PNPInverse Weekly Development Writeup
Week of February 16, 2026

Jake Weinstein (Project Notes Consolidated)

February 19, 2026

Weekly Focus

This week focused on converting the inverse workflow from isolated scripts into a modular framework, then building a new experiment-style inference path for Robin boundary transfer coefficients from ϕ_{applied}-vs-steady-flux data.

The center of gravity was the new Robin curve inference path: data generation, steady-state detection, adjoint gradient extraction point-by-point, SciPy-based optimization, and recovery logic for forward-solve failures.

Outline of What Was Implemented

1. Unified inverse interface that can plug into compatible forward solvers and infer multiple parameter targets.

2. Always-resilient optimization logic that can recover from nonlinear forward-solve failures.

3. Robin experimental pipeline where the observable is steady-state flux on the Robin boundary.

4. New gradient-based Robin κ inference from flux-curve mismatch using Firedrake adjoints + SciPy.

5. Current-density-proxy inference branch using charge-weighted flux $\sum_i z_i F_i$, with Faraday scaling intentionally omitted pending unit calibration.

6. Regenerated report set for 0%, 2.5%, and 5% noise in both data-generation and inference-fit formats, plus a 2.5% convergence GIF artifact.

1. Unified Inverse Interface

What was built

A shared inference core was established so new inverse problems can be configured through:

- a forward-solver adapter,

- a parameter target definition,

- a request object containing true value, initial guess, noise, and optimizer settings.

This unifies diffusion, Dirichlet BC, and Robin BC inference under one architecture while preserving compatibility wrappers for existing helper imports.

Mathematical template

For state $u(m)$ produced by the forward solve under control/parameter m, the generic objective template is:

$$J(m) = \frac{1}{2} \int_\Omega \|\mathcal{O}(u(m)) - d\|^2 \, dx,$$

where $\mathcal{O}$ selects the measured field(s) (concentration, potential, or derived observable), and d is synthetic/experimental target data.

2. Resilient Optimization for Forward-Solve Failures

What was built

The optimization loop was made always-resilient by adding staged recovery rather than hard-failing on the first diverged PDE solve. Recovery phases were implemented in this order:

1. Increase `snes_max_it`.

2. Apply anisotropy reduction to the trial parameter vector (with reset of relaxed tolerances).

3. Relax nonlinear/linear tolerances (`snes_atol`, `snes_rtol`, `ksp_rtol`) and line-search strategy.

The restart logic uses the last known feasible parameter state, so retries do not repeatedly jump back to the original initial guess.

Anisotropy reduction concept

For a multi-component parameter vector m, anisotropy is treated as a max/min magnitude ratio. If too large, the vector is blended toward an isotropic surrogate based on geometric-mean magnitude:

$$m_{\text{new}} = (1 - \beta)m + \beta m_{\text{iso}},$$

where $\beta \in [0, 1]$ controls flattening strength.

3. Robin Experimental Flux Pipeline

What was built

A new workflow mirrors an experiment where:

- control input is applied voltage ϕ_{applied},

- measured output is steady-state flux through the Robin boundary.

Utilities were added for:

- steady-state probe sweeps,

- synthetic ϕ_{applied}-flux data generation,

- CSV serialization and reuse across fitting runs.

Steady-state flux definition

The Robin boundary law in this implementation is:

$$J_i \cdot n = \kappa_i(c_i - c_{\infty,i}).$$

Species flux across the Robin electrode boundary is assembled as

$$F_i = \int_{\Gamma_{\text{electrode}}} \kappa_i(c_i - c_{\infty,i})\, ds.$$

The default scalar observable used this week is total species flux:

$$F_{\text{obs}} = \sum_i F_i.$$

From Modeled Flux to Experimental Current Density

Experimentally, the directly measured quantity is typically current density (mA/cm^2), not molar flux. In this workflow, the key relation is:

$$\text{charge-weighted flux } \propto \sum_i z_i F_i,$$

so to move from total species flux to a current-like observable, we weight each species contribution by its charge and sum.

If physical current density units are needed, this weighted sum is then scaled by constants (Faraday factor and boundary-area conversion). For fitting, the important structural step is the charge weighting itself.

Current implementation note (temporary): the new current-density inference branch currently omits the Faraday factor on purpose and fits a charge-weighted flux proxy $\sum_i z_i F_i$ in arbitrary units. This keeps curve magnitudes in the same $\mathcal{O}(1)$ range as the flux-based studies while unit conventions are being finalized. Therefore, the present proxy scaling is not yet a physically calibrated mA/cm^2 mapping.

Steady-state criterion

At each time step n, with per-species boundary flux $F_i^{(n)}$:

$$\Delta_i^{(n)} = |F_i^{(n)} - F_i^{(n-1)}|,$$

$$\mathrm{rel}^{(n)} = \max_i \frac{\Delta_i^{(n)}}{\max(|F_i^{(n)}|, |F_i^{(n-1)}|, \varepsilon_{\mathrm{abs}})}, \quad \mathrm{abs}^{(n)} = \max_i \Delta_i^{(n)}.$$

A step is steady if

$$\mathrm{rel}^{(n)} \leq \varepsilon_{\mathrm{rel}} \quad \text{or} \quad \mathrm{abs}^{(n)} \leq \varepsilon_{\mathrm{abs}},$$

and steady state is declared after the required number of consecutive steady steps.

How κ Shapes the ϕ_{applied} Curves

To isolate κ-dependence, a no-noise overlay was generated for:

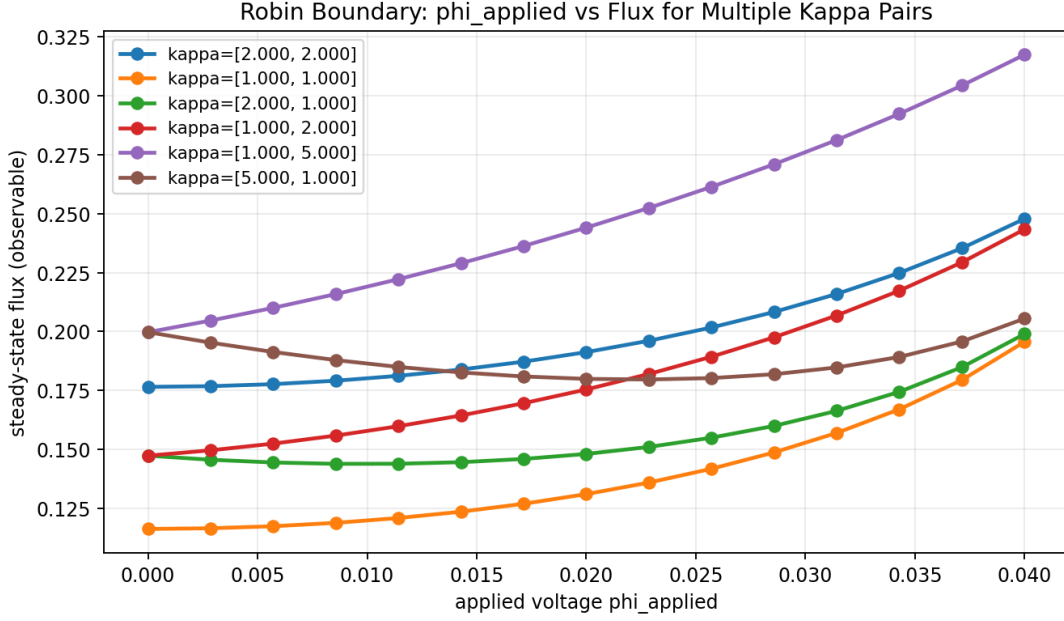$$\kappa \in \{[2,2], [1,1], [2,1], [1,2], [1,5], [5,1]\}.$$

Figure 1: No-noise ϕ_{applied}-vs-steady-flux curves for multiple Robin κ pairs (original Figure 1).
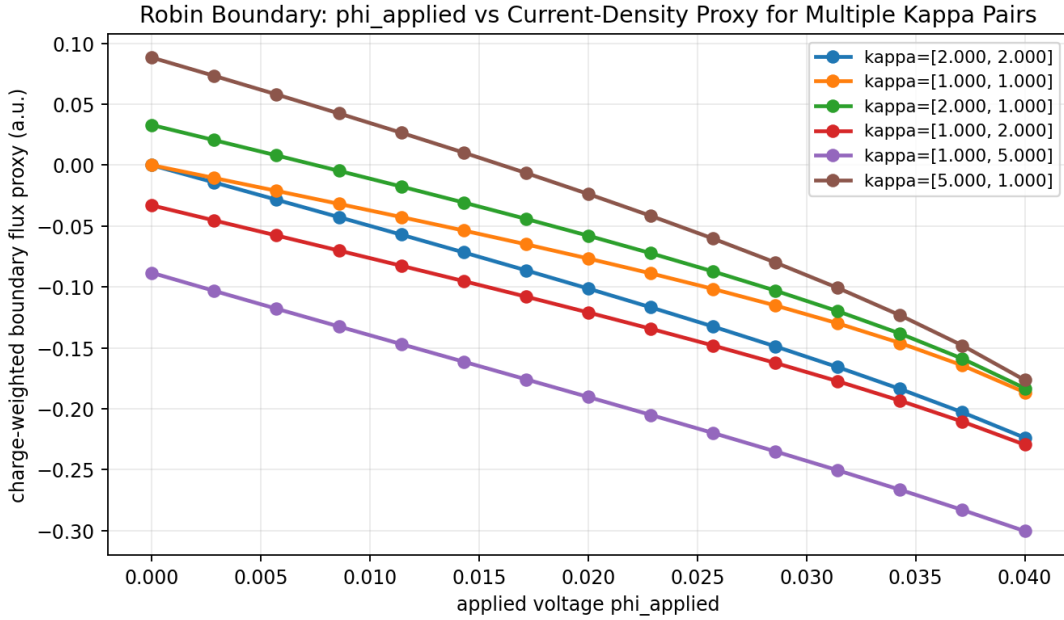

Figure 2: No-noise ϕ_{applied}-vs-current-density-proxy curves for the same Robin κ pairs (added below Figure 1).

Interpretation from the current-density-proxy overlay (second figure):

- In this voltage window, all curves have negative slope: increasing ϕ_{applied} drives the charge-weighted proxy to more negative values.

- Zero-voltage intercepts depend strongly on anisotropy: $[1, 5]$ and $[1, 2]$ start negative, $[5, 1]$ and $[2, 1]$ start positive, while $[1, 1]$ and $[2, 2]$ start near zero.

- Changing which species gets the larger κ ($[1, 5]$ vs $[5, 1]$) changes both intercept and slope, consistent with asymmetric species contributions to the charge-weighted observable.

Using endpoint slope $\big(P(0.04) - P(0)\big)/0.04$ for proxy $P = \sum_i z_i F_i$, representative values are:

$$[1, 1]: -4.661, \quad [2, 2]: -5.594, \quad [2, 1]: -5.402, \quad [1, 2]: -4.911, \quad [1, 5]: -5.301, \quad [5, 1]: -6.619.$$

4. Robin κ Inference from Flux Curves

What was built

A dedicated script and helper pipeline were implemented for inferring κ from a full ϕ_{applied}-vs-steady-flux curve.

Per voltage point ϕ_j, a steady-state forward solve is run and compared to target flux $F_j^\star$.

Objective and gradient

Per-point loss:

$$L_j(\kappa) = \frac{1}{2} \big(F_j(\kappa) - F_j^\star \big)^2.$$

Global loss over m sweep points:

$$J(\kappa) = \sum_{j=1}^{m} L_j(\kappa).$$

Firedrake adjoint is used at each point to compute

$$\nabla_\kappa L_j(\kappa),$$

and gradients are accumulated over converged points:

$$\nabla J(\kappa) = \sum_{j \in \mathcal{C}} \nabla_\kappa L_j(\kappa).$$

SciPy minimize pipeline

SciPy receives:

- `fun`: curve loss $J(\kappa)$,

- `jac`: analytic adjoint gradient $\nabla J(\kappa)$,

- bounds on κ,

- method/tolerance/options (e.g. `L-BFGS-B`, `ftol`, `gtol`, `maxiter`),

- callback for iteration diagnostics and plot updates.

5. Noise Model Clarification

Noise is Gaussian with standard deviation tied to RMS signal magnitude:

$$\sigma = \left(\frac{p}{100}\right) \mathrm{RMS}(F_{\mathrm{clean}}),$$

where p is noise_percent.

So 5% means $\sigma = 0.05 \cdot \mathrm{RMS}$, not a hard $\pm 5\%$ pointwise cap. Individual points can exceed 5% in magnitude.

6. Representative Results and Plots

Summary Tables

Case	Seed	Mean $|\Delta F/F|$ (%)	Max $|\Delta F/F|$ (%)	Best κ_0	Best κ_1	Final Loss
0% noise	20260220	0.000	0.000	1.0015	1.9980	1.4901×10^{-8}
2.5% noise	20260222	1.790	8.263	1.0165	1.9927	1.6312×10^{-4}
5% noise	20260221	3.908	7.918	1.4533	1.7969	4.0573×10^{-4}

Case	Target κ_0	Target κ_1	Fit κ_0	Fit κ_1	RMSE (Flux)	MAE (Flux)
0% noise	1.0000	2.0000	1.0015	1.9980	4.4573×10^{-5}	3.8524×10^{-5}
2.5% noise	1.0000	2.0000	1.0165	1.9927	4.6636×10^{-3}	3.0610×10^{-3}
5% noise	1.0000	2.0000	1.4533	1.7969	7.3551×10^{-3}	5.3404×10^{-3}

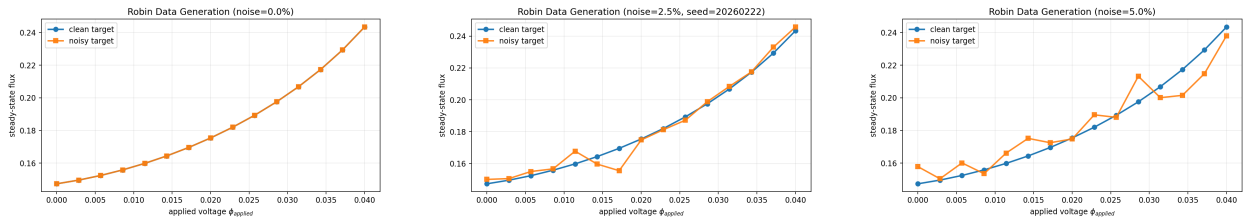| Case	Objective Evals	SciPy Success	Termination
0% noise	20	True	Proj. grad $\leq$ PGTOL
2.5% noise	20	True	Proj. grad $\leq$ PGTOL
5% noise	16	True	Proj. grad $\leq$ PGTOL

Data Generation Curves

Figure 3: Synthetic Robin flux-curve data generation ordered from least to most noise: 0%, 2.5%, 5%.
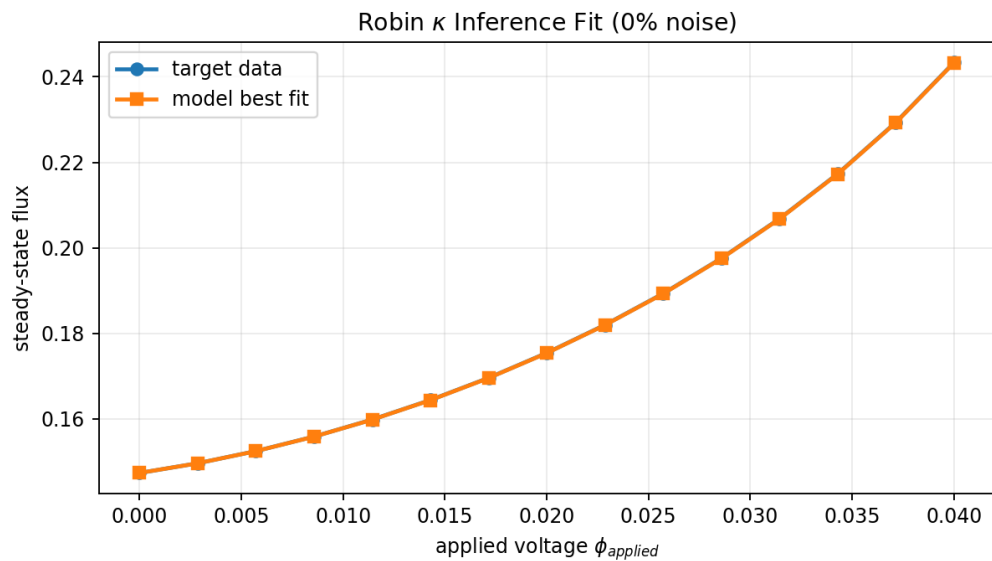
Inference Fits
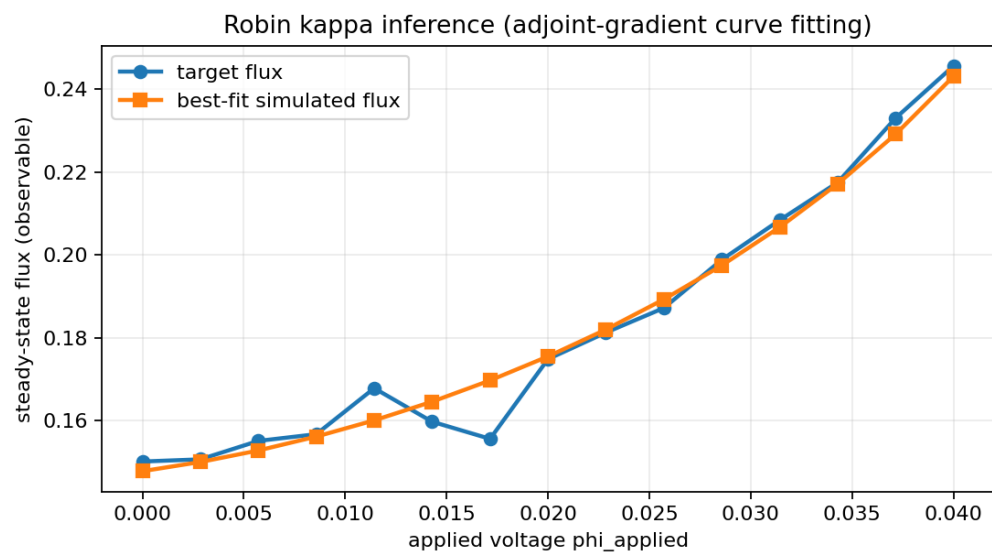

Figure 4: Robin κ curve fit at 0% noise.
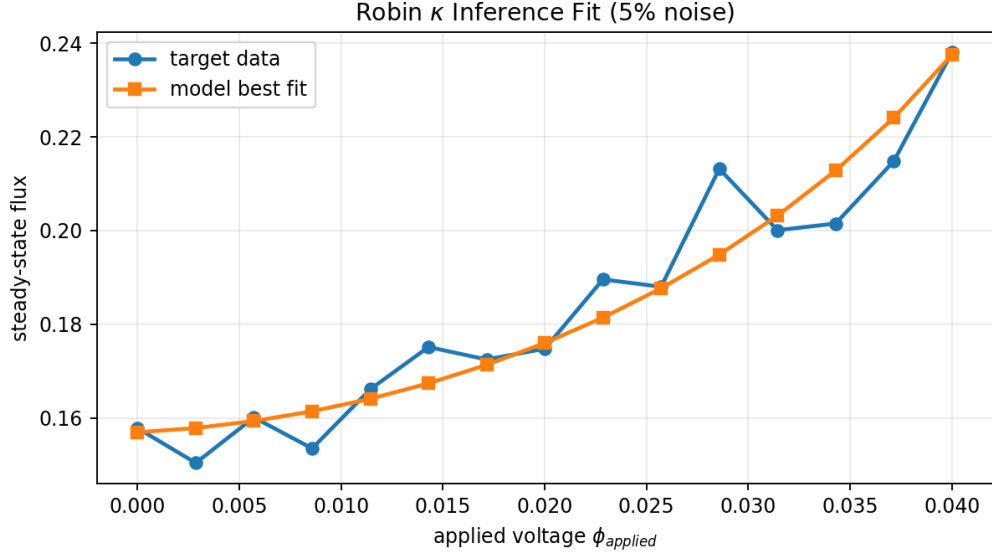

Figure 5: Robin κ curve fit at 2.5% noise.

Figure 6: Robin κ curve fit at 5% noise.

Current-Density Counterparts (Proxy Scaling)

An analogous set of plots was generated for the current-density inference path. In this revision, the observable is the charge-weighted flux proxy $\sum_i z_i F_i$ (Faraday scaling intentionally omitted), so the y-axis is in arbitrary units. The objective and optimizer are unchanged.

Case	Seed	Best κ_0	Best κ_1	Final Loss
0% noise	20260220	0.9989	1.9982	1.1347×10^{-8}
2.5% noise	20260222	1.0064	2.0014	9.0801×10^{-5}
5% noise	20260221	1.9494	2.6694	5.1787×10^{-4}

Table 1: Current-density-proxy inference summary (arbitrary units, no Faraday scaling).
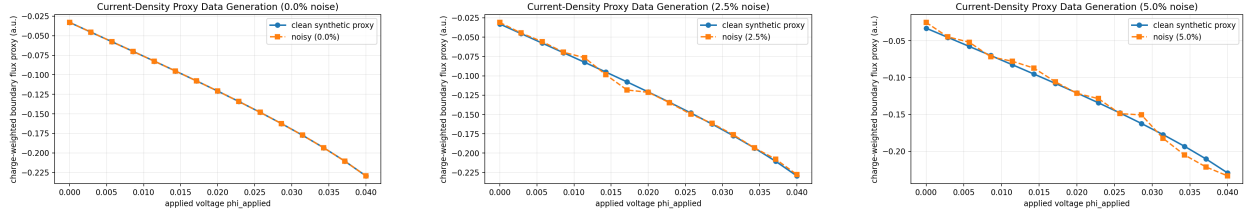

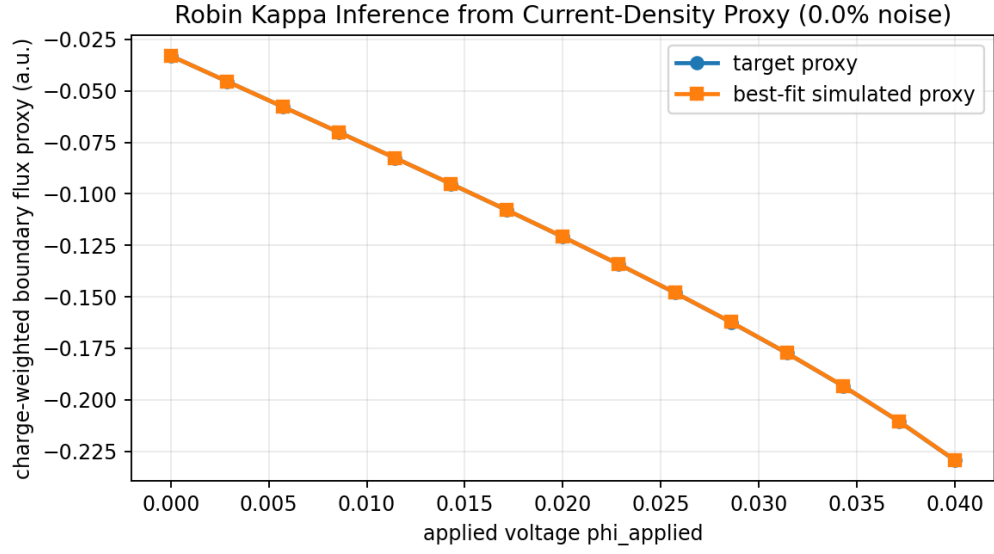Figure 7: Synthetic current-density-proxy data generation (0%, 2.5%, 5% noise; arbitrary units).

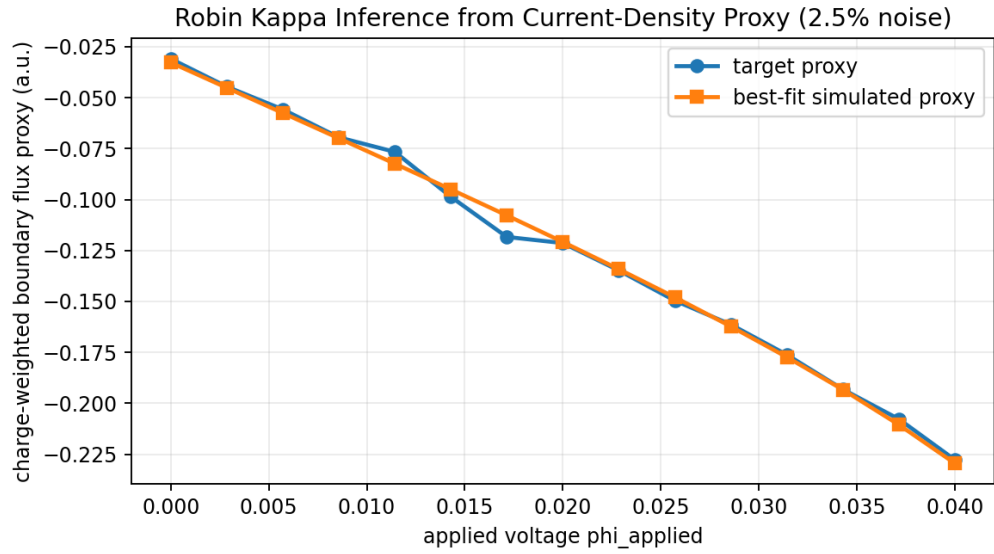Figure 8: Robin κ current-density-proxy fit at 0% noise (arbitrary units).

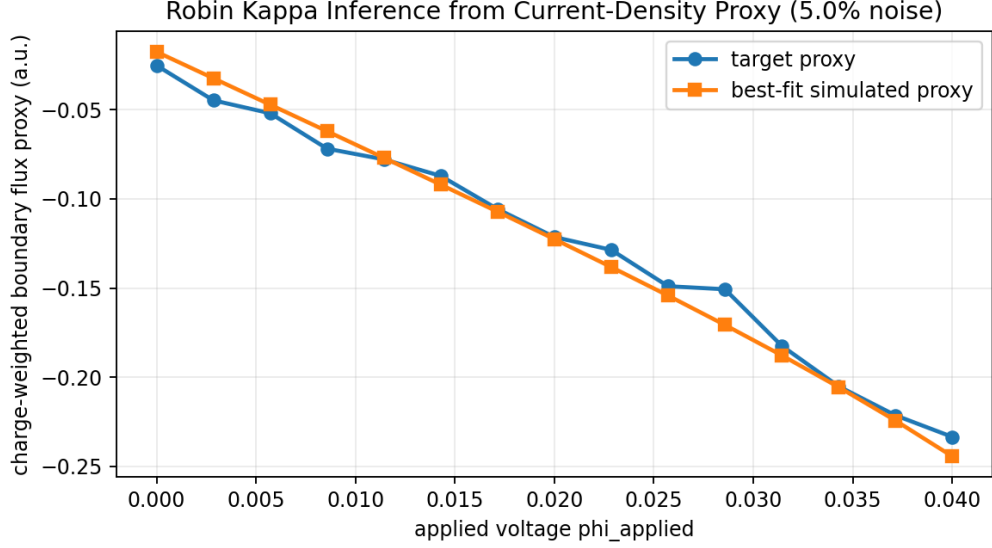Figure 9: Robin κ current-density-proxy fit at 2.5% noise (arbitrary units).

Figure 10: Robin κ current-density-proxy fit at 5% noise (arbitrary units).

7. Practical Takeaways from This Week

- The Robin flux-based inverse workflow is now fully implemented end-to-end and experimentally interpretable.

- A parallel current-density-proxy path is now in place: charge weighting is enforced, while absolute current scaling is explicitly deferred until unit calibration is finalized.

- The unified interface significantly reduced duplication and made target/solver swapping straightforward.

- Resilient retry mechanics now handle many nonlinear solve failures that previously aborted optimization.

- Regenerated proxy runs show strong recovery at 0% and 2.5% noise, with expected degradation and basin sensitivity at 5% noise.

- Reporting artifacts are now synchronized with the new branch: side-by-side data-generation plots, per-noise fit plots, summary table, and a 2.5% convergence GIF.