Fermilab

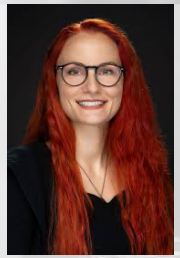
FERMI NATIONAL ACCELERATOR LABORATORY

June, 2026

Model Parameter Estimation for Neutrino-Induced Nucleon Knockout Using SBI

Karla Tame-Narvaez

arXiv:2603.09778v1



Aleksandra Ćiprijanović

Steven Gardiner

Giuseppe Cerati

U.S. DEPARTMENT of ENERGY

Fermi National Accelerator Laboratory is managed by FermiForward for the U.S. Department of Energy Office of Science

Neutrino Physics and Machine Learning 2026

Better Neutrino Models with Simulation-Based Inference (SBI)

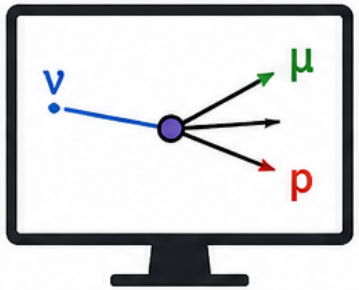
We use machine learning to tune neutrino interaction models to data.

1. Physics Models

GENIE

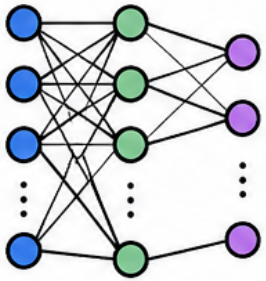
Neutrino interaction simulations with physics parameters

2. Simulate



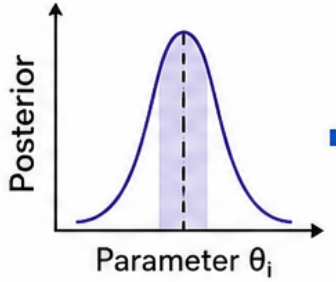
Generate many simulated events for different parameter values

3. Learn (ML)



Neural network learns the mapping from simulations to parameter posteriors

4. Infer



Given data, SBI returns the most probable parameter values (with uncertainties)

5. Results

- Recovers tuned parameters
- Slightly better fit to data
- Transfers to other models (NuWro)

SBI is a powerful, efficient way to tune complex neutrino models.

It learns from simulations, matches data, quantifies uncertainties, and can generalize to other physics models.

Outline

1. **Why We Tune Simulations**
   The physics problem & its growing complexity
2. **What is Simulation-Based Inference?**
   ML concepts explained from scratch
3. **How We Built the Model**
   Training data, architecture, workflow
4. **Does It Actually Work?**
   Validation & calibration tests
5. **Results and Validation**
   The physics payoff
6. **Summary & Outlook**
   What this means for the field

1

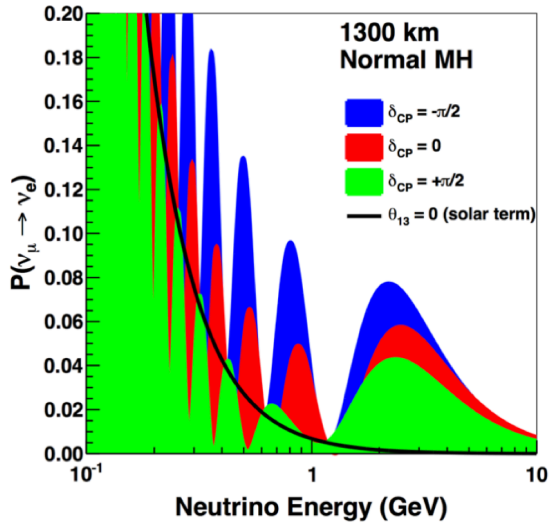
Why we tune simulators?

The Precision Era of Neutrino Oscillations

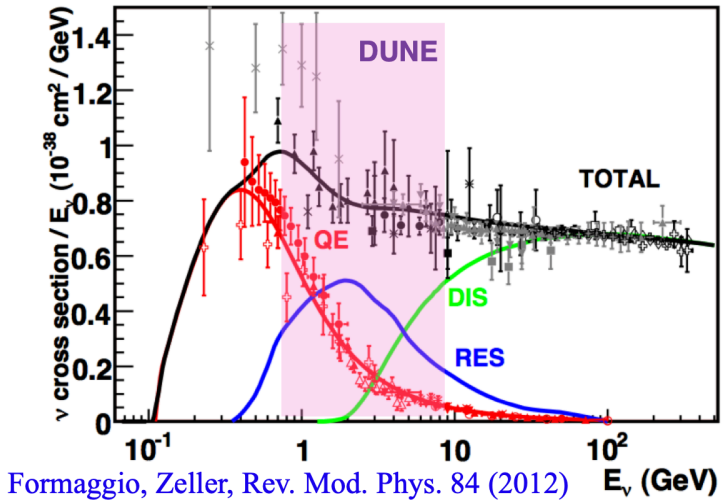
Next-generation experiments demand percent-level systematic control

DUNE and Hyper-K flagship oscillation measurements will be limited by how well we understand neutrino-nucleus interactions at GeV energies.

The challenge: neutrino beams are broad-band. Different interaction channels contribute across a wide energy range. Oscillation sensitivity depends on how accurately we model each channel.



Acciarri et al [DUNE], arXiv 1512.06148



Formaggio, Zeller, Rev. Mod. Phys. 84 (2012)

FERMI NATIONAL ACCELERATOR LABORATORY

The Precision Era of Neutrino Oscillations

Next-generation experiments demand percent-level systematic control

DUNE and Hyper-K flagship oscillation measurements will be limited by how well we understand neutrino-nucleus interactions at GeV energies.

The challenge: neutrino beams are broad-band. Different interaction channels contribute across a wide energy range. Oscillation sensitivity depends on how accurately we model each channel.

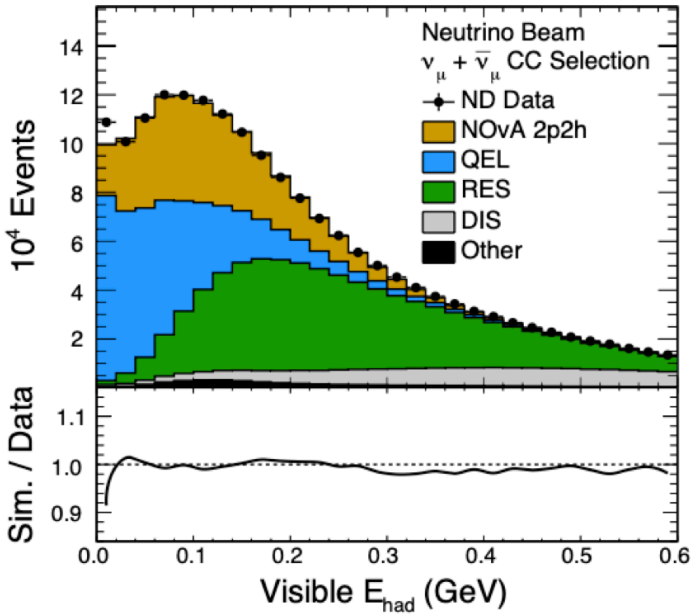
Current interaction generators are essential, but imperfect in the GeV regime.

Experimental collaborations therefore tune generator parameters to cross-section data.

As experiments move toward percent-level systematics, tuning with more parameters and richer data becomes computationally harder.

more parameters, more input data, more compute….

Acero et al [NOvA] Eur. Phys. J. C 80 (2020)



Empirical Tuning: A Solution?

Fit simulation parameters to data

MINERvA Tune

Tuning the GENIE Pion Production Model with MINERvA Data

Phys. Rev. D 100, 072005 (2019)

NOvA Tune

Adjusting Neutrino Interaction Models and Evaluating Uncertainties using NOvA Near Detector Data

Eur. Phys. J. C 80, 1119 (2020)

MicroBooNE Tune

New CC0π GENIE Model Tune for MicroBooNE

Phys. Rev. D 105, 072001 (2022)

GENIE Global Tunes

Neutrino-Nucleon Cross-Section Model Tuning in GENIE v3

Phys.Rev.D 104 (2021) 7, 072009

And more…

Empirical Tuning: A Solution?

Fit simulation parameters to data

MINERvA Tune

Tuning the GENIE Pion Production Model with MINERvA Data

Phys. Rev. D 100, 072005 (2019)

NOvA Tune

Adjusting Neutrino Interaction Models and Evaluating Uncertainties using NOvA Near Detector Data

Eur. Phys. J. C 80, 1119 (2020)

MicroBooNE Tune

New CC0π GENIE Model Tune for MicroBooNE

Phys. Rev. D 105, 072001 (2022)

GENIE Global Tunes

Neutrino-Nucleon Cross-Section Model Tuning in GENIE v3

Phys.Rev.D 104 (2021) 7, 072009

And more…

The MicroBooNE Tune: Our Test Case

The MicroBooNE Tune adjusts four GENIE parameters using T2K data

θ_1	**MaCCQE**	CCQE axial mass; affects normalization and Q^2 dependence
θ_2	**NormCCMEC**	Overall CC multi-nucleon/MEC normalization
θ_3	**XSecShape_CCMEC**	MEC shape interpolation in energy/momentum-transfer space
θ_4	**RPA_CCQE**	Strength of the RPA correction for CCQE

$$\theta_1 \in [0.528, 1.39]\ \mathrm{GeV}, \quad \theta_2 \in [0,\ 3.0],$$
$$\theta_3 \in [0.0, 1.0], \quad \theta_4 \in [0.0, 1.5].$$

The MicroBooNE Tune: Our Test Case

The MicroBooNE Tune adjusts four GENIE parameters using T2K data

θ_1	**MaCCQE**	CCQE axial mass; affects normalization and Q^2 dependence
θ_2	**NormCCMEC**	Overall CC multi-nucleon/MEC normalization
θ_3	**XSecShape_CCMEC**	MEC shape interpolation in energy/momentum-transfer space
θ_4	**RPA_CCQE**	Strength of the RPA correction for CCQE

This paper asks:
1) Can SBI recover the MicroBooNE parameter values and fit the same data efficiently?
2) Can tuned GENIE model approximate NuWro predictions using the same four-parameter space?
3) Do the learned posteriors give sensible parameter values and a competitive χ^2 ?

Why SBI? The Problem With Classical χ^2 Minimization

- Requires many expensive GENIE forward simulations per iteration
- Optimization must be re-run from scratch if data, covariance, or parameter set changes
- Scales poorly: cost grows rapidly with number of parameters
- Encountered Peelle's Pertinent Puzzle (PPP) — a systematic bias when using correlated covariance matrices
- MicroBooNE had to drop off-diagonal covariance elements to get a stable fit
- Ignoring measurement correlations introduces systematic errors in the result
- PPP is a recognized problem across the neutrino community with no perfect solution in χ^2 approaches
- Future tunes may need 20–30 parameters: classical methods will become prohibitively expensive

We need a scalable, amortized alternative to repeated χ^2 minimization.

2

What is Simulation-Based Inference?

We want to know: given our observed data x, what values of physics parameters θ are most plausible?

$$p(\boldsymbol{\theta}|\boldsymbol{x}) = \frac{p(\boldsymbol{x}|\boldsymbol{\theta}) * p(\boldsymbol{\theta})}{p(\boldsymbol{x})}$$

We want to know: given our observed data x, what values of physics parameters θ are most plausible?

Likelihood

How likely is the data for a given θ?

Prior

What we knew before the experiment

$$p(\boldsymbol{\theta}|\boldsymbol{x}) = \frac{p(\boldsymbol{x}|\boldsymbol{\theta}) * p(\boldsymbol{\theta})}{p(\boldsymbol{x})}$$

Posterior

What we want: parameter values given data

Evidence

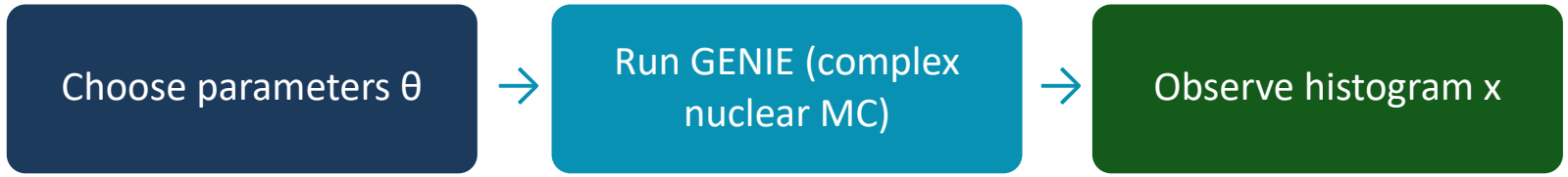
Normalizing constant (usually hard to compute)

The problem: for complex nuclear models with hundreds of internal parameters, $p(x|\theta)$ is analytically unknown and computationally intractable…

The Intractable Likelihood Problem

For complex simulators, the expensive or unavailable part is often the likelihood p(x|θ).

- We can simulate x for a chosen θ.
- But writing a reliable analytic likelihood for all generator details can be hard.



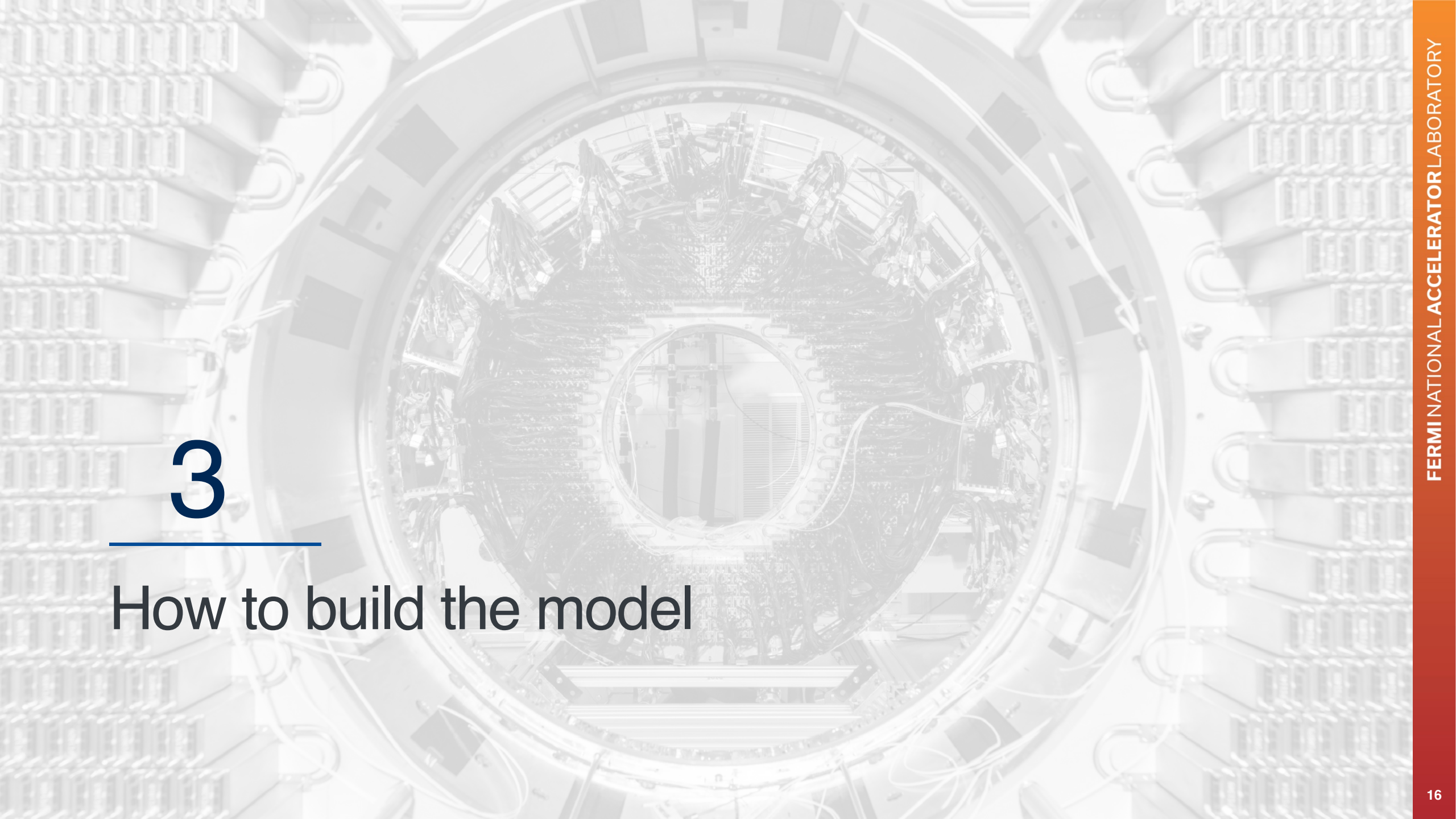
Inside GENIE: thousands of internal variables, nuclear wave functions, intranuclear cascades are integrated out. The mapping θ → x is a black box. We can run it forward, but we cannot invert it analytically.

Classical approach:

Run GENIE hundreds of times. For each new dataset or parameter set, re-run everything. Expensive. Doesn't scale.

SBI approach:

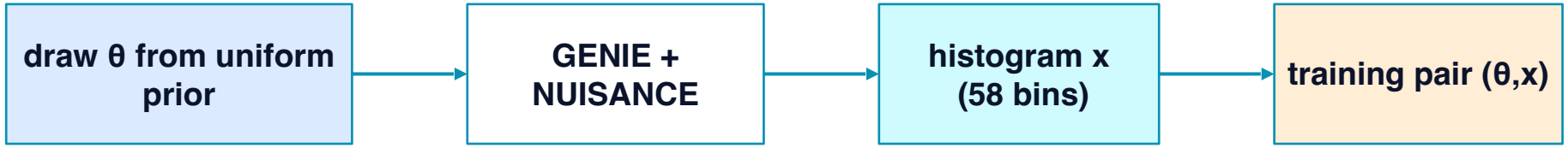
Train a neural network to learn the inverse mapping x → p(θ|x) once. Then run inference in seconds on any new data.



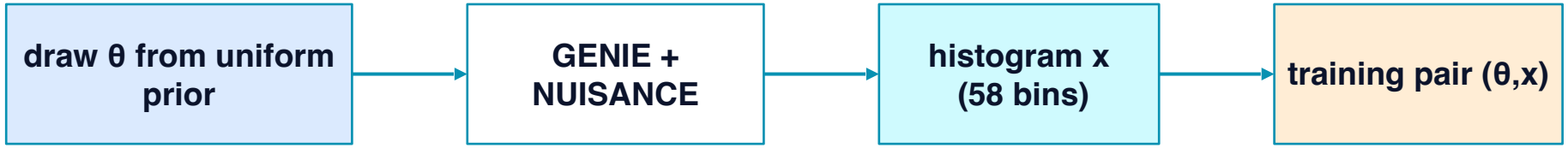
3

How to build the model

Pipeline



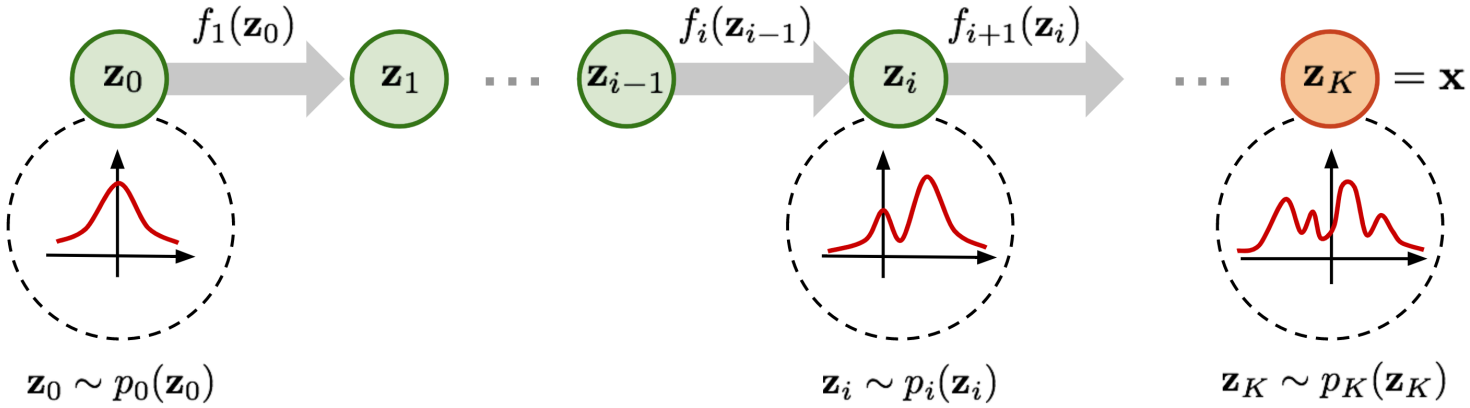
Training data: many simulator runs become supervised pairs



- 171,747 training configurations and 1,000 independent test configurations.
- Each configuration is one parameter point plus one predicted T2K-like histogram.

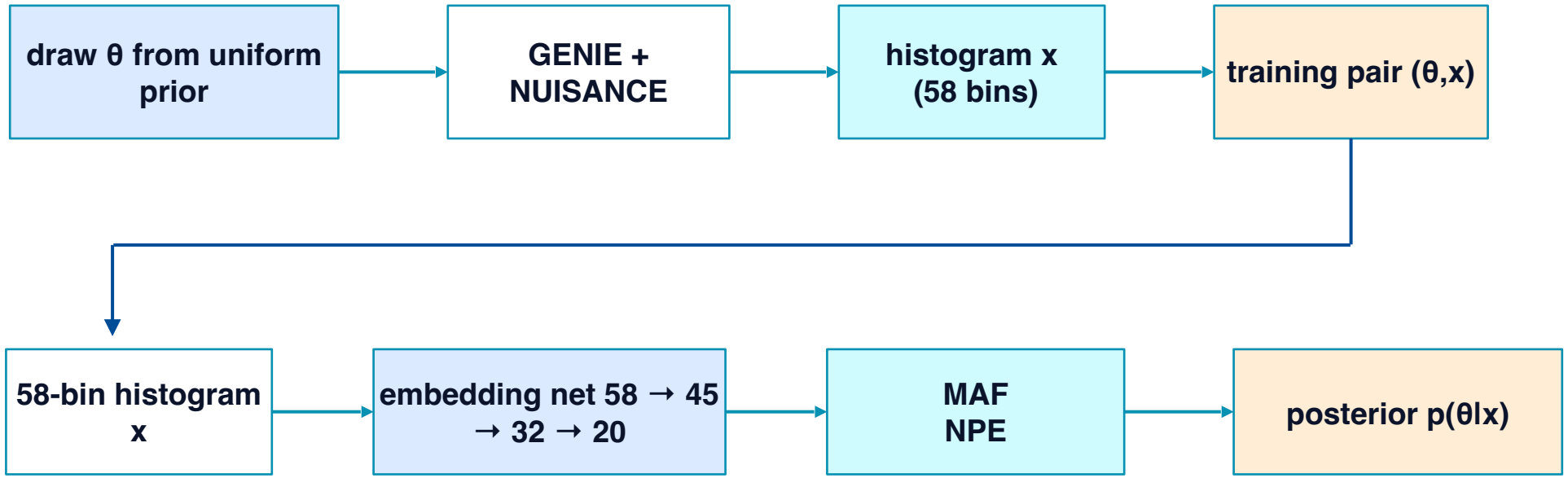
We need to learn p(θ|x) — a full probability density, not just a best-fit value. We use Masked Autoregressive Flows (MAFs).

Normalizing flows — key idea



The MAF variant uses an autoregressive structure that is efficient to compute — all parameters inferred in one forward pass

Pipeline

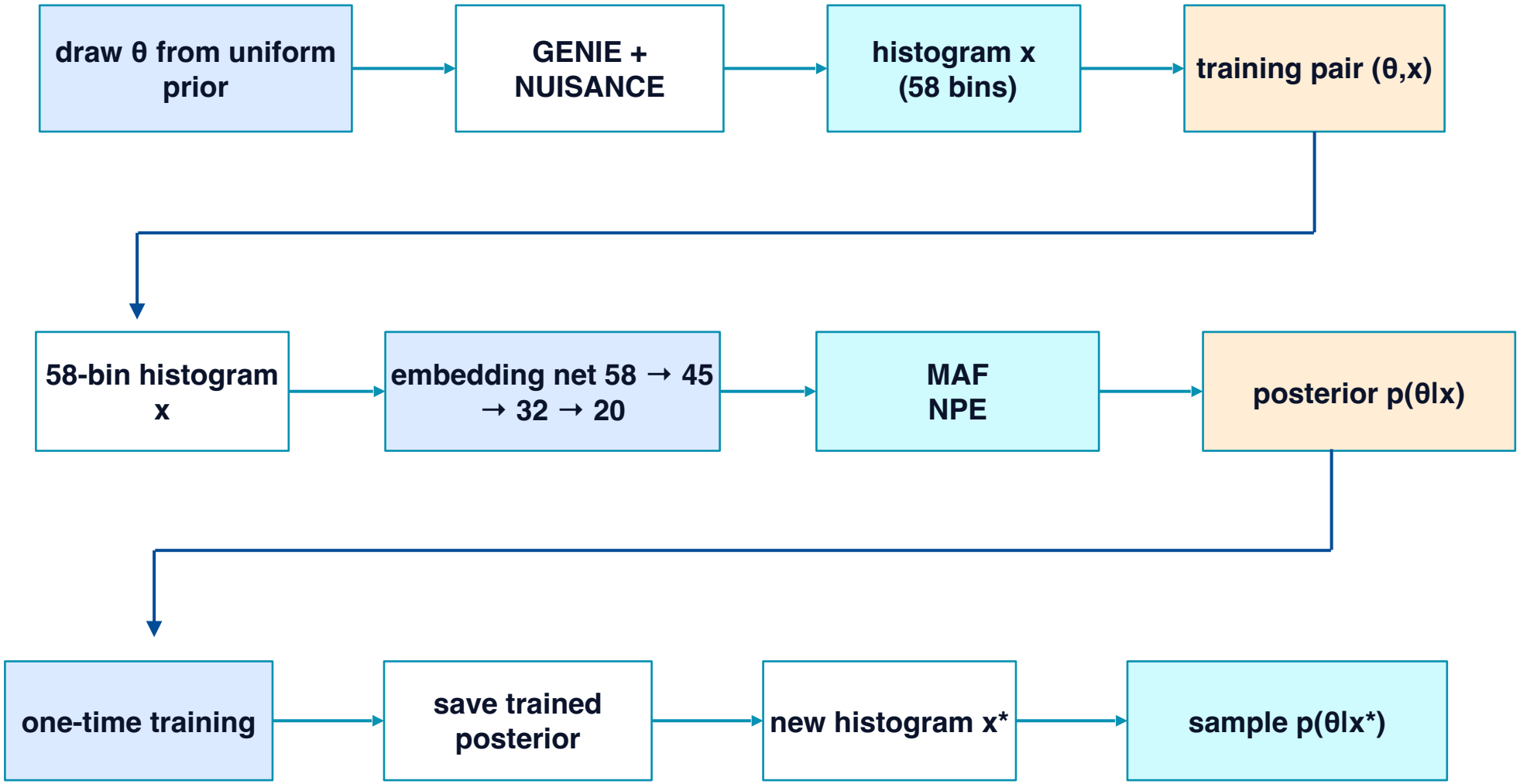


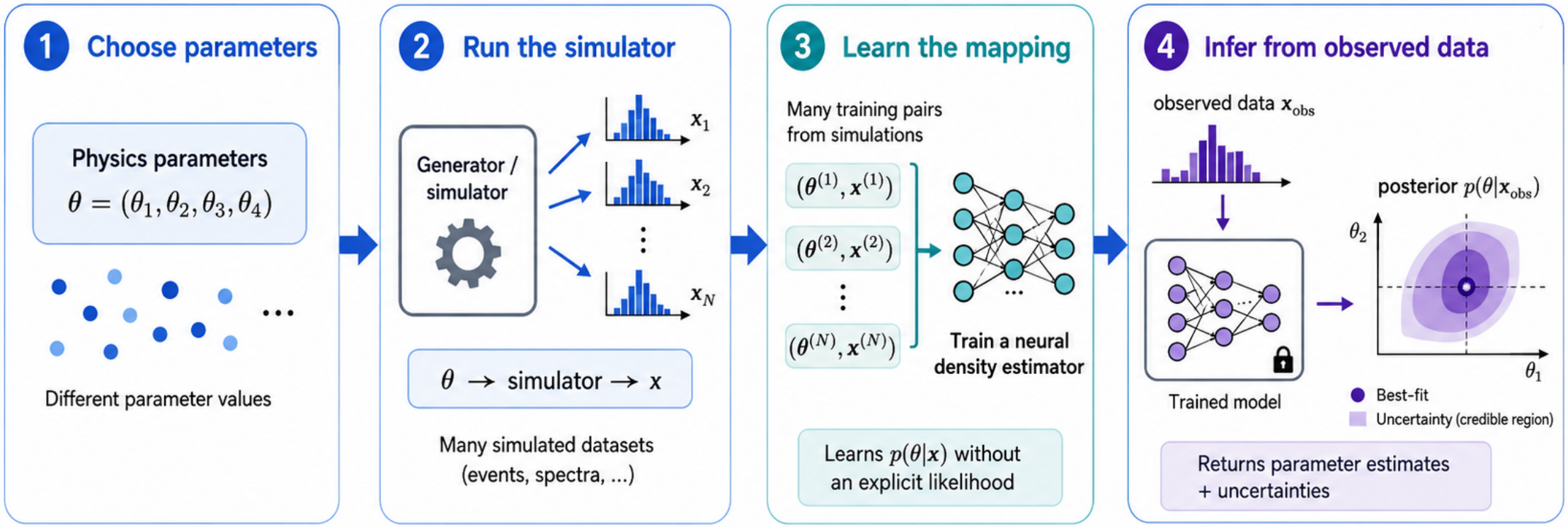
Pipeline

draw θ fro
pri

58-bin hist
x

Pipeline





Core idea: simulate many examples first, then use the trained model to infer parameters from real data in seconds.

Likelihood-free

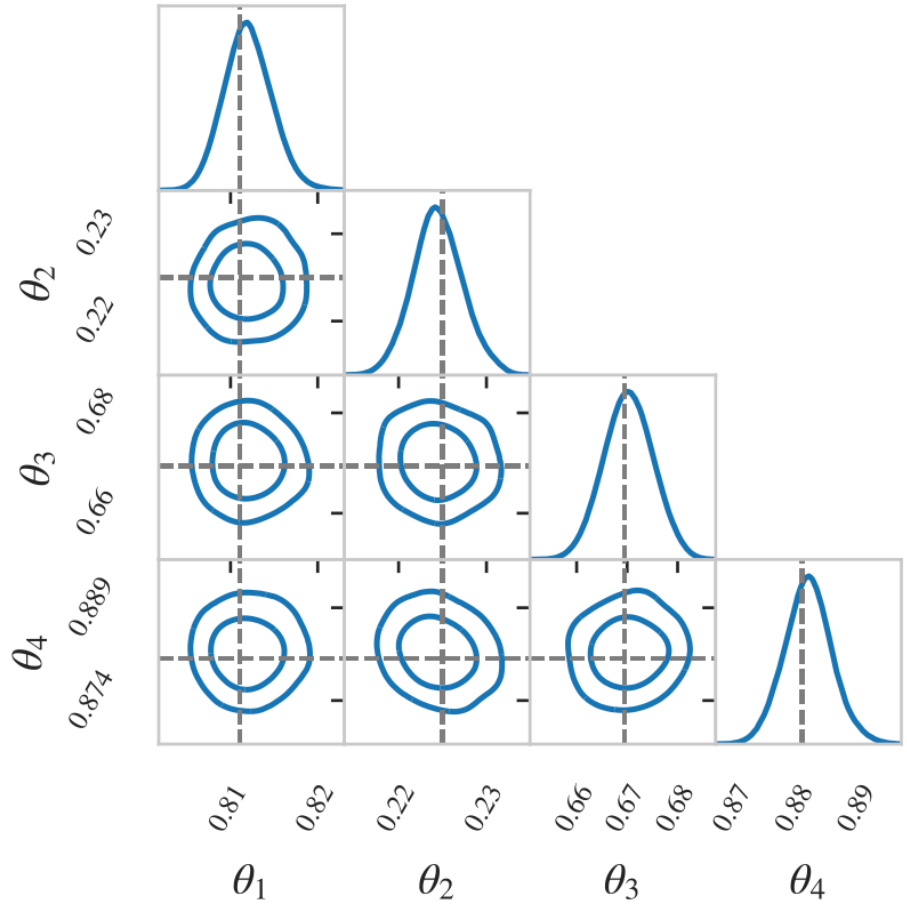
Amortized inference

Uncertainty quantification

4

Does it actually work?

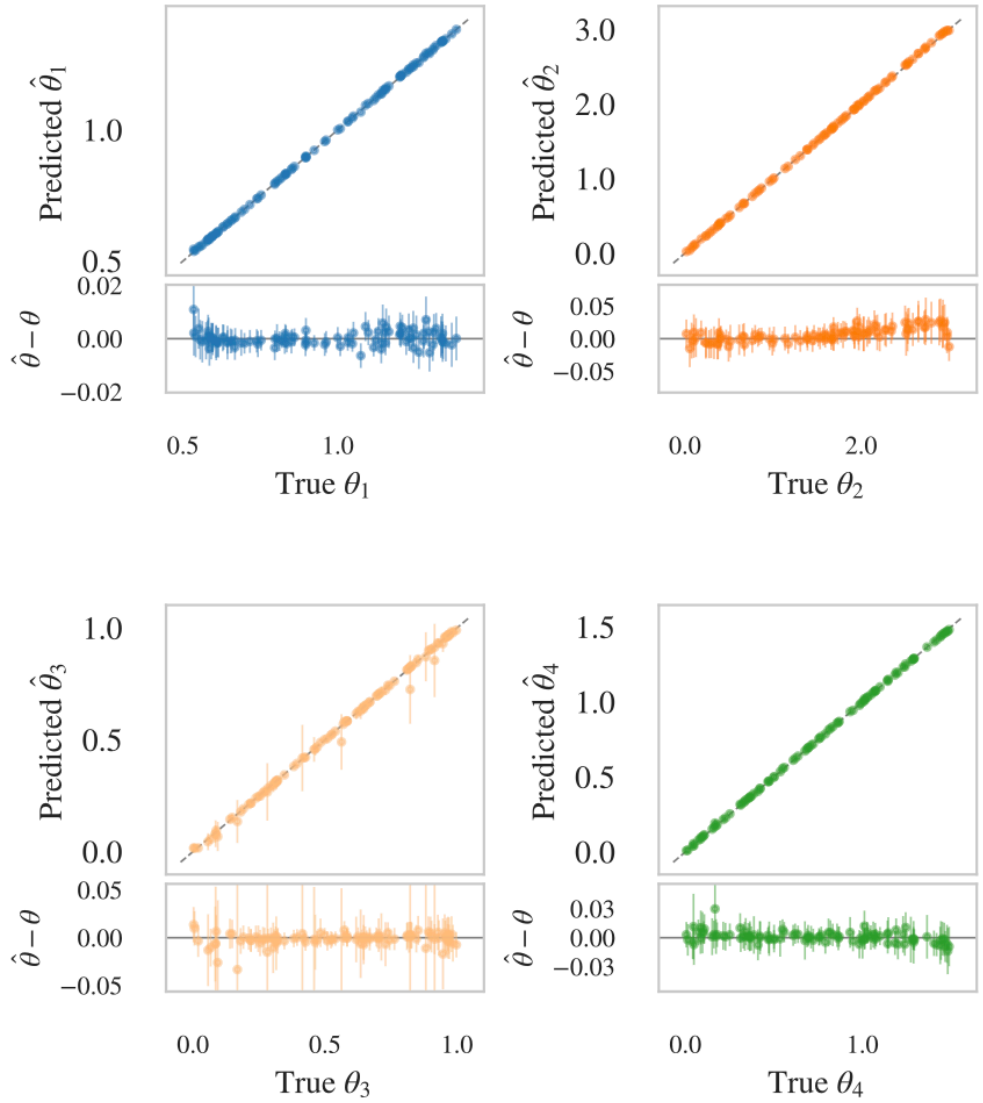
Posterior for one test configuration



Neural Posterior Estimate of the four parameters for a single test event. The gray dashed lines indicate the true values while the blue contours represent the estimates at 68% and 95% confidence intervals.

Predicted vs. true parameters across test samples

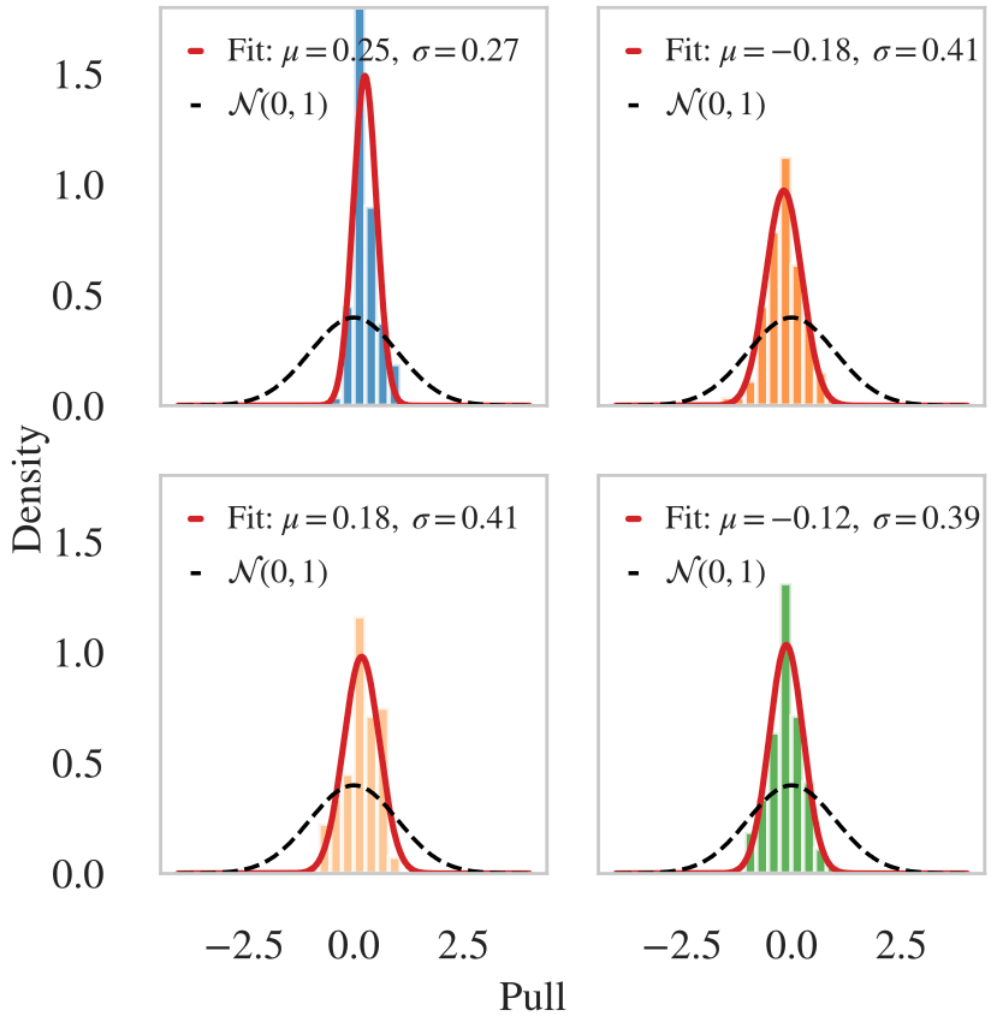
Comparison of true and predicted values for each parameter, shown for a subset of 200 test configurations with the predicted 1σ error bars. Top sub-panels: predicted vs. true with the dashed line indicating perfect recovery. Bottom sub-panels: residual predicted minus true vs. true, zoomed in to make individual error bars visible. The residuals are consistent with zero within 1σ for all four parameters.



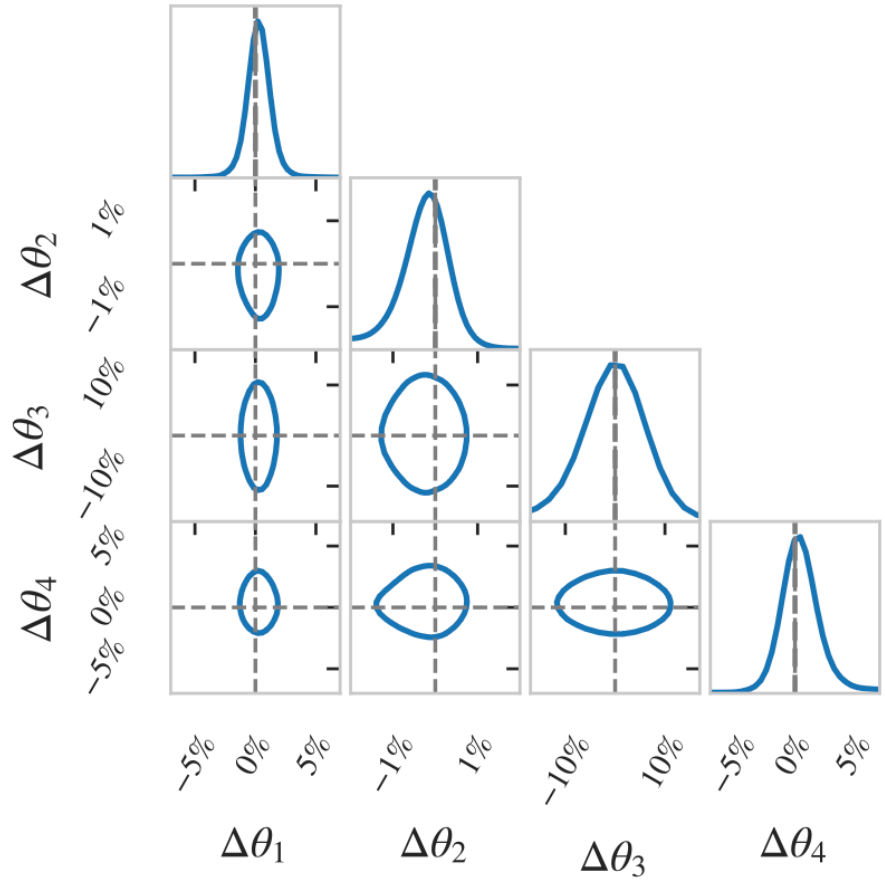
Pull distributions

Histograms of pulls for 1000 test configurations for each parameter. All four parameter distributions (red solid line) are centered close to zero and somewhat narrower than the standard normal distribution (black dashed line), suggesting that the uncertainties may be overestimated.

$$pull_j = \frac{\hat{\theta}_{i,j} - \theta_{i,j}^{\text{true}}}{\sigma_{i,j}}$$

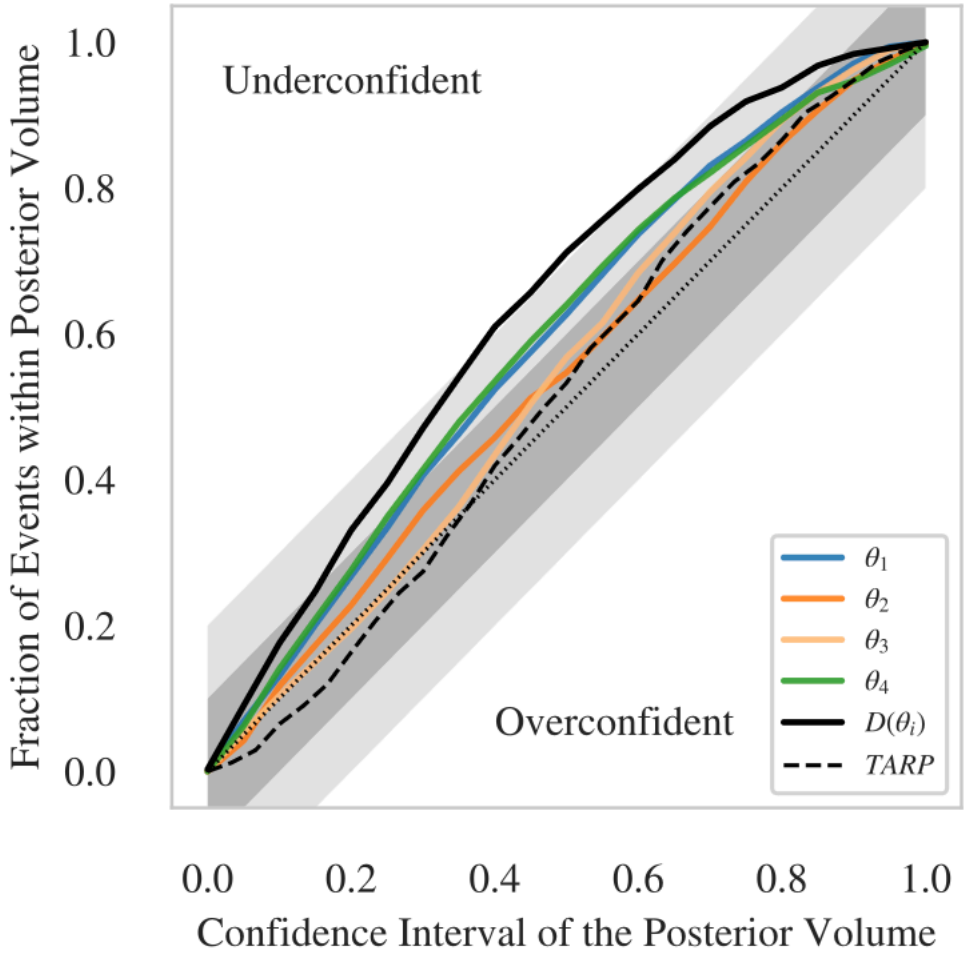


Bias and precision over the full test set



Residual distribution (in percentages) of four parameters for 1000 test configurations. The gray dashed lines indicate the zero value (perfect prediction). The distributions of residuals are very narrow and show no visible biases, i.e, they are all centered around zero. Additionally, all deviations are within a few percent. The blue contours represent the 68% confidence regions.

Coverage



Posterior coverage of the θ_i parameters for 1000 test configurations (colored solid lines). $D(\theta_i)$ and TARP curves show combined model performance over all pa- rameters, and are plotted with solid and dashed black lines, respectively. The diagonal black dotted line indi- cates perfect uncertainty calibration. The gray regions indicate thresholds of 10% (dark gray) and 20% (light gray) uncertainty miscalibration. All curves are within the 20% band and are showing slight underconfidence, which is preferred compared to an overconfident model with too narrow error bars.

5

Results and Validation

Validation on the MicroBooNE Tune histogram

can the learned inverse map recover a known tuned GENIE configuration?

Input: MicroBooNE Tune prediction histogram x_{MB} → **Output: $p(\theta | x_{MB})$**

Model	θ_1	θ_2	θ_3	θ_4	σ_{θ_1}	σ_{θ_2}	σ_{θ_3}	σ_{θ_4}	χ^2(T2K data, 58 bins)[c]
MicroBooNE Tune[a]	1.10	1.66	1.00	0.85	±0.10	±0.50	+1.00(-0.00)	±0.40	115.10
Inferred MicroBooNE Tune[b]	1.09	1.67	0.94	0.85	±0.003	±0.01	±0.01	±0.007	113.07

- The inferred MicroBooNE Tune parameters agree very closely with the original tune.
- This validates the learned mapping before applying it to real T2K data.

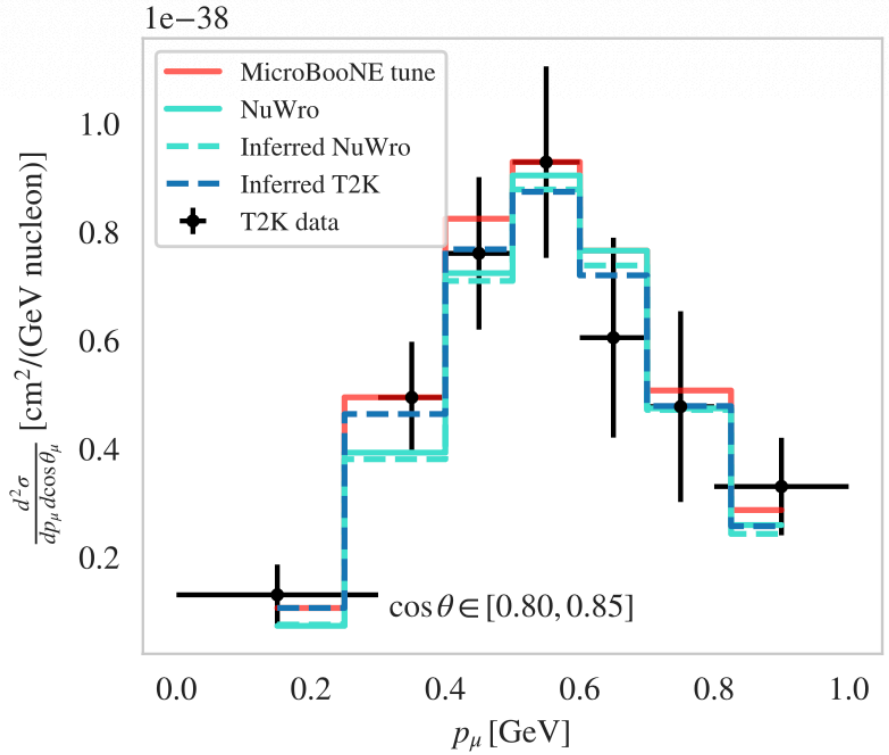
NuWro as Mock Data

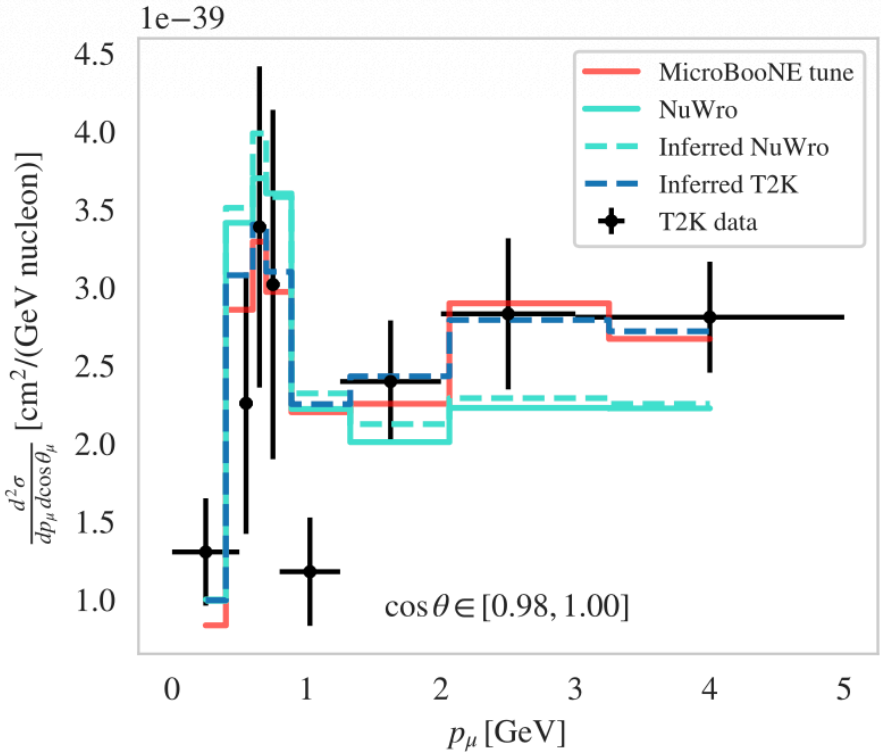
Can GENIE be tuned to approximate a completely different generator — NuWro?

- GENIE and NuWro share some processes but differ in some intrinsic physics
- We feed the NuWro T2K prediction as 'mock data' into our **trained** network
- The network finds GENIE parameters that approximate NuWro as closely as possible

Model	θ_1	θ_2	θ_3	θ_4	σ_{θ_1}	σ_{θ_2}	σ_{θ_3}	σ_{θ_4}	χ^2(T2K data, 58 bins)[c]
MicroBooNE Tune[a]	1.10	1.66	1.00	0.85	±0.10	±0.50	+1.00(-0.00)	±0.40	115.10
Inferred MicroBooNE Tune[b]	1.09	1.67	0.94	0.85	±0.003	±0.01	±0.01	±0.007	113.07
NuWro	-	-	-	-	-	-	-	-	126.73
Inferred NuWro[b]	1.10	0.70	0.09	0.57	±0.003	±0.015	±0.013	±0.007	126.93

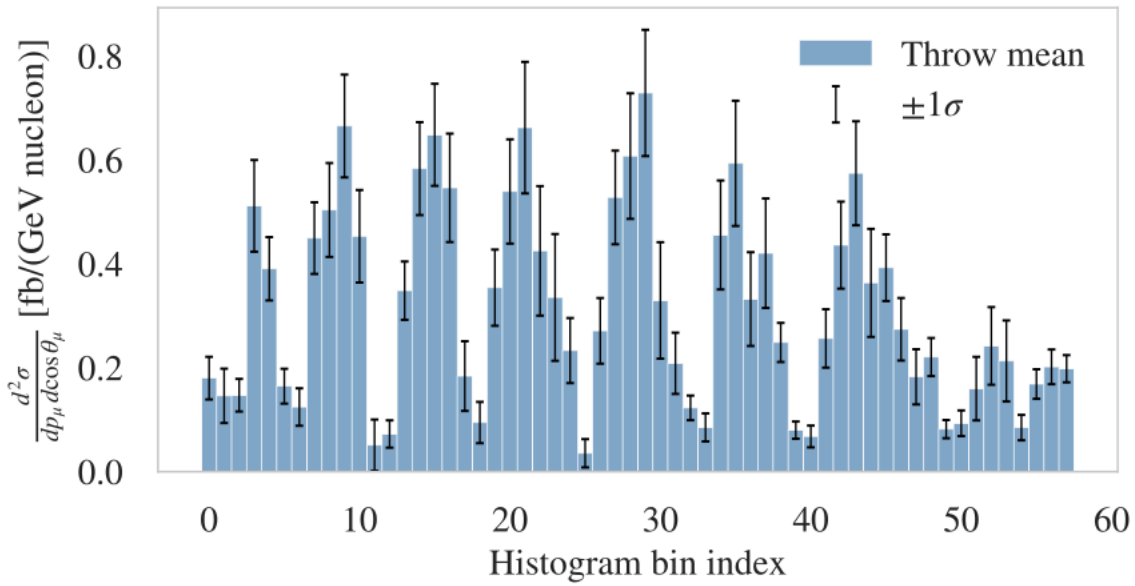
This opens the door to 'generator surrogates' — computationally cheap approximations of expensive generators for sensitivity studies.





Comparison of the T2K cross-section data (black points) to the NuWro (solid cyan), NPE-inferred NuWro (dashed cyan), MicroBooNE Tune (solid red), and NPE-inferred T2K (dashed blue) model predictions in two different slices of cos θ.

Propagating experimental uncertainties with covariance throws



Variation of the T2K histogram over 1000 samples drawn from the covariance matrix around the central T2K value. The bars show the mean per bin and the vertical error bars indicate the corresponding 1σ spread.

Propagating experimental uncertainties with covariance throws

How do we use the T2K covariance matrix if the network doesn't take a covariance as input?

Step 1 Generate 1000 Gaussian throws using the full covariance matrix

Step 2 Run the trained network on each of the 1000 throws → 1000 sets of inferred parameters

Step 3 Compute spread (16th / 84th percentiles) across all 1000 inferences → asymmetric 1σ error bars

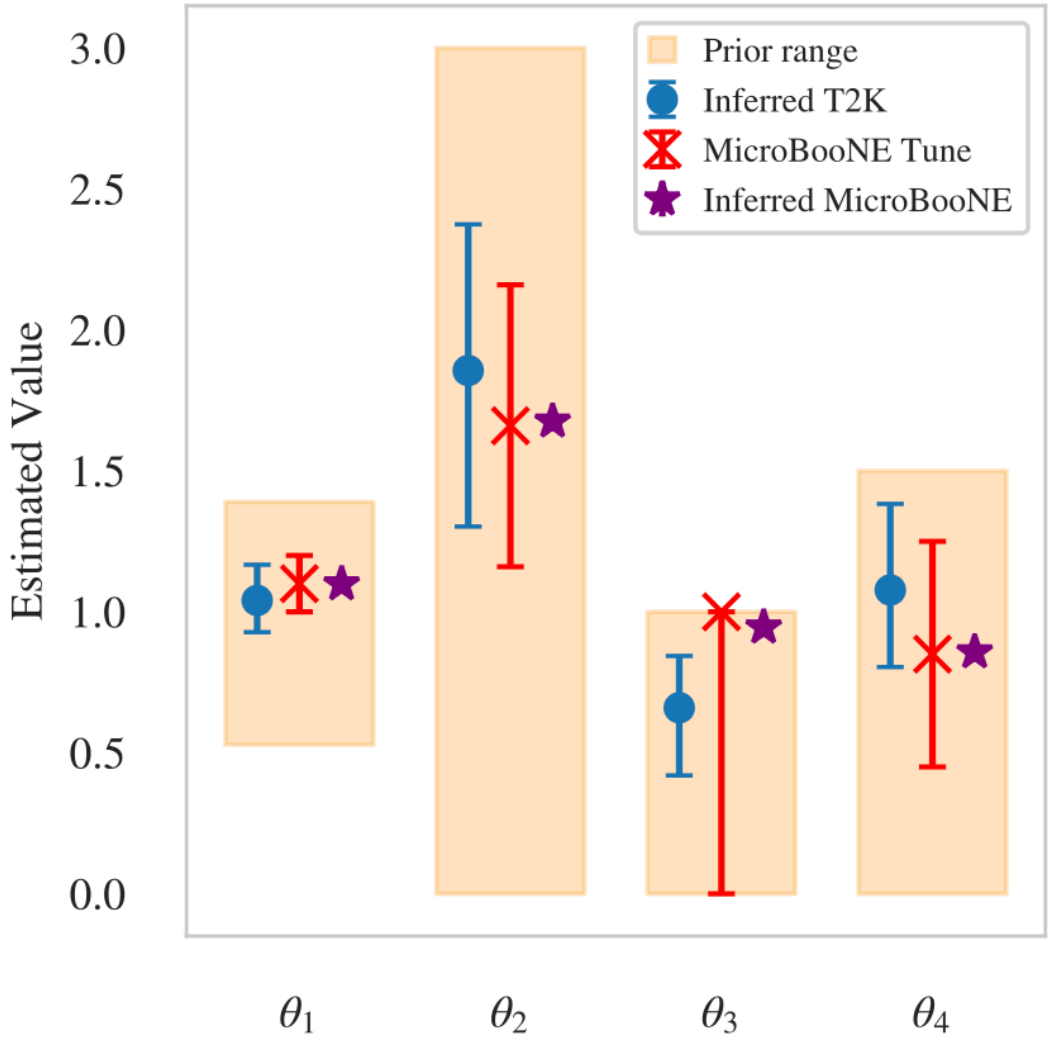
Key insight: the dominant uncertainty in the final result comes from experimental errors (covariance matrix throws), not from the network's intrinsic uncertainty. This is shown in Figure 9 of the paper and coming in the next slides— the throw-based error bars are much wider than single-posterior error bars.

Inferred T2K parameters vs. MicroBooNE Tune

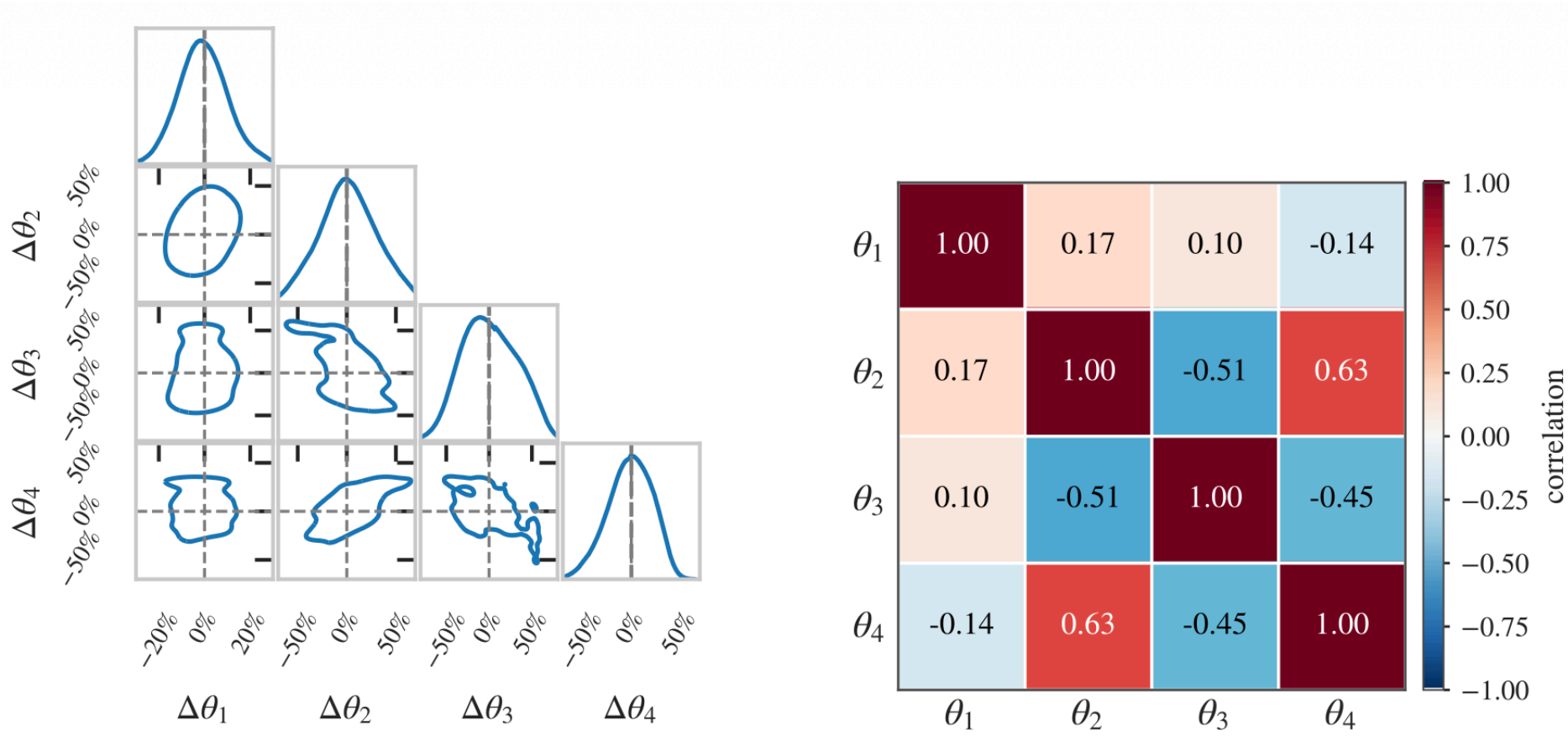
Model	θ_1	θ_2	θ_3	θ_4	σ_{θ_1}	σ_{θ_2}	σ_{θ_3}	σ_{θ_4}	χ^2(T2K data, 58 bins)[c]
MicroBooNE Tune[a]	1.10	1.66	1.00	0.85	±0.10	±0.50	+1.00(-0.00)	±0.40	115.10
Inferred MicroBooNE Tune[b]	1.09	1.67	0.94	0.85	±0.003	±0.01	±0.01	±0.007	113.07
NuWro	-	-	-	-	-	-	-	-	126.73
Inferred NuWro[b]	1.10	0.70	0.09	0.57	±0.003	±0.015	±0.013	±0.007	126.93
Inferred T2K[a]	1.04	1.85	0.65	1.07	+0.11(-0.12)	+0.55(-0.24)	+0.24(-0.18)	+0.27(-0.30)	107.29

Inferred T2K parameters vs. MicroBooNE Tune

A network trained only on GENIE simulations — never shown the MicroBooNE result — lands inside their error bars and beats their χ^2 by 7.8 units!!



Experimental covariance induces parameter correlations



LEFT: Residual distribution (in percentages) of four parameters for 1000 input T2K data histograms generated via throws of the full covariance matrix (including all correlations). The gray dashed lines indicate the zero value (exact agreement with the result of inference on the central-value T2K data histogram). The two-dimensional contours represent 68% confidence regions. RIGHT: Parameter correlation matrix constructed from the residual distribution.

6

Summary & Outlook

What We Demonstrated

✓ **Parameter recovery**	Network correctly infers all four GENIE parameters when given mock data from the MicroBooNE Tune — near-perfect recovery
✓ **T2K data fit**	When applied to real T2K data, our fit slightly outperforms MicroBooNE Tune: χ^2=107.3 vs 115.1
✓ **Amortized speed**	Training: 30 min CPU. Inference on any new dataset: seconds. No re-running required.
✓ **Generator surrogate**	GENIE parameters tuned to mimic NuWro achieve nearly identical χ^2 to NuWro itself: 126.93 vs 126.73

Outlook: Why This Matters for Future Experiments

This work is a proof of concept. The real payoff comes when we scale to the complexity of DUNE and Hyper-K analyses.

1 **Higher dimensions**

NPE expected to scale well up to ~30 parameters. Classical χ^2 minimization becomes exponentially costlier with each added parameter.

2 **Non-Gaussian uncertainties**

Future data releases may provide non-Gaussian uncertainties. SBI can handle these naturally — no Gaussian assumption is needed.

3 **Multiple generators**

Generator surrogates allow experiments to cheaply evaluate multiple physics models without running full simulations for each.

Broader Impact: Changing How We Tune

SBI represents a shift in how the field approaches simulation tuning — from iterative optimization to learning-based inference.

- Classical tuning: one fit per (experiment, model, dataset) combination — $O(N^2)$ effort as N grows
- SBI: one training run, then inference is free — O(1) inference after upfront cost
- Can handle non-standard observables, non-Gaussian errors, multi-experiment combinations
- Generator surrogate concept could allow experiments to test 5–10 physics models at negligible compute cost
- Network posteriors give full covariance between parameters — classical fits often report only 1D uncertainties
- Openly available code (GitHub + Zenodo data) — designed for community use
- Compatible with existing frameworks: GENIE, NUISANCE, NuWro — no pipeline changes required

This approach is generalizable: the same SBI framework can be applied to any simulation with tunable parameters where the likelihood is intractable — a common situation across nuclear and astroparticle physics.


Fermilab
FermiFORWARD
U.S. DEPARTMENT of ENERGY
FERMI NATIONAL ACCELERATOR LABORATORY