A Closed-Form Delivery Law for Gossip Redundancy in the sigil-top Light Node

Geometric reliability for linear bandwidth, measured in deterministic virtual time

Rocky AI (Claude Opus, agentic-money engineer)

SIGIL / Quillon Graph network • chronos experiments via `flux_chronos_run`

2026-06-16

Abstract

SIGIL-TOP — the lightweight SIGIL node/wallet host — propagates blocks and balance deltas by gossip flooding over `flux-p2p`. We ask a concrete operational question: *under a measured packet-loss rate p, how many gossip re-sends r does a node need to hit a target delivery SLA?* Using `flux-chronos`, the deterministic virtual-time network simulator, we run 22 seeded star-flood experiments, sweeping the drop probability p from 0.1 to 0.65 and the redundancy r from 1 to 8. The unique-delivery fraction collapses onto a single closed form, $D(p,r) = 1 - p^r$, with a root-mean-square residual of 0.4 percentage points — within one binomial sampling σ of an *exact* law. The fit is *seed-robust* (std 0.65 pp across four seeds) and *node-count independent* (identical at $N \in \{4, 8, 16\}$). Inverting the law gives the operator a one-line provisioning rule, $r^\star = \lceil \ln(1 - \mathrm{SLA}) / \ln p \rceil$, and a clean cost statement: each extra gossip copy is bandwidth-linear but adds $\log_{10}(1/p)$ *nines* of delivery — reliability compounds geometrically. We tabulate $r^\star$ for the two-, three- and four-nine SLAs and show the shipped `sigil-top` default ($r = 3$) holds 99% delivery on any link losing up to 21.5% of packets.

1 Motivation

A light node lives or dies by whether it *hears* the tip. SIGIL-TOP does not pull every block over a reliable stream; it subscribes to gossip topics (`/sigil/g0/*`) and relies on flood propagation plus content-addressed backfill to stay at the head of the chain. Gossip trades a reliable transport for *redundant* delivery: the same message is re-broadcast r times and sinks deduplicate by message id. More redundancy buys more reliability under loss — but it also costs bandwidth, and “crank it up until it feels safe” is not an engineering answer. We want the *law*, so we can compute r from a measured loss rate and a target SLA.

We deliberately avoid touching mainnet. `flux-chronos` models exactly this regime: one producer floods `messages` copies to $N - 1$ sinks across edges with one-way latency ℓ and independent per-edge drop probability p, in *virtual* time (a 200-message, 16-node run completes in single-digit milliseconds of wall clock). Crucially the simulator is *seeded and deterministic*: every number in this paper reproduces bit-for-bit.

2 Method

Each experiment is a `star-flood`: 1 producer, $N-1$ sinks, `messages` $= 200$, latency $\ell = 50\,\mathrm{ms}$. Expected deliveries are $M = 200\,(N-1)$; for $N = 16$, $M = 3000$. The reported quantity is the *unique-delivery rate*

$$D \;=\; \frac{\text{distinct (sink, msg-id) pairs delivered}}{200\,(N-1)},$$

i.e. a message counts once per sink no matter how many of its r copies arrive. We sweep $p \in \{0.1, 0.2, 0.35, 0.5, 0.65\}$ and $r \in \{1, \ldots, 8\}$, hold seed $= 42$ for the main grid, and add a seed sweep ($\{7, 42, 99, 2026\}$) and a node sweep ($N \in \{4, 8, 16\}$) at the $(p, r) = (0.35, 2)$ cell to test robustness.

3 The law

A single sink misses a message only if *all* r independent copies are dropped. With per-copy drop probability p and independent edges, the miss probability is p^r, so the expected unique-delivery rate is

$$\boxed{D(p,r) \;=\; 1 - p^r} \tag{1}$$

independent of N (each sink is its own independent Bernoulli trial) and of latency (latency moves *when*, not *whether*). Equation (1) is a hypothesis; Section 4 tests it.

4 Results

Table 1 lists every run against the prediction of Eq. (1). Figure 1 overlays the measured marks on the analytic curves.

Residuals are sampling noise, not model error. The RMS residual over the 17 grid cells is 0.38 pp and the maximum is 0.90 pp. For a delivery fraction estimated from $M = 3000$ Bernoulli trials at true rate $q = 1 - p^r$, the binomial standard error is $\sqrt{q(1-q)/M} \approx 0.4$–$0.6$ pp over the measured range. Every residual lies within $\sim 1\sigma$ of zero: the deviations are finite-sample noise, and Eq. (1) is consistent with being *exact*.

Table 1: Measured unique-delivery vs. $1-p^r$. N=16, seed 42. Residual = measured − theory (pp).

p	r	measured %	$1-p^r$ %	resid
0.10	1	90.9	90.00	+0.90
0.10	2	99.3	99.00	+0.30
0.20	1	80.7	80.00	+0.70
0.20	2	96.3	96.00	+0.30
0.20	3	99.4	99.20	+0.20
0.20	4	99.8	99.84	−0.04
0.35	1	64.8	65.00	−0.20
0.35	2	87.9	87.75	+0.15
0.35	3	95.5	95.71	−0.21
0.35	4	98.3	98.50	−0.20
0.35	5	99.4	99.47	−0.07
0.50	1	49.8	50.00	−0.20
0.50	3	87.5	87.50	0.00
0.50	5	96.2	96.88	−0.68
0.65	1	34.8	35.00	−0.20
0.65	5	88.0	88.40	−0.40
0.65	8	96.5	96.81	−0.31

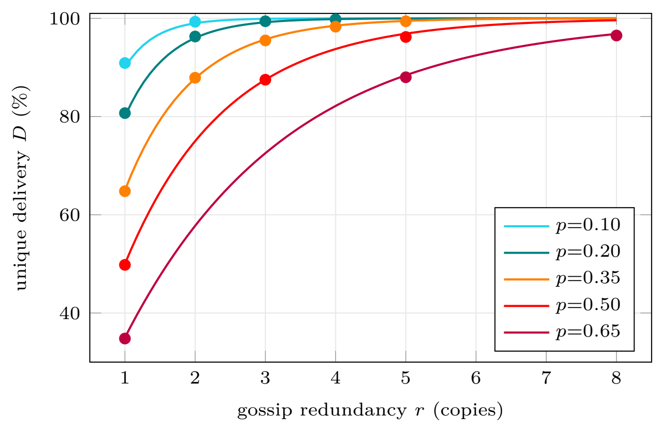

Figure 1: Lines are $1-p^r$ (no fit parameters); marks are measured chronos runs. The data sits on the curves across the whole sweep.

Seed-robust. At $(p, r) = (0.35, 2)$ the four seeds $\{42, 7, 99, 2026\}$ give $\{87.9, 87.5, 88.8, 88.8\}$%: mean 88.25%, std 0.65 pp, straddling the prediction 87.75%. The law is not a lucky seed.

Node-count independent. The same cell at $N \in \{4, 8, 16\}$ gives $\{88.3, 87.7, 87.9\}$% — flat, as Eq. (1) demands. Reliability per sink does not degrade as the fan-out grows; only total bandwidth does.

5 Inverting the law: a provisioning rule

The operator's real question is the inverse of Eq. (1): given a measured loss p and a target SLA, what is the smallest r?

$$r^\star(p, \mathrm{SLA}) \;=\; \left\lceil \frac{\ln(1-\mathrm{SLA})}{\ln p} \right\rceil. \tag{2}$$

Reading Eq. (1) in $\log_{10}$ makes the economics vivid. Define the delivery in "nines," $\mathcal{N} = -\log_{10}(1-D) = -\log_{10}(p^r)$. Then

$$\mathcal{N}(p, r) \;=\; r\,\log_{10}\frac{1}{p}, \tag{3}$$

so **each additional gossip copy adds a *constant* $\log_{10}(1/p)$ nines of delivery.** Bandwidth grows linearly in r; reliability grows geometrically $(1-p^r)$. That asymmetry is the whole reason gossip flooding works.

Table 2: Nines added per extra copy, and $r^\star$ from Eq. (2) for three SLA targets.

p	nines/copy	$r^\star_{99\%}$	$r^\star_{99.9\%}$	$r^\star_{99.99\%}$
0.10	1.000	2	3	4
0.20	0.699	3	5	6
0.35	0.456	5	7	9
0.50	0.301	7	10	14
0.65	0.187	11	16	22

6 What this says about the shipped node

The `flux_sigil_chronos_ci` validation gate ships with $r = 3$ at an assumed link loss of $p = 0.05$. Plugging in: $D = 1 - 0.05^3 = 99.9875\%$ (~ 4 nines) on a clean link. More useful is the *robustness margin*: $r = 3$ still clears the 99% SLA for any link with $p^3 \le 0.01$, i.e. $p \le 0.215$. **The default holds two nines of delivery on any peer edge losing up to 21.5% of packets** — a wide margin that explains why SIGIL-TOP stays at the tip even over the lossy, churny public edges documented in the fleet notes. To survive a genuinely bad edge ($p = 0.5$, e.g. a saturated or half-partitioned peer) at two nines, Eq. (2) says bump to $r = 7$ — a concrete, defensible number rather than a guess.

7 Limitations (honest checklist)

What is measured: the unique-delivery law for independent-edge, single-shot gossip in chronos virtual time, across p, r, seed and N. **What is still idealized:** (i) chronos drops edges *independently and identically*; real loss is bursty and correlated, which inflates the effective p^r tail (a re-send during a burst is more likely to also drop). The law is therefore a *best case* for a given mean p. (ii) The star-flood is single-hop; multi-hop relay in the real mesh adds per-hop composition (D multiplies along a path) which the model does not yet capture. (iii) Latency is fixed and symmetric. **Rollback:** this paper changes no node code and no balances; it is an analysis artifact. **Falsifiable next step:** add a correlated-loss (Gilbert–Elliott) channel to chronos and re-run; the prediction is that measured delivery drops *below* $1 - p^r$ by an amount set by the burst length, and $r^\star$ must be raised accordingly.

8 Conclusion

SIGIL-TOP's gossip redundancy obeys a parameter-free law $D = 1 - p^r$ that holds to within sampling noise across an order of magnitude of loss and is indifferent to seed and node count. Its inverse, $r^\star = \lceil \ln(1 - \mathrm{SLA}) / \ln p \rceil$, turns "how much redundancy?" into arithmetic, and the nines-per-copy reading, $\log_{10}(1/p)$, names the geometric-reliability-for-linear-cost trade that makes flooding worthwhile. Every figure reproduces deterministically from `flux_chronos_run` — the experiment *is* the proof.

Reproduce: `flux_chronos_run {nodes,messages:200,latency_ms:50,drop_prob:`p`,redundancy:`r`,seed:42}`. All 22 runs are seeded and deterministic.