arXiv: 2309.04639
Accepted by MLST

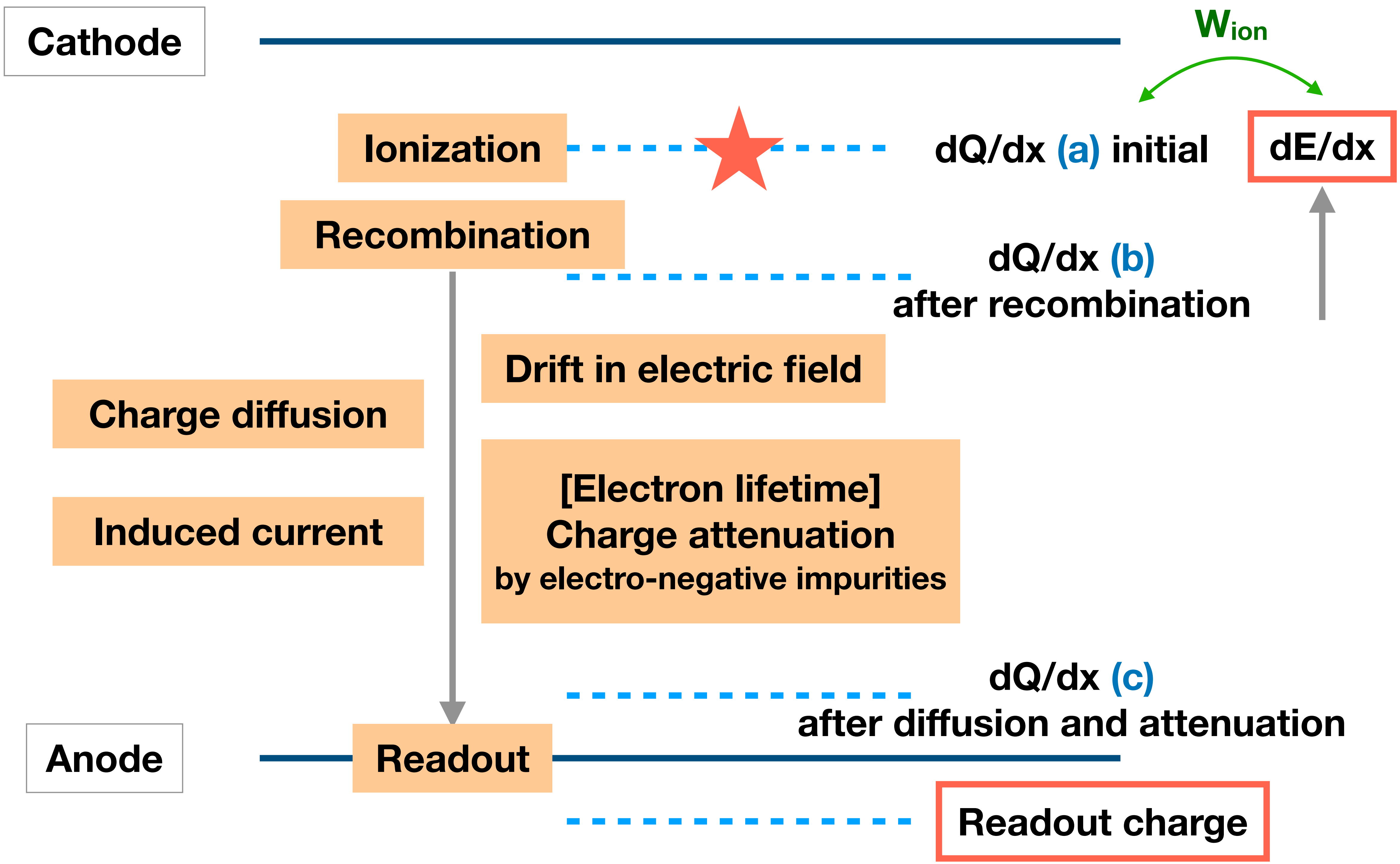
SLAC NATIONAL ACCELERATOR LABORATORY

Differentiable Simulation of a Liquid Argon Time Projection Chamber for High-dimensional Detector Calibration

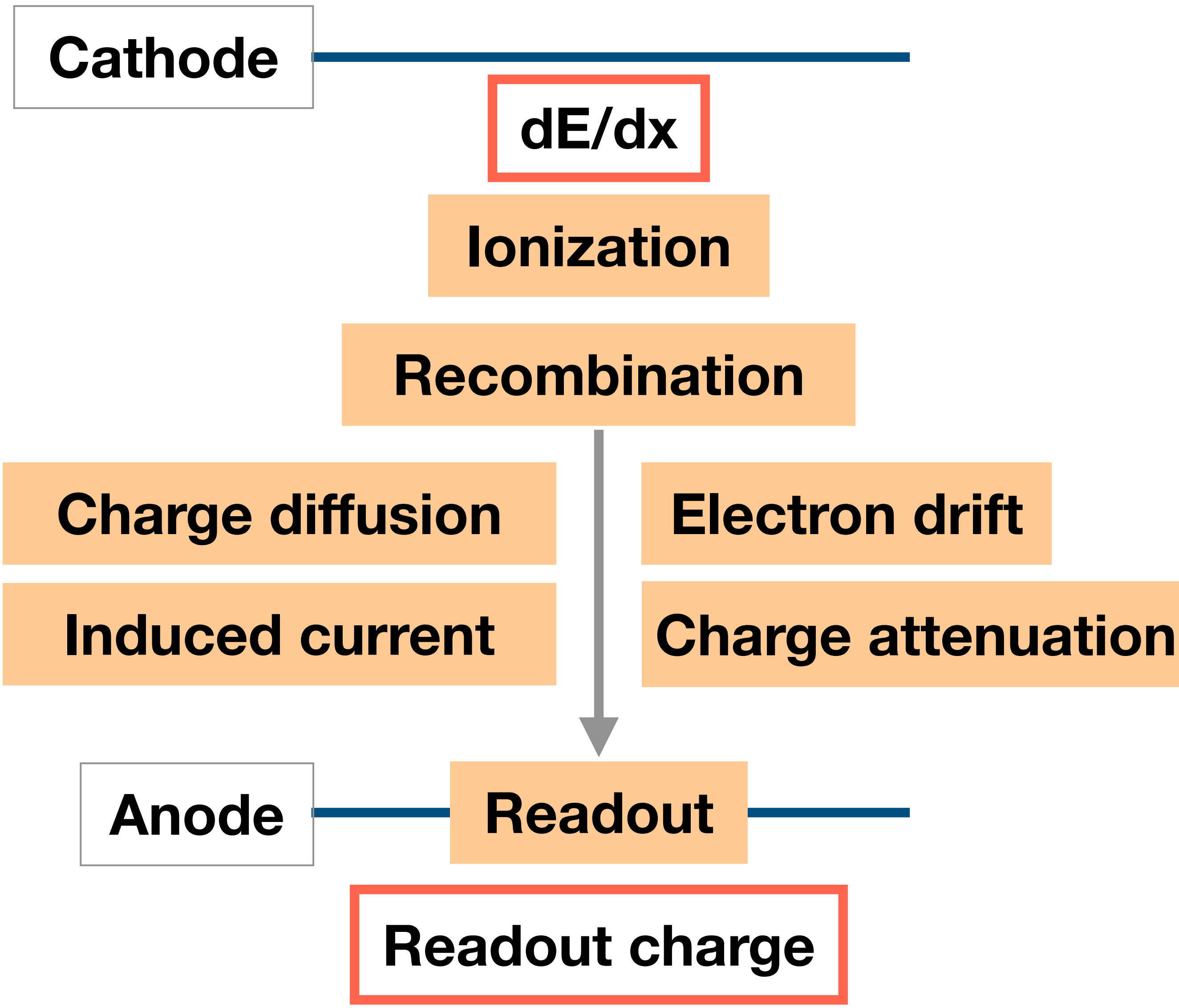
Yifan Chen
SLAC, Stanford University

The 2nd Wire-Cell Summit
BNL
12 April 2024

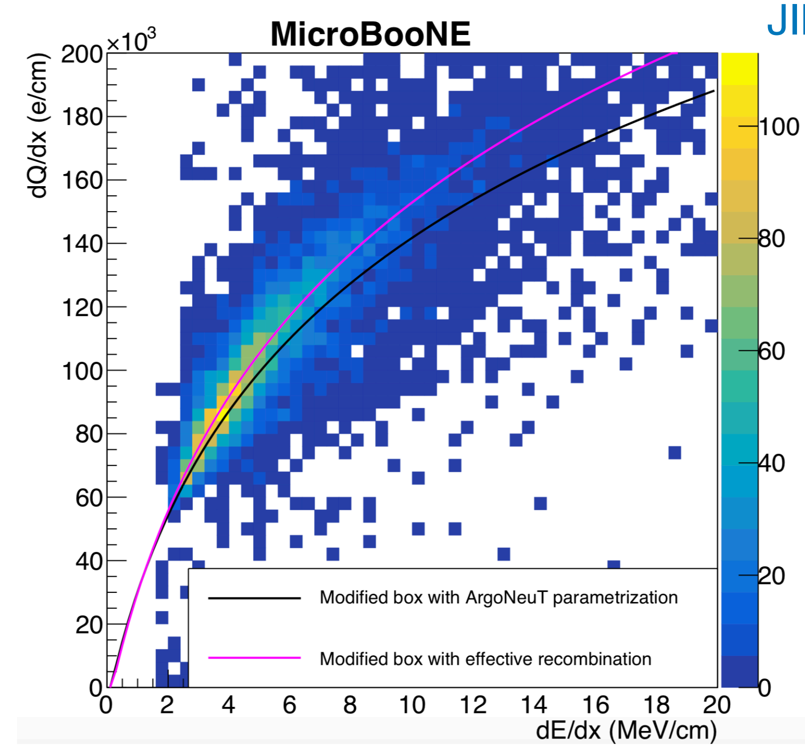
Simulation and Calibration for the Charge Signal

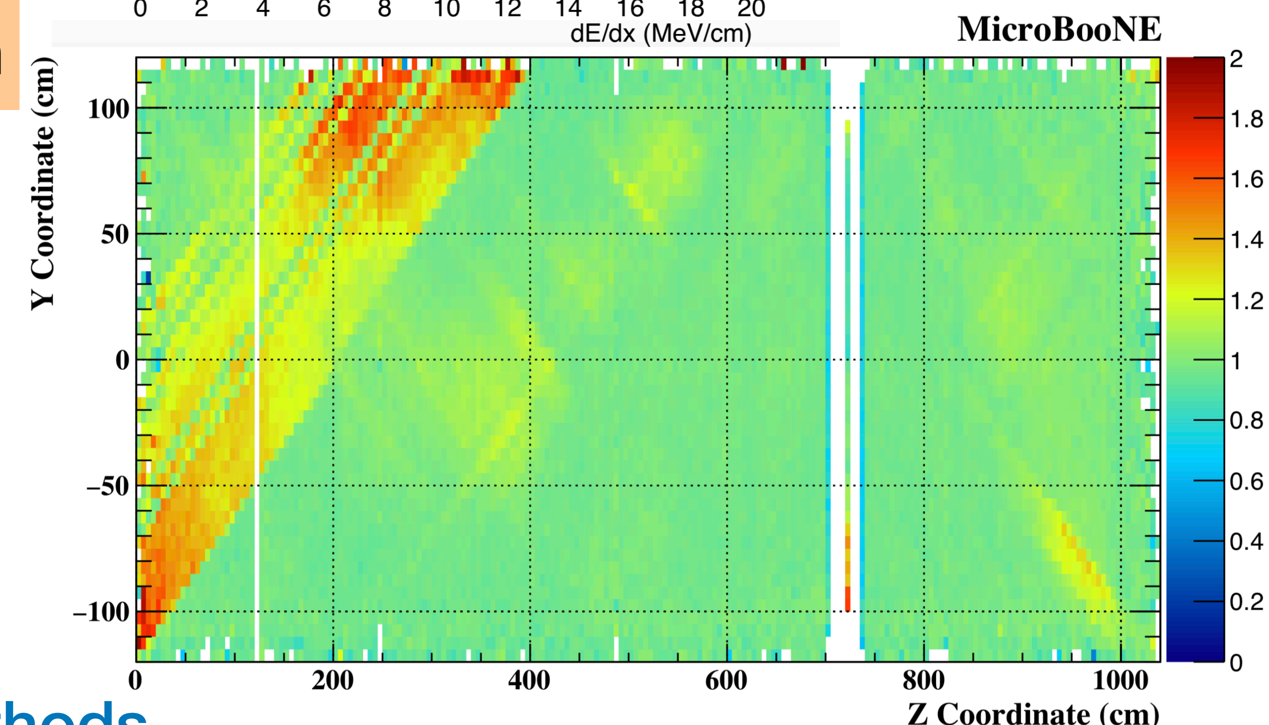


Calibration with Conventional Approaches



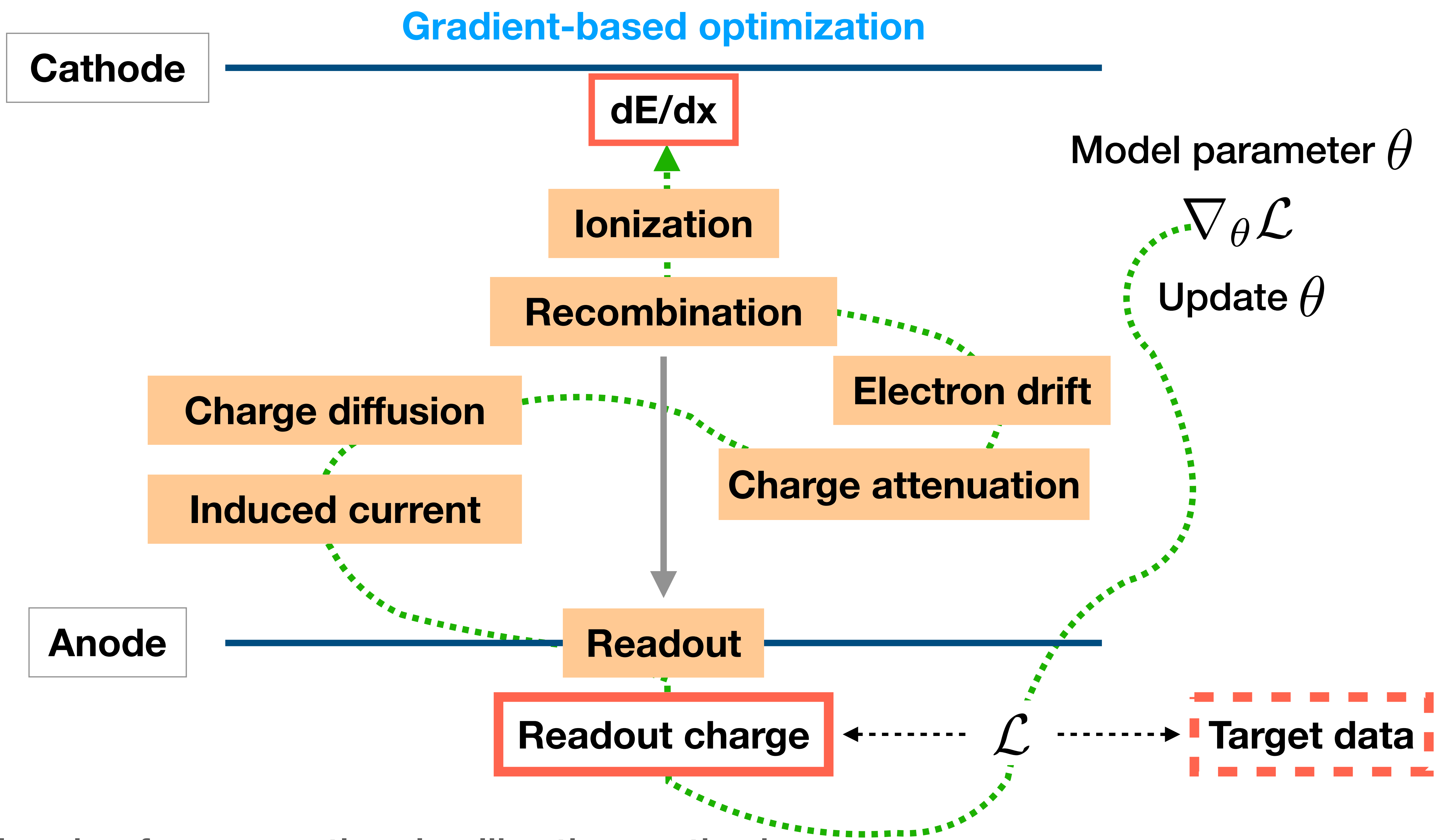
JINST 15, P03022 (2020)





- Challenging for conventional calibration methods
- Led to development of a differentiable simulation for high dimensional calibration
  - Simultaneous optimization for multiple model parameters
  - Straightforward application of the calibration

Calibration with a Differentiable Simulation



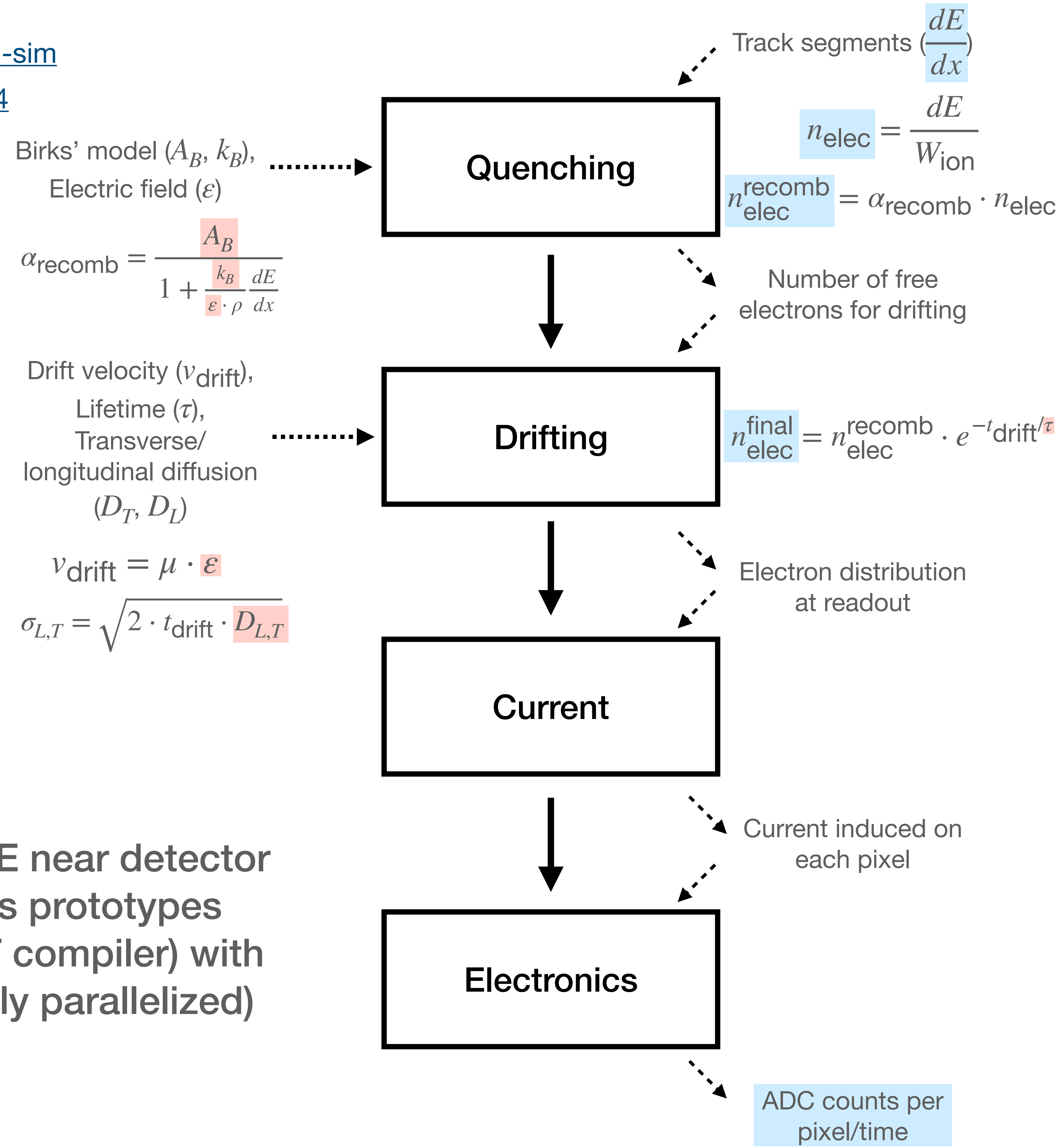
- Challenging for conventional calibration methods
- Led to development of a differentiable simulation for high dimensional calibration
    - Simultaneous optimization for multiple model parameters
    - Straightforward application of the calibration

Starting with a Default Simulation: *larnd-sim*

larnd-sim

https://github.com/DUNE/larnd-sim

JINST 18 (2023) 04, P04034

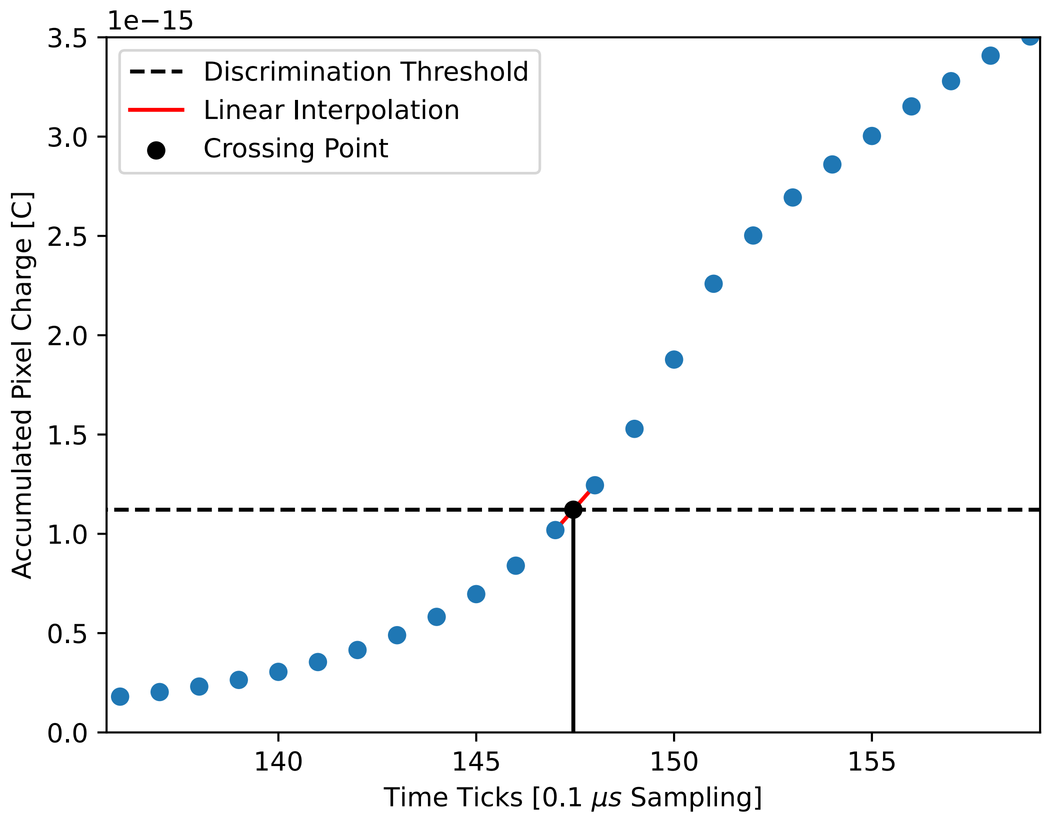


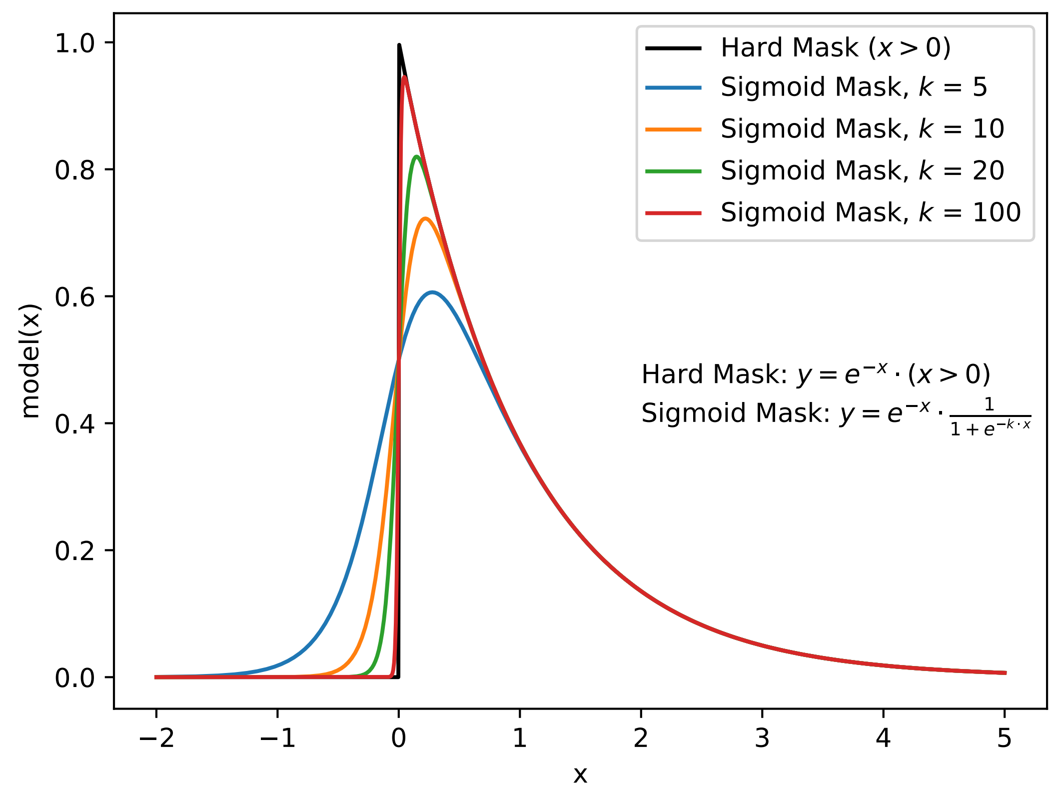
- Applicable for DUNE near detector LArTPC and all of its prototypes
- Using Numba (a JIT compiler) with CUDA kernels (highly parallelized)

Make *larnd-sim* Differentiable

https://github.com/ynashed/larnd-sim

- Rewrote using EagerPy (agnostic to automatic differentiation backend)
- Used PyTorch for this demonstration
- Wrote in a vectorized way to benefit from these frameworks
- Drawbacks:
  - Operations with dense tensors
  - Dropped from CUDA JIT compiled kernels



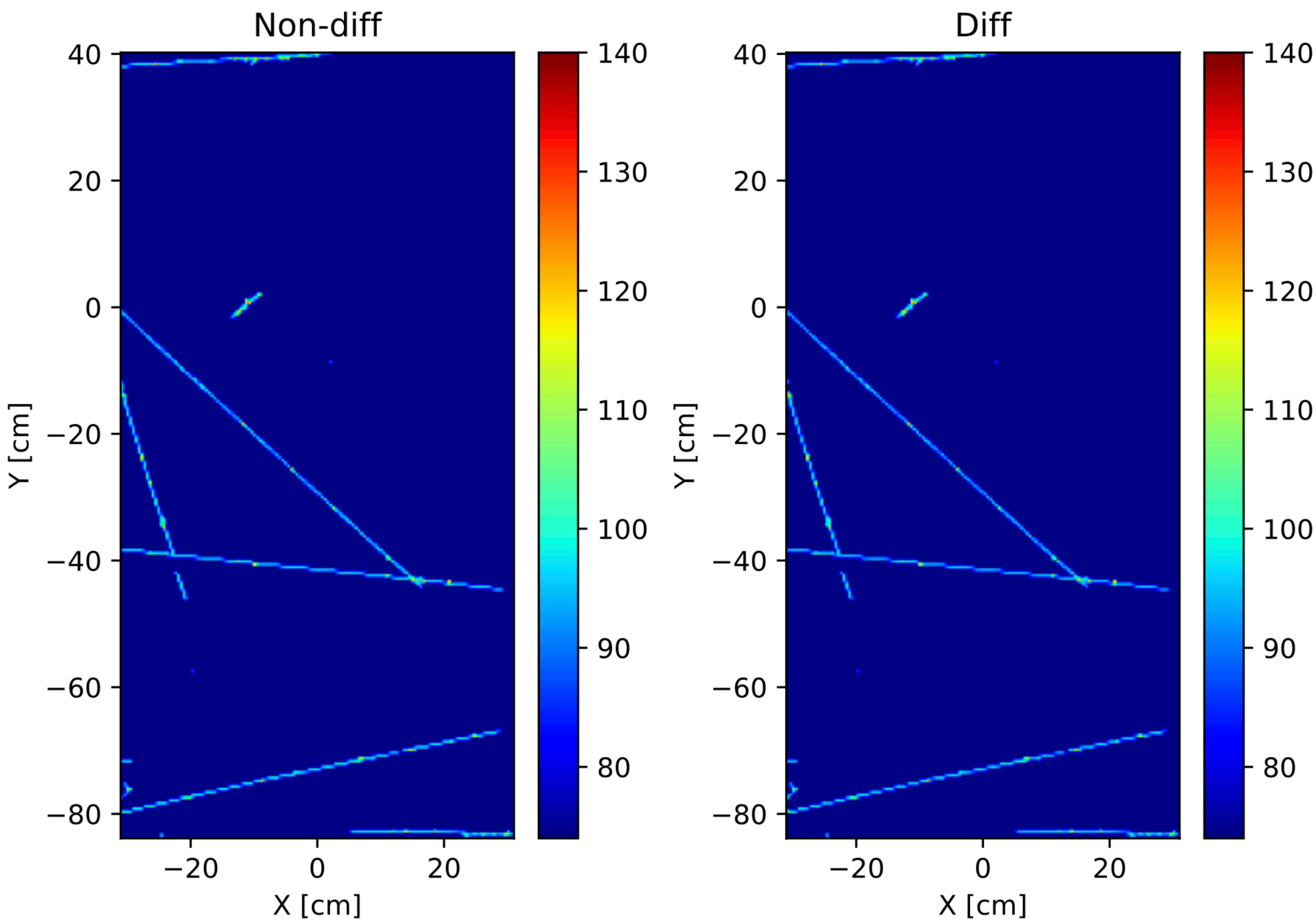


Differentiable relaxation

Important for parameter related operations

- Integer operations → floating point
- Discrete sampling → interpolation
- Conditional operation (hard mask) → sigmoid threshold

Fidelity of the Differentiable Simulation to *larnd-sim*



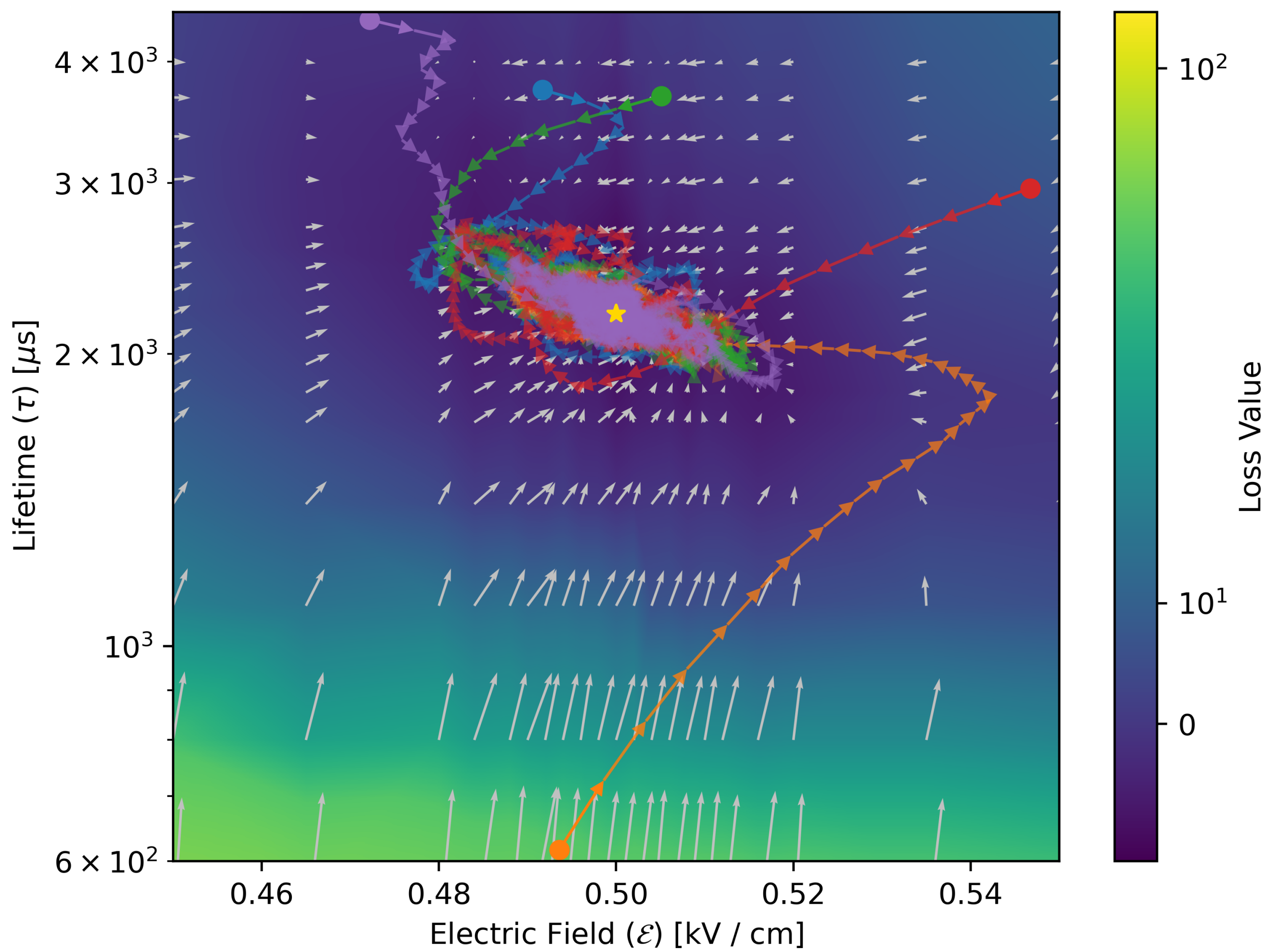
Average deviation: 0.04 ADC counts per pixel
Far below the typical noise level of a few ADC counts

Calibration: Optimization of the Model Parameters

Input particle segments (position and energy deposition): χ
Model parameters: θ
Differentiable simulation: $f(\chi, \theta)$
Target data: F_{target}

1. Choose the initial parameter values θ_0
2. Run the forward simulation $f(\chi, \theta_0)$
3. Compare the simulation output and the target data with a loss function $\mathcal{L}(f(\chi, \theta_0), F_{\text{target}})$
4. Calculate gradients for the parameters $\nabla_\theta \mathcal{L}(f(\chi, \theta_0), F_{\text{target}})$
5. Update parameter values $\theta_0 \rightarrow \theta_i$ to minimize the loss

Iterate step 2. to 5.



For gradient descent, the parameter update takes form of

$$\theta_{i+1} = \theta_i - \eta \cdot \nabla_\theta \mathcal{L}(f(\chi, \theta_i), F_{\text{target}})$$

For this demonstration, we use Adam

Loss Function: soft DTW

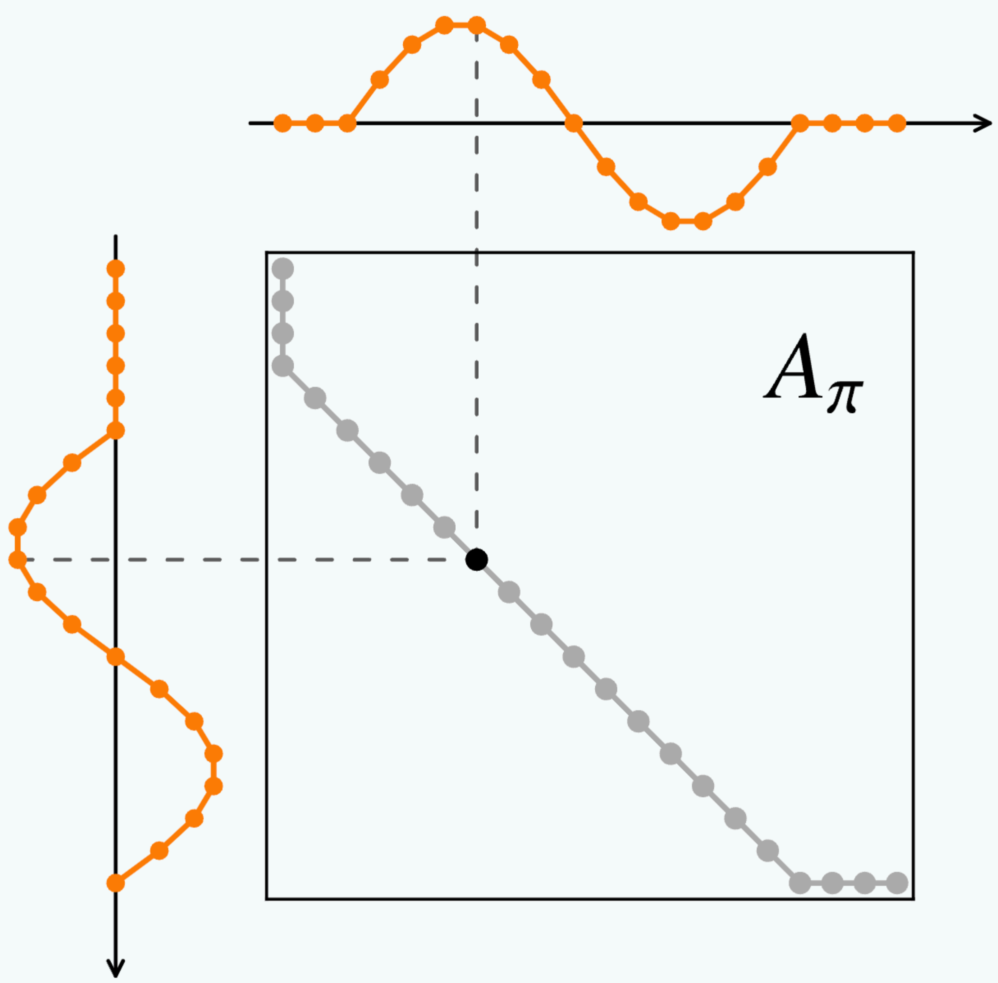
The choice of the loss function is important to the fit performance.

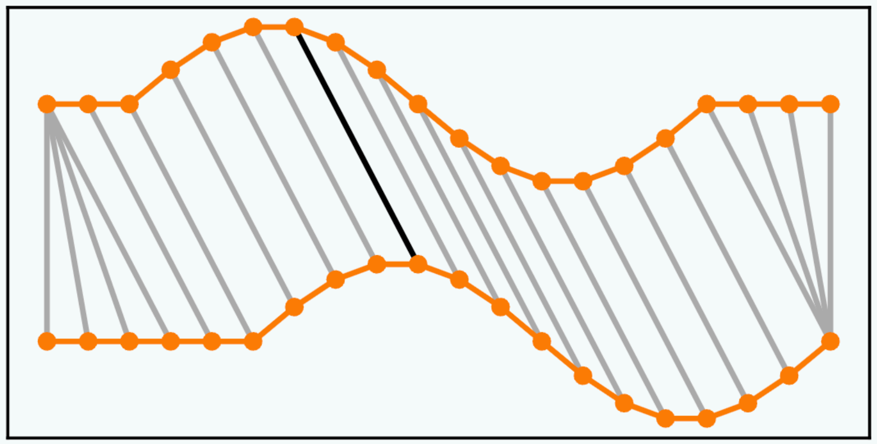
- High discriminating power of the best parameter values
- Differentiable

Challenges

- Potentially different length of the simulation output and the target data
- Obscure correspondence between hits from the two sets

Dynamic Time Warping (DTW) addresses the alignment challenges.
Soft DTW is a differentiable version of it.





Romain Tavenard
Post: DTW
Post: Soft DTW

Fitting Considerations

Parameter [Units]	Nominal Value	Range
A_B	0.8	[0.78, 0.88]
k_B $[kV.g/cm^3/MeV]$	0.0486	[0.04, 0.07]
$\mathcal{E}$ $[kV/cm]$	0.5	[0.45, 0.55]
τ $[\mu s]$	2200	[500, 5000]
D_L $[cm^2/\mu s]$	4×10^{-6}	$[2 \times 10^{-6}, 9 \times 10^{-6}]$
D_T $[cm^2/\mu s]$	8.8×10^{-6}	$[4 \times 10^{-6}, 14 \times 10^{-6}]$

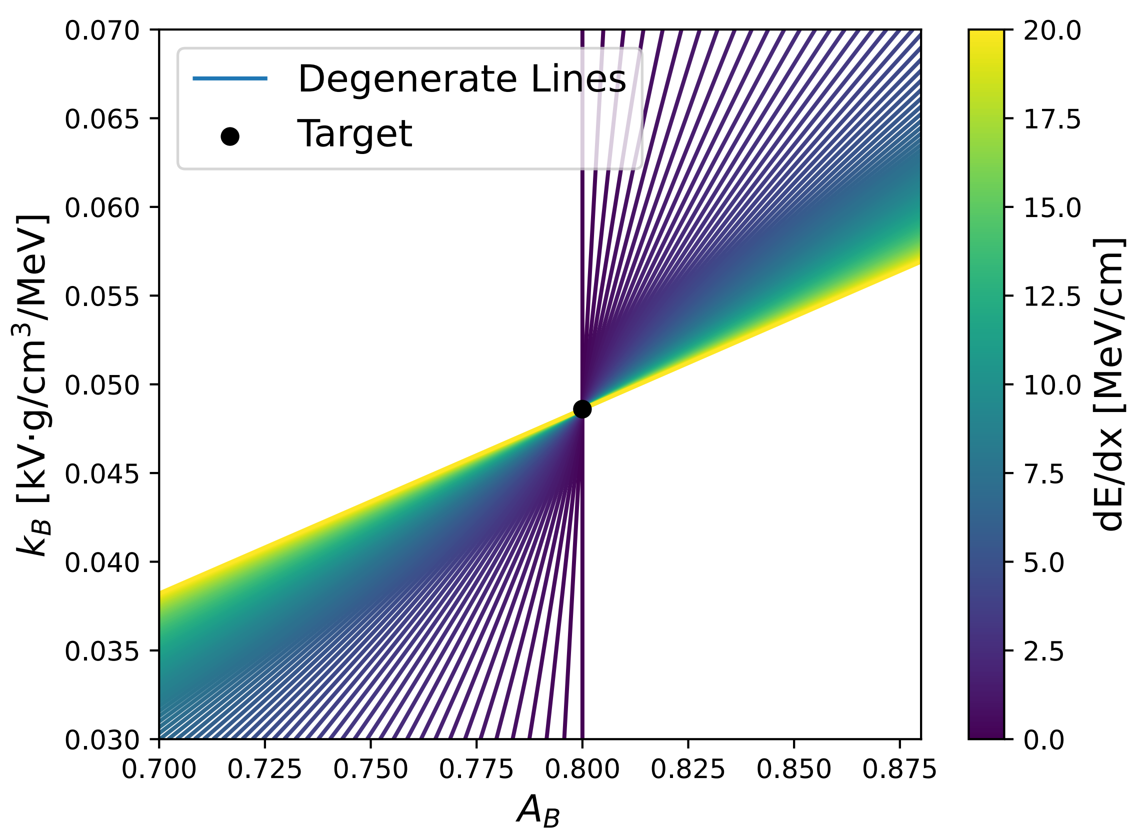
- Normalize the parameters with their nominal values for gradient calculation
- Gradient clips on normalized gradient
- Use the same learning rate for all parameters
- Exponential learning rate decay
- Recover the parameter values for the forward simulation

Potential degeneracy in the backward model

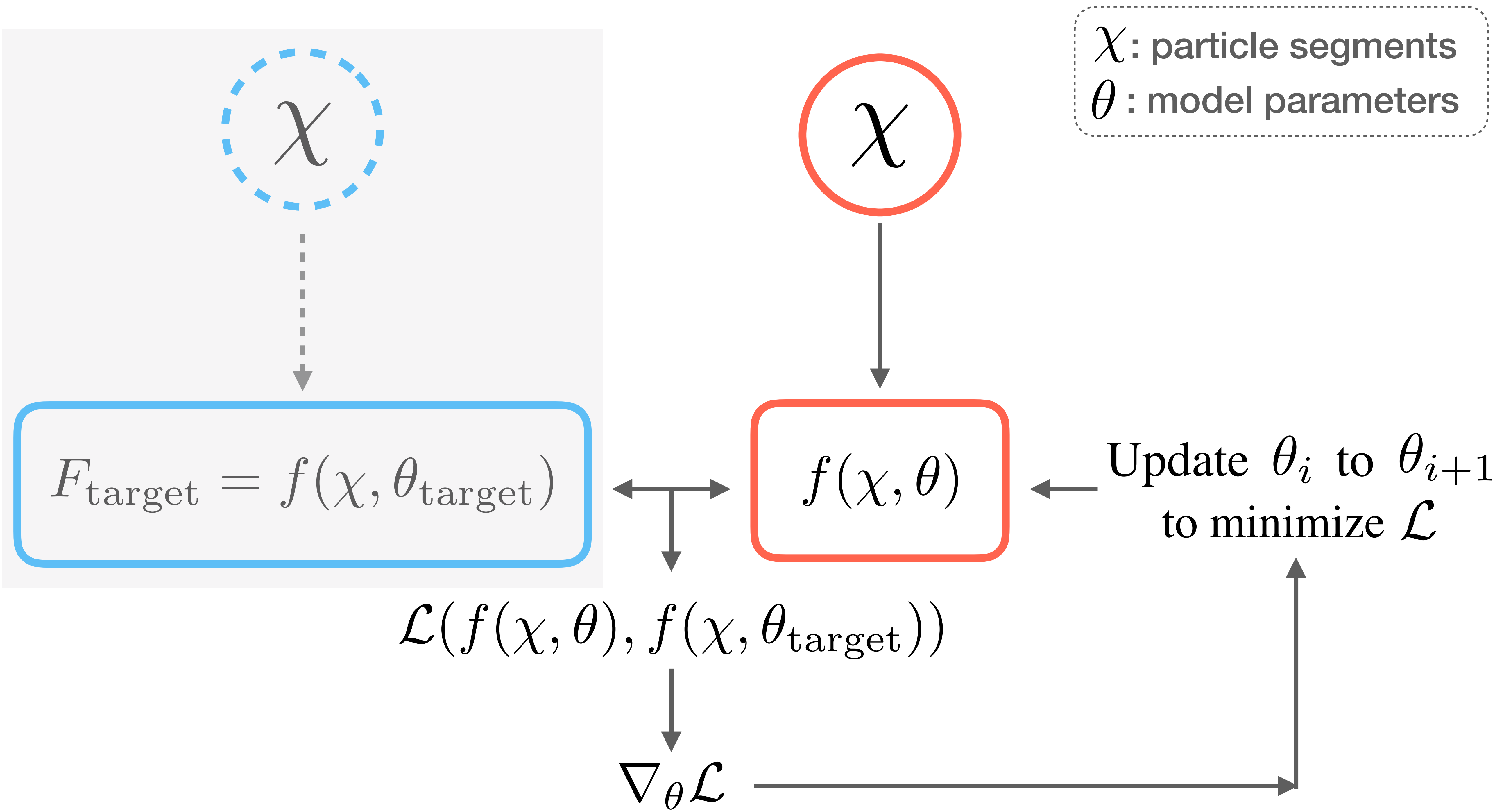
Electron recombination - Birks model

$$\alpha_{\text{recomb}} = \frac{A_B}{1 + \frac{k_B}{\mathcal{E}\rho}\frac{dE}{dx}}$$

$$A_B = \alpha^*_{\text{recomb}} \cdot \left(1 + \frac{k_B}{\mathcal{E} \cdot \rho}\frac{dE}{dx}\right)$$



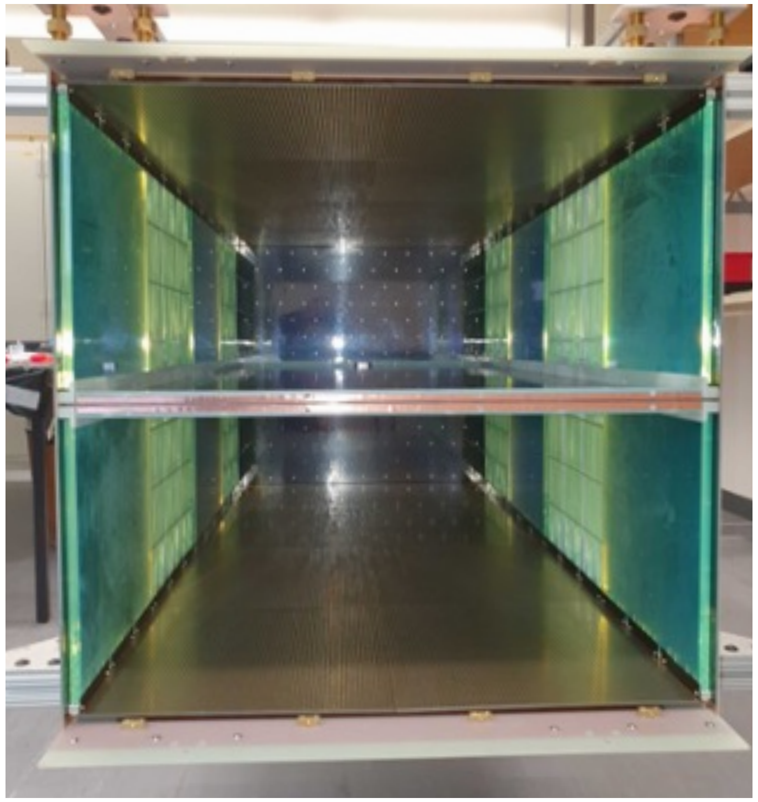
Closure Test for the Fit



Evaluate if θ converge to θ_{target}

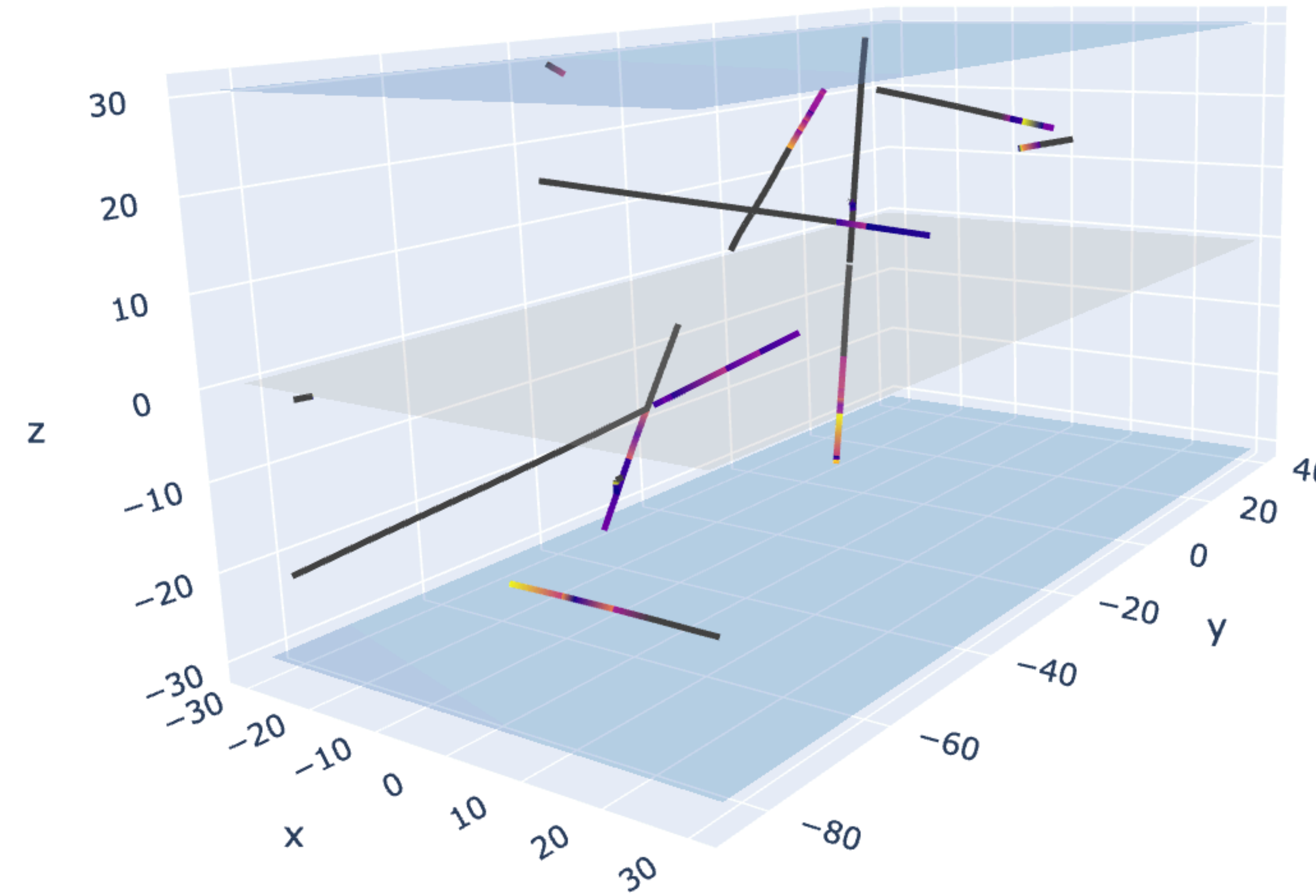
Samples and Selection

DUNE



Default sample

Particle segments



Default sample:

100 events of ~10 muons with 1 GeV kinetic energy (K.E.)

Alternative sample:

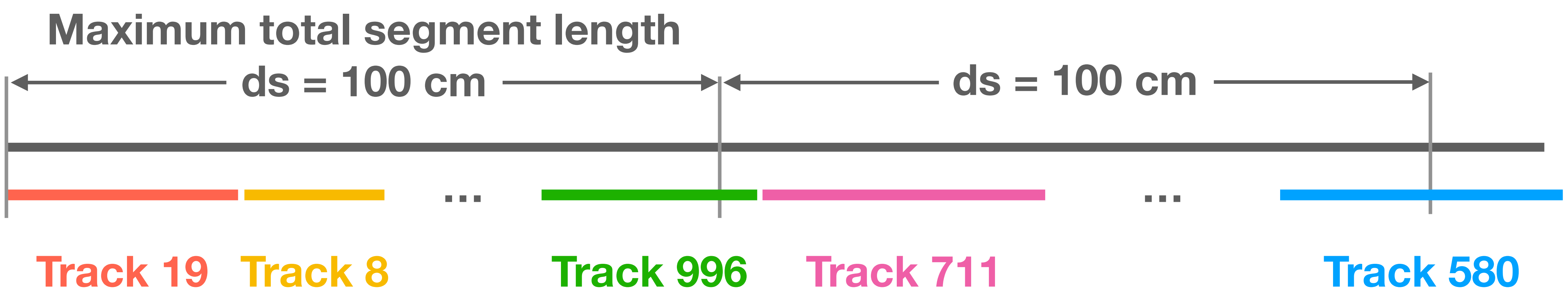
100 events of ~10 mixed particles (muons, charged pions, protons) with [0.1, 2] GeV K.E.

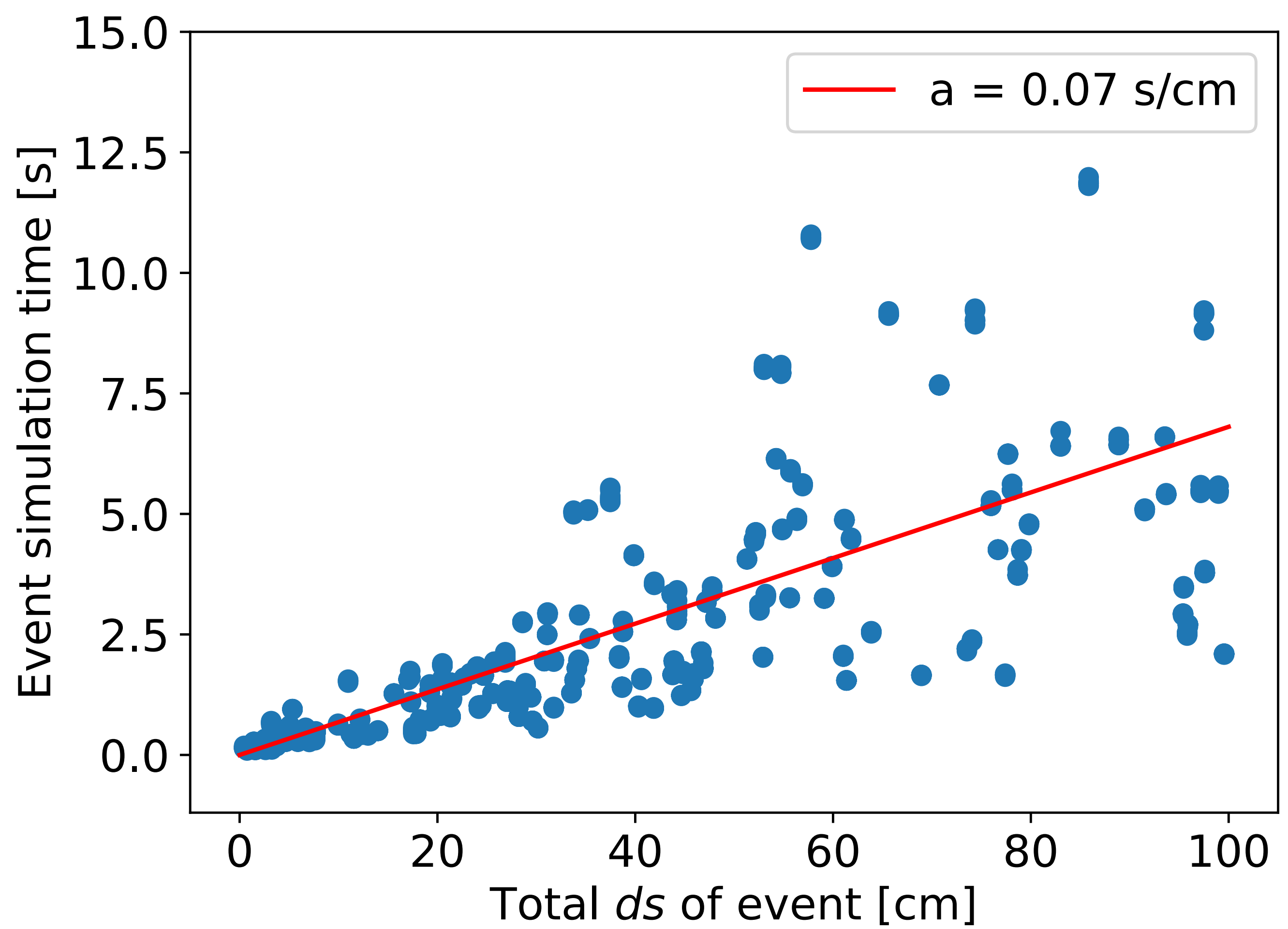
Selection:

- Track length > 2 cm
- abs(track segment z) in [15, 28] cm (near the anodes)
- Track angle wrt. z larger than 15°

Mini-batching

Each fitting iteration runs on a mini-batch





Memory Usage

Peak memory usage in track current calculation

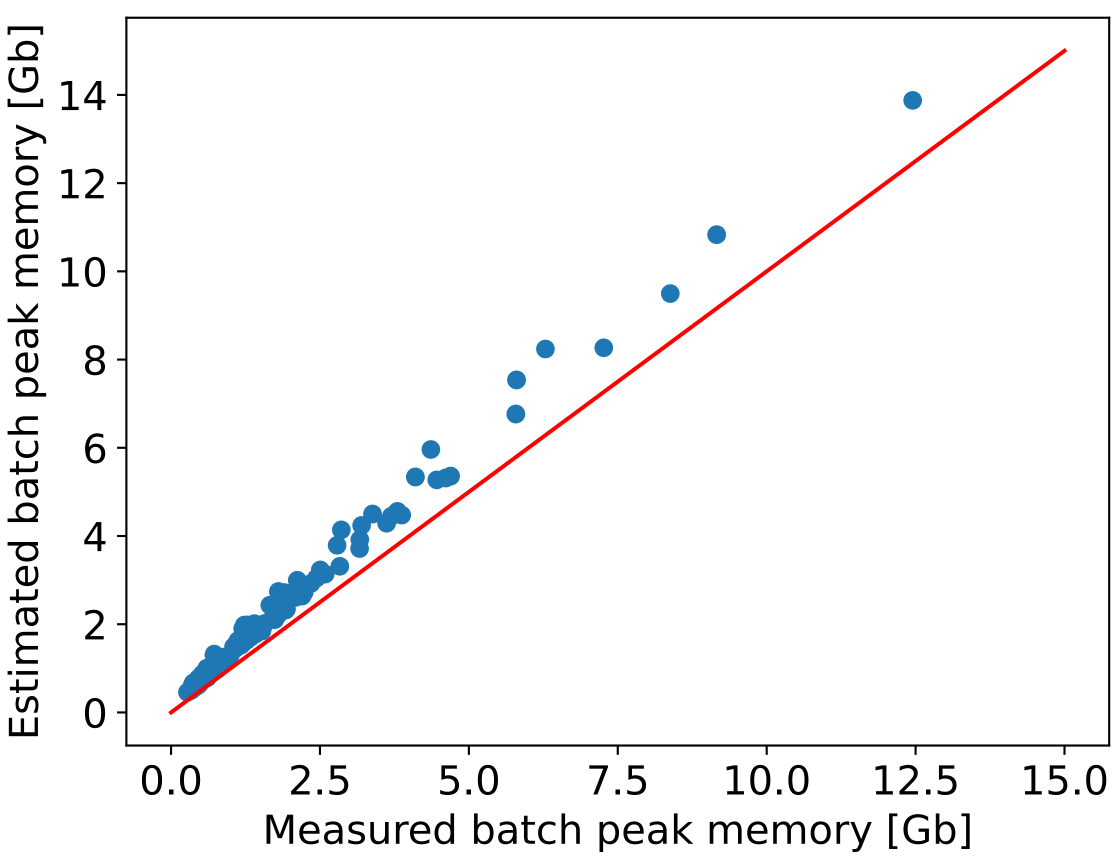
loop (over N_{segments}) — loop (over N_{pixels})

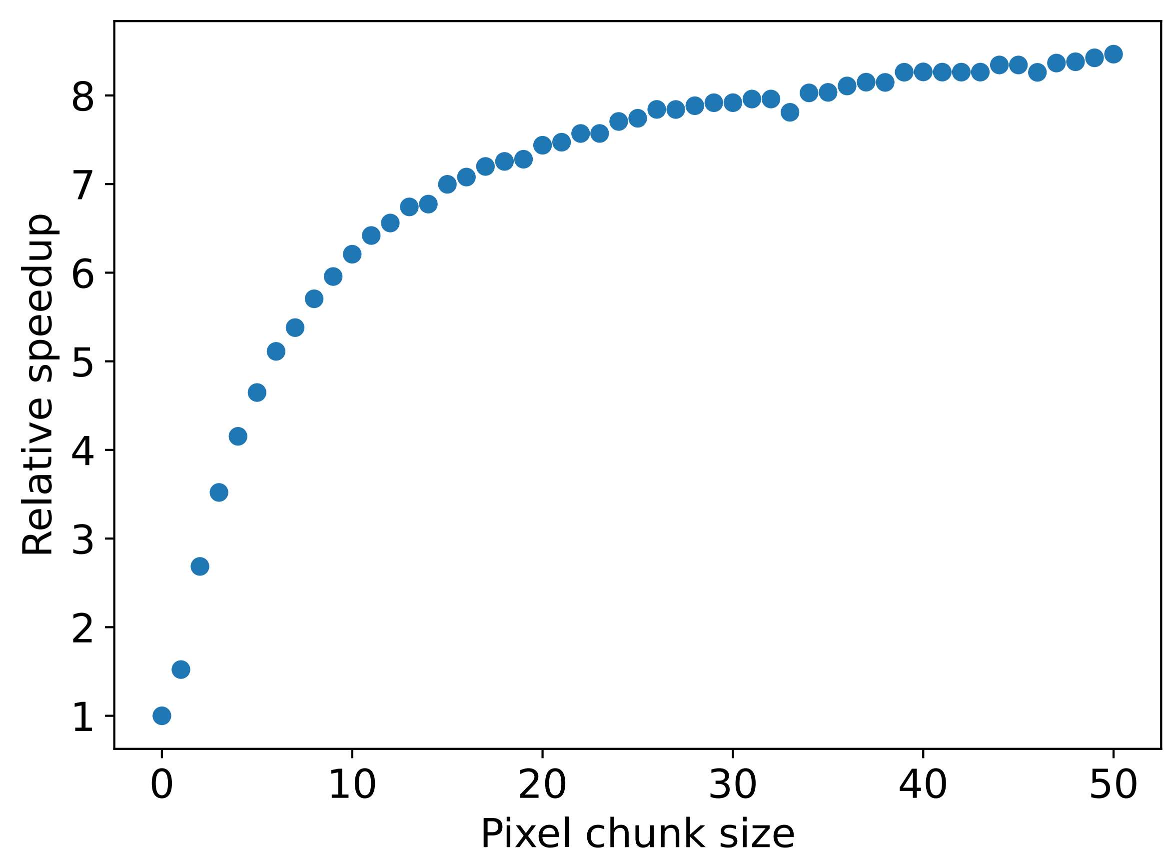
$$\mathcal{M} = N_{\text{segments}} \times N_{T_f} \times N_{T_0} \times N_x \times N_y \times N_{\text{pixels}}$$

chunk size 1 — adjustable chunk size

Trade-off between memory and computation time

- Shrink the tensor dimension by loops
- “Chunking” operation to reduce loop iterations
- Gradient checkpoint



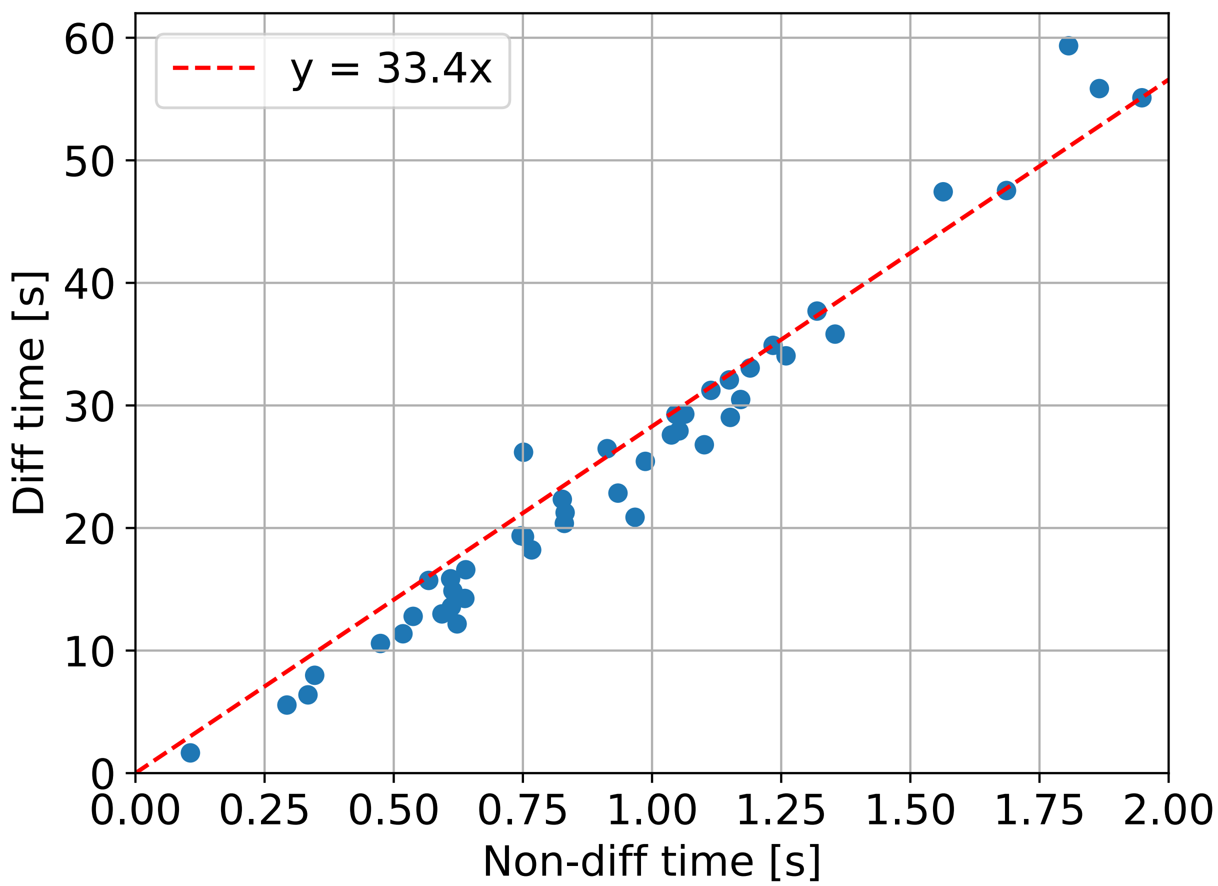


Computation Time

Modular nature of the detector — highly parallelizable

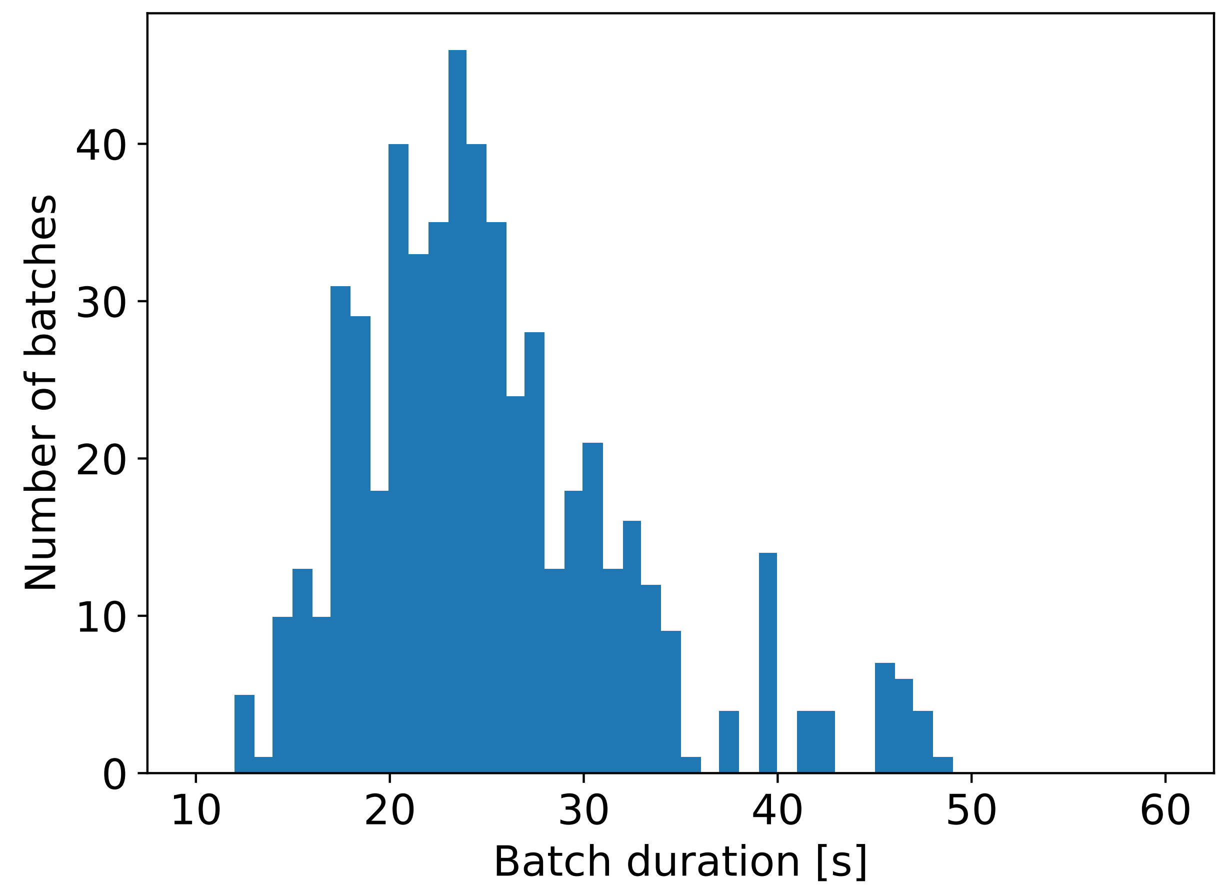
Working on translate the diff-sim using JAX to speed up the simulation

Simulation time per event



Forward simulation

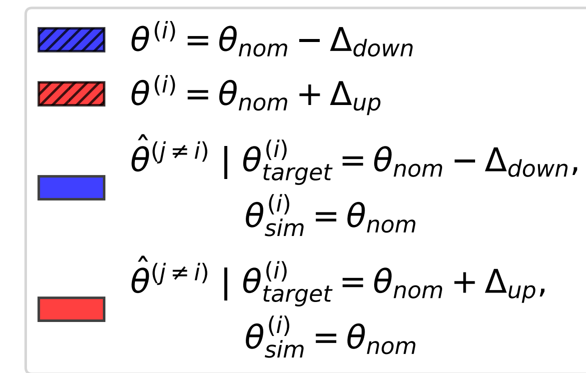
Time per min-batch iteration



Forward simulation + gradient calculation

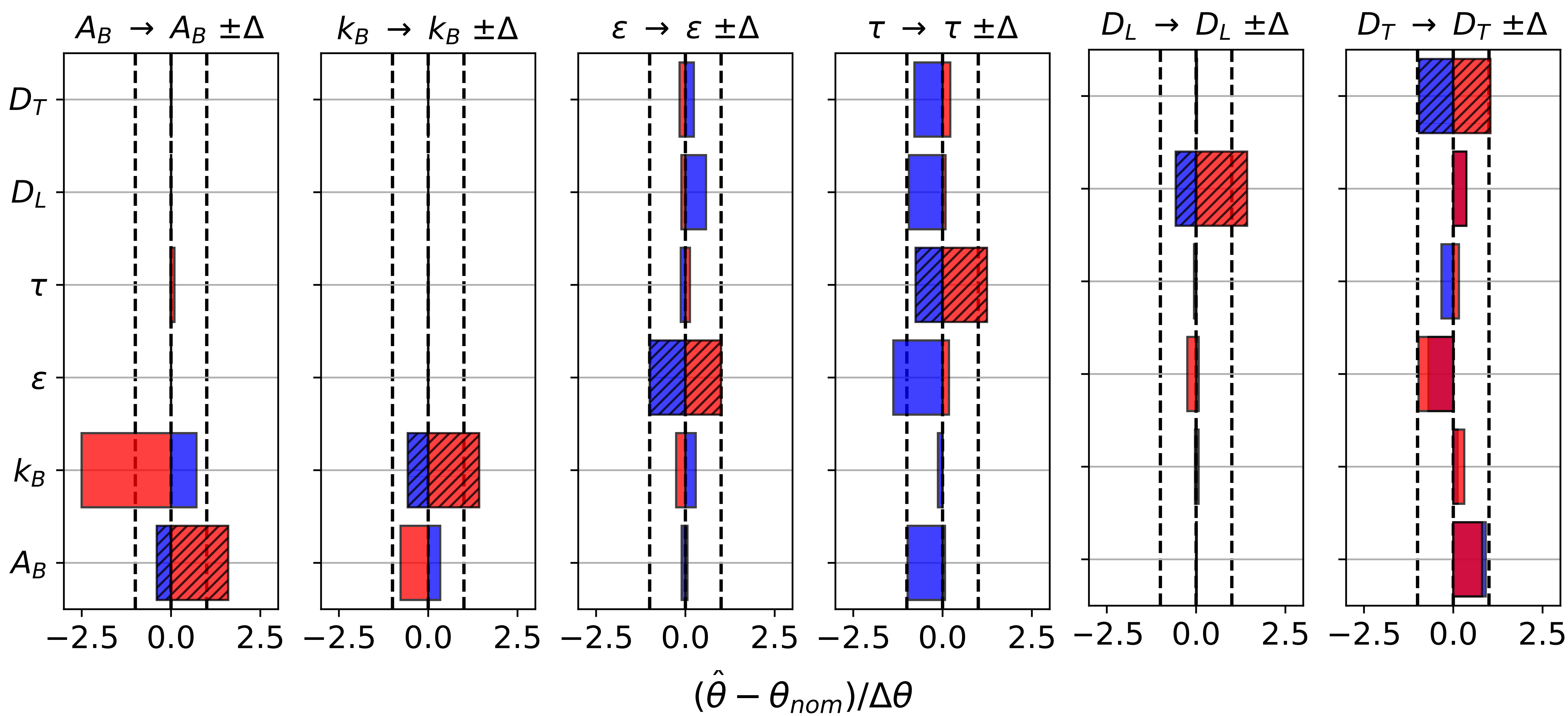
No Fully Independent Parameters

Parameter [Units]	Nominal Value	Range
A_B	0.8	[0.78, 0.88]
k_B $[kV.g/cm^3/MeV]$	0.0486	[0.04, 0.07]
$\mathcal{E}$ $[kV/cm]$	0.5	[0.45, 0.55]
τ $[\mu s]$	2200	[500, 5000]
D_L $[cm^2/\mu s]$	4×10^{-6}	$[2 \times 10^{-6}, 9 \times 10^{-6}]$
D_T $[cm^2/\mu s]$	8.8×10^{-6}	$[4 \times 10^{-6}, 14 \times 10^{-6}]$



$$\Delta\theta = \frac{\Delta_{up} + \Delta_{down}}{2}$$

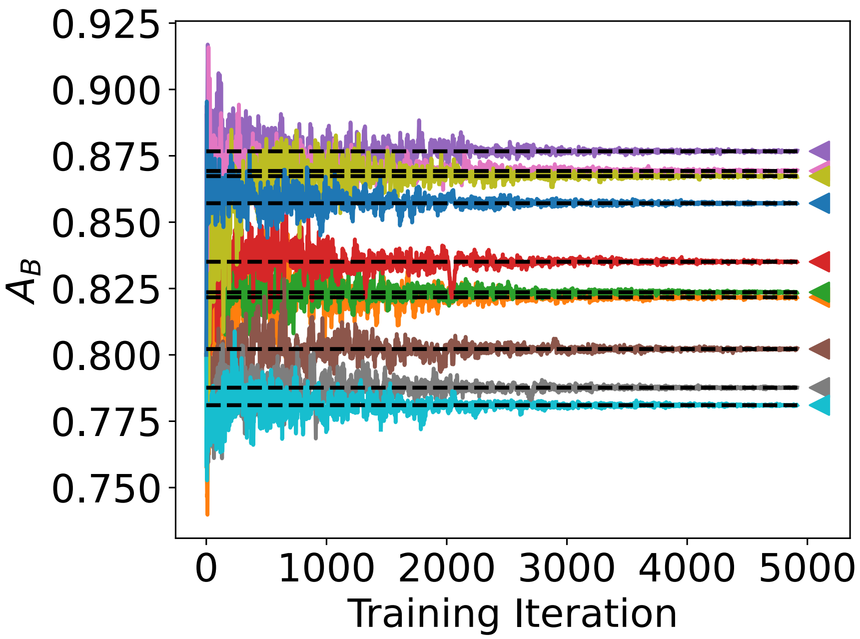
$\hat{\theta}$ = Fitted Value



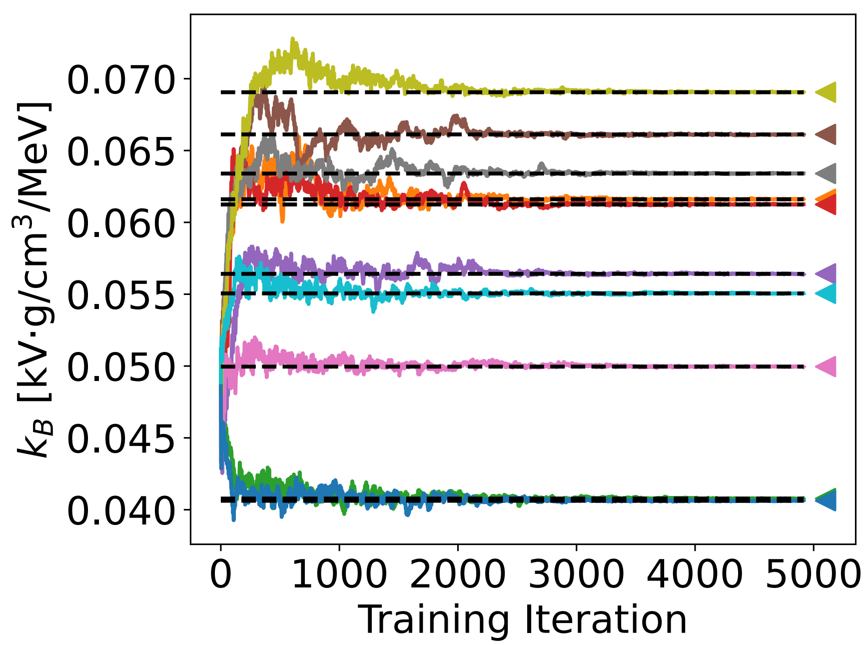
Simultaneous Fit Result

10 fits with different targets in 6D phase space
All 6 parameters of interest converge to the target values

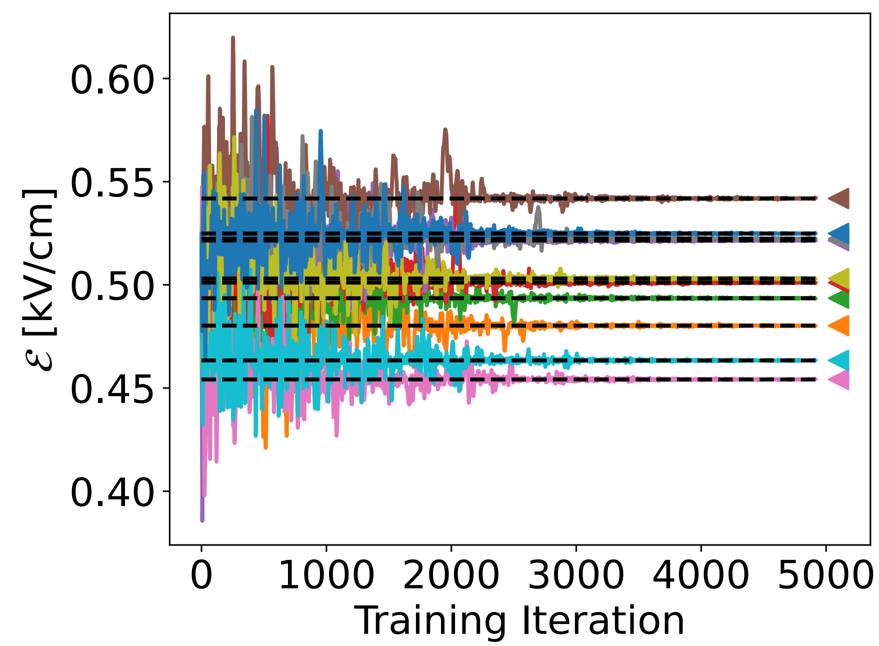
Recombination model A_B



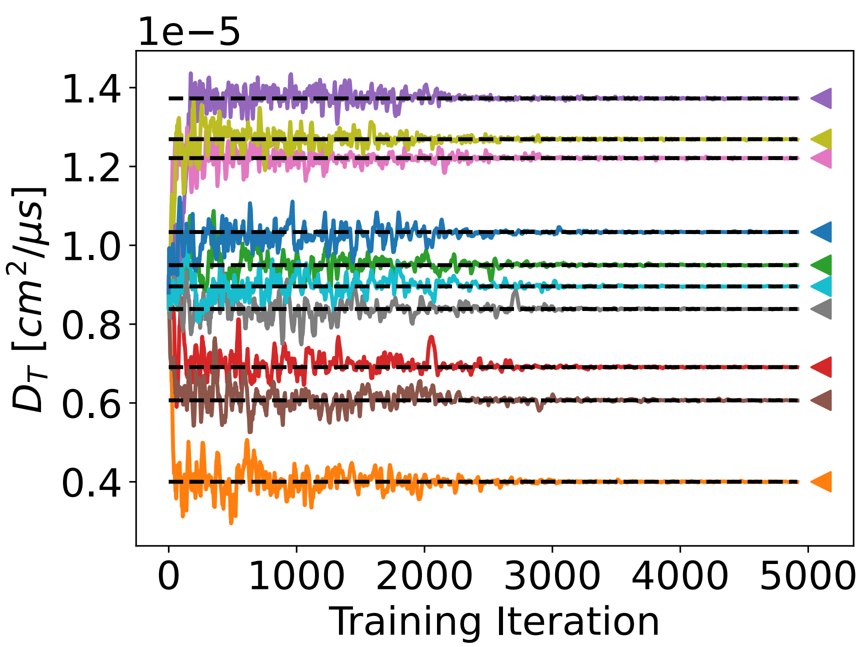
Recombination model k_B



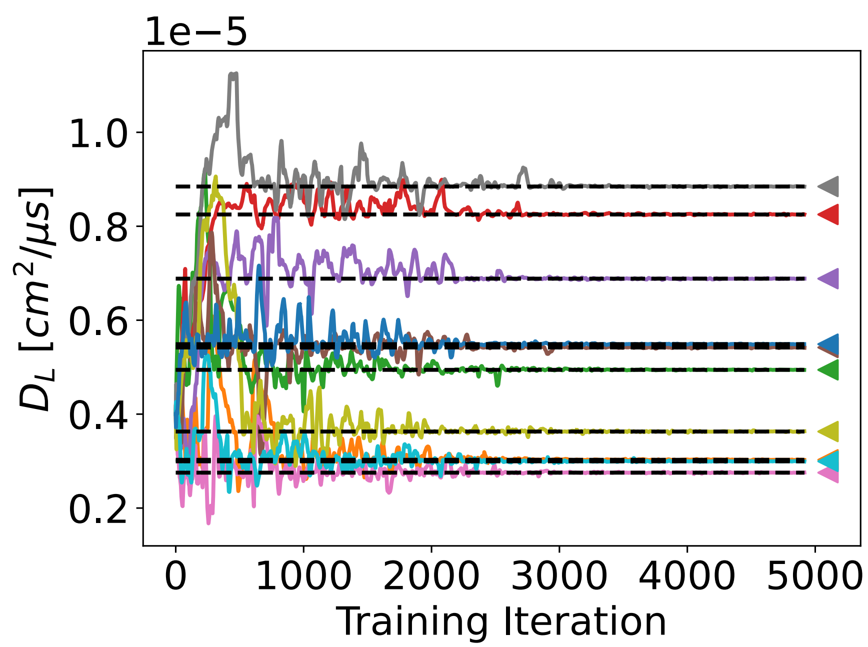
Electric field



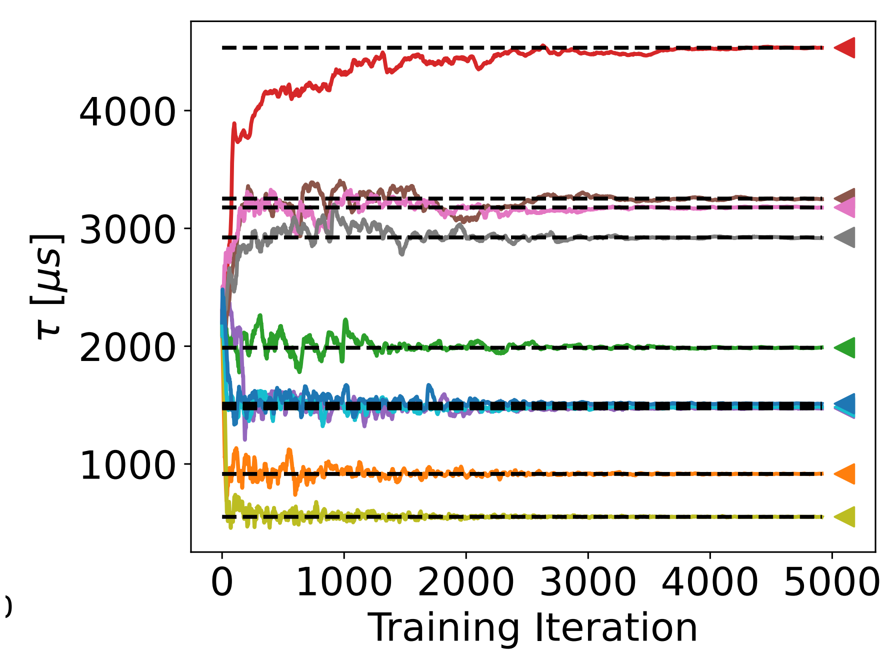
Transverse diffusion coefficient



Longitudinal diffusion coefficient

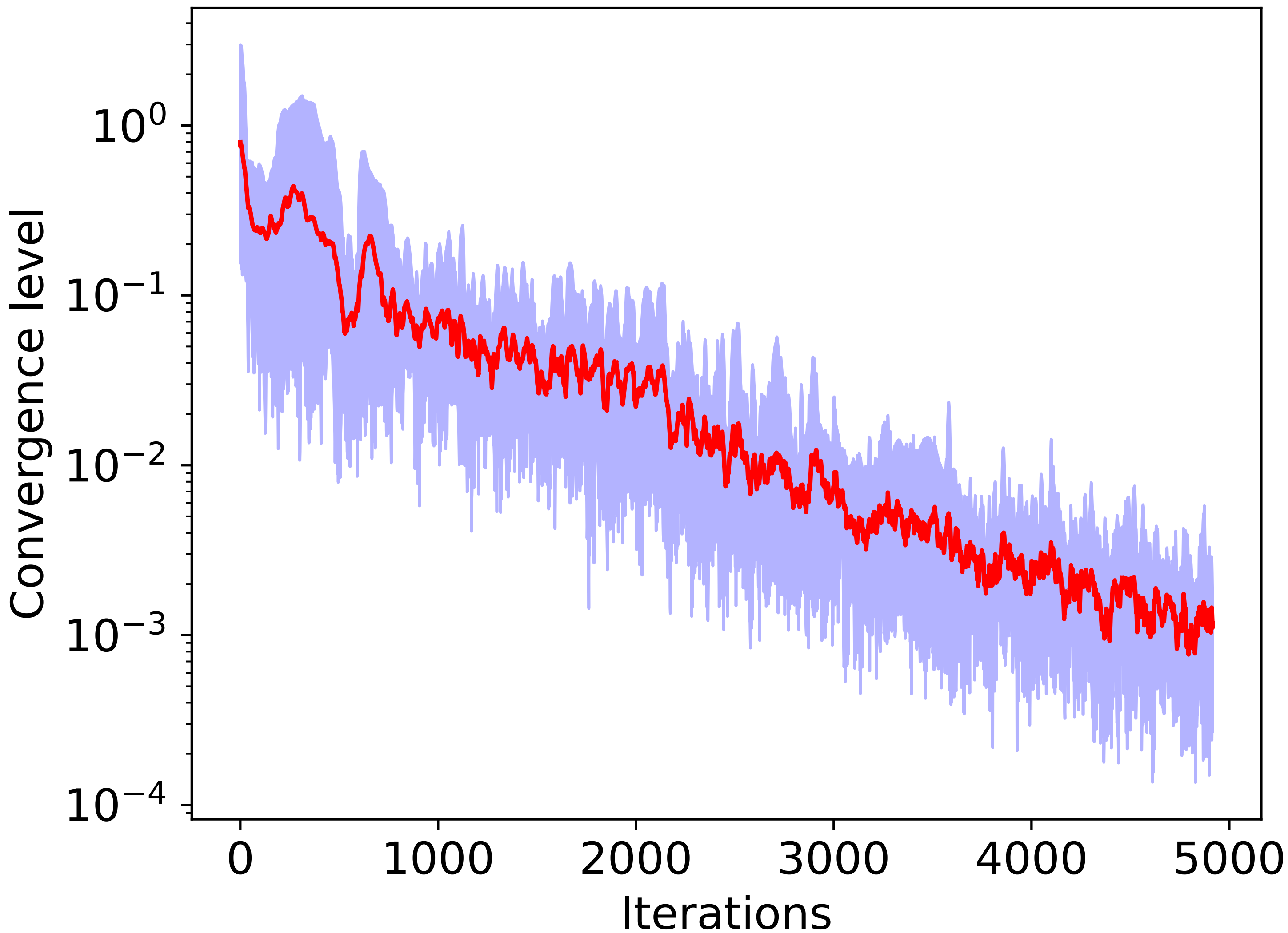


Electron lifetime

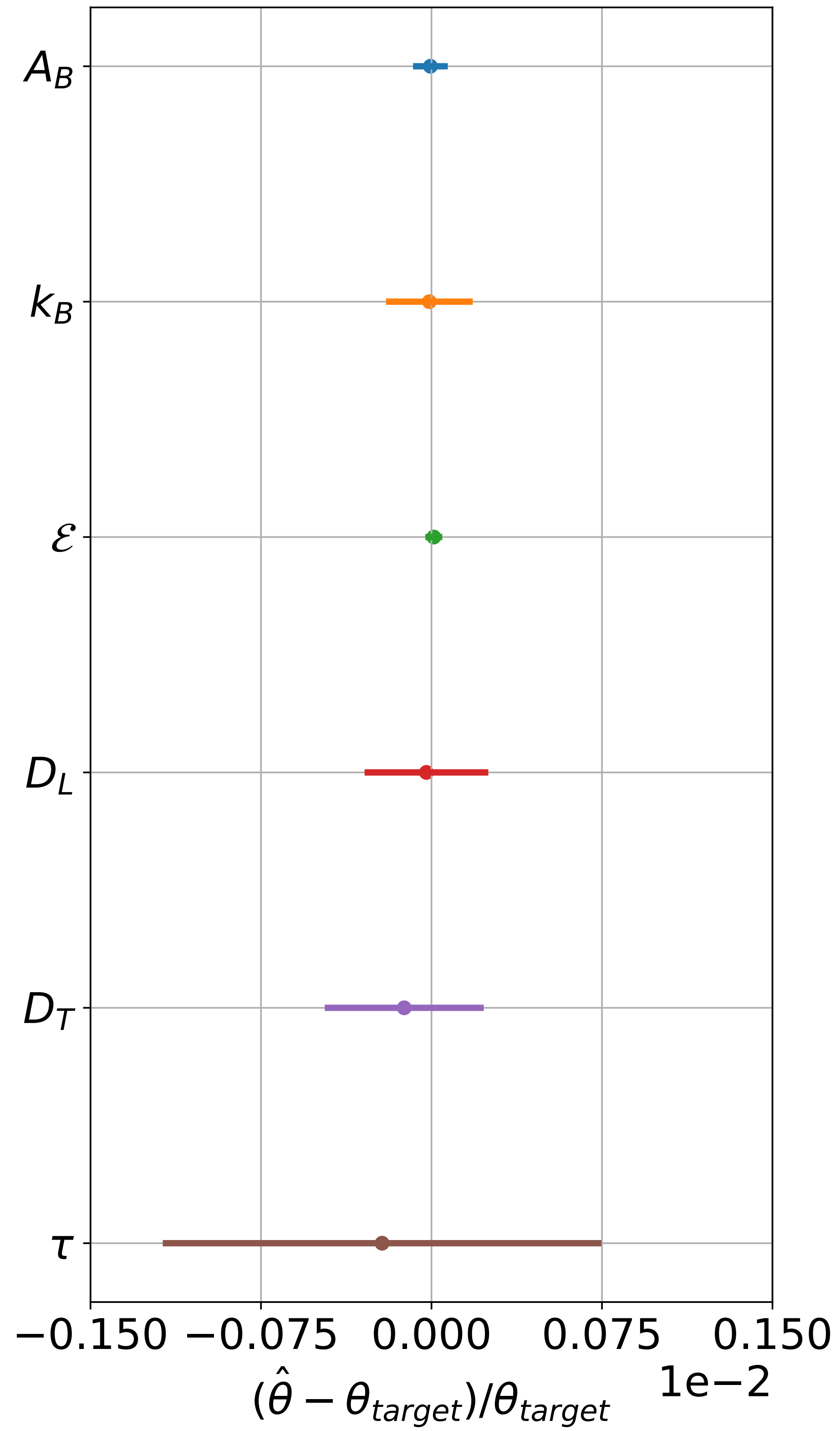


Convergence level

Convergence evolution



Final convergence



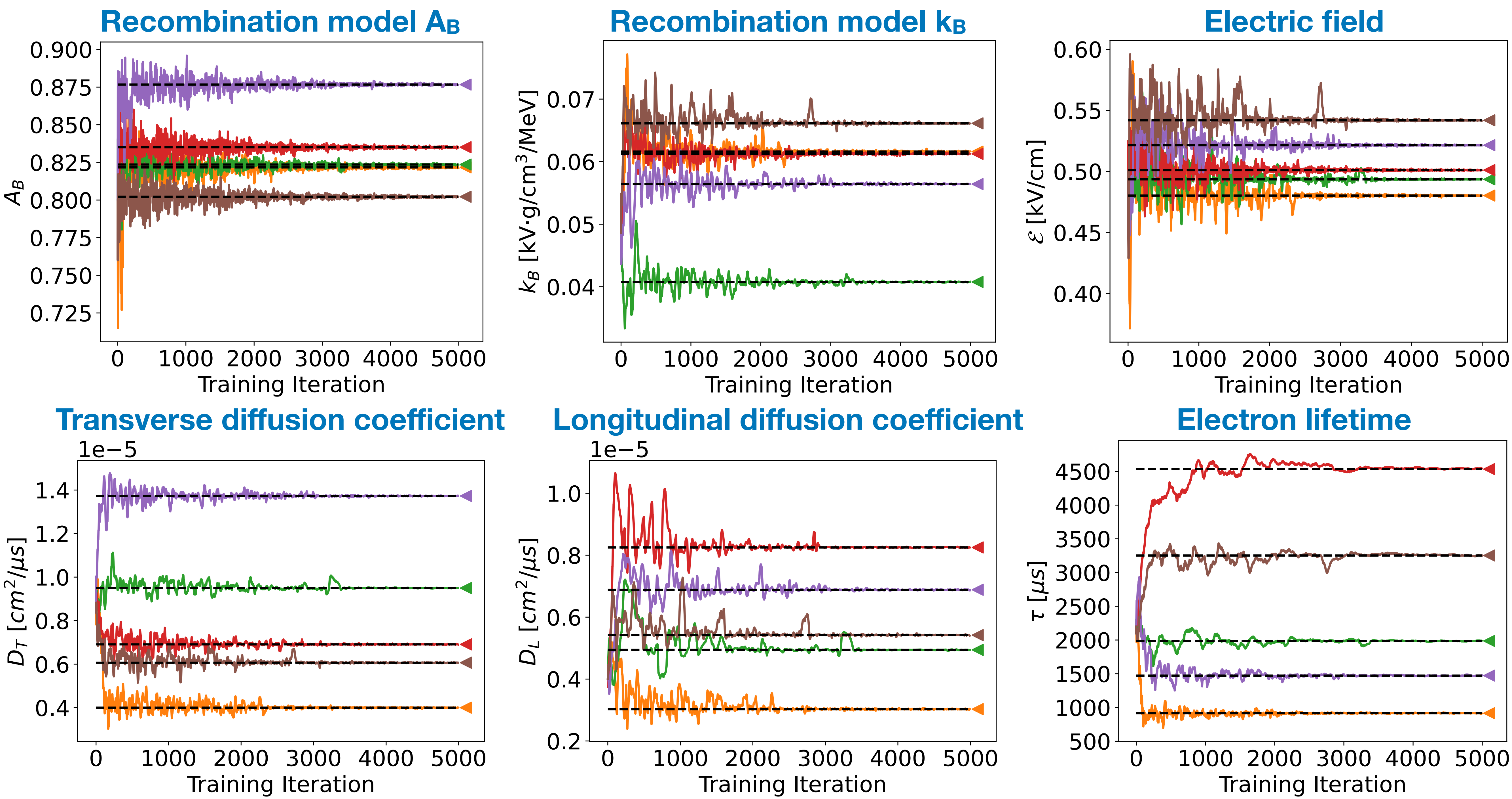
Result with the Alternative Sample

5 different targets in 6D phase space

All 6 parameters of interest converge to the target values

Relaxed requirement on Inputs

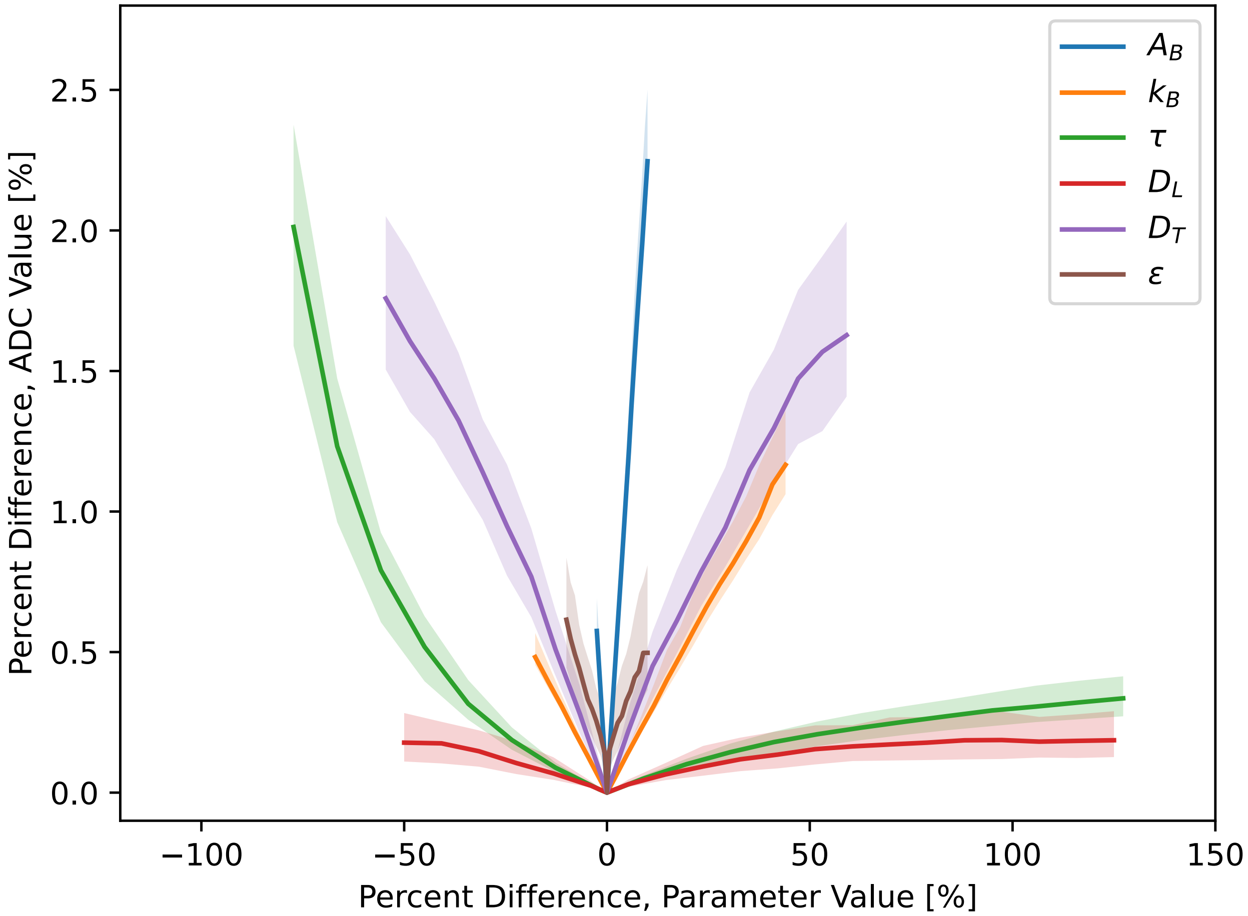
(Muons, charged pions and protons of [0.1, 2] GeV kinetic energy)



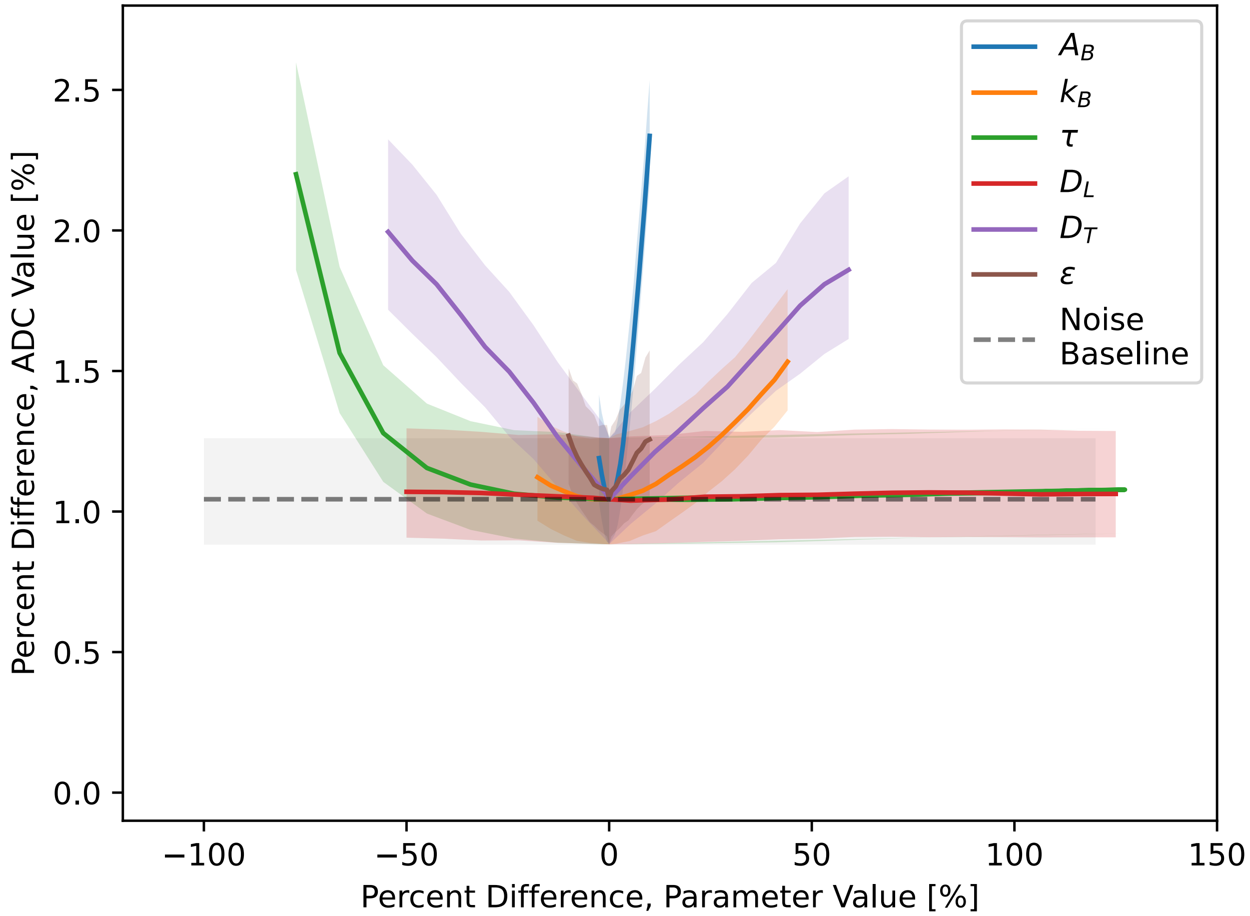
Remaining Challenges: Electronic Noises

The data (the target) will always contain electronic noises.
We will fit without noise in the forward simulation to reduce stochasticity

Target without noise



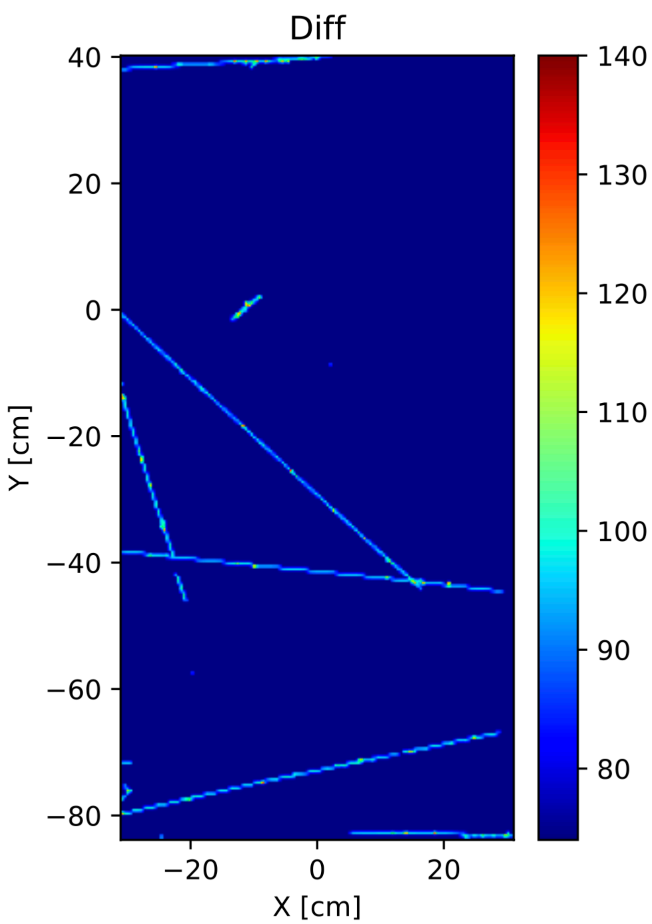
Target with noise

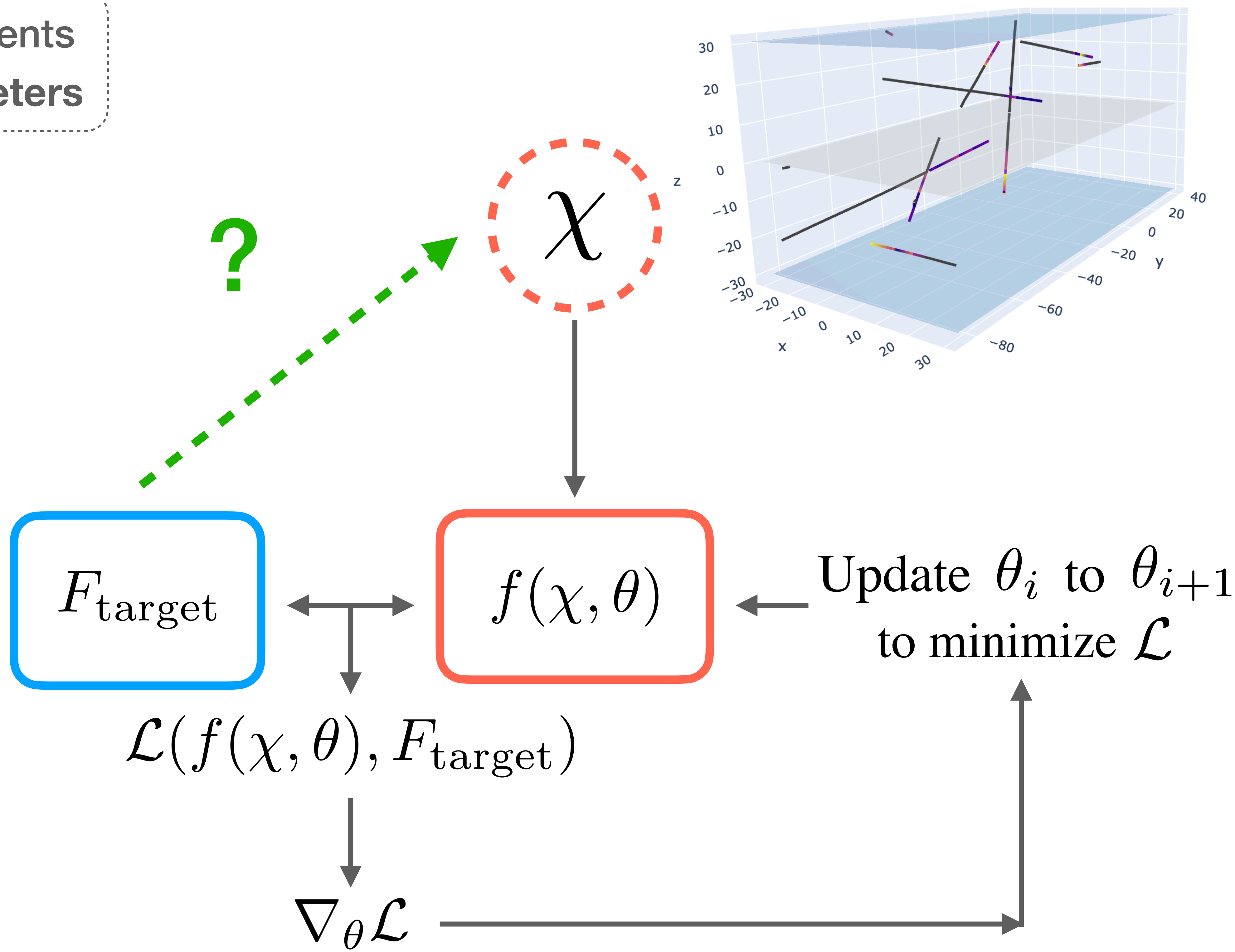


Stochasticity enhances batch-to-batch variations.
Require the impact of the parameters to exceed the one from the noises.

Remaining Challenges: Input Estimation

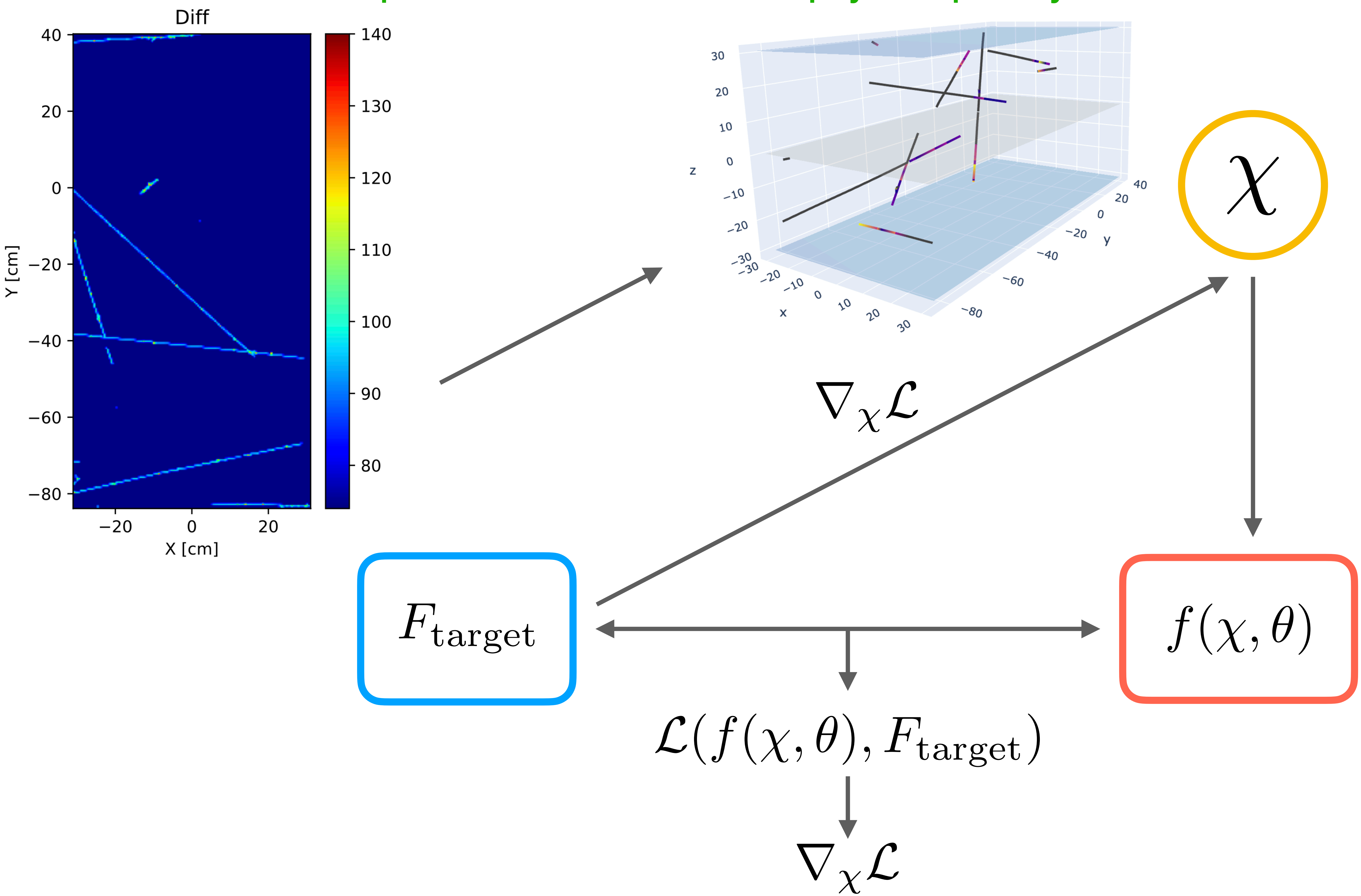
χ: particle segments
θ: model parameters





Future Application: Inverse Solver

Map from detector readout to physics quantity



Summary

- Emerging area: New approaches to conduct experiments, including detector optimization, simulation, calibration, reconstruction, analysis…
- A demonstrator of differentiable simulation using LArTPCs (a DUNE near detector prototype)
- Aim for high-dimensional detector calibration
- Made *larnd-sim* (a simulator for DUNE near detector LArTPC and its prototypes) differentiable
- A successful closure test for a simultaneous calibration fit of 6 detector parameters
- Future development towards data application