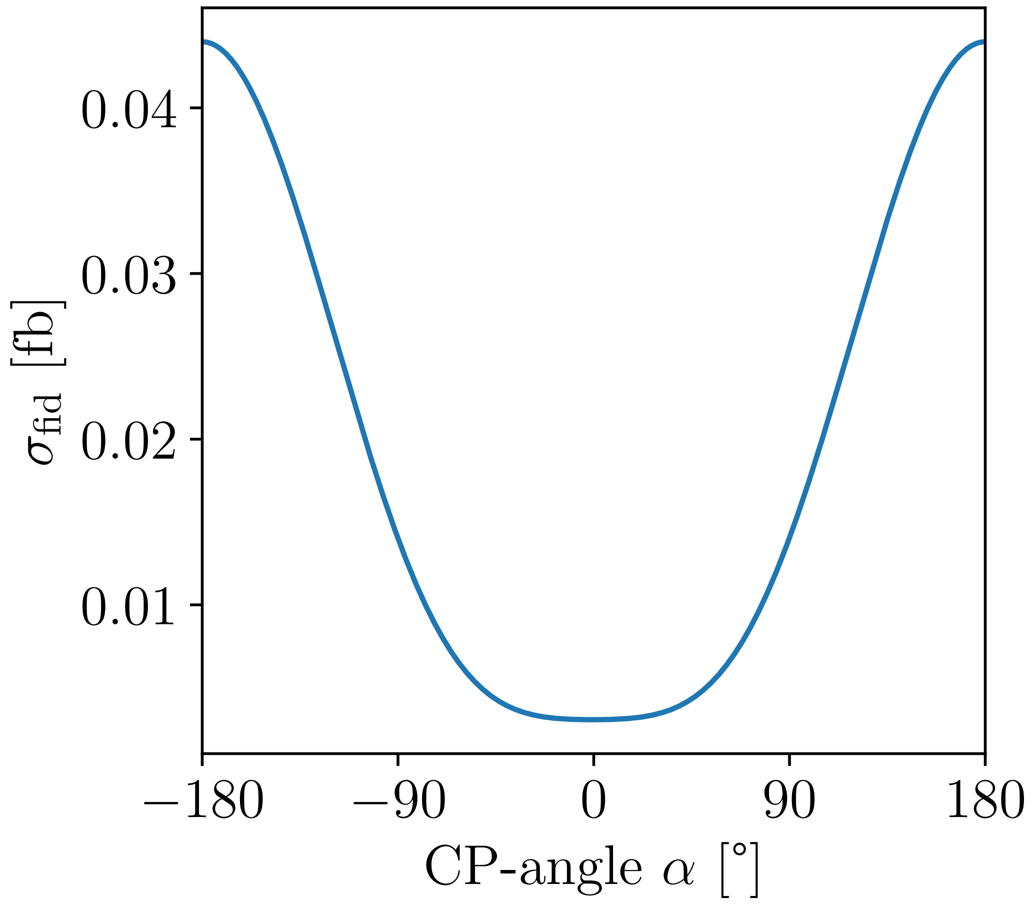


[Figures taken from Butter et al arXiv:2210.00019]

Why is this a problem here?

Cross-section:

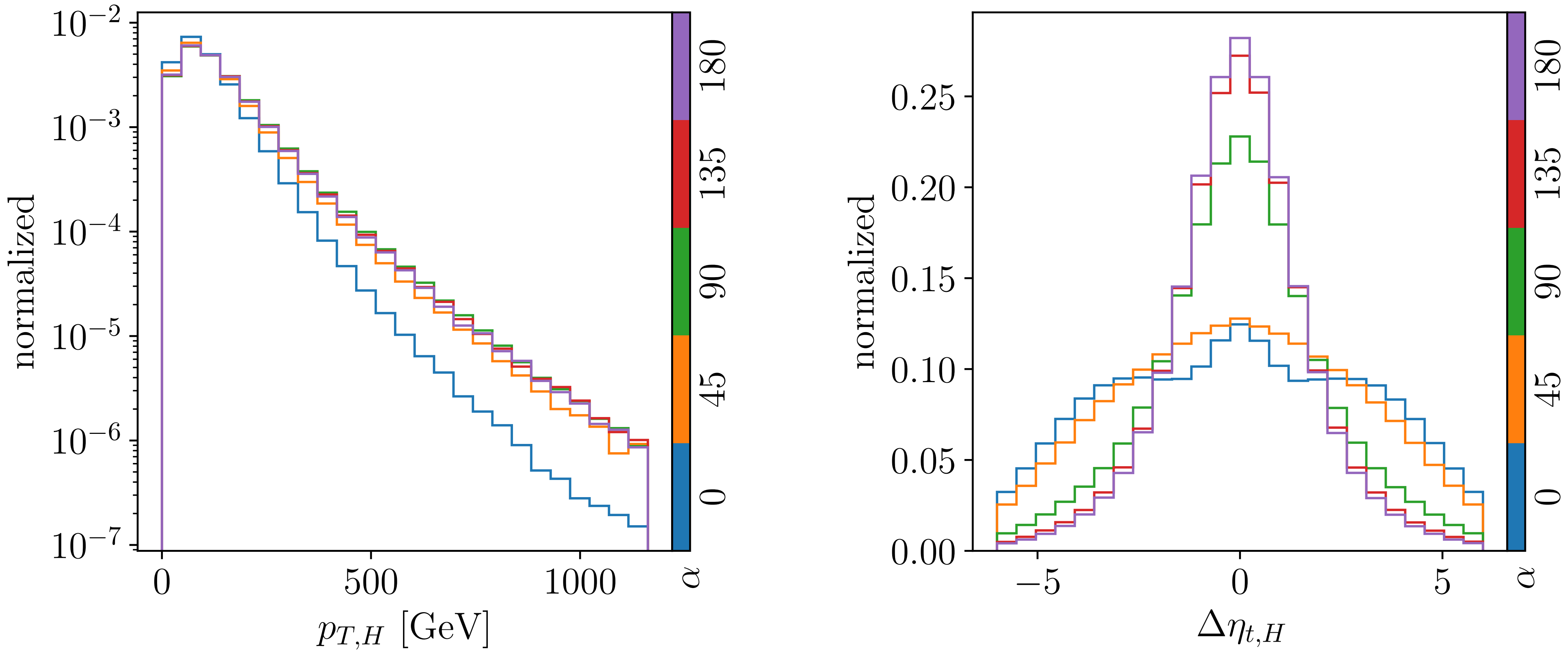
- very small
- insensitive to variations in α



Hard-scattering $p(x_{hard} | \alpha)$ kinematics are sensitive

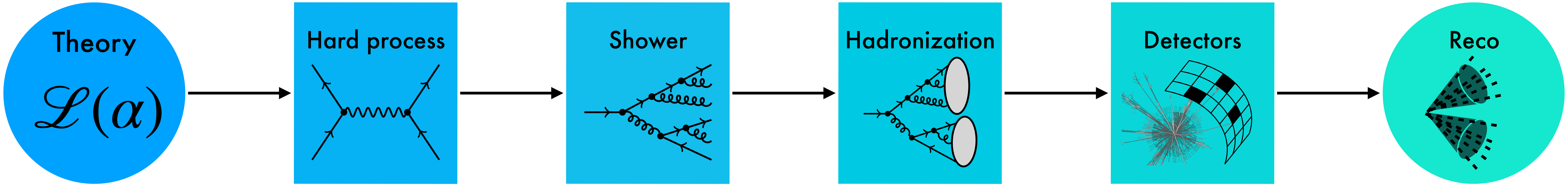
Need analysis method based on **kinematics**

Likelihood ratio ideal test statistic according to Newman-Pearson lemma



[Figures taken from Butter et al arXiv:2210.00019]

Going for the likelihood ratio

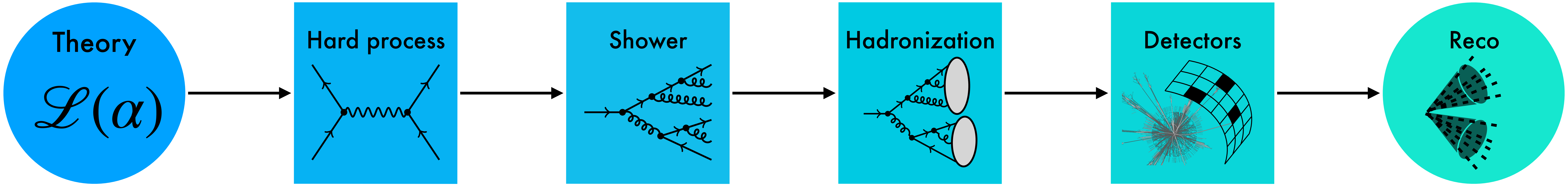


$$p(x_{hard} | \alpha)$$

At hard-scattering level likelihood given by differential cross-section $p(x_{hard} | \alpha) = \dfrac{1}{\sigma(\alpha)} \dfrac{d\sigma(\alpha)}{dx_{hard}}$

Unfortunately we do not measure at hard-scattering level

Going for the likelihood ratio

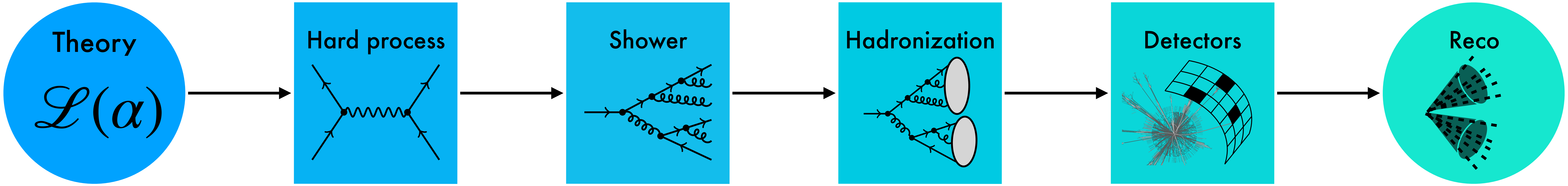


$p(x_{hard} | \alpha)$

$p(x_{reco} | \alpha)$

Need access to the likelihood at reconstruction level!

Going for the likelihood ratio

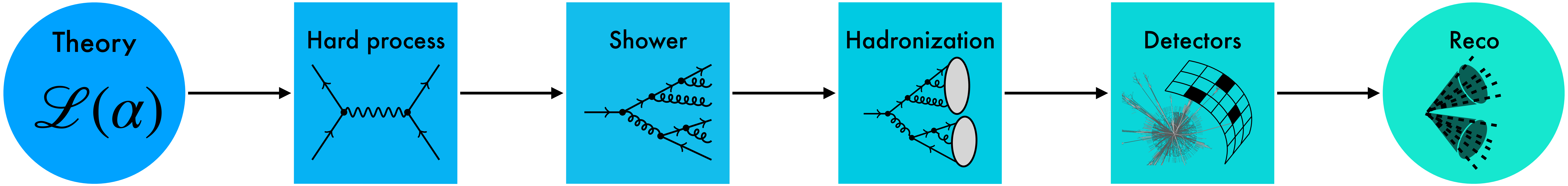


$p(x_{hard} | \alpha)$ $\qquad$ $p(x_{reco} | x_{hard}, \alpha)$ $\qquad$ $p(x_{reco} | \alpha)$

Hard-scattering and reconstruction linked by forward transfer probability

Forward transfer probability not known, encoded implicitly in forward simulation chain

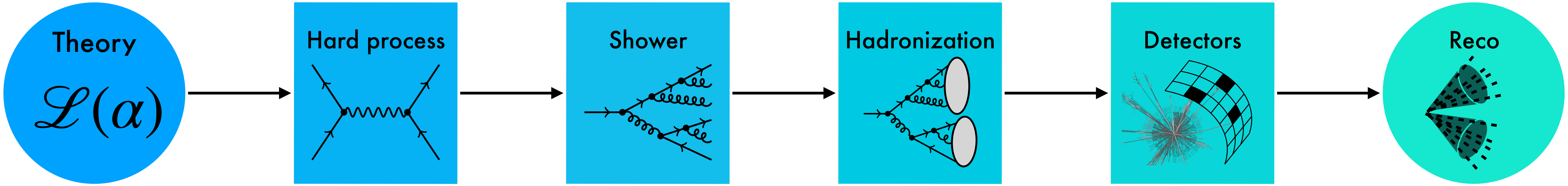
Going for the likelihood ratio



$$\int dx_{hard} \quad p(x_{hard}\,|\,\alpha) \quad \underbrace{\hphantom{xxxxxxxxxxxx}}_{p(x_{reco}\,|\,x_{hard},\,\alpha)} \quad p(x_{reco}\,|\,\alpha)$$

Integrate over all possible hard-scattering configurations

Going for the likelihood ratio

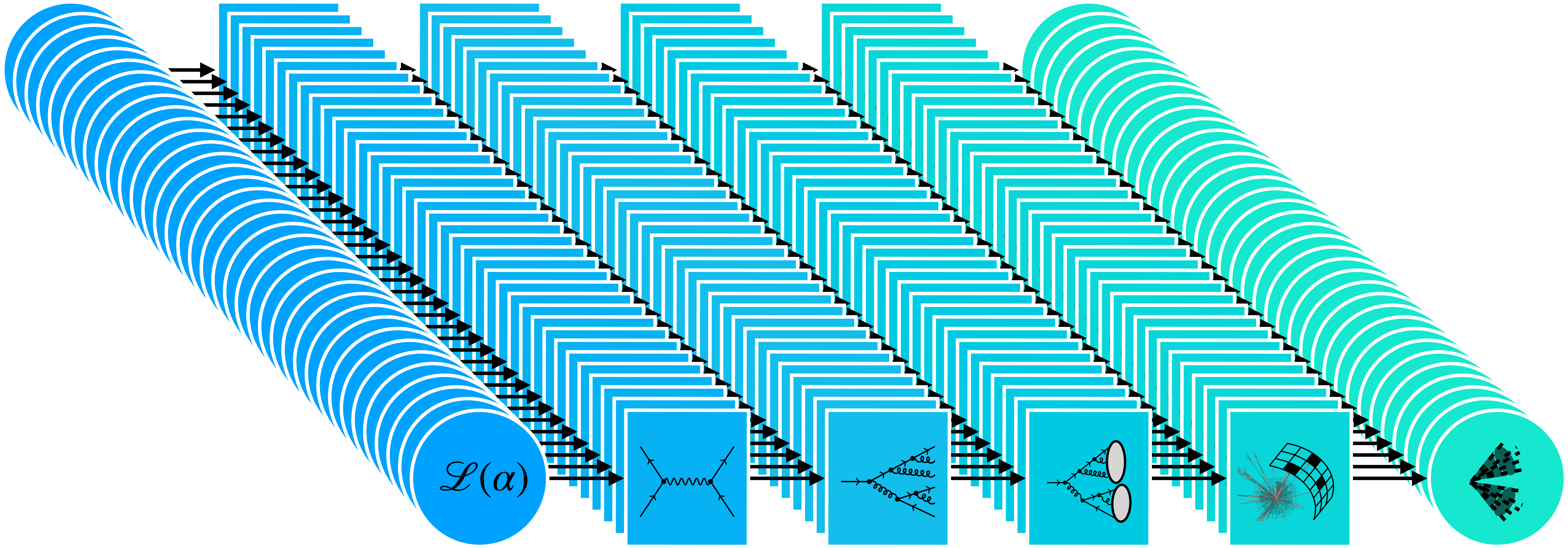


$$\int dx_{hard} \quad p(x_{hard}\,|\,\alpha) \quad p(x_{reco}\,|\,x_{hard},\alpha)\,\epsilon(x_{hard},\alpha) \;=\; p(x_{reco}\,|\,\alpha)$$

Include an efficiency term to account for acceptance of events

Encodes the probability that hard-scattering level configuration will pass reco level cuts

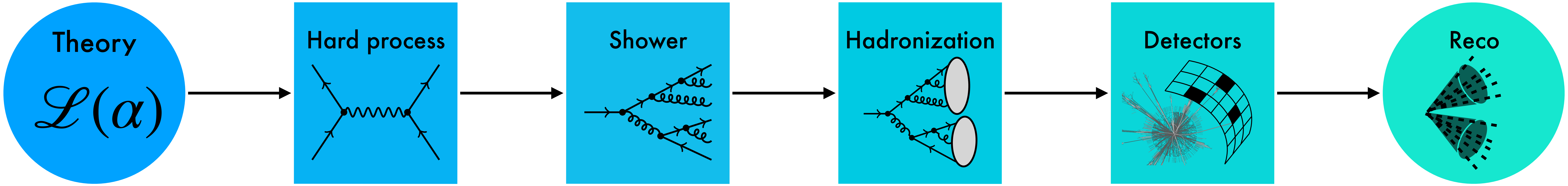
Going for the likelihood ratio



$$\prod_{events\ i} p(x_{reco,i}\,|\,\alpha)$$

Event likelihoods are combined into sample likelihoods

The Matrix Element Method



$$\int dx_{hard} \quad p(x_{hard}\,|\,\alpha) \quad p(x_{reco}\,|\,x_{hard}, \alpha)\,\epsilon(x_{hard}, \alpha) \;=\; p(x_{reco}\,|\,\alpha)$$

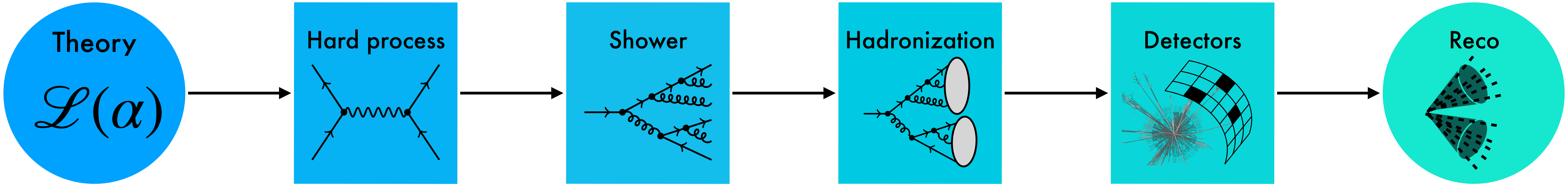
+++ Unbinned and multivariate by design

+++ Optimal use of information derived from Newman-Pearson lemma

— — Transfer probability and efficiency not known

— — Integral numerically very challenging

The Matrix Element Method



$$\int dx_{hard} \qquad p(x_{hard}\,|\,\alpha) \qquad p(x_{reco}\,|\,x_{hard}, \alpha)\, \epsilon(x_{hard}, \alpha) \; = \; p(x_{reco}\,|\,\alpha)$$

+++ Unbinned and multivariate by design

+++ Optimal use of information derived from Newman-Pearson lemma

— — ~~Transfer probability and efficiency not known~~ **USE MACHINE LEARNING**

— — ~~Integral numerically very challenging~~ **USE MACHINE LEARNING**

The Transfer Network

Intractable

$$\int dx_{hard}\ p(x_{hard}|\alpha)\ \boxed{p(x_{reco}|x_{hard},\alpha)}\ \epsilon(x_{hard},\alpha) \ = \ p(x_{reco}|\alpha)$$

The Transfer Network

Intractable

$$\int dx_{hard}\; p(x_{hard}|\alpha)\; \boxed{p(x_{reco}|x_{hard})}\; \epsilon(x_{hard},\alpha) \;=\; p(x_{reco}|\alpha)$$

The Transfer Network

Intractable

$$\int dx_{hard}\; p(x_{hard}|\alpha)\; \boxed{p(x_{reco}|x_{hard})}\; \epsilon(x_{hard},\alpha) \;=\; p(x_{reco}|\alpha)$$

Transfer probability is analytically intractable

Transfer can be simulated to generate paired data x_{hard} , x_{reco}

Generative Neural Network to encode the transfer probability

The Acceptance Network

$$\int dx_{hard}\; p(x_{hard}|\alpha)\; p(x_{reco}|x_{hard})\; \overset{\text{Unknown}}{\boxed{\epsilon(x_{hard}, \alpha)}} = p(x_{reco}|\alpha)$$

The Acceptance Network

Unknown

$$\int dx_{hard}\; p(x_{hard}|\alpha)\; p(x_{reco}|x_{hard})\; \boxed{\epsilon(x_{hard})} = p(x_{reco}|\alpha)$$

The Acceptance Network

Unknown

$$\int dx_{hard}\; p(x_{hard}\,|\,\alpha)\; p(x_{reco}\,|\,x_{hard})\; \boxed{\epsilon(x_{hard})} = p(x_{reco}\,|\,\alpha)$$

Efficiency at hard-scattering level is unknown

Transfer can be simulated to generate labeled data $x_{hard} \rightarrow x_{reco}(x_{hard})$ ✅ ❌

Classifier Neural Network to encode the acceptance probability

The Sampling Network

$$p(x_{reco}|\alpha) = \int dx_{hard}\; p(x_{hard}|\alpha)\, p(x_{reco}|x_{hard})\, \epsilon(x_{hard})$$

The Sampling Network

$$
\begin{aligned}
p(x_{reco} | \alpha) &= \int dx_{hard}\; p(x_{hard} | \alpha)\; p(x_{reco} | x_{hard})\; \epsilon(x_{hard}) \\
&= \left\langle \frac{1}{q(x_{hard})} p(x_{hard} | \alpha)\; p(x_{reco} | x_{hard})\; \epsilon(x_{hard}) \right\rangle_{x_{hard} \sim q(x_{hard})}
\end{aligned}
$$

The Sampling Network

$$p(x_{reco} | \alpha) = \int dx_{hard}\; p(x_{hard} | \alpha)\; p(x_{reco} | x_{hard})\; \epsilon(x_{hard})$$

$$= \left\langle \frac{1}{q(x_{hard})} p(x_{hard} | \alpha)\; p(x_{reco} | x_{hard})\; \epsilon(x_{hard}) \right\rangle_{x_{hard} \sim q(x_{hard})}$$

Integral becomes trivial if : $q(x_{hard}) \propto p(x_{hard} | x_{reco}, \alpha) \epsilon(x_{hard})$

The Sampling Network

$$p(x_{reco}|\alpha) = \int dx_{hard}\ p(x_{hard}|\alpha)\ p(x_{reco}|x_{hard})\ \epsilon(x_{hard})$$

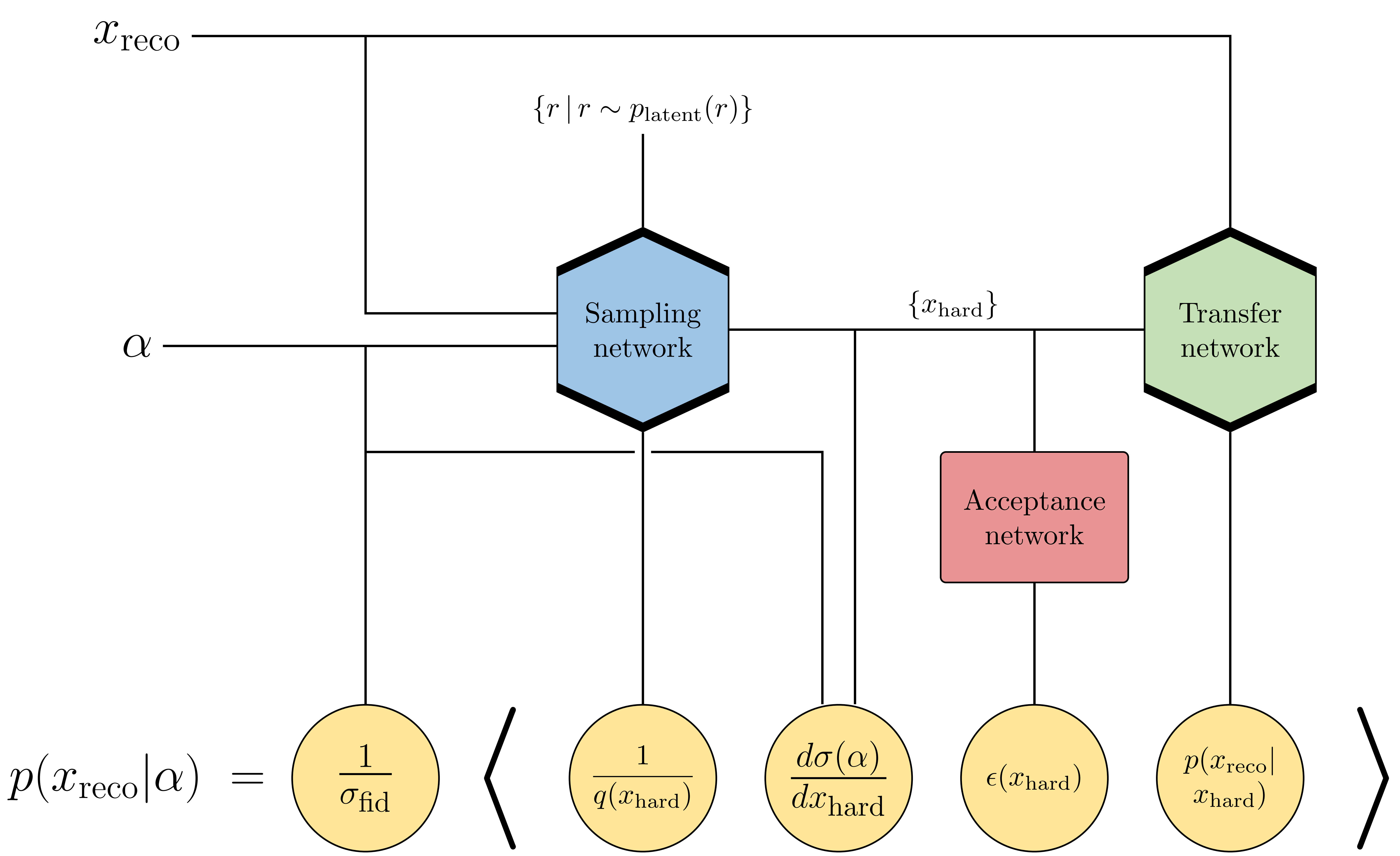
$$= \left\langle \frac{1}{q(x_{hard})} p(x_{hard}|\alpha)\ p(x_{reco}|x_{hard})\ \epsilon(x_{hard}) \right\rangle_{x_{hard}\sim q(x_{hard})}$$

Integral becomes trivial if : $q(x_{hard}) \propto p(x_{hard}|x_{reco}, \alpha)\epsilon(x_{hard})$

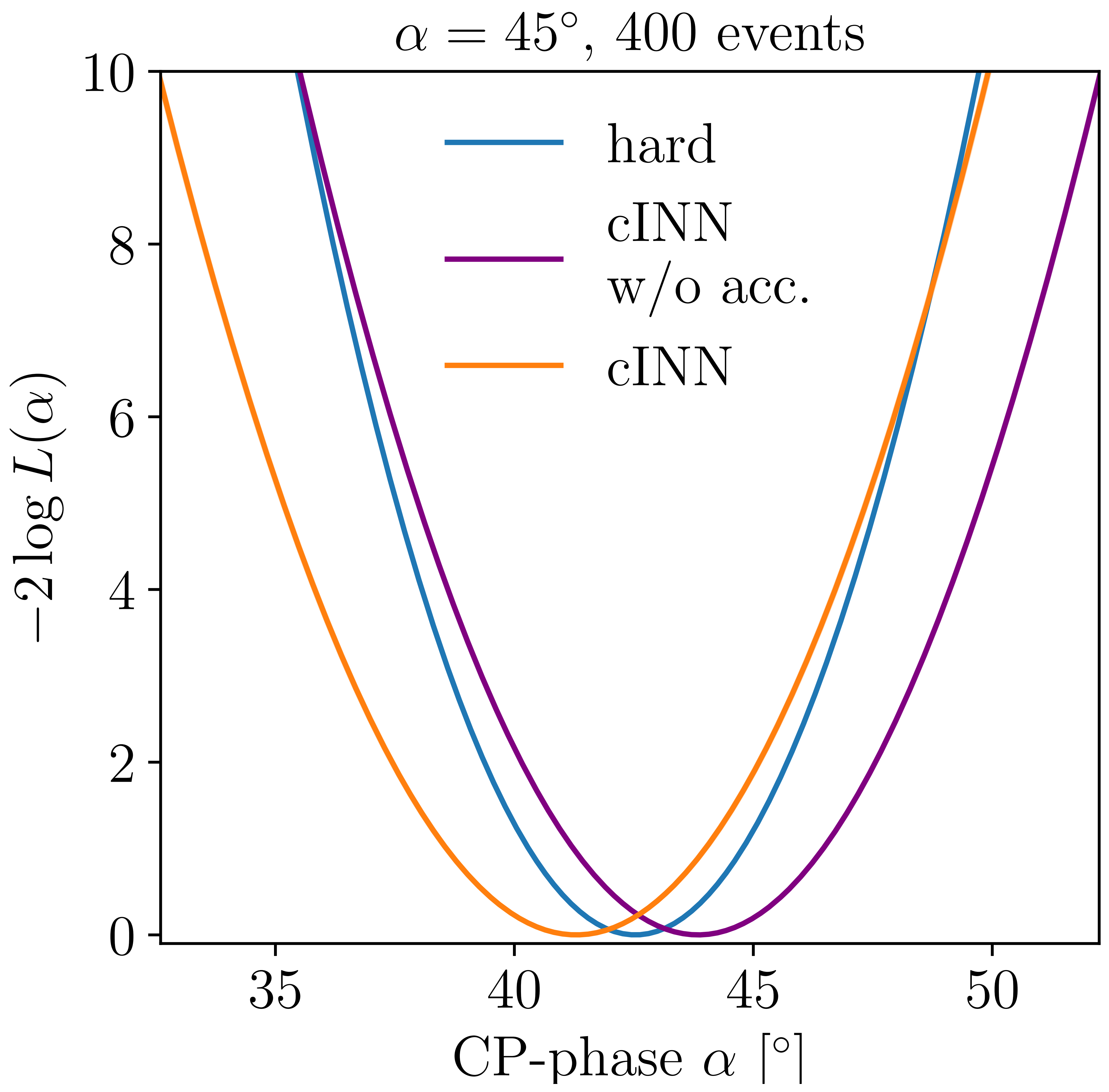
Generative Neural Network to encode sampling distribution

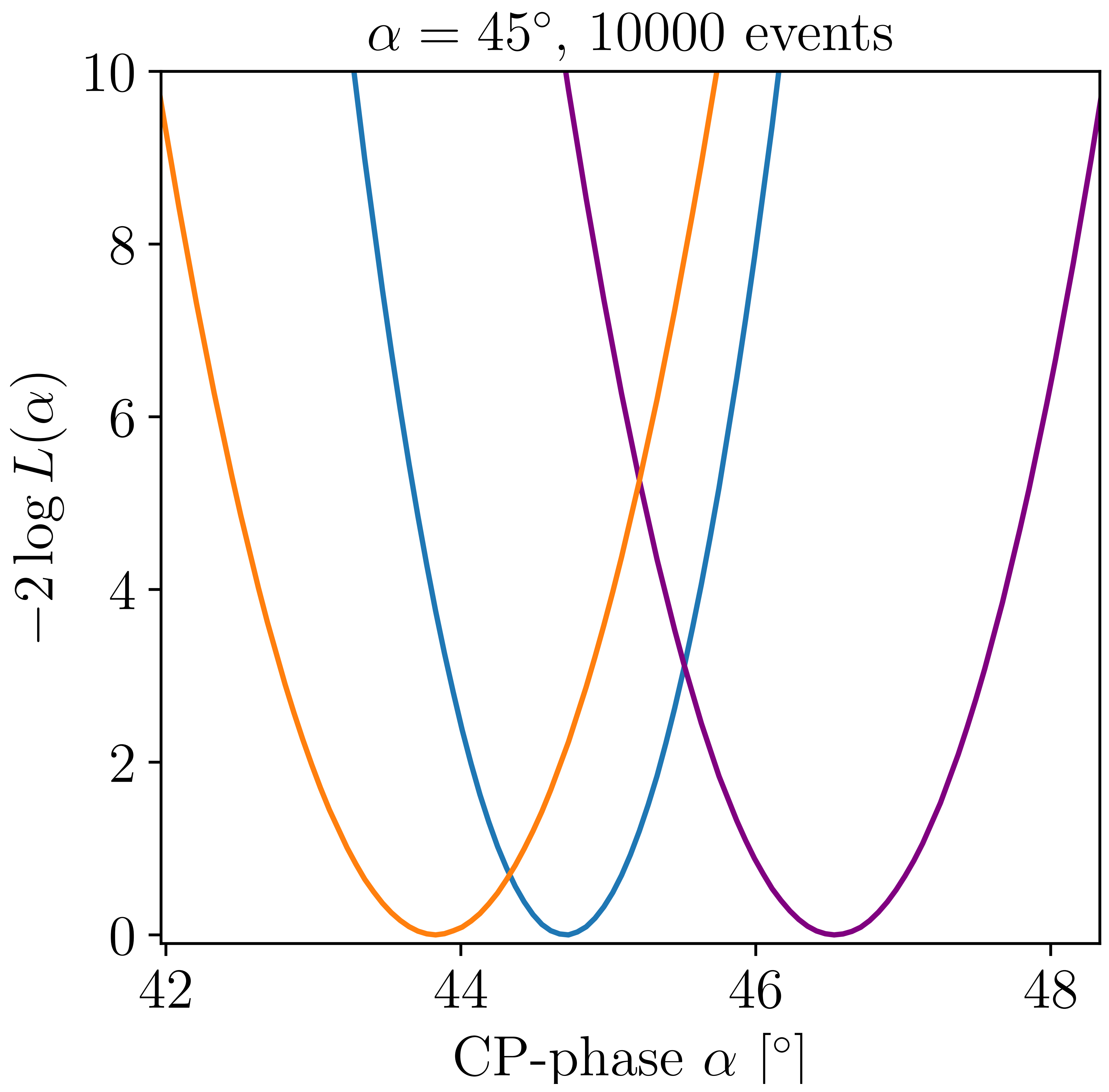
$$r \sim p_{latent}(r) \longleftrightarrow x_{hard}(r) \sim q_\phi(x_{hard})$$

Machine-learned MEM

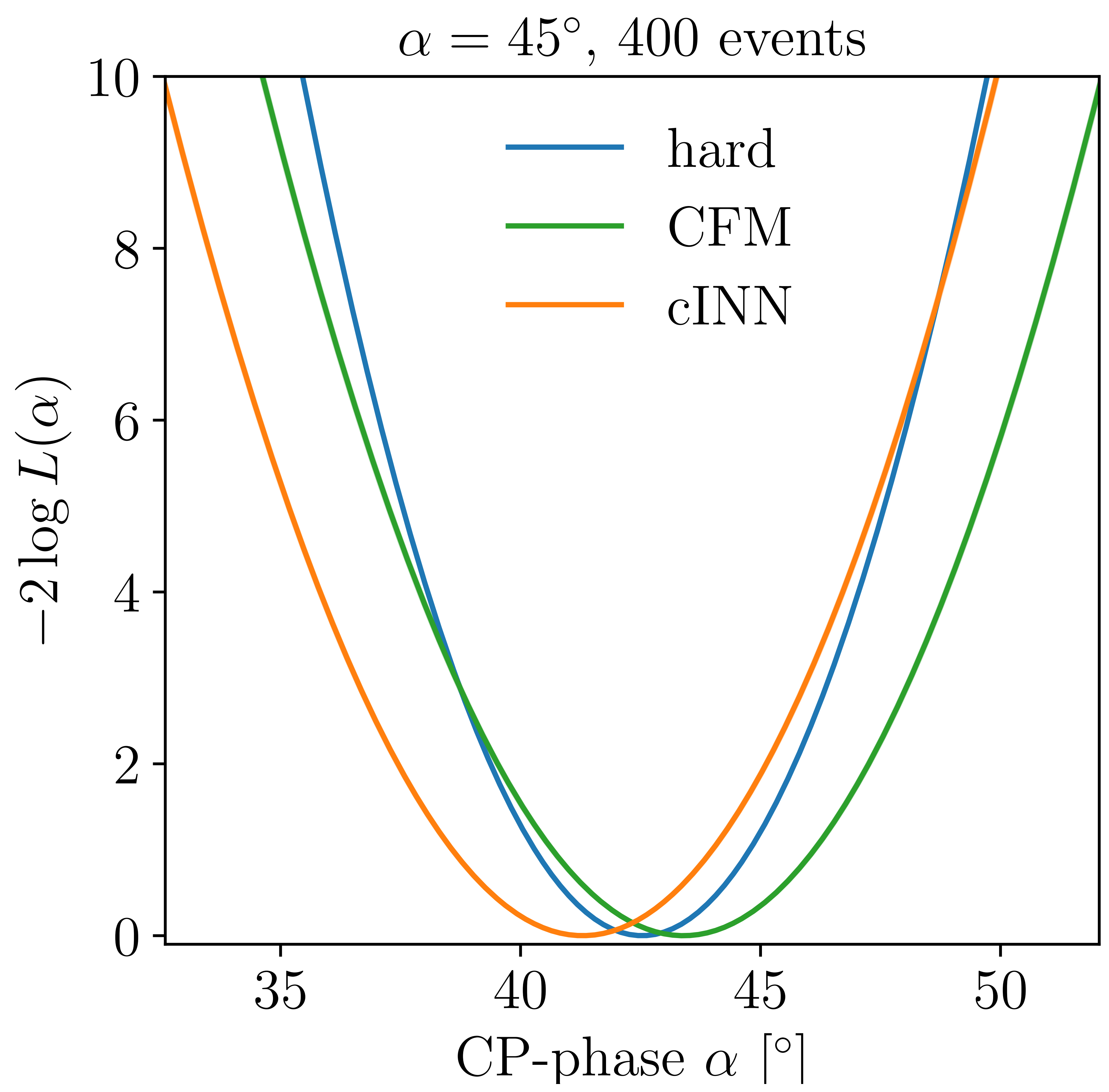


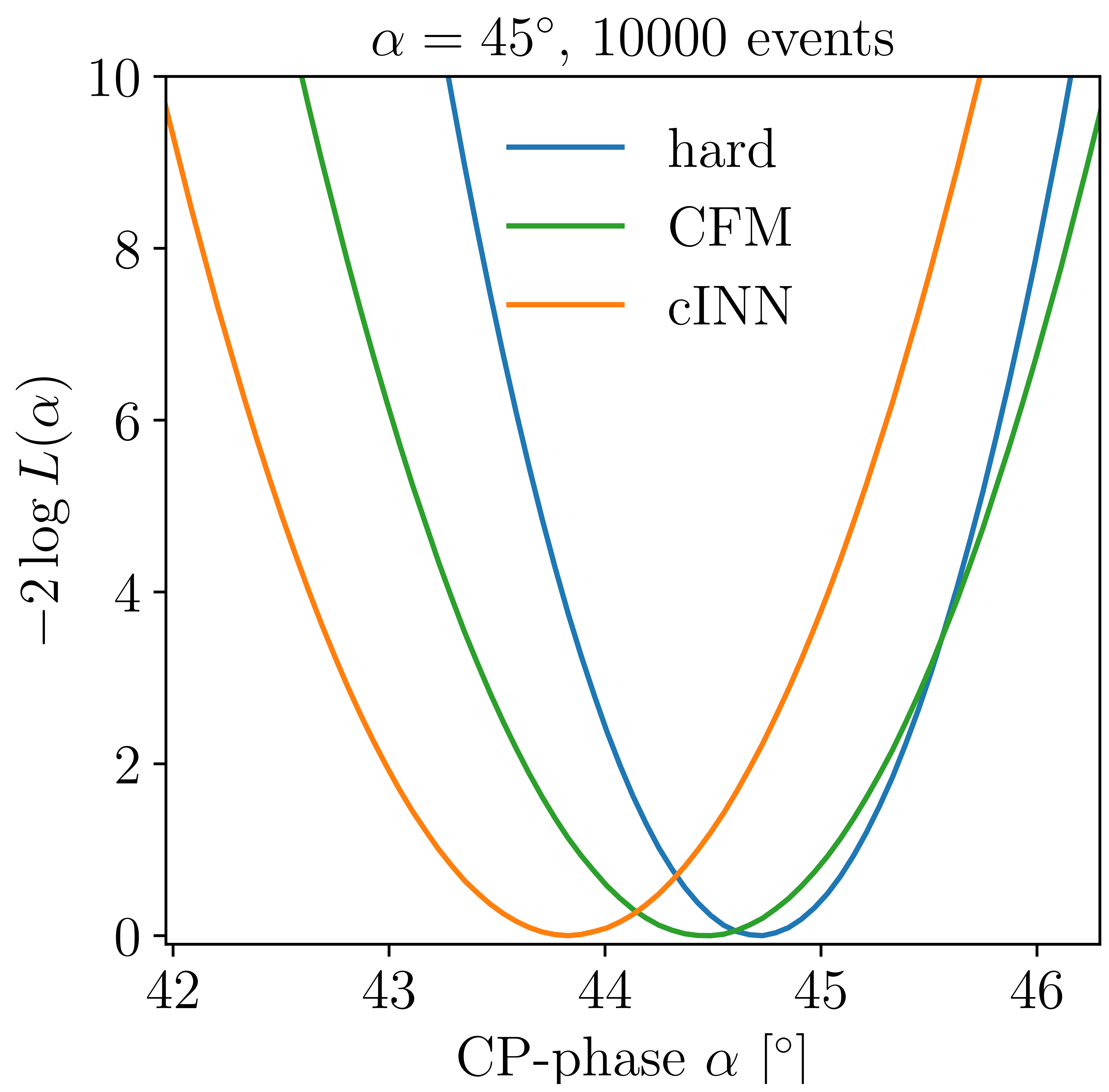
Baseline Results



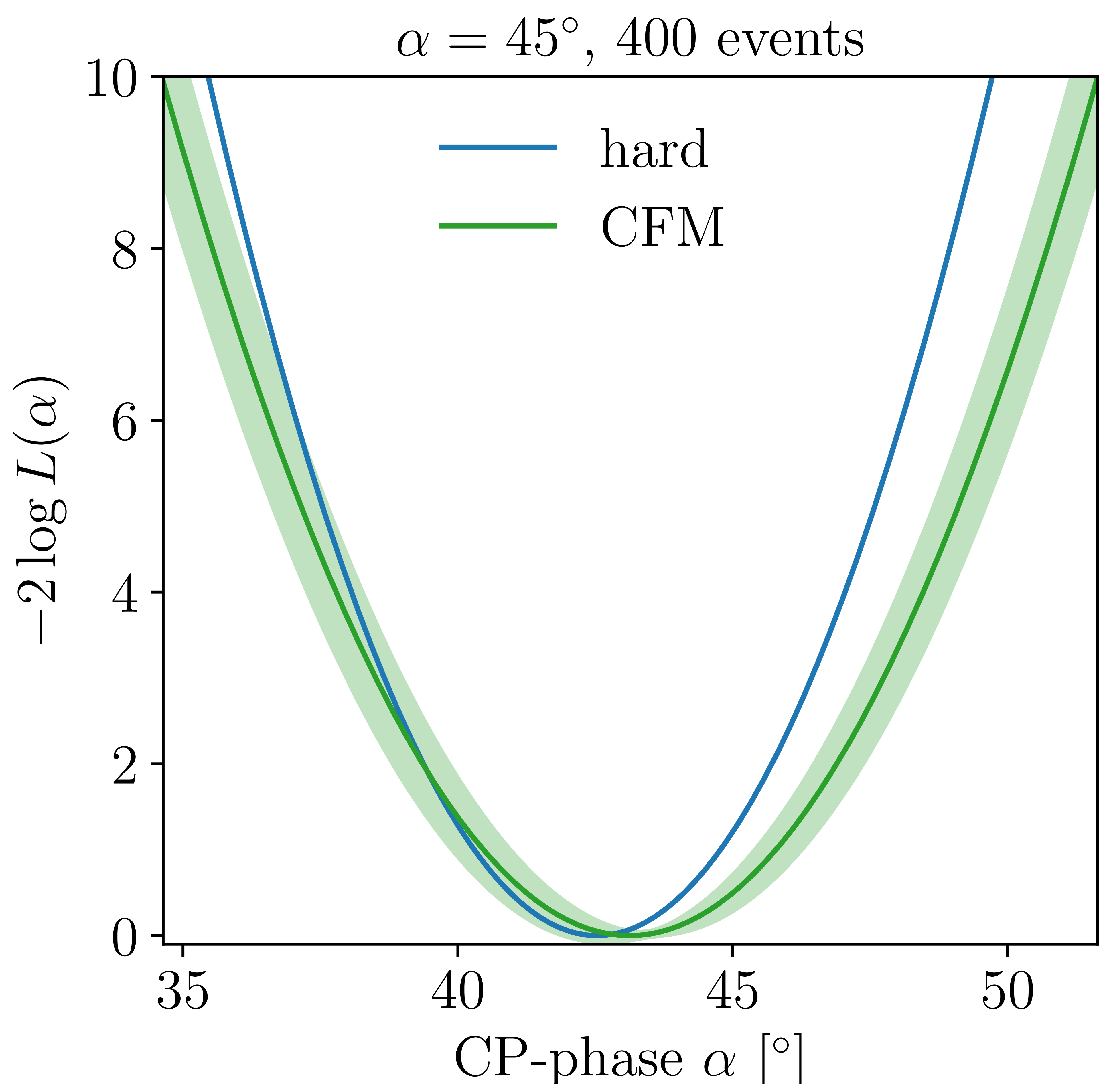


Transfer-Diffusion Results





Estimating Uncertainty

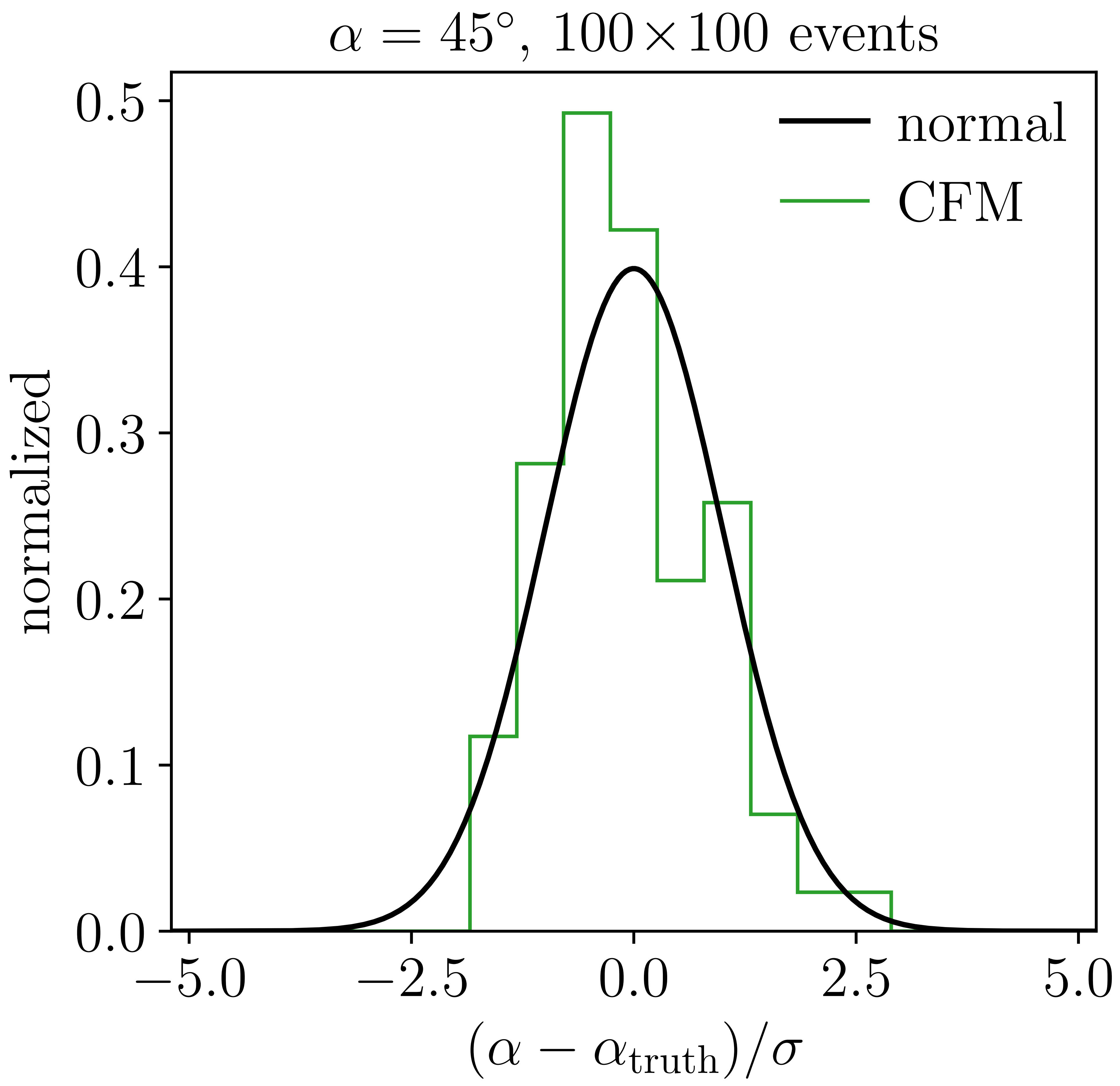


Estimate uncertainty using replicas:

Integration uncertainty:
Resample the MC points using bootstrapping

Network uncertainty:
Use Bayesian NN and resample for each replica

Estimating Calibration



Divide the 10k events into 100 samples of 100 events

Look at this distribution of the minima around the true α

Summary and Outlook

Modern machine learning makes the MEM tractable and scaleable

We present a state-of-the-art three network setup consisting of

1) Transfer-Network encoding the transfer probability $p(x_{reco} | x_{hard})$
2) Acceptance-Network encoding the efficiency $\epsilon(x_{hard})$
3) Sampling-Network encoding the proposal distribution $q(x_{hard})$

Bootstrapping and Bayesian NNs allow us to capture the uncertainties

Extend our formalism to NLO, both on the physics and the ML side

Test our setup on an actual analysis and/or more challenging processes