PNP Inverse Optimization Method Study

Automated Benchmark Report

February 11, 2026

What We Ran

Goal

We're comparing optimization methods for two inverse problems: inferring diffusion coefficients (D) and inferring the Dirichlet BC potential (ϕ_0).

Setup

Methods: `BFGS`, `L-BFGS-B`, `CG`, `SLSQP`, `TNC`, `Newton-CG`. Noise levels: $\sigma \in \{0, 0.005, 0.02\}$. Seeds used for $\sigma = 0$: [20260211]; seeds used for $\sigma > 0$: [20260211, 20260212, 20260213]. Total runs: 420.

How to read the plots

Each plot title states the optimization direction explicitly (*higher is better* or *lower is better*).

Method Comparisons

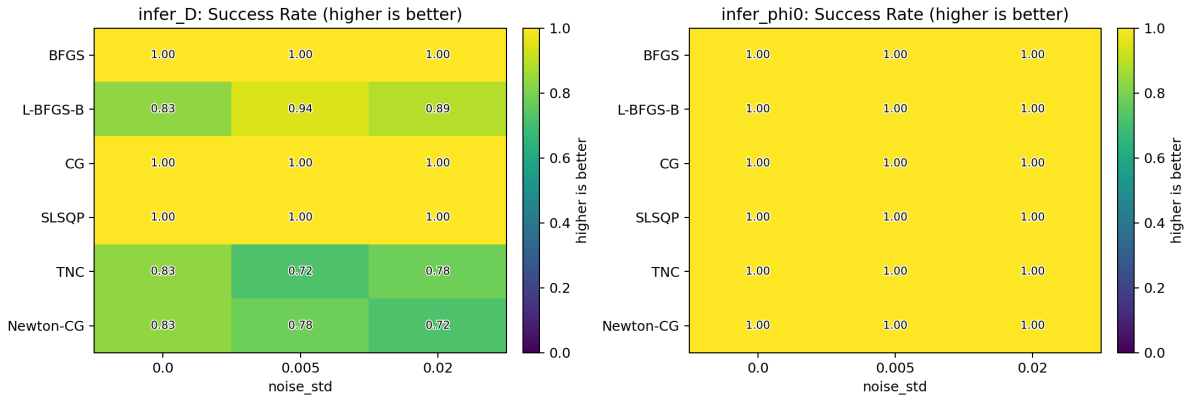

Figure 1: Success-rate heatmaps by problem, method, and noise level (higher is better).

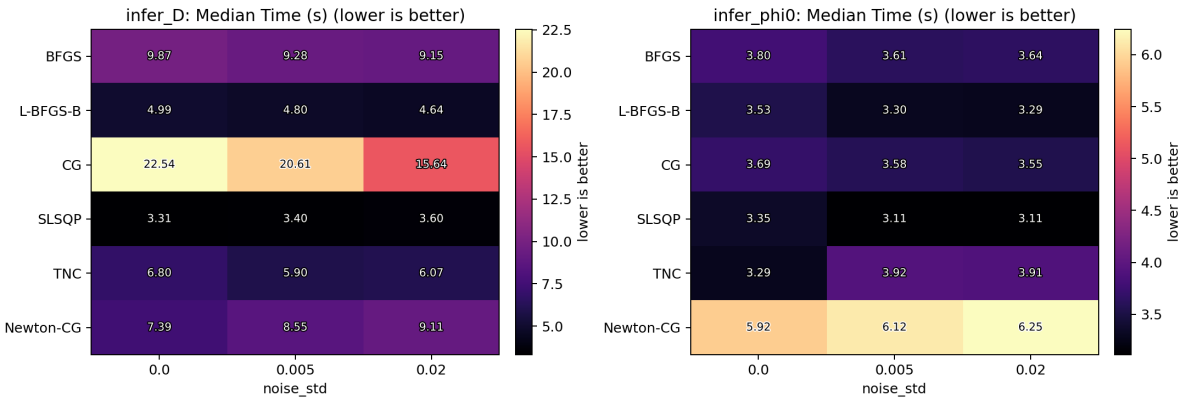

Figure 2: Median wall-clock time (seconds) for successful runs (lower is better).

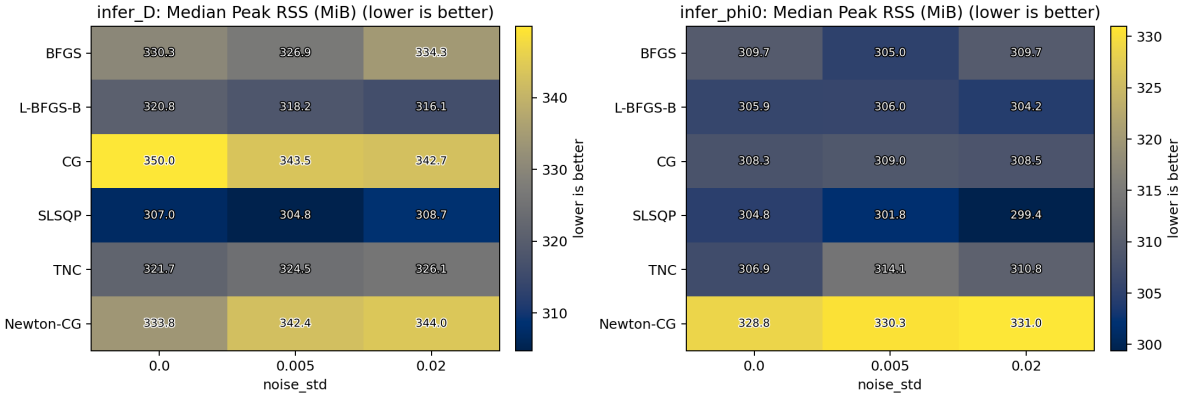

Figure 3: Median peak RSS memory (MiB) for successful runs (lower is better).

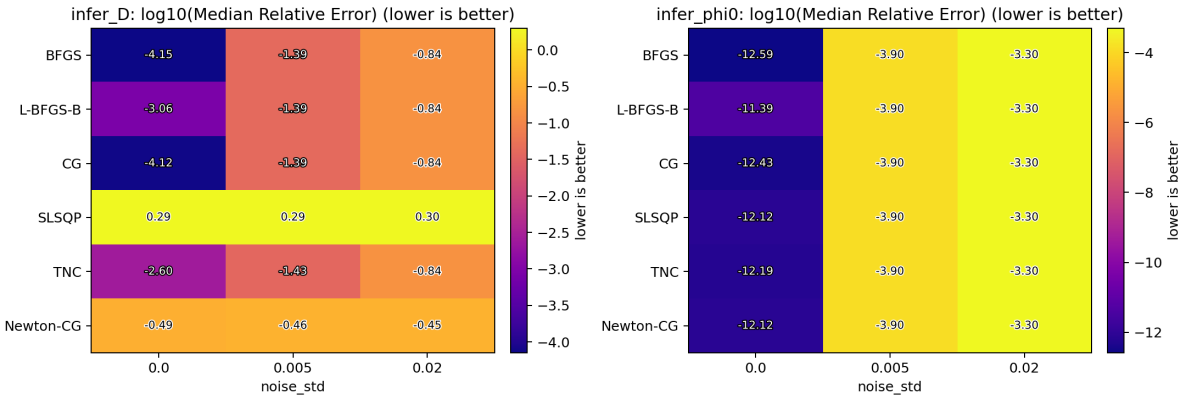

Figure 4: $\log_{10}$ median relative error by method and noise (lower is better).

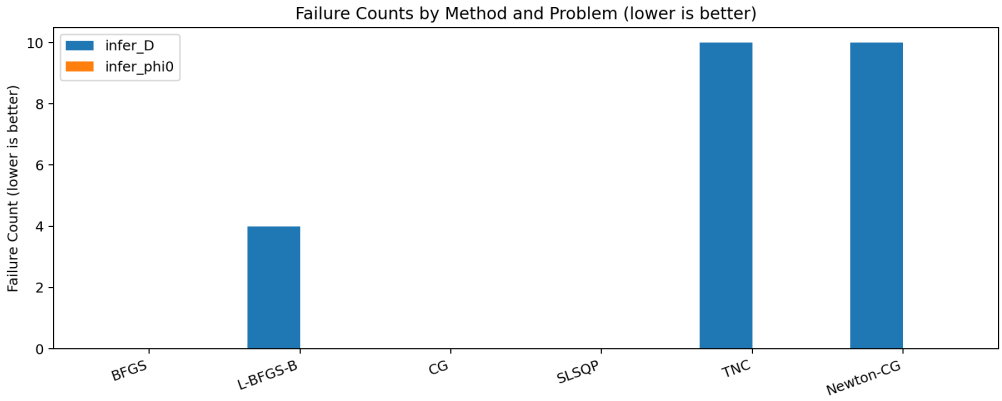

Figure 5: Failure counts by method and problem (lower is better).

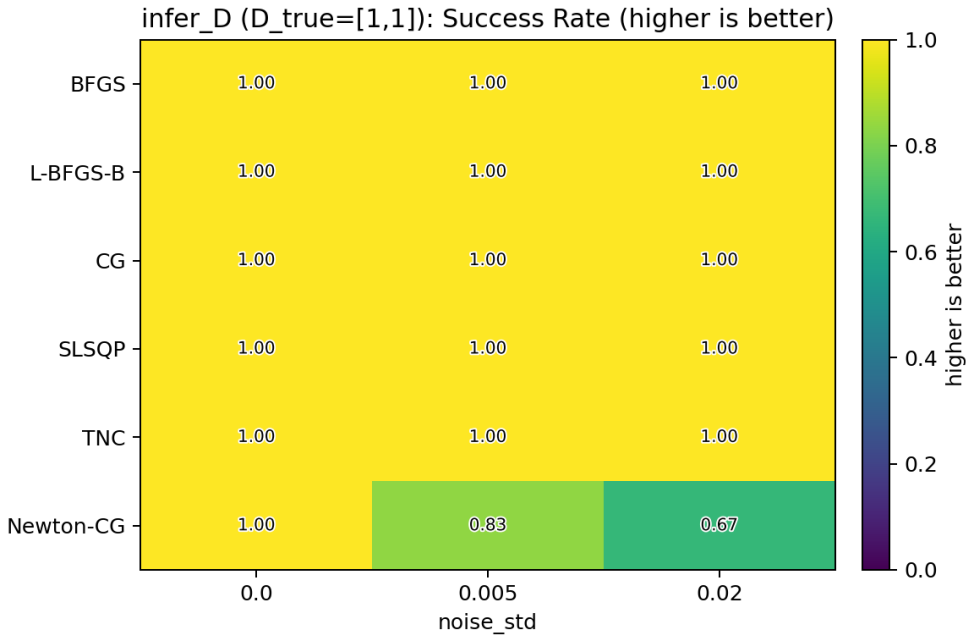

Figure 6: Symmetric diffusion truth case ($D_{\text{true}} = [1, 1]$): success rate by method and noise (higher is better).

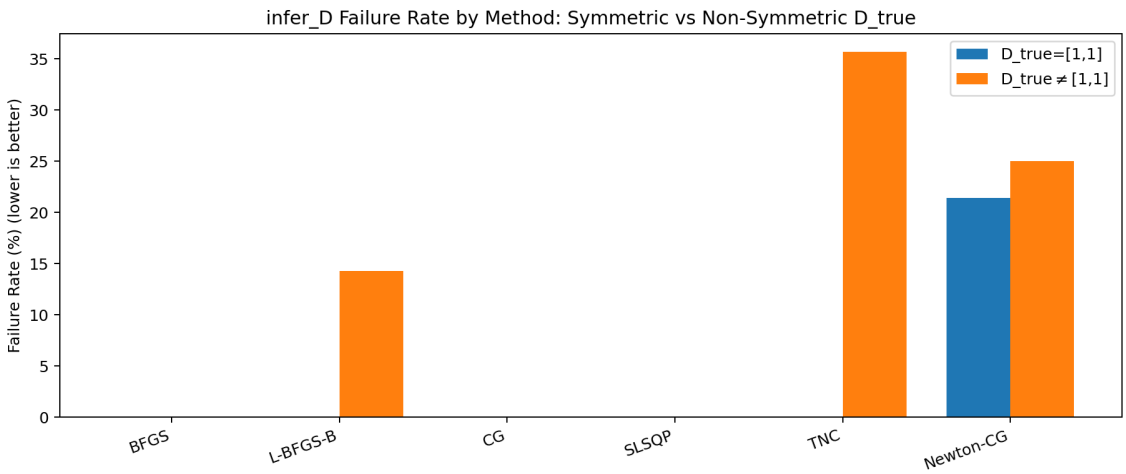

Figure 7: infer_D failure-rate comparison by method: symmetric case vs all non-symmetric cases (lower is better).

Aggregated Results Across Noise Levels

Infer D

Method	Success (%)	Median Time (s)	Median RSS (MiB)	Median RelErr
BFGS	100	9.22	332	0.044
CG	100	18.3	343	0.044
L-BFGS-B	90.5	4.77	318	0.0431
Newton-CG	76.2	8.6	342	0.349
SLSQP	100	3.46	307	2
TNC	76.2	5.99	325	0.0388

Infer ϕ_0

Method	Success (%)	Median Time (s)	Median RSS (MiB)	Median RelErr
BFGS	100	3.66	308	0.000224
CG	100	3.58	309	0.000224
L-BFGS-B	100	3.31	306	0.000224
Newton-CG	100	6.12	330	0.000224
SLSQP	100	3.14	301	0.000224
TNC	100	3.89	310	0.000224

Crash Analysis

What failed?

Observed failures: 24 total. `infer_D` failures: 24; `infer_phi0` failures: 0.

The dominant failure reason was Firedrake nonlinear forward-solve divergence: `DIVERGED_LINE_SEARCH`. Most common recorded reason count: 16 (DIVERGED_LINE_SEARCH).

Was this caused by unconstrained $\phi_0 < 0$?

No. Evidence: all failures occurred in the diffusion-inference problem, not the BC-inference problem; the BC runs completed successfully for all methods/noise levels. Across successful BC runs, estimated ϕ_0 ranged from 0.499144 to 1.5006 with 0 negative estimates. For diffusion inference, controls are parameterized as $\log D$ and exponentiated in the forward model, so D remains strictly positive by construction. The crashes are therefore consistent with unstable trial iterates that make the PDE nonlinear solve hard for Newton line search, not sign violations of ϕ_0.

Symmetric Diffusion Case: $D_{\text{true}} = [1, 1]$

Symmetric-case failures: 3/84 (3.57%). Non-symmetric failures: 21/168 (12.5%). Dominant symmetric-case failure reason: SciPyConvergenceError (3 runs).

Method	Failures / Runs	Failure Rate (%)	Comment
BFGS	0/14	0	no failures
L-BFGS-B	0/14	0	no failures
CG	0/14	0	no failures
SLSQP	0/14	0	no failures
TNC	0/14	0	no failures
Newton-CG	3/14	21.4	unstable in symmetric case

Why L-BFGS-B Failed While BFGS Did Not (`infer_D`)

In `infer_D`, `L-BFGS-B` had 4 forward-solve failures, while `BFGS` had 0. Failed `L-BFGS-B` cases used initial guesses [10, 10] and true-D cases [0.5, 2], [1, 3]. In failed runs, logged trial iterates show sharp drops in one or both diffusivities before SNES terminated with `DIVERGED_LINE_SEARCH`.

σ	Seed	Initial guess D_0	SNES iters	Min logged trial D	Last logged D	Matched BFGS estimate
0	20260211	[10, 10]	25	[0.2336, 0.369]	[1.893, 2.318]	[1, 3.001]
0.005	20260211	[10, 10]	29	[0.02232, 0.8507]	[1.425, 2.151]	[0.9606, 2.663]
0.02	20260212	[10, 10]	76	[0.1579, 0.6134]	[0.4391, 1.221]	[0.4632, 1.632]
0.02	20260213	[10, 10]	80	[0.234, 0.3683]	[1.901, 2.324]	[1.003, 3.006]

Across-Seed Stability ($\sigma > 0$)

To test whether failures are random or method-specific, the noisy cases were repeated across multiple seeds and aggregated by method.

Method (`infer_D`)	Failures / Runs	Failure Rate (%)	Seeds with failures
BFGS	0/36	0	none
L-BFGS-B	3/36	8.33	[20260211, 20260212, 20260213]
CG	0/36	0	none
SLSQP	0/36	0	none
TNC	9/36	25	[20260211, 20260212, 20260213]
Newton-CG	9/36	25	[20260211, 20260212, 20260213]

Methods with any noisy-case failures in `infer_D`: L-BFGS-B, TNC, Newton-CG. If the same method fails across multiple seeds while others remain stable, that is evidence of method-specific robustness differences rather than pure random noise effects.

Practical Recommendations

Takeaways

- For `infer_phi0`: `SLSQP` or `L-BFGS-B` gave the best speed with full robustness.
- For `infer_D`: `BFGS` had best reliability (100% success) and good accuracy; `L-BFGS-B` was faster but had occasional forward-solve failures.
- `TNC` and `Newton-CG` were less robust for `infer_D` under this setup.