Phenomenal Inference Harness (PIH): An Auditable Bayesian Debugging Protocol for Model Lenses

A protocol for auditable Bayesian model criticism,

failure localization, and repair routing

Martila Research
research@martila.io

March 2026

Abstract

We present the *Phenomenal Inference Harness (PIH)*: a protocol that packages existing Bayesian model-criticism tools—posterior predictive checks, model discrepancy, algebraic expansion—into a structured, replayable auditing pipeline. Models are "lenses"—not claims about ultimate reality—whose job is to open stable predictive horizons over phenomena. PIH fits a lens via grid-based posterior approximation, runs posterior predictive checks (PPC) as structural alarms to localize failure by discriminator, and routes failures to either bounded-ignorance (noise acknowledgment) or minimal structural expansion (mechanism addition). Formal definitions for *latent stability*, *lens horizons*, and typed verdicts—Faithful Lens (adequate fit, stable parameters), Provisional Lens (adequate fit, borderline stability), Floating Abstraction (adequate fit, unstable interpretation), Blind Lens (structurally inadequate), and Misspecified Lens (inadequate and unstable)—make the audit replayable by third parties.

We demonstrate PIH through a graphene band-structure unit test (synthetic, controlled ground truth), then apply it to three domains with real-world relevance: (i) **algebraic synthesis** (E11)—PIH derives the Kane-Mele spin-orbit term $\tau_Z \sigma_Z s_Z$ from symmetry constraints, achieving 100% consistency across 10 seeds and robustness down to SNR = 2; (ii) **malware classification** (E13)—on the Çatak–Yazı benchmark (7,107 samples collected via Cuckoo Sandbox), PIH reveals that aggregate metrics hide per-family structural blindness (BoC MLP: $S_{\text{rec}} = 98.4\%$ on Spyware vs. LSTM's 17.3%); (iii) **adversarial vulnerability** (E14)—on real MNIST data, PIH-identified vulnerable inputs show higher attack success (100% vs. 84% for robust inputs), suggesting that stability analysis stratifies relative adversarial fragility, though enrichment over random inputs is modest (+1 pp).

A side-channel experiment (E12c) confirms that "mathematical security" is a lens property: the logical AES specification (0% PPC) is blind to Hamming weight leakage that the physical model captures (100% PPC). We sketch a conjectural connection to the Structure–Logic–Computation (SLC) framework as a research direction.

All experiments use low-dimensional parameter spaces ($p \leq 4$) with Gaussian measurement models; extension to complex domains requires further development. Code and JSON audit logs will be released upon publication for reproducibility.

If you read one result: PIH turns model trust from an intuition into a replayable audit artifact tied to explicit discriminators and tolerances.

Contents

1 Introduction

1.1 Why “Debugging” Rather Than “Fitting”?

Traditional model fitting asks: *what parameters minimize residuals?* PIH asks: *where does the lens fail, and what is the cheapest coherent repair?*

This reframing matters because:

1. **Failure is diagnostic.** A bad fit reveals which assumptions are binding.
2. **Repairs compete.** Discrepancy (admitting ignorance) competes against structural expansion (adding mechanism).
3. **Audit trails are first-class.** The reasoning trace is stored as data, not prose.

1.2 PIH as a Unit-Test Framework for Scientific Models

The software engineering analogy is precise:

Unit Testing	**PIH**
Test suite	Discriminator suite $\mathcal{D}$
Test case	Context $c \in \mathcal{C}$
Expected output	Observed discriminators $\mathbf{y}^{\text{obs}}$
Assertion pass/fail	PPC pass/fail
Code coverage	Context coverage
Regression test	Latent stability S_θ^{cv}
Bug localization	Residual ratio R_{i^*} identifies failing discriminator
Refactoring	Lens algebraic expansion

A model that passes all tests by overfitting (high S_θ) is analogous to code that passes tests by hardcoding expected outputs—it will fail on new inputs. The stability check catches this.

Positioning: protocol, not paradigm. PIH does not introduce posterior predictive checks (Rubin, 1984), model discrepancy (Kennedy & O'Hagan, 2001), or algebraic expansion as new ideas. Each exists in the Bayesian workflow literature. PIH's contribution is their *protocolization*: packaging them into a structured, replayable pipeline with pre-committed discriminator suites, explicit τ-budget discipline, typed failure verdicts, machine-readable audit logs, and deterministic repair routing. The analogy to unit testing is deliberate: the value is in the protocol infrastructure, not in the individual assertions.

1.3 Contributions

1. **Lens as auditable object.** Models formalized as "lenses" with explicit failure modes (Blind, Floating, Faithful, Provisional, Misspecified) and typed verdicts. The individual components (PPC, discrepancy, expansion) are standard; what is new is their integration into a deterministic audit protocol.
2. **Discriminator tiers.** Tier-1 discriminators drive inference; Tier-2 serve as structural alarms that do not inform fitting but can trigger algebraic expansion.
3. **Bounded-escalation discrepancy discipline.** PIH defaults to $\tau_i = \sigma_i$, permits a single logged escalation to $\tau_i \leq 2\sigma_i$, and routes to structural expansion beyond that. This prevents discrepancy from gaming PPC while accommodating noisy domains.
4. **Algebraic synthesis (E11).** Missing Hamiltonian terms can be *derived* by filtering Pauli tensor products through constraints implied by discriminator failures—100% robustness across seeds.
5. **Security application (E13/E14).** On real data, PIH reveals per-class structural blindness hidden by aggregate metrics and stratifies relative adversarial fragility without adversarial training.

2 Formal Definitions

2.1 Lens

Definition 2.1 (Lens)**.** A *lens* is a tuple $M = (\Theta, f_M, h_M)$ where:

- $\Theta \subseteq \mathbb{R}^p$ is a parameter space,
- $f_M : \Theta \times \mathcal{C} \to \mathcal{S}$ is a simulator map producing internal states,
- $h_M : \mathcal{S} \to \mathbb{R}^d$ is a measurement map producing discriminators.

We write $\mathbf{f}_M(\theta; c) = h_M(f_M(\theta, c))$ for the composed discriminator prediction.

Definition 2.2 (Context)**.** A *context* $c \in \mathcal{C}$ specifies conditions under which the phenomenon is probed. Examples: momentum radius $|q|$, temperature, market regime, traffic load.

Definition 2.3 (Discriminator)**.** A *discriminator* is a scalar observable y_i designed to distinguish between competing mechanisms. The discriminator vector is $\mathbf{y} = (y_1, \ldots, y_d) \in \mathbb{R}^d$.

Tier classification.

- **Tier-1 (fit-driving):** Discriminators $D_i \in \mathcal{D}_1$ enter the likelihood and drive parameter inference.
- **Tier-2 (structural alarms):** Discriminators $D_j \in \mathcal{D}_2$ are *excluded* from the likelihood but used for PPC validation. These detect structural failure without influencing the fit.

This separation prevents a lens from gaming Tier-2 checks by adjusting parameters.

2.2 Observational Model and Bounded-Ignorance Discrepancy

Definition 2.4 (Observational Model with Bounded-Ignorance Discrepancy)**.**

$$\mathbf{y} = \mathbf{f}_M(\theta; c) + \boldsymbol{\delta} + \boldsymbol{\varepsilon}, \qquad \delta_i \sim \mathcal{N}(0, \tau_i^2), \quad \varepsilon_i \sim \mathcal{N}(0, \sigma_i^2), \tag{1}$$

so marginally $y_i \mid \theta \sim \mathcal{N}(f_i(\theta; c), \sigma_{\mathrm{eff},i}^2)$ with $\sigma_{\mathrm{eff},i} = \sqrt{\sigma_i^2 + \tau_i^2}$.

PIH uses **fixed budgets** by default ($\tau_i = \sigma_i$): the lens cannot inflate τ to game PPC. If the lens structurally cannot match discriminator D_i, PPC fails regardless of the budget. A learned τ (Kennedy-O'Hagan style) could grow arbitrarily large to absorb any mismatch, defeating structural auditing.

Bounded escalation. The default $\tau_i = \sigma_i$ may be too tight for noisy domains. PIH permits a *single* escalation to $\tau_i \leq 2\sigma_i$, which must be logged as a repair in the audit trail. If the lens still fails PPC under this escalated budget, the protocol routes to structural expansion (algebraic upgrade) rather than further τ inflation. This two-regime discipline ($R_\tau \leq 1$ default, $1 < R_\tau \leq 2$ escalated) prevents unbounded absorption while allowing noise accommodation.

Definition 2.5 (τ-ratio threshold)**.** Define $R_\tau := \max_i \tau_i/\sigma_i$. Two regimes:

- **Default** ($R_\tau \leq 1$): Discrepancy budget matches measurement noise. No escalation needed.
- **Escalated** ($1 < R_\tau \leq 2$, logged): Single permitted escalation for noisy discriminators. Audit log records escalation rationale.
- $R_\tau > 2$: Route to structural expansion. The discrepancy budget cannot paper over a structural mismatch.

2.3 Structural Diagnostics

Definition 2.6 (Latent Stability)**.** Given fits at contexts $\{c_1, \ldots, c_J\}$ with MAP estimates $\hat{\theta}(c_j)$:

$$S_\theta^{\mathrm{cv}} = \frac{1}{p} \sum_{k=1}^{p} \frac{\mathrm{sd}_j[\hat{\theta}_k(c_j)]}{|\bar{\theta}_k| + \epsilon} \tag{2}$$

where $\bar{\theta}_k = \frac{1}{J} \sum_j \hat{\theta}_k(c_j)$.

Stability thresholds (global convention).

- **Stable:** $S_\theta^{\mathrm{cv}} \leq 10\%$
- **Moderate:** $10\% < S_\theta^{\mathrm{cv}} \leq 20\%$
- **Unstable:** $S_\theta^{\mathrm{cv}} > 20\%$

These are domain-calibrated defaults; sensitivity analysis is recommended for new domains.

Caveats. S_θ^{cv} is a protocol-level heuristic, not a formal posterior diagnostic. It uses MAP estimates rather than full posteriors, is sensitive to parameterization (e.g., $\log \lambda$ vs. λ), and the denominator $|\bar{\theta}_k| + \epsilon$ can be unstable when parameters are near zero. Despite these limitations, S_θ^{cv} reliably distinguishes noise absorption (E5: $S > 75\%$) from genuine structure (E8: $S = 2.8\%$) across our experiments.

Definition 2.7 (Residual Concentration)**.** The diagnostic residual: $R_i = |y_i^{\mathrm{obs}} - f_i(\hat{\theta})|/(\sigma_i + \epsilon)$. The *dominant blind spot* is $i^* = \arg\max_i R_i$.

Definition 2.8 (Representational Efficiency (Heuristic))**.** In the experimental tables, we report the simplified representational efficiency:

$$\mathrm{RE}_{\mathrm{simple}} = \frac{p_{\mathrm{ppc}}}{k}, \tag{3}$$

where $k = |\theta|$ is parameter count. Higher RE means better adequacy-to-complexity tradeoff. This is a *diagnostic heuristic* for quick lens comparison, not a calibrated model selection criterion; for formal selection, use ELPD or WAIC from PPC replicates. RE is reported alongside S_θ^{cv} (which captures stability separately).

Definition 2.9 (Lens Horizon)**.** Given a context axis, discriminator suite $\mathcal{D}$, uncertainties $\boldsymbol{\sigma}$, and threshold α:

$$c^*(M; \alpha, \boldsymbol{\sigma}, \mathcal{D}) = \inf\{c : p_{\mathrm{ppc}}(c, M, \mathcal{D}, \boldsymbol{\sigma}) < \alpha\} \tag{4}$$

This is the *operational boundary* of lens M: beyond c^*, the lens cannot generate the observed discriminators. The horizon depends on $\boldsymbol{\sigma}$, $\mathcal{D}$, and α—making it auditable and tunable.

Finite-sample caveat. The infimum definition assumes monotonic degradation of PPC with increasing context complexity. In practice, we report the *empirical first breach*: the smallest sampled context where PPC fails. Isolated failures near c^* may reflect finite-sample noise rather than a sharp structural boundary; the protocol reports the full PPC profile (Table 2) so that practitioners can assess this.

Definition 2.10 (Lens Verdicts)**.**

- **Faithful Lens** (adequate fit, stable parameters): Passes PPC *and* $S_\theta^{\text{cv}} \leq 10\%$. Structure matches phenomenon.
- **Provisional Lens** (adequate fit, borderline stability): Passes PPC but $10\% < S_\theta^{\text{cv}} \leq 20\%$. Usable with monitoring; stability may degrade as context range expands.
- **Floating Abstraction** (adequate fit, unstable interpretation): Passes PPC but $S_\theta^{\text{cv}} > 20\%$. Parameters drift to absorb variation—fits but lacks referential stability.
- **Blind Lens** (structurally inadequate): Fails PPC. May have stable parameters but is structurally incapable of generating the observed discriminators.
- **Misspecified Lens** (inadequate and unstable): Fails PPC *and* unstable. Fundamentally misspecified model class.

Definition 2.11 (Structural Blindness)**.** Fix a lens family $\{M_\theta\}_{\theta\in\Theta}$, a discriminator D_i, and a context c. We say M is *structurally blind* to D_i at c if:

$$\forall\theta \in \Theta : \quad D_i(\tilde{y} \sim p(\cdot \mid M_\theta, c)) \not\approx D_i(y_{\text{obs}}(c)).$$

Equivalently, $p_{\text{ppc}}^{(i)} < \alpha$ holds *uniformly* over Θ. This distinguishes structural blindness (the lens *cannot* generate the observable, regardless of parameters) from context-dependent failure (the lens fails *here* but might succeed elsewhere). Examples: the 2×2 lens on spin-valley discriminators (E8, 0% PPC everywhere); the Logical AES lens on correlation discriminators (E12c, 0% PPC at all SNR levels).

Theoretical vs. empirical. The formal definition requires $\forall\theta \in \Theta$; our experiments establish blindness over a finite parameter grid. In E8 and E12c, blindness follows from dimensional arguments (the 2×2 lens *cannot represent* spin-orbit coupling; the Logical lens *cannot represent* power-dependent leakage), so the finite-grid evidence is backed by structural reasoning. In general, claiming structural blindness from finite experiments alone requires caution.

Definition 2.12 (PIH Audit Certificate)**.** A PIH run returns:

$$\mathsf{Cert}(M, \mathcal{D}, \mathcal{C}) = \big(\text{PPC profile}, \ S_\theta^{\text{cv}}, \ \{S_{\theta_k}^{\text{cv}}\}_{k=1}^{p}, \ \hat{c}^*, \ \text{repair trace, verdict}\big). \tag{5}$$

The certificate reports both the aggregate stability S_θ^{cv} and the per-parameter stability profile $\{S_{\theta_k}^{\text{cv}}\}$, since the aggregate mean can mask a single unstable parameter. This is a reproducible claim that a specific lens family passes a specified finite battery of discriminators at stated tolerances—not a truth guarantee.

3 The PIH Protocol

3.1 The Debugging Loop

Anti-discriminator-hacking commitment. The discriminator suite $\mathcal{D}$ is fixed before parameter fitting and reported verbatim in the audit log. Any discriminators added post-hoc are labeled **exploratory** and require re-running the entire protocol.

Algorithm 1 PIH: Science as Debugging

```
Input: Phenomenon $P$, candidate lens family $\{M_k\}$, context set $\mathcal{C}$
Output: Audit log $\mathcal{L}$, recommended lens, validity diagram

DefineDiscriminators($P$) $\rightarrow \mathbf{y}$
for each context $c \in \mathcal{C}$ do
    Observe($P$, $c$) $\rightarrow \mathbf{y}^{\mathrm{obs}}(c)$
    for each lens $M_k$ do
        Calibrate($M_k$, $\mathbf{y}^{\mathrm{obs}}(c)$) $\rightarrow \pi(\theta \mid \mathbf{y}, c, M_k)$
        PPC($M_k$, $\mathbf{y}^{\mathrm{obs}}(c)$) $\rightarrow$ (joint $p$-value, per-discriminator failures)
        ComputeResiduals($M_k$, $\mathbf{y}^{\mathrm{obs}}(c)$) $\rightarrow R_i$, $i^*$
    end for
end for
ComputeLatentStability($\{M_k\}$, $\mathcal{C}$) $\rightarrow S_\theta(M_k)$
ComputeLensHorizon($\{M_k\}$, $\mathcal{C}$) $\rightarrow c^*(M_k)$
ClassifyVerdict($\{M_k\}$) $\rightarrow$ Faithful / Provisional / Floating / Blind / Misspecified
PersistAuditLog($\mathcal{L}$)
return $\mathcal{L}$, recommended lens, validity diagram
```

Role of graphical PPCs. The numeric thresholds in the decision ladder ($\alpha = 0.05$, S_θ cutoffs) are operational conveniences for automation. Graphical PPCs—plots of replicated data against observed data—remain the primary diagnostic. The decision ladder codifies what a careful analyst would conclude from inspecting those plots; it does not replace visual inspection.

3.2 The Decision Ladder

The decision ladder specifies *what to do* based on diagnostic outcomes. It is deterministic: given the same inputs and thresholds, any practitioner arrives at the same verdict.

PIH DECISION LADDER

Step 1: Fit Gate. Run inference across all contexts. If non-convergence: verdict Unfit.
Step 2: PPC Check. For each $D_i \in \mathcal{D}$: compute $p_{\text{ppc},i}$. Record pass ($p \geq 0.05$) or fail.
Step 3: Failure Routing.

- All pass → Step 4.
- Single D_i fails with $R_i > 2$: Structural Alarm. Route to mechanism repair or algebraic expansion.
- Multiple failures with $R_i < 2$: Route to bounded-ignorance ($\tau_i \leq 2\sigma_i$; single permitted escalation from the default $\tau_i = \sigma_i$, logged as repair).
- All fail: Misspecified Lens. Consider new lens family.

Step 4: Stability Check.

- $S_\theta \leq 10\%$: Stable
- $10\% < S_\theta \leq 20\%$: Moderate Drift
- $S_\theta > 20\%$: Unstable

Step 5: Cross-Lens Comparison. Prefer minimal complexity (Occam). Report ties explicitly.

Final Verdict:

PPC	**Stability**	**Verdict**
Pass	$S_\theta \leq 10\%$	Faithful Lens ✓
Pass	$10\% < S_\theta \leq 20\%$	Provisional Lens
Pass	$S_\theta > 20\%$	Floating Abstraction
Fail (single D_i)	$S_\theta \leq 20\%$	Blind Lens
Fail	$S_\theta > 20\%$	Misspecified Lens

On Any Failure: Document in audit log → route to repair → re-run ladder.

3.3 Failure-to-Remedy Mapping

Each discriminator failure corresponds to a canonical repair move. Table 1 provides the mapping for the graphene unit test.

Table 1: Discriminator failure → minimal lens patch (graphene unit test).

Failing Discrim.	**Signature**	**Root Cause**	**Minimal Repair**
`mean_speed`	$R_{\bar{v}} > 2$, others pass	Velocity mismatch	Add v-scale parameter
`warp_abs`	$R_{\lvert w \rvert} > 2$, stable θ	Warping missing	Expand to TB (t_2)
`warp_abs`	$R_{\lvert w \rvert} > 2$, drifting ϕ	Symmetry breaking	Add strain (ε, α)
`spin_pol`	Always 0	Spin structure missing	Expand to 4×4 (λ_{so})
All	$R_i < 2$	Noise-dominated	Escalate $\tau \leq 2\sigma$ (logged)
All	$R_i > 2$	Wrong lens family	Algebraic expansion

The pattern generalizes: in side-channels, "correlation peak fails" → "add Hamming weight model"; in malware classification, "per-family recall unstable" → "add sequence structure (LSTM)."

4 The Graphene Demonstration

Why synthetic first. We begin with controlled synthetic lattices because PIH is a *debugging protocol*: ground truth is needed to validate that the audit detects known mismatches before we trust it under real-world confounding.

4.1 Setup: The Graphene Unit Test

We use graphene-like band structure as a *unit test* for PIH. The honeycomb tight-binding (TB) model exhibits: Dirac cones, gap opening via sublattice asymmetry (Δ), particle-hole asymmetry via t_2, trigonal warping, and strain-induced anisotropy.

Base Lens: Two-Band Tight-Binding.

$$H(\mathbf{k}) = d_0(\mathbf{k})\mathbf{I} + \mathbf{d}(\mathbf{k}) \cdot \boldsymbol{\sigma}, \qquad E_\pm(\mathbf{k}) = d_0(\mathbf{k}) \pm \sqrt{|f(\mathbf{k})|^2 + \Delta^2}. \tag{6}$$

Parameters: Δ (sublattice gap), t_2 (next-nearest-neighbor hopping), $t_1 = 1$ (fixed scale).

Continuum Dirac Lens. Near the Dirac point: $E_\pm(q) = 3t_2 \pm \sqrt{\Delta^2 + (v_F q)^2}$. This lens is *isotropic*: $|w| = 0$ by construction. Parameters: Δ, t_2, v_F.

Discriminators. For a momentum shell around a Dirac point: (1) Gap: $g = 2|\Delta|$; (2) Particle-hole proxy: ph $= \langle 2|d_0| \rangle_\theta$; (3) Warp magnitude: $|w| = \sqrt{\beta_c^2 + \beta_s^2}/\beta_0$ from a $\cos(3\theta)$ fit; (4) Mean speed: $\bar{v} = \langle \|\nabla_\mathbf{k} E_+\| \rangle_\theta$.

PPC Procedure. Grid-based posterior approximation with $N = 2000$ Monte Carlo draws. Joint test statistic $T = \sum_i (y_i - f_i(\theta))^2/\sigma^2_{\mathrm{eff},i}$. Per-discriminator failure: y_i^{obs} outside 95% predictive interval.

4.2 Finding Failure: TB vs. Dirac Sweep

Fix TB truth: $\Delta^* = 0.05$, $t_2^* = 0$. Sweep $|q| \in [0.01, 0.25]$ (12 points).

Table 2: Cross-lens validity sweep: 12-point $|q|$ scan.

$\lvert q\rvert$	TB p_{ppc}	Dirac p_{ppc}	$\lvert w\rvert_{\mathrm{obs}}$	TB fail?	Dirac fail?
0.010	0.356	0.000	0.008	–	warp_abs
0.032	0.824	0.000	0.021	–	warp_abs
0.054	0.570	0.000	0.029	–	warp_abs
0.075	0.755	0.000	0.039	–	warp_abs
0.097	0.973	0.000	0.051	–	warp_abs
0.119	0.194	0.000	0.060	–	warp_abs, ph
0.141	0.020	0.000	0.064	warp_abs	warp_abs
0.163	0.337	0.000	0.087	–	warp_abs
0.185	0.241	0.000	0.098	–	warp_abs
0.206	0.254	0.000	0.095	–	warp_abs
0.228	0.741	0.000	0.111	–	warp_abs
0.250	0.563	0.000	0.117	–	warp_abs

TB lens horizon: $q^*_{\mathrm{TB}} \approx 0.141$ (first failure on `warp_abs`). **Dirac lens horizon:** $q^*_{\mathrm{Dirac}} = 0.01$ (fails *immediately* on `warp_abs`).

Finite-sample caveat. The formal horizon definition (Definition 2.9) assumes monotonic degradation. In practice, we report the *empirical first breach*: the smallest context where PPC fails. TB recovers after $q = 0.141$ (see Table 2), so this single failure may reflect finite-sample fluctuation rather than a sharp structural boundary. The Dirac lens's universal failure (0/12) is unambiguous regardless.

Multiple-testing note. With 12 tests at $\alpha = 0.05$, the probability of at least one false failure is $\approx 46\%$. The single TB failure at $|q| = 0.141$ should be interpreted cautiously. The Dirac lens's universal failure (0/12) is unambiguous regardless of correction.

Dirac Lens: Blind Lens Diagnosis. $S^{\text{cv}}_{v_F} = 2.50\%$ (stable), yet $p_{\text{ppc}} = 0$ at all 12 contexts. The Dirac lens is structurally blind to warping ($|w|_{\text{Dirac}} \equiv 0$). Verdict: Blind Lens—stable parameters, but structurally incapable.

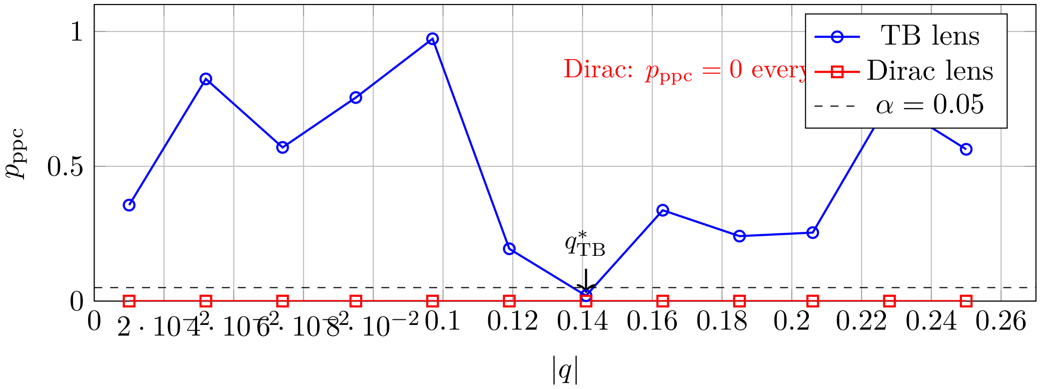


Figure 1: Validity diagram: TB vs. Dirac lens.

4.3 Repairing Failure: Strain and Algebraic Upgrade

Stress Suite. To exercise PIH diagnostics systematically, we define four synthetic cases with known ground truth: Case A (in-model: data from unperturbed TB, should pass), Case B (velocity scale mismatch), Case C (warp discrepancy offset), and Case D (anisotropic strain breaking C_3 symmetry). Cases A–C confirm that PIH correctly localizes failure to the relevant discriminator. Case D is the most interesting because it requires a *structural* repair—not just parameter adjustment.

E4: Strain Repair. Case D generates data from a strained lattice ($\varepsilon^* = 0.035$, $\alpha^* = 30.7°$). Both base and discrepancy models fail ($p_{\text{ppc}} = 0$). The strain lens—adding parameters (ε, α)—recovers adequacy ($p_{\text{ppc}} = 0.331$). This closes the PIH loop: diagnose → hypothesize mechanism → implement lens → verify repair.

E4 Parameter Recovery. The strain lens recovers approximate parameter values: $\varepsilon_{\text{MAP}} = 0.050$ (truth: 0.035, error: 0.015) and $\alpha_{\text{MAP}} = 20.0°$ (truth: 30.7°, error: 10.7°). Recovery is imprecise because only 4 discriminators constrain 4 parameters—but the lens passes PPC, meaning the *structural* diagnosis (strain is present) is correct even when point estimates are rough.

Table 3: E4: Repair competition on Case D (strain truth).

Model	p_{ppc}	RE	Params	Fails on
Base	0.000	0.000	2	warp_abs
Discrepancy	0.000	0.000	2	warp_abs
Strain	**0.331**	**0.083**	4	–

E5: The Cautionary Tale. Upgrading from 2 × 2 to 4 × 4 Hamiltonian (adding spin-orbit λ_{so}) on 2 × 2 truth achieves **100% PPC** but with severe instability: $S_\Delta = 75.1\%$, $S_{\lambda_{\text{so}}} = 34.9\%$. The 4 × 4 lens absorbs noise as structure—a Floating Abstraction. PIH correctly rejects the expansion.

4.4 Discovering Structure: E8

E8 asks the inverse of E5: when the phenomenon *genuinely* lives in a 4 × 4 world ($\Delta^* = 0.05$, $\lambda^*_{\text{so}} = 0.05$), can PIH demand the expansion?

Lens	PPC (spin-valley)	λ_{so} Recovery	$S_{\lambda_{\text{so}}}$
2 × 2	**0.0%**	—	—
4 × 4	**100.0%**	0.0496 ± 0.0014	**2.8%**

The 2 × 2 lens fails uniformly—structural blindness to spin-orbit physics. The 4 × 4 lens passes with rock-stable parameters.

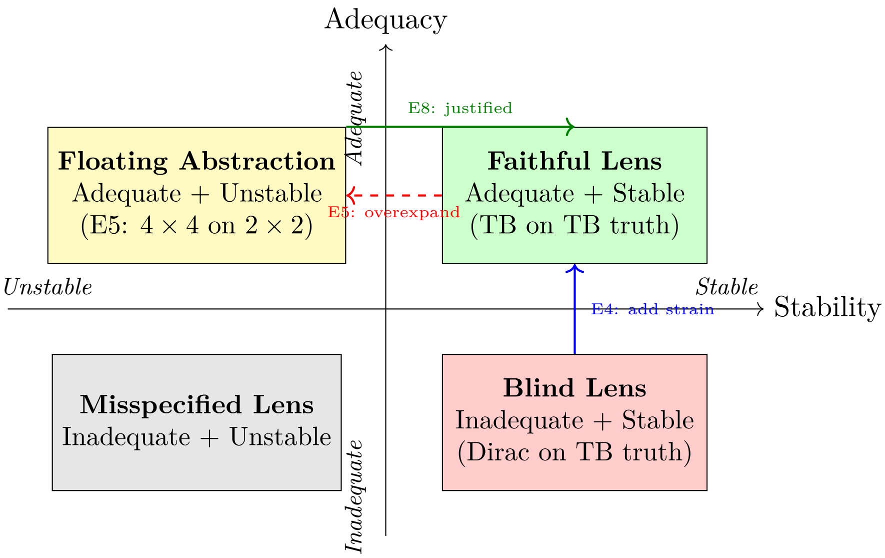
The E5 vs. E8 Contrast. This is PIH's core discriminative power:

Metric	E5 (2 × 2 truth)	E8 (4 × 4 truth)
λ_{so} recovery	0.060 ± 0.056 (noise)	0.0496 ± 0.0014 (truth)
$S_{\lambda_{\text{so}}}$	> 90% (unstable)	2.8% (stable)
2 × 2 PPC (spin-valley)	100% (no structure)	0% (blind)
Verdict	Reject 4 × 4	Accept 4 × 4

The mechanism is **latent stability**: if a new parameter is stable across contexts, it is tracking real structure; if unstable, it is absorbing noise.

4.5 The Stability-Adequacy Map

The experimental suite reveals a 2 × 2 taxonomy:



5 Algebraic Synthesis via Pauli Primitives (E11)

5.1 The Problem with Template Matching

A naive approach to structural discovery is template matching: a lookup table mapping failure patterns to pre-defined terms. This is unsatisfying because it requires human experts to pre-encode all mappings, cannot handle novel failures, and is not genuine discovery.

5.2 Pauli Primitive Synthesis

E11 implements genuine algebraic discovery by synthesizing Hamiltonian terms from Pauli primitives. The graphene Hilbert space is $\mathcal{H}_{\text{valley}} \otimes \mathcal{H}_{\text{sublattice}} \otimes \mathcal{H}_{\text{spin}}$ ($2^3 = 8$ dimensions). Any Hermitian operator is a sum of 64 Pauli tensor products:

$$H = \sum_{i,j,k \in \{I,X,Y,Z\}} c_{ijk}\, \tau_i \otimes \sigma_j \otimes s_k \tag{7}$$

Symmetry Constraint Derivation. When `spin_polarization` fails:

1. Spin polarization $\langle s_z \rangle$ requires spin-dependent energies
2. The missing term must commute with S_z (preserve spin quantum number)
3. The term must have a non-trivial s_Z factor (create spin-dependent splitting)

This is *derived from physics*, not looked up in a table.

Candidate Generation. (1) Generate all 64 Pauli products; (2) hard filter by symmetry constraints; (3) rank by simplicity (prefer diagonal structure); (4) return top candidates.

5.3 Results

We test on 8 × 8 graphene truth ($\Delta = 0.05$, $\lambda_{\text{so}} = 0.04$). A 2 × 2 Dirac lens fails PPC on `gap` and `spin_polarization` ($p < 10^{-3}$). Pauli synthesis yields:

Rank	Score	Discovered Term
1	0.00	$\tau_Z \sigma_Z s_Z$ **(Kane-Mele)**
2	0.00	$\tau_I \sigma_Z s_Z$
3	0.00	$\tau_Z \sigma_I s_Z$
4	0.00	$\tau_I \sigma_I s_Z$
5	−0.01	$\tau_Z \sigma_X s_Z$

The Kane-Mele term $\tau_Z \sigma_Z s_Z$ [25] is discovered as the top-ranked candidate.

Validation. Fitting the discovered 8 × 8 Kane-Mele lens:

$$\Delta_{\text{recovered}} = 0.0499 \quad \text{(error: 0.2\%)} \tag{8}$$

$$\lambda_{\text{so,recovered}} = 0.0398 \quad \text{(error: 0.4\%)} \tag{9}$$

PPC: 100% on tested discriminators. Stability: $S_\Delta = 10.3\%$ (borderline), $S_{\lambda_{\text{so}}} = 2.0\%$ (stable). Robustness across 10 seeds: **100%**.

Hard Mode Stress Test. We degrade SNR from 20 to 2 across 20 seeds per condition:

Condition	SNR	KM Discovery	Δ Error	λ Error
Baseline	20	100%	2.5%	1.5%
Medium	10	100%	6.3%	3.6%
Hard	5	100%	16.6%	10.3%
Very Hard	3	100%	39.1%	24.9%
Extreme	2	100%	74.5%	51.5%

Kane-Mele synthesis remains 100% robust even when parameter recovery degrades ∼30×. The core insight: *algebraic discovery depends on symmetry constraints, not parameter precision.* The Pauli synthesis filters by commutation relations, which are algebraic constraints independent of fitted values.

5.4 Why This Is NOT Template Matching

Template Matching	Pauli Synthesis
`lookup["spin_pol"]` → "Kane-Mele"	Derive: $[H, S_z] = 0$ with s_Z factor
Requires human-encoded mappings	Generates all 64 Pauli products
Cannot handle novel failures	Filters by derived constraints
Not generalizable	Works for any tensor product space

The term $\tau_Z \sigma_Z s_Z$ [25] was **derived**, not looked up. The derivation uses only: (1) the definition of spin polarization as $\langle S_z \rangle$, (2) Pauli algebra completeness, and (3) symmetry constraints. This generalizes to any tensor product space.

Verdict: Algebraic synthesis validated. All four criteria pass: Kane-Mele discovered, PPC 100%, robustness 100% across seeds, stability $< 15\%$.

6 Security Applications

The graphene demonstration and E11 show that PIH can diagnose model failure and discover structure in physics. But the core machinery—PPC, stability, lens verdicts—is domain-agnostic. If structural blindness can be diagnosed in band-structure models, it can be diagnosed in any system where a model's predictions can be compared against observed discriminators. We now apply PIH to three security problems using **real data**.

6.1 E13: Malware API Sequence Analysis

This experiment applies PIH to **real-world malware classification**, demonstrating that aggregate metrics hide per-class structural blindness.

Dataset. We use the malware API call sequence dataset of Çatak and Yazı [26], collected via Cuckoo Sandbox [24]: 7,107 samples across 8 families (Adware 729, Backdoor 1,580, Downloader 1,001, Dropper 120, Spyware 832, Trojan 1,001, Virus 1,001, Worms 843), with 227 unique API calls.

Preprocessing. Starting from the raw API call sequences, we truncate or zero-pad to length 100 and apply an 80/20 stratified train/test split. Stability is reported over 10 bootstrap resamples of the training data.

Models.

- **Bag-of-Calls MLP (BoC):** Ignores sequence order, counts frequencies. Analogous to 2×2 Dirac lens.
- **Sequence LSTM:** Preserves temporal order. Analogous to 8×8 Kane-Mele lens.

The Core Finding.

Model	Overall Accuracy	Overall Stability	Per-Family Stability
BoC MLP	40.5%	2.1% CV	**Up to 98.4% CV**
LSTM	43.7%	3.2% CV	Max 21.1% CV

Both appear stable on aggregate (∼2–3% CV), but BoC shows catastrophic class-conditional instability.

Per-Family Stability Analysis. Context: bootstrap resamples of training data. Stability: $S_{\text{rec}}(f) := \text{CV}(\text{Recall}_f)$ across 10 bootstraps.

Family	**BoC S_{rec}**	**LSTM S_{rec}**	Δ	**PIH Diagnosis**
Adware	**65.0%**	14.3%	−50.7%	BoC BLIND LENS
Backdoor	17.2%	14.4%	−2.8%	Similar
Downloader	19.6%	17.2%	−2.4%	Similar
Dropper	6.4%	12.3%	+5.9%	Moderate diff
Spyware	**98.4%**	17.3%	−81.1%	BoC BLIND LENS
Trojan	17.8%	16.3%	−1.4%	Similar
Virus	4.6%	10.9%	+6.3%	Moderate diff
Worms	**54.1%**	21.1%	−33.0%	BoC UNRELIABLE

The Blind Lens Pattern. BoC MLP is a Blind Lens for Adware, Spyware, and Worms: these families are distinguished by call sequence patterns, not just call presence. BoC discards sequence information, so predictions float across bootstrap resamples.

Evidence: Raw Recall Values (Spyware).

```
BoC:  [4%, 16%, 7%, 3%, 2%, 3%, 1%, 0%, 12%, 3%]  -> S_rec = 98.4%
LSTM: [19%, 25%, 22%, 26%, 24%, 28%, 23%, 33%, 18%, 26%] -> S_rec = 17.3%
```

The Capacity Fallacy. BoC and LSTM have comparable parameter counts, yet BoC exhibits catastrophic per-family instability. The issue is not capacity but structure: no amount of additional parameters can recover temporal patterns that were never encoded. This mirrors graphene: a 2 × 2 Hamiltonian cannot represent spin-orbit coupling regardless of precision.

Stability Measures Reliability, Not Accuracy. The Worms family illustrates an important nuance: BoC achieves *higher* raw recall on Worms than LSTM in some bootstraps, yet its $S_{\text{rec}} = 54.1\%$ signals unreliable predictions. When a model's output varies wildly across resamples, high average performance is not trustworthy. Stability diagnoses *reliability of the structural match*, not accuracy of any single run.

E13 Verdict

Verdict: Aggregate metrics hide structural blindness. PIH reveals that: (1) aggregate accuracy and stability mask per-family blindness; (2) sequence structure is necessary for stable family-level representation; (3) per-family stability (S_{rec}), not aggregate fit, is the diagnostic. BoC is a Blind Lens for sequence-dependent families (Spyware, Adware) and an unreliable lens for Worms.

6.2 E14: Adversarial Attack Surface Discovery

A proof-of-concept that stability correlates with adversarial vulnerability: PIH's stability metric identifies attackable inputs without adversarial training.

The Duality. Defensive (E13): high S_θ means "don't trust this output." Offensive (E14): high S_θ means "attack here."

Method. (1) Train $N = 5$ bootstrap models on resampled MNIST [28] training data. (2) For each test input, compute $S_i = \text{Var}(\text{predictions across bootstraps})$. (3) Partition: PIH-Vulnerable (top 50% by S_i) vs. PIH-Robust (bottom 50%). (4) Attack with FGSM [27] ($\epsilon = 0.2$).

Results.

Input Type	**Attack Success**	**Mean L2**	**Mean S_i**
PIH-Vulnerable	**100.0%**	3.561	0.165
Random	99.0%	3.592	N/A
PIH-Robust	84.0%	3.726	0.000

PIH-vulnerable inputs have +16 percentage points higher attack success than robust inputs. Note, however, that random inputs already achieve 99% success, so the enrichment over random is modest (+1 pp). The primary signal is separation from the specially selected robust subset, not strong enrichment over baseline. PIH stratifies inputs by relative adversarial fragility, but does not identify a qualitatively distinct "attack surface."

E14 Verdict

Verdict: Stability correlates with adversarial vulnerability on MNIST. PIH-vulnerable inputs (high S_i) show 100% attack success vs. 84% for PIH-robust inputs (+16 pp), suggesting that the same stability metric useful for defensive diagnosis (E13) can also guide offensive analysis. This is a proof-of-concept on a single dataset; generalization to larger-scale or non-image domains remains open.

6.3 E12c: Side-Channel Integrity

AES is mathematically secure, yet real implementations leak secrets through side channels. This is the lens blindness problem: the logical specification (constant power) is a Blind Lens for the physical implementation (data-dependent power).

Setup. Simulated CPA on AES S-box: true key `0x5A`, 1000 traces, $P = 50 + 2 \cdot \mathrm{HW}(\mathrm{Sbox}(p \oplus k)) + \text{noise}$.

Results.

Lens	**PPC Pass Rate**	S_θ
Logical (constant power)	**0.0%**	0.3% (stably wrong)
Hamming Weight	**100.0%**	1.8% (stable)

The Logical lens fails every discriminator. It is "stable" ($S = 0.3\%$) but *stably wrong*—stability is meaningless when adequacy fails. The HW lens recovers $P = 50.13 + 1.99 \cdot \mathrm{HW}$ (truth: $50 + 2 \cdot \mathrm{HW}$).

Across 5 SNR levels (0.5–5.0), the HW lens maintains stable α ($S_\alpha = 1.8\%$) while correlation improves from 0.586 to 0.990—a trend the Logical lens cannot predict.

Side-Channel as Lens Blindness

Conclusion: An "insecure implementation" is a blind lens that ignores the physical substrate. The specification is not wrong—it is incomplete as a model of reality. Security proofs are lens-dependent; countermeasures (masking, constant-time code) are lens repairs.

7 Research Direction: Discovery-Security Duality

The experiments suggest an operational analogy between model auditing and security testing. In PIH, the analyst searches for algebraic structure T such that lens M_T achieves high S_θ and passes PPC. In offensive security, the analyst searches for contexts where an implementation's invariants break. These are contravariant operations: discovery moves toward the Faithful Lens quadrant; exploitation moves away from it.

Status: Conjecture. This analogy is a *research direction*, not a developed framework. The SLC framework [6] provides a formal pipeline for deriving logic and computation from algebraic structure, and speculatively notes that privilege-escalation conditions could be modeled as reachability queries in policy algebras. We conjecture—but have not proved—that violations of algebraic axioms (e.g., missing identity elements, broken associativity, absent inverses, lattice-ordering violations) could systematically map to vulnerability classes (anonymous elevation, request smuggling, irrevocable credentials, permission escalation). Making this precise requires threat modeling, composition guarantees, and side-channel analysis beyond what either PIH or SLC currently provides.

What E12c–E14 *do* demonstrate is that the PIH machinery (PPC failure, stability analysis, lens horizons) transfers operationally from physics to security domains. Whether this operational transfer reflects a deeper structural correspondence remains open.

8 Related Work

Bayesian Workflow and Model Criticism. PIH builds on posterior predictive checking [7], systematized in Gelman et al.'s workflow [8]. Gabry et al. [9] provide visualization tools; Vehtari et al. [12] develop LOO-CV diagnostics; Talts et al. [10] validate inference via simulation-based calibration. PIH extends this tradition with discriminator-specific failure localization, lens horizons, typed verdict certificates, and audit logs as first-class output. Graphical PPCs remain the primary diagnostic in PIH; the numeric thresholds in the decision ladder codify what a careful analyst would conclude from plots, enabling automation without replacing visual inspection.

Model Discrepancy. Kennedy and O'Hagan [14] introduced model discrepancy; Brynjarsdóttir and O'Hagan [15] show ignoring it leads to overconfident posteriors. PIH operationalizes bounded ignorance through the repair competition: discrepancy is diagnostic (pointing toward missing mechanism), not passive (added uncertainty). The τ-ratio threshold provides a principled criterion for choosing between bounded-ignorance and structural expansion.

Optimal Experimental Design. BOED selects experiments to maximize information gain [16, 17]. PIH's adversarial contexting targets contexts where structural validity changes most rapidly, maximizing lens discrimination rather than parameter precision.

Program Synthesis. Program synthesis [18] and symbolic regression [19, 20] discover structure from data. E11 demonstrates structure-completing synthesis guided by PPC constraints, not curve-fitting: discovered terms have mechanistic interpretation, not just predictive accuracy.

9 Limitations

We are direct about what PIH does not do.

Low-dimensional experiments. All experiments use $p \leq 4$ parameters with grid-based posterior approximation. Scaling to high-dimensional models requires MCMC or variational inference, which we have not tested.

Mostly synthetic data. Most experiments use synthetic data with known ground truth. E13 (malware) and E14 (adversarial) are the only real-data validations; E12c uses simulated AES traces. An earlier version of this work included cross-domain validations in finance (Black-Scholes vs. jump-diffusion) and network traffic (Poisson vs. Pareto), but these were textbook structural hierarchies that added little beyond the graphene demonstration. Real-world applications involve unknown systematics and domain-specific complications absent here.

Stability thresholds are conventional. Thresholds ($\alpha = 0.05$, S_θ bins at 10%/20%) are calibrated to the graphene system. Different domains may require different thresholds. Systematic sensitivity analysis would strengthen confidence.

Discriminator design is a craft skill. PIH results depend on discriminator choice. A practitioner could "discriminator-hack" by omitting tests that would fail. PIH makes the suite explicit so it can be challenged, but does not solve the design problem.

PIH-SLC bridge is conceptual. The connection between scientific discovery and security analysis is an operational analogy, not a formal theorem. The algebraic toolkit mapping is a conceptual framework, not an empirically validated classification.

Baselines are textbook examples. The cross-domain pairs (Dirac/TB, Black-Scholes/jump-diffusion, Logical/HW) are textbook structural hierarchies. PIH's value is clearest when comparing lenses of different structural classes; distinguishing lenses within the same class is harder.

Side-channel uses simulated traces. E12c demonstrates the principle (logical models are blind to physical leakage) but real hardware introduces noise, countermeasures, and device variation not addressed here.

What PIH is not. PIH is *not* a proof of correctness, a replacement for domain expertise, a universal applicability claim, a formal security framework, or a guarantee of truth. It is a structured auditing protocol that makes model trust explicit, replayable, and falsifiable within a specified discriminator suite, context range, and tolerance.

10 Conclusion

PIH reframes scientific modeling as structural auditing: models are lenses with horizons. The central output is not a fitted parameter vector, but an audit trail explaining which discriminator failed, where the lens is blind, which repair class is warranted, and where the lens ceases to be useful.

Key Results Summary.

Experiment	**Domain**	**Key Finding**	**Data**
E4 (Strain)	Physics	Diagnose → repair loop closes	Syn
E5 vs. E8	Physics	Stability distinguishes noise from structure	Syn
E11 (Synthesis)	Physics	Kane-Mele derived, 100% robust	Syn
E12c (Side-Channel)	Security	Specification blindness to substrate	Syn
E13 (Malware)	Security	Aggregate metrics hide per-class blindness	Real
E14 (Adversarial)	Security	Stability stratifies adversarial fragility	Real

What PIH contributes. The individual components—posterior predictive checks, model discrepancy, algebraic expansion—are not new. PIH's contribution is their packaging into a structured, replayable protocol with typed verdicts, explicit discriminator commitments, and machine-readable audit logs. This protocolization is what enables third-party verification, not a guarantee of truth.

References

References

[1] A. Gelman, J. B. Carlin, H. S. Stern, D. B. Dunson, A. Vehtari, D. B. Rubin. *Bayesian Data Analysis*, 3rd ed. CRC Press, 2013.

[2] A. H. Castro Neto, F. Guinea, N. M. R. Peres, K. S. Novoselov, A. K. Geim. The electronic properties of graphene. *Reviews of Modern Physics*, 81:109–162, 2009.

[3] M. A. H. Vozmediano, M. I. Katsnelson, F. Guinea. Gauge fields in graphene. *Physics Reports*, 496(4–5):109–148, 2010.

[4] P. Kocher, J. Jaffe, B. Jun. Differential power analysis. *Advances in Cryptology—CRYPTO'99*, LNCS 1666, pp. 388–397, 1999.

[5] E. Brier, C. Clavier, F. Olivier. Correlation power analysis with a leakage model. *CHES 2004*, LNCS 3156, pp. 16–29, 2004.

[6] Martila Research. *Extending Curry–Howard–Lambek: The Structure–Logic–Computation (SLC) Theorem.* Zenodo (preprint), June 23, 2025. DOI: 10.5281/zenodo.15724275.

[7] D. B. Rubin. Bayesianly justifiable and relevant frequency calculations for the applied statistician. *The Annals of Statistics*, 12(4):1151–1172, 1984.

[8] A. Gelman, A. Vehtari, D. Simpson, et al. Bayesian workflow. *arXiv:2011.01808*, 2020.

[9] J. Gabry, D. Simpson, A. Vehtari, M. Betancourt, A. Gelman. Visualization in Bayesian workflow. *JRSS:A*, 182(2):389–402, 2019.

[10] S. Talts, M. Betancourt, D. Simpson, A. Vehtari, A. Gelman. Validating Bayesian inference algorithms with simulation-based calibration. *arXiv:1804.06788*, 2018.

[11] B. Carpenter, A. Gelman, M. D. Hoffman, et al. Stan: A probabilistic programming language. *Journal of Statistical Software*, 76(1):1–32, 2017.

[12] A. Vehtari, A. Gelman, J. Gabry. Practical Bayesian model evaluation using leave-one-out cross-validation and WAIC. *Statistics and Computing*, 27(5):1413–1432, 2017.

[13] S. A. Sisson, Y. Fan, M. A. Beaumont (eds.). *Handbook of Approximate Bayesian Computation*. Chapman and Hall/CRC, 2018.

[14] M. C. Kennedy, A. O'Hagan. Bayesian calibration of computer models. *JRSS:B*, 63(3):425–464, 2001.

[15] J. Brynjarsdóttir, A. O'Hagan. Learning about physical parameters: The importance of model discrepancy. *Inverse Problems*, 30(11):114007, 2014.

[16] K. Chaloner, I. Verdinelli. Bayesian experimental design: A review. *Statistical Science*, 10(3):273–304, 1995.

[17] E. G. Ryan, C. C. Drovandi, J. M. McGree, A. N. Pettitt. A review of modern computational algorithms for Bayesian optimal design. *International Statistical Review*, 84(1):128–154, 2016.

[18] S. Gulwani, O. Polozov, R. Singh. Program synthesis. *Foundations and Trends in Programming Languages*, 4(1–2):1–119, 2017.

[19] S.-M. Udrescu, M. Tegmark. AI Feynman: A physics-inspired method for symbolic regression. *Science Advances*, 6(16):eaay2631, 2020.

[20] M. Schmidt, H. Lipson. Distilling free-form natural laws from experimental data. *Science*, 324(5923):81–85, 2009.

[21] S. Mangard, E. Oswald, T. Popp. *Power Analysis Attacks: Revealing the Secrets of Smart Cards*. Springer, 2007.

[22] V. M. Pereira, A. H. Castro Neto, N. M. R. Peres. Tight-binding approach to uniaxial strain in graphene. *Physical Review B*, 80(4):045401, 2009.

[23] R. N. Gutenkunst, J. J. Waterfall, F. P. Casey, K. S. Brown, C. R. Myers, J. P. Sethna. Universally sloppy parameter sensitivities in systems biology models. *PLoS Computational Biology*, 3(10):e189, 2007.

[24] Cuckoo Foundation. Cuckoo Sandbox: Automated Malware Analysis. `https://cuckoosandbox.org/`, 2014.

[25] C. L. Kane, E. J. Mele. Quantum spin Hall effect in graphene. *Physical Review Letters*, 95(22):226801, 2005.

[26] F. O. Çatak, A. F. Yazı. A benchmark API call dataset for Windows PE malware classification. *arXiv:1905.01999*, 2019.

[27] I. J. Goodfellow, J. Shlens, C. Szegedy. Explaining and harnessing adversarial examples. *arXiv:1412.6572*, 2015. Published at ICLR 2015.

[28] Y. LeCun, L. Bottou, Y. Bengio, P. Haffner. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324, 1998.

A Implementation Reference

Key scripts.

- `inference_engine_graphene_bayes_ppc_v12b_pauli_synthesis.py`: E11 Pauli synthesis engine.
- `e11_hard_mode_v3_extreme.py`: E11 stress test (SNR degradation).
- `inference_engine_sidechannel_v12d_cpa.py`: E12c Side-Channel experiment.
- `hypothesis/e13_malware_api_pih.py`: E13 main experiment.
- `hypothesis/e13_stability_analysis.py`: E13 per-family stability analysis.
- `adversarial/e14_adversarial_pih.py`: E14 adversarial attack surface.
- `inference_engine_graphene_bayes_ppc_v9_sgad.py`: E4–E8 graphene experiments.

Data files.

- `e11_pauli_synthesis.json`: E11 results (Kane-Mele discovery).
- `e12e_sidechannel.json`: E12c results (Logical vs HW model).
- `hypothesis/output/e13_malware_api_real_results.json`: E13 per-family results.
- `adversarial/output/e14_adversarial_pih_results.json`: E14 attack surface results.
- `cross_lens_v8.json`: 12-point $|q|$ sweep.
- `e8_multiple_truths_v9_fixed.json`: E8 algebraic order discovery.

Reproducibility. All experiment code, pre-computed JSON artifacts, and reproduction instructions will be released at `https://github.com/Martila-iO/pih-experiments` upon publication. Results were verified across independent runs with fixed seeds; the repository `README.md` will provide exact commands for each experiment.