SIGIL — The Variational Chain

An Experimental Provenance-Signed BlockDAG Organized by a Conjectural Effective Action

Whitepaper v1.1 · 2026-07-18

`bitknight.dipper688@passmail.net`

SIGIL / Flux Foundation · `sigilgraph.quillon.xyz`

source `d7906ff` · based on `16f05a8` · branch `feat/commons-tithe-routing-0.36.1`

Live experimental network — not production ready.

29M	**2,568 B**	**3 μs**	**0**
live chain height (`sigil-g0`)	fold proof — constant for any chain length	live incremental tip check (`sigil-g0`)	divergence across all adversarial sim gates

Abstract

SIGIL is an experimental DagKnight chain built on the Flux compiler-as-platform, sister network to Quillon Graph. This edition opens with *one equation*: a *conjectural effective action* $\delta\mathcal{S}_{\text{SIGIL}} = 0$ on a Lorentzian 4-manifold, with every protocol component — settlement conservation, emission control, witness attestation, fork topology, the post-quantum floor — corresponding to one field. The correspondence is design discipline and research programme, not accomplished mathematics: Lagrangians are unspecified and the discrete-DAG-to-manifold construction is open (§2). What earns the analogy its keep is measurement: a closed-form gossip delivery law $D(p, r) = 1 - p^r$ (RMS residual 0.4 pp over 22 seeded simulations), a constant-size whole-chain proof, byte-identical multi-producer braid convergence, and 3 μs live incremental tip verification. A live experimental network runs at height ∼29M under accelerated laboratory economics. This paper presents the action, its correspondences, the measured laws, the architecture, and an unusually honest account of what remains conjecture.

Security disclosure — read before interacting with the network. `sigil-g0` is a *live experimental network*, not a production system, running accelerated test economics. Its RPC money routes *are* authenticated — each spend/mint route requires the acting wallet's ed25519 signature plus a monotonic nonce (the 2026-06-10 audit's theft findings are fixed and enforcing). The residual money risk is not at the RPC edge but **deeper in the stack**: block-apply does not yet enforce the producer's signature (consensus item H1 — a height-gated verifier is written and tested but dormant), the live VDF still runs over a forgeable fixed modulus while its trustless replacement sits unwired (C10), and peer-sync ingress is under-authenticated (a stub handshake H2, a follower that trusts peer-declared difficulty H3, keyless wire tip-proofs H4). These are tracked with file:line and fix status in the hardening backlog (§6). Do not route significant value through SIGIL until they close.

● MEASURED ☉ SIMULATION ◇ STAGED ○ OPEN △ *conjectural*

These markers are used consistently throughout — in the field table, the measurements, the architecture, the predictions scoreboard, and the closing checklist. Equations are set in black; color is reserved for results and status.

1 Introduction: why reinvent money, and why a chain needs an action

Money is the technology we are least allowed to get wrong and the one we most take on faith. Its supply is set by discretion, its settlement runs through intermediaries we are required to trust, and the record of who owns what is a private ledger we are shown, not one we can check. Bitcoin's founding insight was that the last property is not a law of nature: a network can agree on a shared ledger with no trusted keeper, and a supply no one can quietly inflate. That insight is now more than fifteen years proven — and more than fifteen years old, and the chains built on it each stopped short of the whole problem.

Bitcoin fixed the money and left the expressiveness: a capped, honest supply and an unforgeable ledger, but deliberately no contracts, no native exchange, no programmable economy — and, like every widely deployed chain, elliptic-curve signatures that a large-scale quantum computer would break. Ethereum added general-purpose programmable state and *already* commits state, transaction, and receipt roots in every execution header. What it does not provide is a node that must *halt* on a root mismatch rather than continue, a fixed 21M-style monetary policy, a post-quantum floor under its signatures, or protocol-level provenance for its released client binaries. So SIGIL's distinction is not "a chain with a state root" — it is the *combination* neither puts in the protocol: domain-separated roots with fail-closed local enforcement, a post-quantum signature floor, build provenance bound to every released binary, and time-normalized capped emission. SIGIL — the experimental sister chain to Quillon Graph — takes the obvious brief: keep what they got right (capped honest money, programmability), add what post-quantum cryptography and a self-signing compiler now make possible, and lose what they got wrong.

Concretely, SIGIL is a chain where:

- **Post-quantum runs from the floor up.** Every release binary carries a *require-both* SQIsign Level-5 *and* Ed25519 provenance proof (~356 B, smaller than one Dilithium-5 signature); the whole chain folds to one constant-size Ajtai/SIS proof with no trusted setup; browser key material is ML-KEM-1024 / ML-DSA-87.
- **Agreement is structural.** Every header commits four state roots — wallet, DEX, event log, contract storage — and a node that drifts on any of them halts itself (exit 78) before it serves a byte. Divergence cannot hide.
- **The economy is programmable.** A contract VM, a constant-product AMM DEX ($x \cdot y = k$), on-chain credit, and a Bitcoin bridge — each committing to its own state root, so the whole programmable economy is inside the structural guarantee, not bolted beside it.
- **It verifies on a potato.** A 2,568-byte whole-chain proof and a 3 μs browser tip-check; the light client downloads zero blocks to hold the tip to account.
- **The money stays honest.** Dual-lane mining (BLAKE-family PoW × Wesolowski VDF), a 21M cap, time-based emission invariant under block cadence, and a fail-loud enforcer that halts the node at ±2% supply drift.
- **One static binary** is node, miner, light client, and wallet.

None of that is new as a *list* — and a feature list is exactly what SIGIL is trying *not* to be. A blockchain is usually presented as a pile of mechanisms: a hash here, a signature there, a consensus rule, an emission curve, each justified on its own and the whole trusted because the parts individually check out. SIGIL is built on the opposite conviction: that a chain should follow from a single organizing object, the way a physical theory follows from an *action*, and that agreement about state should be *structural* — written into the object every node checks, not reassembled from conventions each node is merely trusted to honor. That is why *what the network agrees the state is* became a first-class, committed quantity (the four roots above): agreement stops being faith in a linear ledger and becomes a checkable property of the header itself.

The v0 whitepaper (2026-05-29) stated these as engineering claims. What v0 lacked — and what Quillon's own quantum-physics whitepaper demonstrated the value of — is a *unifying theoretical object* from which the design follows, rather than a list of mechanisms. That organizing object has now been proposed: a conjectural effective action $\delta\mathcal{S}_{\mathrm{SIGIL}} = 0$ in which every protocol component — settlement conservation, emission control, witness attestation, fork topology, the post-quantum floor — corresponds to one field. It was developed in the SIGIL master-equation programme (2026-05-30/31) using the `flux-science` modules already in the workspace, and this edition finally puts it where it was always intended: first (full companion note published standalone; provenance of every measured claim: Appendix A).

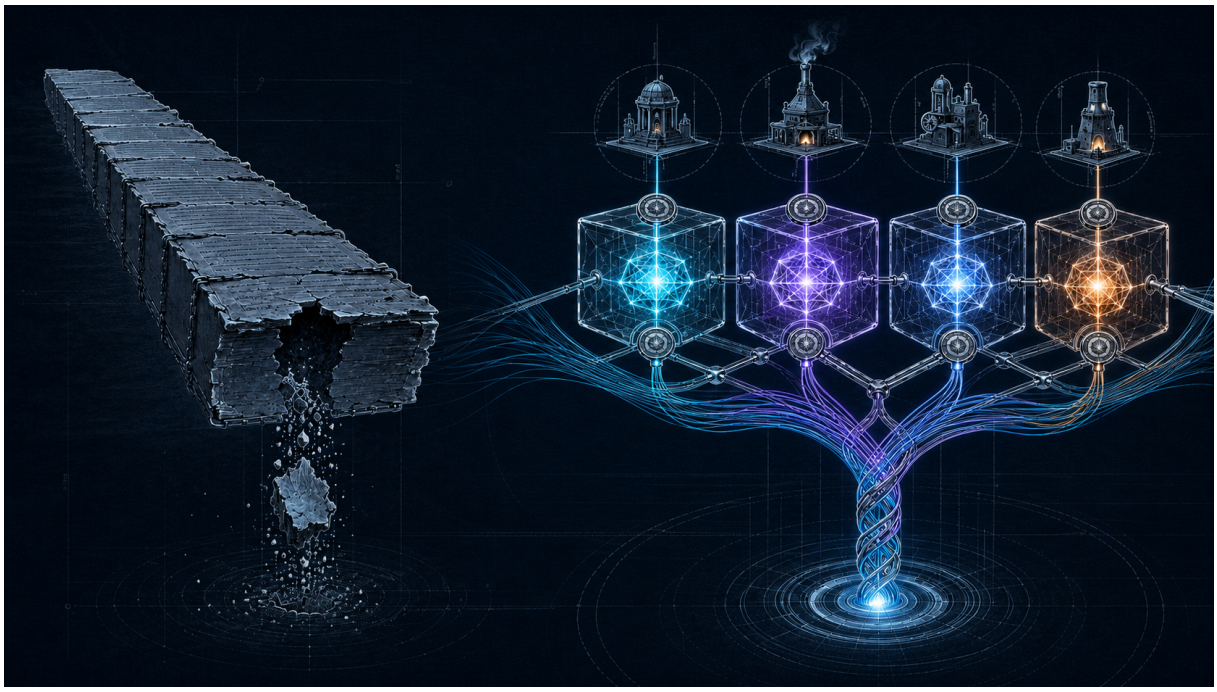

Figure 1: **Structural agreement.** Left: the old way — a linear ledger trusted on faith, with no structural statement of what state the network agrees on, so drift in its latest section can pass unnoticed. Right: SIGIL's answer — transparent blocks carrying luminous state-root structures, each sealed to the machine that produced it, their causal strands braiding down into one verified foundation. Color code as everywhere in this paper: cyan = measured and operational, violet = staged, amber = open or unfinished protection.

Three cultural commitments carry over from v0 and bind everything below:

1. **Measure the live path first.** A microbenchmark reading millions/s next to a live number in hundreds/s *is* the bug. Claims below carry the status markers from the legend above; unmarked prose is description, not claim.
2. **Provenance is not optional.** Every release binary carries an SQIsign Level-5 `.proof` emitted by the compiler itself.
3. **Honesty sections are load-bearing.** Every physics analogy and every engineering claim gets a "what's still pretend" entry (§6).

2 The Master Equation

SIGIL's BlockDAG braid $\mathcal{G}$ — vertices are blocks, edges are causal links — is modeled as a Lorentzian 4-manifold (3 spatial + 1 temporal degrees of freedom). We propose, as a *conjectural effective action*, that its dynamics are organized by one variational principle:

$$\boxed{\delta \mathcal{S}_{\text{SIGIL}}[g,\, A,\, \varphi,\, J,\, K,\, \Sigma] \;=\; 0} \tag{1}$$

$$\mathcal{S}_{\text{SIGIL}} = \int_{\mathcal{G}} d^4\sigma\, \sqrt{|g|}\, \Big[\tfrac{1}{2\kappa}\big(R - 2\Lambda_{\text{emission}}(\varphi)\big) + \mathcal{L}_{\text{settle}}(J, g) + \mathcal{L}_{\text{witness}}(A, g) + \mathcal{L}_{\text{topo}}(K) + \mathcal{L}_{\text{crypto}}(\Sigma, g) \Big] \tag{2}$$

IN PLAIN WORDS — **the one equation**

Picture the whole chain as a stretchy fabric floating in time. Every block is a stitch; every fork is a wrinkle. The equation above is a single sentence claiming the fabric always settles into its least-wrinkled shape — money flowing, miners mining, watchdogs watching, even the cryptography, each one dial on the same machine. We have *not* proven the machine works this way (that's the honest part). But designing *as if* it does keeps every piece of SIGIL pulling in the same direction — and it hands us predictions we can actually test.

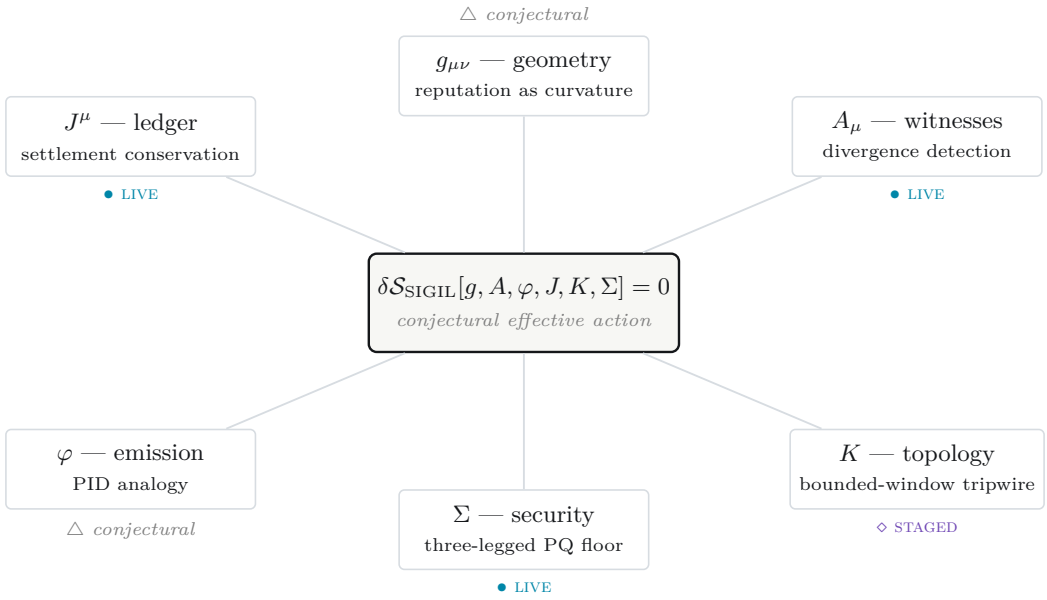

Figure 2: **The Master Equation Map.** Each field of the action names one protocol component; the label states its current epistemic status. LIVE means the corresponding mechanism runs on `sigil-g0` today; STAGED means designed and partially landed; *conjectural* means the correspondence is analogy awaiting derivation or empirical fit.

2.1 The fields

Field	Protocol correspondence	Status
$\mathcal{G}$	The discrete BlockDAG braid, conjecturally modeled as a Lorentzian 4-manifold; the fabric of the chain.	△ *conjectural*
$g_{\mu\nu}$	Conjectured (heuristic) to be induced by agent-reputation density: high-reputation producers bend the metric, consensus flows toward them — reputation as curvature.	△ *conjectural*
R	Consensus tension; forks raise it, convergent histories minimize it.	△ *conjectural*
$\Lambda_{\text{emission}}(\varphi)$	The emission "cosmological constant," PID-controlled toward target supply.	△ *conjectural*
A_μ	Witness attestations; the curvature 2-form $F = dA + A \wedge A$ is the witness-disagreement field. State-root divergence check is live.	● LIVE
J^μ	Settlement flow; conservation enforced at the state chokepoint.	● LIVE
K	The link obtained by closing the bounded-window braid; consensus-equivalent histories share its invariants. Header commitment staged.	◇ STAGED
Σ	Three disjoint post-quantum hardness assumptions, one per layer guarded (§2.2).	● LIVE
κ	The gravitational constant of the agent economy; empirical.	△ *conjectural*

2.2 Correspondences: one field per protocol component

Each variation of $\mathcal{S}_{\text{SIGIL}}$ with respect to one field *names* one subsystem of the running chain. The paper's central organizational claim is deliberately modest: *SIGIL is designed as if it were one action varied six ways*, and that discipline — every new mechanism must say which term it belongs to — is what the equation buys today. What it does not yet buy is derivation: the component Lagrangians ($\mathcal{L}_{\text{settle}}$, $\mathcal{L}_{\text{witness}}$, $\mathcal{L}_{\text{topo}}$, $\mathcal{L}_{\text{crypto}}$) are unspecified beyond their field content, and no discretization of $\mathcal{G}$ into a lattice field theory has been constructed, so the "field equations" below are *correspondences* between standard equations and shipped mechanisms, not consequences extracted from the action. Making even one of them a genuine derivation is the physics programme's headline open problem (§6).

> IN PLAIN WORDS — **"correspondence, not derivation"**
>
> We are saying the protocol *rhymes* with famous physics — not that we've done the mathematics forcing it to. Think of a city map drawn in pencil: genuinely useful for navigating, honest about being redrawable. Each pencil line gets inked only when a measurement makes it earn the ink. One line already has (§3.1); the rest are queued on the scoreboard (§5).

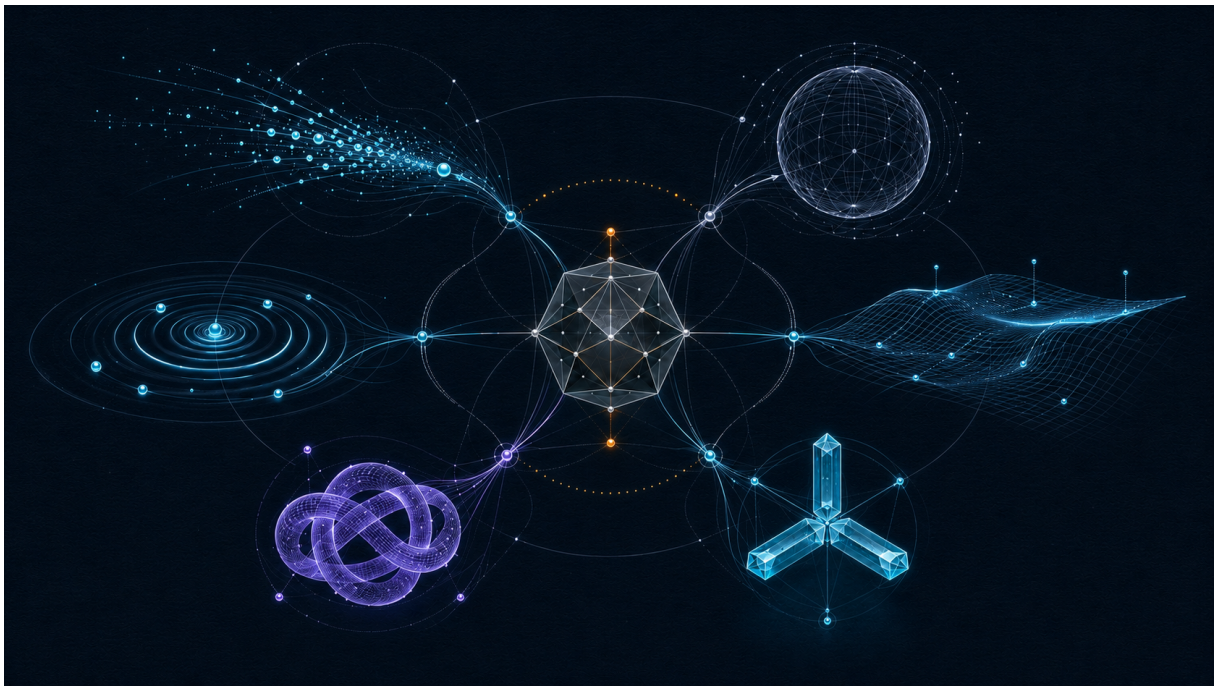

Figure 3: **The conjectural effective action, visualized.** The faceted object at center is the action; the six phenomena around it are its fields — particle streams (settlement flow J), expanding ripples (emission φ), a warped surface (reputation curvature g), a wireframe sphere (global geometry), a violet closed braid (topology K), and a three-legged cyan crystal (the post-quantum floor Σ). Thin orbital curves connect the fields to the center; deliberately, none is drawn as *derived* from it — which is precisely this paper's epistemic claim. Solid cyan = working mechanism, violet = staged, pale/amber = conjectural theory.

$\delta\mathcal{S}_{\text{SIGIL}}/\delta J^{\mu} = 0$ — **conservation (the ledger).**

$$\nabla_{\mu} J^{\mu} = 0 \quad \Longleftrightarrow \quad \partial_t \rho_{\text{value}} + \nabla \cdot \mathbf{J}_{\text{settle}} = S_{\text{emission}}(\varphi) \tag{3}$$

Value is neither created nor destroyed except by the emission source term. In code this is the `commit_state_transition` chokepoint (§4): the *only* function permitted to mutate chain state, with conservation checked per typed delta. The 21M supply cap is the integral form; the live laboratory network reads 21,000,000 of 21,000,000 SIGIL minted under its accelerated test economics.

$\delta\mathcal{S}_{\text{SIGIL}}/\delta\varphi = 0$ — **Klein–Gordon emission (monetary policy).**

$$\ddot{\varphi} + 3H\dot{\varphi} + V'(\varphi) = 0, \qquad V(\varphi) = K_p\, e^2 + K_i\, e \int e\, dt + K_d\, (\dot{e})^2 \tag{4}$$

The PID emission controller is *cast as* a scalar field rolling in a potential; the Hubble friction term $3H\dot{\varphi}$ is network expansion (new blocks arriving). Inflationary cosmology as a mnemonic for economic policy. The displayed V is *schematic*: e, $\int e\,dt$, and $\dot{e}$ are controller states, not functions of φ alone, so V is not an ordinary local potential — a faithful treatment introduces auxiliary fields (or defines $e(\varphi)$ explicitly). The mapping is exact only for quadratic potentials. Stability prediction, from the linearized characteristic equation $K_d s^2 + K_p s + K_i = 0$: the controller is non-oscillatory iff $K_p^2 \geq 4K_dK_i$ (§5; this corrects the discriminant stated in the v0.1 master-equation note, which had the roles of K_p and K_d interchanged — a fresh derivation against the implemented controller is part of the test).

$\delta\mathcal{S}_{\text{SIGIL}}/\delta K = 0$ — **topological invariance (fork detection).**

$$\frac{d}{dt}\,V_K(q)\Big|_{\text{consensus}} = 0 \tag{5}$$

where K is the link obtained by closing the DAG's braid over a bounded window. Valid consensus moves are Reidemeister moves on the braid; invariants are conserved. A fork *may* change the bounded-window invariant — when it does, divergence is proven; equality of invariants never proves equality of histories. **Two honesty notes.** *Computational:* the Jones polynomial is #P-hard in general — nobody runs "Jones consensus at 10k blocks/s." SIGIL's tractable programme (QTFT lanes) commits a *cheap* invariant per block over a bounded window — linking number (O(crossings)) or the Alexander polynomial (polynomial-time via the Burau determinant) — reserving Jones/Khovanov machinery for rare deep-fork adjudication off the hot path. *Logical:* cheap invariants fingerprint, they do not classify; the topology check is a one-way tripwire complementing (never replacing) the state-root comparison. The braid ordering layer this attaches to is live: the DagKnight braid shipped to the live experimental network's producer 2026-07-02 (§3.3).

$\delta\mathcal{S}_{\text{SIGIL}}/\delta A^{\mu} = 0$ — **Yang–Mills witnesses (divergence detection).**

$$\nabla_{\nu}F^{\mu\nu} = J^{\mu}_{\text{witness}}, \qquad F = dA + A \wedge A \tag{6}$$

When two state-root tracks diverge, the curvature F becomes locally non-zero; divergence detection is the measurement of $|F|$ above threshold. (Healing is *re-synchronization dynamics* that drives F back toward zero — not a gauge transformation, which merely relabels the connection and cannot repair genuine disagreement; curvature is gauge-covariant.) In code: four state roots per header, computed independently by every node, with mismatch ⇒ self-halt (exit code 78, no auto-restart). The adversarial simulation gate measured divergence = 0 across all six scenarios (§3.3) — the flat-connection condition holds on the paths tested.

$\delta\mathcal{S}_{\text{SIGIL}}/\delta\Sigma = 0$ — **crypto-agility (the security floor).** The security connection has three components living on three disjoint hardness assumptions, each placed where its strength matters and its weakness doesn't:

Component	Primitive	Layer	Why there
Σ_{iso}	isogeny — SQIsign L5	genesis / identity / provenance	tiny sigs (292 B), rare keys: sign once, permanent
Σ_{lat}	lattice — ML-DSA / ML-KEM	per-transaction / transport	high volume → fast verify, GPU-batchable
Σ_{hash}	FRI-STARK / Ajtai-SIS fold	state / tip-proof	whole-chain succinct verification, no trusted setup

The correspondence, taken with the weakest-component constraint, motivates the dispatch rule: retire Σ_i the instant its hardness margin drops; the other legs carry what they guard. **Composition, stated carefully:** because the three primitives protect *different layers*, the security of any *single property* is the floor of the primitives guarding *that* property — an adversary targeting, say, transaction forgery attacks the lattice leg alone and need not touch the others. The "break-all-three" bar applies only to forging a block's *complete* commitment stack (provenance + authorship + state proof) as a coherent whole, and a formal joint security model with a composition bound is open work (§6). The lattice transport leg is measured live: ML-KEM-1024 key encapsulation over the browser↔bridge noise channel, proven end-to-end at 115 ms. The hash leg is the fold (§3.2). The isogeny leg signs every release `.proof` and the DNS tip anchor.

The constraint surface — the wall. SIGIL imposes the engineering feasibility condition $\Gamma_{\text{verify}} \geq \Gamma_{\text{state}}$; we call the boundary "the wall." It is an operational constraint, not a proven condition for the existence of an action extremum. At the 2026-05-30 measurement the ratio was ∼1840:1 against verification. The resolution came exactly where the crypto term predicted: not faster per-signature verification but a change of Σ_{hash} collapsing whole-chain verification to $O(1)$ (§3.2). *The wall was a cryptographic choice, not a hardware limit* — and it moved.

3 Measured laws

This section is what distinguishes SIGIL's physics programme from decoration: closed-form laws with residuals, gates with pass/fail, and live numbers.

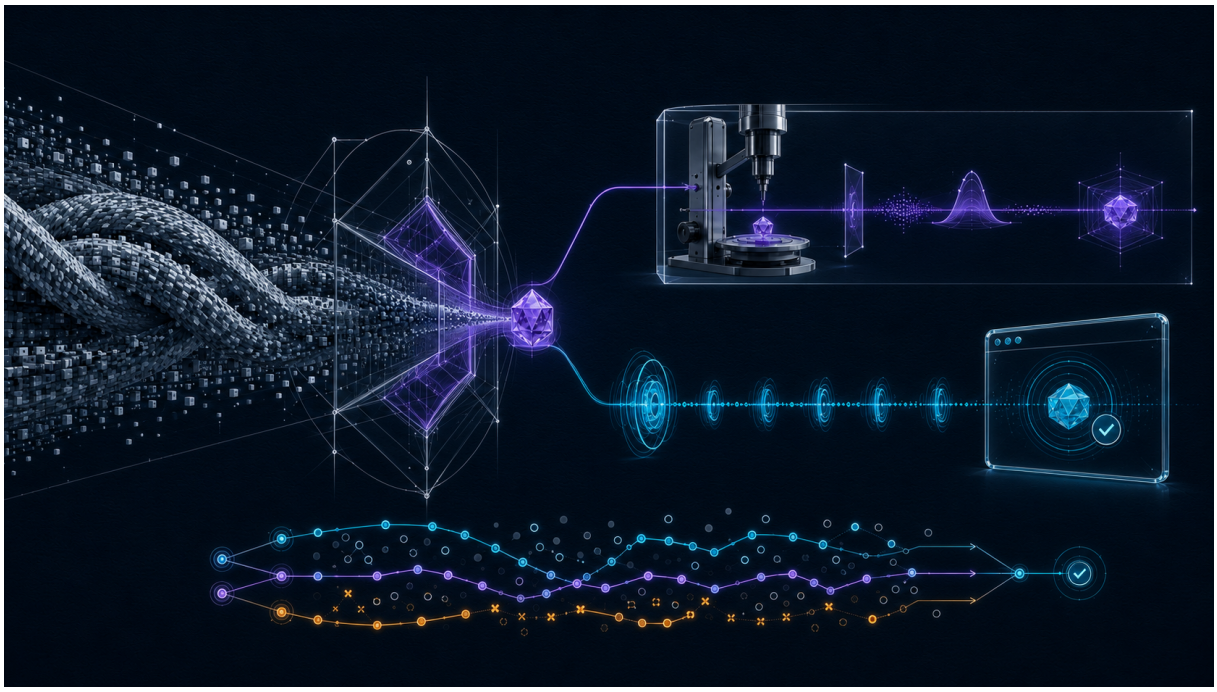

Figure 4: **Measured laws.** A vast braid of blocks is compressed through a geometric aperture into one small proof crystal. Two verification paths leave it: the violet lane enters a laboratory instrument — *full fold verification under benchmark conditions* — while the cyan lane pulses rapid checks to a lightweight glass client — *live incremental tip verification against an already-verified state*. Along the bottom, redundant gossip routes cross a field of simulated packet loss: individual transmissions vanish, delivery survives — the $1 - p^r$ law of §3.1. The two lanes are separate experiments, as in Figure 5.

3.1 The gossip delivery law

● **measured**

For a block gossiped with redundancy r under independent per-transmission loss probability p, the probability a node receives it at least once is

$$\boxed{D(p, r) = 1 - p^r} \qquad \text{inverted to the provisioning rule} \qquad r^* = \left\lceil \frac{\ln(1 - \text{SLA})}{\ln p} \right\rceil \tag{7}$$

Derived and validated against 22 seeded flux-chronos star-flood simulations; RMS residual **0.4 percentage points**. The shipped default $r = 3$ satisfies a 99.9% SLA up to $p = 0.1$. This is statistical mechanics of the gossip layer done properly: model, measurement, closed form, residual, and a provisioning rule operators actually use. (Standalone paper published at `https://quillon.xyz/downloads/sigil-top-delivery-law.pdf`.)

IN PLAIN WORDS — **the delivery law**

If a letter gets lost one time in ten, send three copies — the chance all three vanish is one in a thousand. That's the entire law: $1 - p^r$. We didn't guess it; we ran 22 controlled storms of lost messages and the formula matched reality to within half a percentage point. The network now *calculates* how many copies to send instead of hoping.

3.2 The fold: succinct whole-chain verification

● measured

The Σ_{hash} leg materialized as `flux-fold`: an Ajtai/SIS-based folding scheme (post-quantum, no trusted setup) that compresses verification of the entire chain prefix into one constant-size check.

2,568 B	342 ms	3 µs	0
fold proof size — constant, any length	full fold verification (100k-block benchmark)	live incremental tip check	blocks downloaded by the live verifier

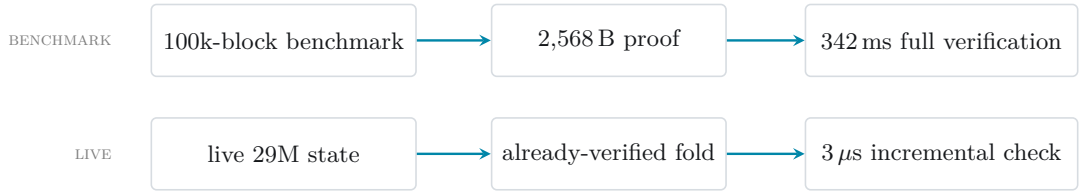

Figure 5: **Fold verification flow — two lanes, two measurements.** The 342 ms full verification was measured at the 100k-block benchmark; the 3 µs incremental check runs on the live 29M-block network against an already-verified fold state. The lanes are separate experiments and must not be read as one pipeline; full verification of the live 29M fold has not yet been separately benchmarked.

The *full fold verification* checks the folded proof of the entire chain prefix from scratch; the *incremental tip check* verifies the latest tip against an already-verified fold state — cheap precisely because the heavy work is folded. The 342 ms figure is now a **measured, reproducible number** (280–342 ms across runs on a loaded 48-core host; run `fold-verify-bench` in `flux-fold` to reproduce), not the poster label it once was. One honest nuance the benchmark surfaces: the *proof* is constant-size (2,568 B, independent of chain length), but full verification is $O(M)$ — the verifier reads all M commitments (∼51 MB at 100k blocks), because `flux_fold::verify` recomputes the random linear combination. Constant-size proof, not constant-cost verification; the incremental tip check is what makes the steady state cheap. Because the proof is constant-size, the chain length covered per unit of verification time grows with the chain; we deliberately do not quote that ratio as a throughput number. The founding claim was tip verification in ≤10 ms in a browser; the live incremental check beats it by three orders of magnitude, and the wall of §2 ceased to be the binding constraint. The precise security statement of the fold — the reduction to Ajtai/SIS with concrete parameters, and what a verifier is entitled to conclude from a 2,568 B proof — is stated as a target, not yet a published proof (§6).

IN PLAIN WORDS — **the fold**

Normally, auditing a blockchain means re-reading the whole book — millions of pages. The fold compresses the *proof of having read it* into a note the size of a text message: 2,568 bytes, the same size whether the book has a hundred thousand pages or twenty-nine million. Checking the note takes a blink (342 ms on our benchmark); after that, each new page costs three *millionths* of a second. Your phone — honestly, a potato — can hold the chain to account.

3.3 DagKnight braid convergence

⊙ simulation

The braid ordering layer (`sigil-dagknight` + `flux-topology`) passed a six-scenario adversarial simulation gate before deployment to the live network:

- ☑ S1 — 10k blocks / 3 producers, dual-instance: divergence = 0
- ☑ S5 — 30% drop / reorder / equivocation: divergence = 0

- ☑ Orderings — 16/16 permutations agree; incremental ≡ batch
- ☑ Withheld-block attacker — finalized prefix byte-identical (1378 blocks)
- ☑ Tamper suite — 5/5 rejected
- ☑ S6 — 100k blocks, max reorg window 258 ≤ 16384 bound

A two-producer live mesh ran 3385 emissions byte-identical before the experimental network's producer flipped to the braid binary at height 17.52M (2026-07-02). In the language of §2: on every path tested, the topological invariants were conserved and the witness curvature stayed flat. Simulation convergence is encouraging evidence; it is not a safety or liveness proof (§4.7).

> IN PLAIN WORDS — **the braid**
>
> When several writers add pages at once, whose ordering wins? DagKnight braids the parallel pages into one agreed story — and we attacked our own braid six ways: dropped a third of the mail, shuffled what survived, had an author lie and withhold pages. Every copy of the story still came out *byte-for-byte identical*. That's stress-test evidence, not a theorem; the theorem is on the to-do list, and we say so out loud.

3.4 The sync ladder and the honest bottleneck

The throughput programme is a case study in the "measure the live path first" rule. Local pipeline stages benchmark at 251k–2.5M blocks/s and a flat append-only skeleton store (72 B/record, two-phase durable, torn-tail safe) sustains ∼10M records/s durable — 2600× the general-purpose LSM write path (∼130 μs/KV ≈ 3.9k blocks/s). An in-process end-to-end harness (pull → stream-verify → commit at 2M records, roots matching) reaches 39k blocks/s. And yet the live sync number sat at ∼800 blocks/s for five days — because the actual caps were a producer-side full-block serve throttle (≤1 per 120 ms) and a serial request→await→apply fetch loop, *not* the local pipeline. Once measured directly, both were removed: the serve throttle is gone (finalized ranges are cache-served by `memcpy`) and a request-ahead window (up to ∼16 ranges in flight, apply-backpressured) replaced the serial loop — **live 2,322 → 10,853 blocks/s** on the running node. The lesson is now Rule 0 of the SIGIL skill and a methodological claim of this paper: **a fast microbenchmark next to a slow live number is a bug report about your understanding, not a victory** — and the fix is the one that moves the *live* number, which this one did. (Snapshot-pull, a further 100× path, is coded and byte-symmetric but stays gated off pending a published signed anchor; §6.) Wire compression is measured at 14.0× (zstd backfill codec, 70.7 B/header live) and 26.3× (gossip frames).

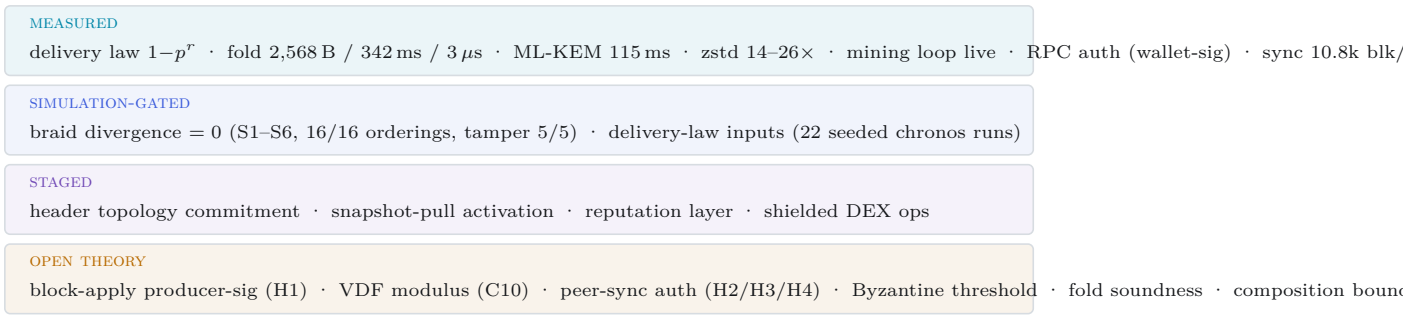

Figure 6: **The evidence ladder.** Where every major claim of this paper currently sits. Items move *up* the ladder only by measurement; the honesty sections (§6) enumerate what would move each one.

4 Architecture

Condensed; `SIGIL_GENESIS_v0.md` remains the full spec. Network `sigil-g0`, libp2p prefix `/sigil/g0/`, P2P :9501, API :8181/:8099, storage `flux-db` (no RocksDB), all builds via `fluxc` with content-hash caching — never raw cargo.

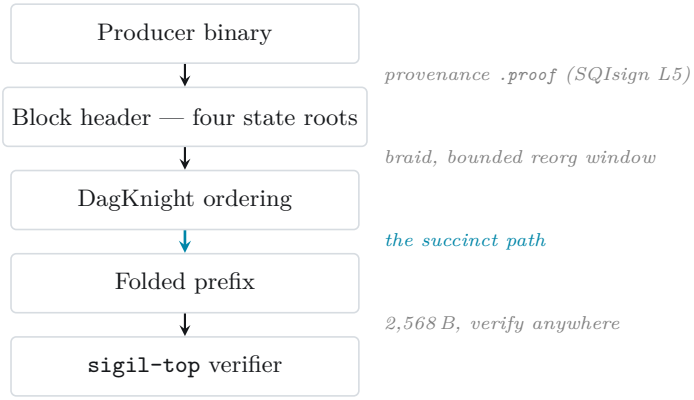

Figure 7: **The verification pipeline.** One cyan edge marks the path that made the wall move: ordered blocks fold into a constant-size proof any client can check.

Figure 8: **Provenance-signed architecture as one motion.** The compiler press at top seals a transparent block of state-root crystals and stamps it with the provenance of the machine that produced it. The causal strands beneath braid into order, the braid folds through the circular aperture, and one small cyan proof crystal travels a single illuminated path to a lightweight glass verifier. Provenance → commitment → ordering → folding → light verification — the same pipeline as Figure 7, drawn as the machine it is.

4.1 The header: three commitments per block

Commitment	Proves	Catches
4 state roots (wallet / DEX / events / contracts)	the state is as claimed	silent state divergence (Quillon's bug)
`fluxc .proof` (SQIsign L5, 292 B)	which binary produced the block	un-attested / forged builds
topology invariant (QTFT lanes, staged)	the committed bounded-window invariant is as claimed	forks that change the selected invariant

Plus dual-lane mining evidence (§4.4) and the producer signature under the crypto-agile `sig_scheme` dispatch. State-root divergence and topology divergence are *independent* halt signals — two different variations of the same action.

> IN PLAIN WORDS — **the four fingerprints**
>
> Every block carries four fingerprints of the *entire* ledger: balances, exchange pools, events, contracts. Every node recomputes all four from scratch and compares. If yours don't match, your node doesn't argue and doesn't guess — it stops serving and raises its hand (exit code 78, alarm fires). Our sister network once lost a user's funds *silently*; SIGIL is built so that silent is structurally impossible.

4.2 State: one chokepoint

● live

`sigil-state::commit_state_transition` is the only function that may mutate state; every other write path is `pub(crate)`, enforced at MIR-time by `flux_ai_audit`. Typed delta variants (`SwapDelta`, `LpDelta`, `LpBurnDelta`) are validated as units — token pairs match, immutable pool fields stay byte-identical, reserves shift exactly as declared, balances cover the operation — before expansion to primitives. Boot-time pre-flight recomputes roots over sampled blocks and refuses to bind ports on mismatch.

4.3 Light client first: sigil-top

● live

The flagship operator experience is `sigil-top`: a single binary whose default invocation opens a TUI light node — tip-proof verification (fold-backed), the four state roots, supply, peers, mining — and serves the embedded browser wallet at `localhost:9800` with a same-origin `/api` proxy, no CORS, no build step. It syncs via the delivery-law-provisioned gossip mesh and the skeleton/snapshot pull path (§3.4), auto-updates from a signed release manifest with a BLAKE3-pinned client key, and verifies the DNS tip anchor: an SQIsign-L5-signed `v=sigil1` TXT record at the apex, so a fresh node can authenticate the tip before speaking to any peer.

4.4 Dual-lane mining: Φ POWER × Ω TIME

● live

A valid share carries *both* lanes: Φ — BLAKE4 proof-of-work, and Ω — a Wesolowski VDF proof of elapsed sequential time. *BLAKE4* is SIGIL's name for its parameterized proof-of-work construction built from the BLAKE3 compression function — not a newly established cryptographic hash. The round count is the parameter: at $R = 7$ it is KAT-verified byte-identical to BLAKE3 (inheriting BLAKE3's security analysis wholesale); reduced-round variants are an experimental, measured speed lever (2.9× at $R = 1$, scalar) with no security claim attached. Consensus stays at $R = 7$ unless a reduced round count is ever cryptanalytically validated. The TUI miner and the node share one verification function and one cap-enforced crediting chokepoint; the loop is proven live (first credited shares $0 \to 1250$ on `sigil-g0`).

> IN PLAIN WORDS — **proof of work × proof of time**
>
> A valid share must show two different things: that you *burned effort* (a hash puzzle) and that you *genuinely waited* (a sequential timer that cannot be fast-forwarded — not even with a warehouse of GPUs). Effort can be bought in bulk; honest elapsed time cannot. A block needs both, which is why the two lanes are multiplied, not added.

4.5 Privacy by default

● live

There is no opt-in path to transparency: every send is shielded (commitment + nullifier), with real post-quantum STARK proving/verification behind the `real-zk` feature (FRI-verified rejection of tampered proofs, 31 tests). Current limits are stated, not hidden: single-input/single-output sends; Phase-0 commitments are BLAKE3-based pending Pedersen; shielded DEX operations are staged.

4.6 Economy

21M hard cap with a halving schedule whose *production parameters* span ∼257 years of emission. The current `sigil-g0` run shows supply fully minted because the laboratory network runs *accelerated test economics* — dev-cadence block production compressed the schedule; a production genesis would restart emission under the long-horizon parameters. The two numbers describe the lab run and the design, respectively, not a contradiction. On top of the base ledger: the DEX (constant-product AMM, checked u128 arithmetic, reserve floors), sigil-bank credit vault (50% LTV against staked SIGIL), the SSHARE treasury instrument, the Laniakea University credential economy (treasury transfers, never mints — conserves the cap), and the agent economy: SIGIL's entire development is settled on-chain between AI agents and the operator, claim → ship → settle, with the swarm message bus as the audit trail. This paper itself was produced under that regime.

4.7 Consensus model and open formal work

Consensus status at a glance.

Current topology	one producer + verifying followers
Formal Byzantine threshold	open
Multi-producer convergence	simulation-gated (S1–S6, divergence = 0)
Fold soundness reduction	open (Ajtai/SIS target stated)
Fork choice	DagKnight ordering over the braid
Finality	operational rule (fold coverage), not a theorem

A technical reader deserves the conventional statements, so here they are, at their current maturity. **Network model:** partial synchrony is assumed informally; no explicit GST bound is claimed. **Producer topology:** the current `sigil-g0` run is a single-producer network with verifying followers; the braid layer is multi-producer-capable (§3.3) but no Byzantine fault threshold is claimed for the live deployment. **Fork choice:** DagKnight ordering over the braid; the finalized prefix advances with the anchored fold checkpoint. **Finality:** a block is treated as final when covered by the fold; this is an operational rule, not yet a theorem — the fold's security reduction (Ajtai/SIS, concrete parameters) is a stated target. Note the distinction: the fold proves the *validity* of a selected prefix; it does not by itself prove that the prefix is *canonical*. On the current single-producer `sigil-g0` network, canonicality comes from the signed anchor and producer trust; multi-producer finality remains open formal work. **Sybil resistance and validator admission:** dual-lane mining prices block production; the reputation layer (`sigil-scoring`) is staged and no reputation-based admission claim is made yet. **Status of evidence:** the six-gate adversarial simulation record of §3.3 is exactly that — a reproducible gate record, encouraging but not a safety/liveness proof. Producing the formal statements (threshold theorem for the braid, fold soundness bound, joint-crypto composition model) is the protocol-theory half of the research programme, sequenced after the measured-law half precisely because the culture here is to measure first and formalize what survives.

IN PLAIN WORDS — **where consensus honestly stands**

Today, one author writes and everyone else verifies — that's the truth of the current network, stated in the panel above. The machinery for many simultaneous authors exists and passed its stress tests, but "passed tests" and "mathematically safe" are different sentences, and this paper refuses to blur them. The scoreboard below tells you exactly which sentence each claim has earned so far.

5 Predictions: the research scoreboard

The master equation earns its place only if it predicts. Status as of 2026-07-18:

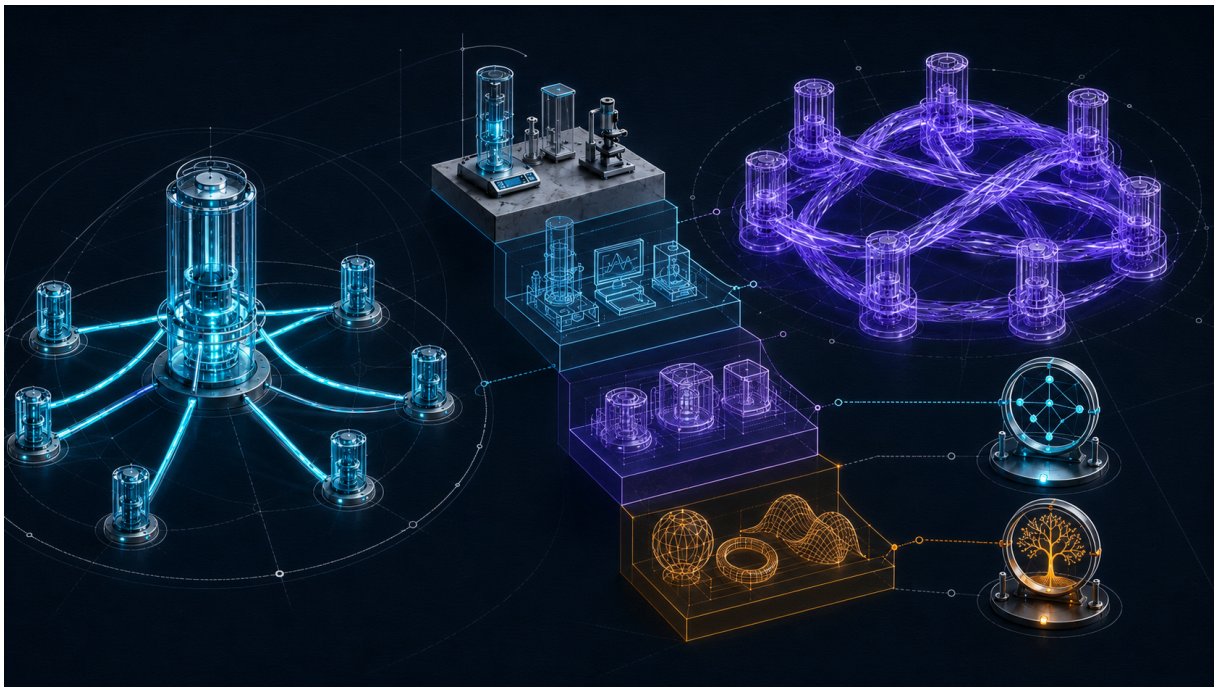

Figure 9: **Consensus maturity, honestly drawn.** Left: the live network as it actually is — one solid cyan producer tower feeding verified information to smaller follower nodes. Upper right: the staged multi-producer braid, rendered translucent because it is simulation-gated, not live. Between them, the evidence ladder of Figure 6 in physical form: measured mechanisms sit solid on the top platform, simulation-gated and staged assemblies grow increasingly transparent, open theory remains amber wireframe on the lowest level. The two instruments at right are the independent divergence detectors: state-root comparison and the topology tripwire. Solid is live; translucent is tested; wireframe is not yet earned.

Prediction	Status	Evidence	Next falsification test
1. Fork survival $\propto \exp(-\Delta R\,\tau_{\text{block}})$	○ OPEN	none yet	fork-survival statistics vs. producer reputation
2. PID non-oscillatory iff $K_p^2 \geq 4K_dK_i$ (corrected; char. eq. $K_ds^2+K_ps+K_i=0$)	○ OPEN	none yet	derive against the implemented controller, then tune to critical value on testnet
3. Topology commitment detects order-tampering forks	◇ STAGED	braid gates: 16/16 orderings agree; tamper 5/5 rejected	land the header commitment; measure detection latency vs. state-root-only
4. The verification wall scales with the crypto term, not hardware	● MEASURED	the fold moved the wall to $O(1)$/2,568 B; per-sig SIMD did not	publish the fold soundness reduction
5. Bulk/boundary duality (settlement throughput $\sim$ boundary entropy production)	○ OPEN	none yet	compare settlement rate to reputation-graph entropy
6. Validator-departure entropy law (Hawking analogue)	○ OPEN	none yet	tip-proof rate vs. validator-set entropy
7. Gossip delivery follows $1-p^r$ *(new since v0)*	● MEASURED	22 simulations, RMS 0.4 pp; shipped as the r=3 default	re-fit under correlated (non-independent) loss

Prediction 7 deserves emphasis: it is the first SIGIL law that went the full arc — hypothesis, seeded simulation campaign, closed form, residual, operator-facing rule, published paper. That arc is the template the remaining predictions are held to.

6 Honest status

6.1 What's real (with evidence)

Live braid ordering on the experimental network's producer; four state roots computed and committed (wallet root live and non-trivial; DEX/event/contract roots currently zero pending those subsystems' activation); fold tip verification at 3 μs on the real chain; dual-lane mining credited live; delivery-law-provisioned gossip; zstd wire compression live at 70.7 B/header; skeleton store and snapshot-pull code landed and harness-proven; ML-KEM-1024 browser transport proven; SQIsign-signed releases and DNS tip anchor operational; 21M supply minted with conservation enforced at the chokepoint.

6.2 What's still pretend (physics)

The headline gap first: $\mathcal{S}_{\text{SIGIL}}$ is a *conjectural effective action*. Its component Lagrangians are unspecified beyond field content, no discretization of the BlockDAG into a lattice field theory has been constructed, and consequently none of the §2 correspondences is currently a derivation. The reputation-induced metric $g_{\mu\nu}$ is a heuristic — the functional form awaits fitting to measured consensus drift. Jones-polynomial machinery is #P-hard; only the bounded-window linking-number/Alexander programme is claimed tractable, cheap invariants are one-way tripwires (they fingerprint, not classify), and the header topology commitment is staged, not shipped. AdS/CFT on a discrete graph is an analogy pending rigorous discretization. The PID/Klein–Gordon mapping is schematic (controller states are not local in φ) and exact only for quadratic potentials. The layered-hardness security design states intent per property; a joint security model with a formal composition bound is open, as is the fold's concrete soundness reduction.

6.3 What's still pretend (engineering)

The RPC edge is authenticated (per-request wallet signature + nonce; the audit's theft findings are fixed and enforcing), so the residual money risk is deeper. The highest-severity open item is **H1**: block-apply does not yet enforce the producer signature — a height-gated verifier is implemented and unit-tested on a branch but dormant, not deployed. Two more sit at the consensus-crypto layer with their fix half-built: the VDF still runs over a forgeable fixed modulus though a trustless replacement exists unwired (**C10**), and the state-layer replay-nonce primitive is committed + signed but not yet called on the live apply path (**H5**). Peer-sync ingress is under-authenticated: a length-only handshake stub (**H2**), a follower that trusts peer-declared difficulty (**H3**), and keyless wire tip-proofs (**H4**). Each carries a verdict (fixed / partial / open) with file:line in the hardening backlog (`docs/SIGIL_HARDENING_BACKLOG.md`) — that document, ground-truthed against the code rather than the stale audit, is what the "open" rung of Figure 6 points at. (The audit's C9 bridge-mint and H6 event-root findings were re-verified as already fixed.) Snapshot-pull live activation awaits a pinned signed anchor on the fleet default plus its M2 crypto (SQIsign + fold in the trailer); until then nodes fall through to the compatible crawl path. The serve-throttle and serial-fetch caps of §3.4 were fixed (throttle removed, request-ahead window live); the remaining `sigil-top` adaptive multi-substream scheduler is coded but dark.

7 What would convince a skeptic

The v0 contract, updated with results — one demonstrated, two partial, two open. Legend: ☑ demonstrated ◧ partially met ☐ open.

☐ **End-to-end committed swap** — a real swap settles with the event committed to the event-log root. DEX math and typed deltas shipped; event-root activation pending.

◧ **Fresh-node proof path** — join via tip-proof, answer verified balance queries, never subscribe to the block topic. Partially met: full fresh-node fold verification is 342 ms; subsequent incremental tip checks take 3 μs with zero blocks downloaded. The complete verified balance-query demonstration remains to be published.

◧ **Public divergence halt** — a tampered node self-halts on root divergence (exit 78, operator alarm). Mechanism live; the staged public tamper-demo remains to be run as theater.

☑ **Signed rolling upgrade** — a gossiped release upgrades running nodes with SQIsign-verified binaries and no block-production pause. Operational (BLAKE3-pinned manifest auto-update in the field since v0.20.x).

☐ **Browser badge verification** — a swap-quality badge verifies in-browser against the committed root within one frame. Depends on the first item.

References

- *SIGIL Whitepaper v0* (2026-05-29) — engineering baseline this edition supersedes; https://quillon.xyz/downloads/SIGIL_WHITEPAPER_v0.pdf
- *The Master Equation of SIGIL — a Conjectural Effective Action* v0.2 (2026-07-18) — standalone companion note (claim provenance: Appendix A); https://quillon.xyz/downloads/SIGIL_MASTER_EQUATION_v0.2.pdf
- *SIGIL × QTFT* (2026-05-30) — `SIGIL_QTFT_TOPOLOGY_v0.md`; the tractable-invariant topology programme
- *A Closed-Form Delivery Law for Gossip Redundancy in the sigil-top Light Node* (2026-06-17) — https://quillon.xyz/downloads/sigil-top-delivery-law.pdf
- *Flux Foundation* whitepaper — the substrate; https://quillon.xyz/downloads/flux-foundation-whitepaper-v0.17.0.pdf
- *SIGIL Genesis v0* — `SIGIL_GENESIS_v0.md`; the architectural spec
- Quillon Graph quantum-physics whitepaper — the sibling tradition this edition answers; https://quillon.xyz/downloads/quantum-physics-whitepaper-full.pdf
- The SQInstructor: a guide to SQIsign and the Deuring Correspondence — arXiv:2603.09899
- High-Throughput GPU Implementation of Dilithium — arXiv:2211.12265
- A Comparative Analysis of zk-SNARKs and zk-STARKs — arXiv:2512.10020

Acknowledgments

This edition integrates the master-equation and QTFT programmes with the measured record of the June–July 2026 development sessions. Foundation work by the SIGIL development swarm, settled on-chain per the agentic-money protocol. Built on Quillon Graph's designs — the consensus, storage, DEX, bank, and mixer lineage this chain inherits, the physics-first whitepaper tradition it answers, and the operating rules (measure live, honest checklists, provenance always) that shaped every section above.

v1.1 — 2026-07-18. The chain is modeled as a 4-manifold with curvature. Reputation is mass. Settlements are matter. Witnesses are gauge fields. Topology is the knot. Security is a three-legged connection. The equation is conjecture; the measurements are not.

A Claim Provenance and Reproduction Guide

The master-equation companion note (*The Master Equation of SIGIL — a Conjectural Effective Action*, v0.2) is published standalone at https://quillon.xyz/downloads/SIGIL_MASTER_EQUATION_v0.2.pdf; it carries the full field-by-field discussion and the v0.2 correction log (PID discriminant, knot formulation, gauge-healing language, per-property composition, the wall as feasibility condition). This appendix instead records where every measured claim in this paper comes from and how to re-run it.

A.1 Environment

All benchmarks and live readings were taken on the Epsilon-class development host: 48-core x86-64, 10 Gbit NIC, NVMe storage, Linux. All builds go through the `fluxc` orchestrator (content-hash cached; raw `cargo` is never used in these workspaces). The live network is `sigil-g0` (single producer + verifying followers; accelerated laboratory economics). This document builds with `pdflatex` (two passes) from `sigil/docs/research/SIGIL_WHITEPAPER_v1.tex`; source `d7906ff`, based on `16f05a8`, branch `feat/commons-tithe-routing-0.36.1` (matching the cover and the PDF metadata).

A.2 Claim-by-claim provenance

Claim	Artifact / commit	How to reproduce
Delivery law $D = 1 - p^r$, RMS 0.4 pp	standalone paper + `flux-chronos` star-flood harness	re-run the 22 seeded simulations (seeds recorded alongside the harness); fit $1 - p^r$; compare residuals
Braid gates S1–S6, divergence = 0	`sigil-dagknight` sim gate, workspace commits `689730f` (gate, 34/34 tests) + `a97ba0e` (late-join)	run the `sigil-chronos` gate binaries; all six scenarios must report divergence = 0
Fold: 2,568 B proof, 280–342 ms full verify	`flux-fold` `fold-verify-bench` bin	`fold-verify-bench 100000` (release); prints proof size + verify ms; note verify is $O(M)$ over the commitments
3 μs incremental tip check, 0 blocks downloaded	live `sigil-top` against `sigil-g0`	launch `sigil-top`; the tip-verify panel reports verify time and blocks downloaded
Skeleton store ∼10M records/s durable	commit `615351c` (`skeleton_store.rs`)	run the store's append benchmark; 72 B/record, two-phase durable
End-to-end snapshot path, 39k blocks/s at 2M records	commit `f52373d` (in-process harness)	set `SIGIL_SNAP_BENCH=N` and run; roots must match
ML-KEM-1024 transport, 115 ms	browser↔bridge noise channel	exercise the wallet connection; timing logged end-to-end
zstd wire: 14.0× backfill (70.7 B/header live), 26.3× gossip	codec=1 backfill + gossip frame lanes	compare wire byte counters against the lz4 baseline on real headers
Mining loop live (shares $0 \to 1250$)	`sigil-rpcd` challenge/submit + `sigil-miner`	mine against the public endpoint; credited balance is the evidence
Live height / supply readings	`sigil-g0` node	any `sigil-top` instance reads them from the verified feed

Workspace commit hashes refer to the SIGIL workspace repository (`sigilgraph`); simulation seeds and gate logs live alongside their harnesses in-tree. Live readings are point-in-time observations of `sigil-g0` and drift with the network; benchmark numbers are reproducible on comparable hardware.

The equation is conjecture; its discipline, and the table above, are not.