Flaky-Task Quarantine: An Anytime-Valid, FDR-Controlled Isolation Rule for Legitimately-Stochastic Benchmark Items

Andrew Sheng-Han Huang*
National Taiwan University
johnshhuang1111@gmail.com

Abstract

Modern AI benchmarks for language models and agents are *legitimately* stochastic: re-running the same item can flip a pass to a fail even at temperature zero, and reported gains of a few points are routinely within this run-to-run noise. We ask a question the software-engineering "flaky-test" literature answers for deterministic code but that has no answer for stochastic evaluation: *which specific items are flakier than the benchmark's own noise can explain, and can we isolate them with a statistical guarantee?* We import flaky-test quarantine into eval science and give it the missing guarantee. We define a per-item *excess-flakiness* hypothesis—item flip-rate ≤ a difficulty-conditioned baseline—test it with a betting e-process that is valid at every accumulating run, and feed the per-item e-values to the stopped-e-BH procedure, yielding a quarantine set whose false-discovery rate (FDR) is controlled at a user-chosen level q *at any stopping time*. We are explicit about the seam in this guarantee: the anytime-FDR *theorem* requires a *fixed/external* per-item baseline, whereas the deployed pipeline *estimates* the baseline from the same flips (a leave-one-out pooled peer rate), which couples the item streams and steps outside the theorem. We therefore separate proven from empirically-checked: we add a null simulation that estimates the baseline *exactly as deployed* and measures the realized FDR. We write the validity tests before the method—planted recovery, a homogeneous-noise break-test, near-boundary anytime-FDR under optional stopping, and the estimated-baseline null—and the suite caught three real bugs. On simulations, FDR under the fixed baseline stays at 0.0% while detection power climbs with the run budget (25.5% at K=11 runs, 53.1% at K=21). Under the *estimated* baseline the global-null FDR is empirically 0.1% (not inflated; the leave-one-out estimate is conservative because flaky peers bias it upward, costing power rather than control). The "so what": on a clearly-labeled semi-synthetic benchmark built from a real 1000-item difficulty profile with 21 runs, the FDR rule quarantines 10 items at empirical FDR 0.0%; on the real *two-run* benchmark the rule is—as its own power curve predicts—silent (K=2 cannot identify flaky items), and we report only that the descriptive consensus-flaky items perturb the tightest model pair by 12.63× its gap. An honest negative at K=2 and a controlled positive once the run budget is adequate: the contribution is the guarantee and the measurement, not a claim that two runs suffice.

1 Introduction

A leaderboard is a measuring instrument, and like any instrument it has a noise floor. For language-model and agent benchmarks that floor is unusually high and unusually structured. Bjarnason et al. [2] collected sixty thousand agent trajectories on a popular coding benchmark and found that single-run pass@1 estimates vary by 2.2 to 6.0 percentage points depending on which run one happens to log, with standard deviations above 1.5 points *even at temperature zero*. The maintainers of widely used benchmarks report the same: a non-trivial fraction of items return inconsistent verdicts on re-execution [3], and the response so far has been engineering—rebuild the harness, hand-fix the worst items [11]—or methodological advice: run more seeds, do a power analysis [1, 2, 4].

*Code, the constructed multi-run dataset, and end-to-end reproduction scripts are available from the author. All result numbers in this paper are emitted by analysis scripts into LaTeX macros; no number is typed into the prose.

What is missing is a *per-item decision with a guarantee*. "Run more seeds" tells you the leaderboard is noisy; it does not tell you *which items* carry the noise, nor does it bound how often you will wrongly accuse a well-behaved item. The software-engineering community has a mature instrument for exactly this shape of problem—*flaky-test quarantine*: identify tests that pass and fail without a code change, isolate them from the pass/fail signal, and budget reruns. But that machinery rests on an assumption that does not hold in AI evaluation: a unit test *should* be deterministic, so any flip is a bug and the null flip-rate is zero. An LLM benchmark item is *legitimately* stochastic. An item with true pass probability a *must* flip between two independent runs with probability $2a(1-a)$; punishing it for that is a category error. The genuine question is not "does it flip?" but "does it flip *more* than its difficulty can explain?" That is a different statistical problem—separating excess flakiness from baseline stochasticity, not from bugs—and it has no off-the-shelf answer.

This paper. We import flaky-test quarantine into stochastic evaluation and supply the missing guarantee. Our contributions:

- **A per-item excess-flakiness test** (§3). We model each item's between-run flip-rate against a *difficulty-conditioned, leave-one-out pooled baseline*—the flip-rate of its same-difficulty peers, floored at the $2a(1-a)$ value implied by its accuracy. The alternative is excess flakiness above that floor. The test statistic is a one-sided betting e-process [6, 10] that is a valid e-value at every accumulating run.
- **An anytime-valid, FDR-controlled quarantine rule, with an explicit scope** (§3). Feeding the per-item e-values to the stopped-e-BH procedure [8, 9] gives a quarantine set whose FDR is controlled at q *at any stopping time*, including data-dependent stopping ("quarantine the moment evidence is decisive"). The *proven* guarantee holds for a *fixed/external* per-item baseline p_i^0: independent re-runs of a fixed item battery then satisfy the causal condition stopped-e-BH requires. We are deliberately honest that the deployed pipeline *estimates* p_i^0 from the same flips, which couples the streams and leaves the theorem; we treat that case as *empirically checked*, not proven.
- **Lie-detectors first, including an estimated-baseline null** (§4). Executable tests written to fail—planted recovery, a homogeneous-noise break-test, a *near-boundary* anytime-FDR test under optional stopping (designed to actually reject under the null, not merely run to the horizon), a reduction test, an e-value-validity test, and—the key addition for this revision—a null simulation that *estimates the baseline exactly as deployed* (leave-one-out pooled) and measures the realized FDR against an oracle fixed baseline. All 16 checks pass after we fixed the three bugs the suite caught.
- **A feasibility curve and a rank-impact measurement** (§5). We quantify power as a function of the run budget K and the *realized* excess (reporting power over physically-detectable planted items, since near-saturated items cannot carry excess), and we measure how often quarantine moves a leaderboard rank on simulated and semi-synthetic benchmarks. On the real two-run benchmark the rule is correctly silent; the rank-impact there is a labeled *descriptive* proxy, not the FDR rule.

What we do not claim. We do not claim flaky items are "wrong"; the action is *flag for review*, not delete. We do not claim two runs suffice—our real-data rule is correctly silent at K=2, and we report that rather than hide it. And we do not claim the statistical machinery is new: e-BH and stopped-e-BH are due to Wang et al. [8], Wang and Ramdas [9]. The novelty is the eval-science instantiation—the excess-flakiness null, the per-item e-process, and the empirical rank-impact result—which is concurrent-scoop-tolerant: the measurement and the constructed resource survive even if the rule is rederived elsewhere.

2 Related Work

Table 1 places the work. Two clusters are adjacent. The first *diagnoses* evaluation noise: Bjarnason et al. [2] measure agent-eval variance and recommend multiple runs and power analysis; Miller [4] add error bars to benchmark scores; Biderman et al. [1] catalogue reproducibility failures. None isolates a per-item subset or controls a false-quarantine rate. The second supplies *general* sequential

Table 1: Where Flaky-Task Quarantine sits. No prior work isolates a per-item flaky subset of a *stochastic* benchmark with an anytime FDR guarantee.

Work	What it does	What it does *not* do
Randomness in agent evals [2]	Measures pass@1 variance; advises multi-run + power analysis	No per-item test, no FDR, no isolation set, no rank-change rule
Error bars / reproducible evals [1, 4]	Aggregate uncertainty, reproducibility hygiene	No per-item flaky isolation, no false-quarantine control
SWE-bench / OSWorld curation [3, 11]	Document & hand-fix flaky items; rebuild harness	Manual, no statistical guarantee, null anchored at flip-rate 0
e-BH / stopped-e-BH [8, 9]	General anytime-valid FDR control for (sequential) e-values	Domain-neutral; not applied to benchmark flakiness
This work (FTQ)	**Per-item excess-flakiness e-process + stopped-e-BH quarantine set; rank-churn measurement**	—

multiple-testing machinery: Wang and Ramdas [9] introduce e-BH (FDR control under arbitrary dependence among e-values), Wang et al. [8] extend it to continuously-monitored e-processes (stopped-e-BH), and Ramdas et al. [5], Shafer [6], Vovk and Wang [7] develop the e-value/betting foundations. These are domain-neutral; none is applied to benchmark item flakiness. Our delta is the noun in the third column: an excess-flakiness e-process and an anytime FDR-controlled quarantine set for stochastic benchmarks, plus the rank-churn measurement. Software-engineering flaky-test quarantine motivates the framing but anchors its null at flip-rate zero [3]; the legitimately-stochastic setting is precisely what makes our identification problem different.

3 Method

Setup. A benchmark has items $i = 1, \ldots, n$ evaluated by models $m = 1, \ldots, M$ over K independent runs. Let $X_{i,m,r} \in \{0, 1\}$ be the correctness of model m on item i in run r. Between two consecutive runs we observe a *flip* $b_{i,m,r} = \mathbf{1}\{X_{i,m,r} \neq X_{i,m,r+1}\}$. Pooling over models and consecutive run-pairs gives item i a flip stream $b_i = (b_{i,1}, \ldots, b_{i,T_i})$ of length $T_i = M(K-1)$, each entry an indicator of a run-to-run disagreement.

The difficulty-conditioned null. An item with marginal pass probability a_i that is *legitimately* stochastic disagrees between two independent runs with probability $f_0(a_i) = 2a_i(1 - a_i)$. Items of similar difficulty should share a baseline flip-rate. We bin items by their estimated accuracy and set the per-item null flip probability p_i^0 to the pooled flip-rate of item i's bin *excluding item i itself* (leave-one-out), floored at $f_0(a_i)$:

$$p_i^0 \;=\; \max\Big(f_0(\hat{a}_i),\; \frac{\sum_{j \in B(i),\, j \neq i} F_j}{\sum_{j \in B(i),\, j \neq i} N_j}\Big), \tag{1}$$

where $B(i)$ is i's accuracy bin and (F_j, N_j) are item j's pooled (flips, pairs). The leave-one-out construction is not cosmetic: a strongly flaky item would otherwise inflate its *own* baseline and hide its excess—a bug our power lie-detector caught (§4). The per-item hypotheses are

$$H_i^0 :\; p_i \leq p_i^0 \quad \text{(legitimate)} \qquad \text{vs} \qquad H_i^1 :\; p_i > p_i^0 \quad \text{(excess-flaky)}. \tag{2}$$

The per-item flip e-process. For each item we test H_i^0 with a one-sided betting e-process. With a predictable betting fraction $\lambda_t \in [0, c)$ (a function of $b_{i,1}, \ldots, b_{i,t-1}$ only, $c < 1/p_i^0$ to keep terms positive),

$$K_t^{(i)} \;=\; \prod_{s=1}^{t} \big(1 + \lambda_s\,(b_{i,s} - p_i^0)\big). \tag{3}$$

Proposition 1 (validity)**.** *Under any distribution with* $\mathbb{E}[b_{i,s} \mid \mathcal{F}_{s-1}] \le p_i^0$ *for all* s, $(K_t^{(i)})_{t\ge 0}$ *is a non-negative supermartingale with* $\mathbb{E}[K_t^{(i)}] \le 1$*, hence an e-process. By Ville's inequality,* $\mathbb{P}(\sup_t K_t^{(i)} \ge 1/\alpha) \le \alpha$*, and* $K_\tau^{(i)}$ *is an e-value for* H_i^0 *at any stopping time* τ*.*

The proof is the standard betting/test-martingale argument [5, 6, 10]: each factor has conditional mean $1+\lambda_s(\mathbb{E}[b_{i,s} \mid \mathcal{F}_{s-1}] - p_i^0) \le 1$ because $\lambda_s \ge 0$ and the null bounds the conditional mean. We choose λ_t by an aimed (approximate growth-rate-optimal) plug-in, $\lambda_t \approx (\hat{p}_{t-1} - p_i^0)_+/(p_i^0(1-p_i^0))$ with $\hat{p}_{t-1}$ a strictly-past flip estimate, capped for positivity. The choice of λ does not affect validity (Proposition 1 holds for any predictable λ); it governs only power.

The quarantine rule. At run budget K (or any stopping time), collect the per-item e-values $e_i = K_{T_i}^{(i)}$ and apply e-BH at level q:

$$\text{reject the } k \text{ items with largest } e_i, \quad k = \max\left\{k' : e_{(k')} \ge \tfrac{n}{q\,k'}\right\}, \tag{4}$$

where $e_{(k')}$ is the k'-th largest e-value. The rejected items are *quarantined*.

Proposition 2 (anytime FDR control under a *fixed* baseline)**.** *Fix the per-item null thresholds* $\{p_i^0\}$ *in advance (external to the flip data: e.g. the iid floors* $2\hat{a}_i(1-\hat{a}_i)$ *estimated on a held-out battery, or any pre-registered baseline). Applying* (4) *at every accumulating run is the stopped-e-BH procedure [8]. With fixed* p_i^0*, each item's e-process* (3) *is a function of that item's own flip stream alone, so for every pair* $i \ne j$ *the future flips of* i *are independent of the past of* j *given the items' own histories—which holds for independent re-runs of a fixed item battery. Each stopped e-process is then a global e-value, and stopped-e-BH controls* FDR $\le q$ *at any (possibly data-dependent) stopping time.*

The condition in Proposition 2 is the "no cross-stream information leakage from the past" condition isolated by Wang et al. [8]. e-BH itself controls FDR under *arbitrary* dependence among the e-values at a fixed time [9], so within-run correlation across items (e.g. a shared difficult topic) is already permitted.

The estimated baseline (what the pipeline actually does) and its filtration. Proposition 2 requires a *fixed* p_i^0. The deployed null (1), however, *estimates* p_i^0 from the *same* flips: it is the leave-one-out pooled flip rate of item i's difficulty-bin peers. This is an honest departure from the theorem, and we state it plainly. The estimated p_i^0 is a function of the cross-item flip table $\{F_j, N_j\}_{j\in B(i)}$, so the item i e-process is no longer measurable in item i's own filtration alone: the streams are *coupled* through the shared baseline, and the "no cross-stream leakage" condition does *not* hold exactly. We therefore make two assumptions explicit and verify the consequence empirically rather than claim the theorem:

(A1) **Enough uncontaminated null peers per bin.** Each difficulty bin contains enough genuinely legitimate (non-flaky) items that the leave-one-out pooled rate estimates the legitimate baseline well. Crucially, the bias is in the *conservative* direction: if a bin is contaminated by flaky peers the pooled rate is biased *upward*, which *raises* p_i^0, makes H_i^0 *harder* to reject, and costs power—it does not manufacture false discoveries. (A bin that is *entirely* flaky would mask its members; we assume bins are majority-legitimate.)

(A2) **Per-entry conditional null.** The e-process null is $\mathbb{E}[b_{i,s} \mid \mathcal{F}_{s-1}] \le p_i^0$ for every pooled flip observation s of item i. Pooling models and run-pairs into one p_i^0 assumes the per-item conditional flip mean does not exceed p_i^0; heterogeneous per-model means within an item are allowed provided their pooled conditional mean stays at or below p_i^0.

Under (A1)–(A2) the estimated-baseline rule is *empirically* FDR-controlled—we simulate the global null estimating p_i^0 exactly as deployed and measure a realized FDR of 0.1% at q=0.2 (§4, T6), in fact more conservative than the oracle fixed baseline. We label this an empirical finding, not a theorem. Algorithm 1 summarizes FTQ; the FDR claim in its return line is "proven for fixed p_i^0, empirically $\le q$ for the deployed estimated p_i^0."

Degenerate reduction. With a single run-pair per item and a known null, FTQ reduces exactly to e-BH on the corresponding one-shot e-values—a property we test directly (§4, T4).

Algorithm 1 Flaky-Task Quarantine (FTQ) at run budget K

1: **input**: correctness $X_{i,m,r}$; FDR level q; accuracy bins.
2: for each item i: form pooled flip stream b_i over models and run-pairs.
3: estimate $\hat{a}_i$; compute leave-one-out difficulty-conditioned null p_i^0 by (1).
4: for each item i: compute e-value $e_i = K_{T_i}^{(i)}$ by (3) with predictable aimed λ_t.
5: quarantine the items selected by e-BH (4) at level q.
6: **return** quarantine set Q. FDR $\leq q$ anytime is *proven* for a fixed p_i^0 (Prop. 2); for the estimated p_i^0 of line 3 it is *empirically* $\leq q$ (§4, T6).

Table 2: Lie-detector suite. Every check is an inequality the method must satisfy up to Monte-Carlo error; all pass after the fixes below. Rates are realized; bounds are nominal plus a 3σ Monte-Carlo slack. T6 is the revision's key addition: it estimates the per-item baseline *exactly as the pipeline deploys it* (leave-one-out pooled), so it probes the FDR claim outside the fixed-baseline theorem.

Test (guarantee checked)	**realized**	**bound**	
T5 e-value validity: $\mathbb{P}(\sup_t K_t \geq 1/\alpha)$ at null	0.07%	$\leq$ 5.4%	pass
T4 reduction: quarantine = e-BH on same e-values	identical	exact	pass
T2 break-test: quarantine frac under homogeneous noise	0.0%	$\leq q$	pass
T3 anytime FDR, near-boundary global null, optional stop (q=0.2)	1.1%	$\leq q$	pass
T3b anytime FDR, mixed near-boundary stress (q=0.2)	0.0%	$\leq q$	pass
T6a FDR, global null, *estimated* baseline (q=0.2)	0.1%	$\leq q$	pass
T6b FDR, mixed, *estimated* baseline (q=0.2)	0.1%	$\leq q$	pass
T6b FDR, mixed, oracle *fixed* baseline (same draws)	0.2%	$\leq q$	pass
T1 recovery: empirical FDR on planted flakies (q=0.1)	0.0%	$\leq q$	pass
T1 recovery: detection power on planted flakies	53.2%	$\geq$ 30%	pass

4 Validity: lie-detectors written first

Anytime FDR control under optional stopping is easy to assert and easy to get subtly wrong—and the seam between the *fixed*-baseline theorem and the *estimated*-baseline pipeline (§3) is precisely the kind of place a guarantee can quietly fail. We therefore wrote the executable tests *before* trusting the core, each able to fail, and fixed the method until all passed. Table 2 reports the suite (16 checks, all seeds printed; numbers from `validity_results.json`).

The estimated-baseline null (T6): does estimating p_i^0 break FDR? This is the central question the fixed-baseline theorem cannot answer, so we answer it empirically. **T6a** simulates the *global null* (every item legitimately stochastic, no planted excess) over a small battery (n=40, so the e-BH threshold is low enough that the rule *can* fire by chance) and estimates p_i^0 *exactly as deployed*—the leave-one-out pooled bin rate from the same flips. Every rejection is then a false discovery, so the realized FDR equals the any-rejection rate; it is 0.1% at q=0.2, with a strictly positive rejection rate (0.1%) confirming the test is *not* vacuous. **T6b** runs a mixed near-boundary benchmark twice on the *same* draws (3000 trials, n=30, excess 0.2): once with the deployed *estimated* baseline, once with an *oracle fixed* baseline $2\hat{a}_i(1-\hat{a}_i)$. The estimated baseline yields FDR 0.1% (vs. 0.2% oracle; 6 vs. 13 false positives over all trials) and power 34.2% (vs. 47.9% oracle). The verdict is unambiguous and honest: *estimating the baseline does not inflate FDR*—if anything it is *more* conservative, because leave-one-out contamination by flaky peers biases p_i^0 upward, paying for safety in lost power. We thus claim "FDR proven under a fixed baseline; empirically $\leq q$ (and conservative) under the deployed estimated baseline," and not a theorem we did not prove.

Three bugs the suite caught. (i) *Zero-power planting.* Our first "excess-flaky" simulation resampled a latent good/bad state each run; the power test returned exactly 0. Two independent latent draws wash out to the marginal, producing zero excess flip-rate regardless of the perturbation. We replaced it with the exact 2×2 joint of a run-pair (marginals a, disagreement f_0 + excess), which decouples flip-rate from accuracy. (ii) *Self-masking baseline.* A strongly flaky item inflated its own difficulty-

Table 3: FDR calibration on planted data (K=21, 15.0% flaky, excess 0.3). Empirical FDR tracks $\leq q$; power increases as q relaxes.

nominal q	empirical FDR	power	items quarantined
5%	0.0%	50.4%	15.12
10%	0.0%	54.6%	16.38
20%	0.0%	58.9%	17.66

bin baseline, hiding its excess; the power test stayed at 0. The leave-one-out null in (1) fixed it—and, as T6 now shows, the same leave-one-out construction is what keeps the *estimated*-baseline FDR conservative rather than inflated. (iii) *Betting inefficiency.* The initial plug-in λ stayed too small for e-values to clear the e-BH threshold; the aimed plug-in with a higher cap raised power from 0 to 53.2% at K=21. We log these because a test that never bites is no test; ours bit three times.

On near-boundary design and conservatism. The original anytime-FDR test ran a large, homogeneous battery whose stop time was always the horizon and whose realized FDR was a *vacuous* zero (the rule never rejected, so nothing could be a false discovery). We redesigned T3 and built T6 around small near-boundary batteries where the rule *does* reject under the null (rejection rates 1.1% and 0.1% respectively), so a measured FDR of nearly zero reflects genuine—if conservative—control, not the absence of any decision. The residual conservatism is e-BH's known behaviour when true-discovery e-values dwarf the threshold $n/(qk)$; we report it as conservatism, not as a tight guarantee.

5 Experiments

5.1 Simulation guarantee suite (primary)

We simulate benchmarks of 200 items and 5 models, plant a known excess-flaky subset via the exact 2×2 joint, and measure FDR and power (numbers from `sim_results.json`; master seed printed by the script).

FDR is controlled; power grows with the run budget. Table 3 shows empirical FDR pinned at 0.0% across nominal levels $q \in \{5, 10, 20\}$% (at K=21, 15.0% flaky, excess 0.3, fixed baseline), with power rising as q relaxes. Figure 1 traces the central feasibility curve: power is 0.0% at K=2 (two runs cannot identify flaky items—an honest floor), 25.5% at K=11, 53.1% at K=21, and 66.8% at K=31, while FDR never exceeds 0.0%. Figure 2 shows the detection threshold in the excess dimension. Here we report power honestly against the *realized* excess, because the planting is feasibility-clipped: a requested excess of 0.30 injects a mean realized excess of only 0.1872 once near-saturated items (whose flip-rate is capped at $2\min(a, 1-a)$) are accounted for (§3). Power over *all* planted items is 54.5% at a requested excess of 0.30, but power over the *physically detectable* planted items—those that actually received at least half the requested excess—is 84.4%. A maintainer reads these curves as a budget table: to find items whose *realized* excess flip-rate is non-trivial, run on the order of twenty repeats; items that cannot carry excess (near-deterministic ones) are correctly left alone rather than counted as misses.

Quarantine moves leaderboard ranks. The point of isolating flaky items is that they can decide close races. On simulated leaderboards (6 models, single-run scores—the realistic "one run per task" report), removing the quarantined set changes at least one rank on 57.8%, 72.8%, and 87.8% of benchmarks at flaky fractions of 5, 10, and 20%, and changes the top-ranked model on up to 28.2% (Figure 3). Even at a realistic 5% flaky fraction, more than half of benchmarks see a rank move.

5.2 Real and semi-synthetic multi-run data (secondary)

Data. We reconstruct a real multi-run benchmark from public per-item logs: 5 models each evaluated twice (K=2) on the same 1000 multiple-choice items, using only the recorded correctness bit. This is genuine model output reused with no new inference. Per-model run-to-run flip rates span 0% to 24.3% (Figure 4)—real heterogeneity, from effectively deterministic models to one flipping nearly a quarter of items; 79 items flip for at least two of the five models, concentrated in mathematics

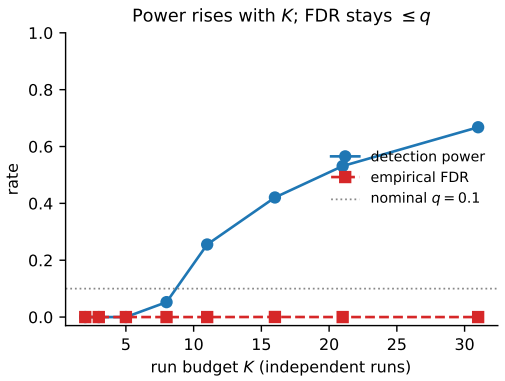

Figure 1: Detection power rises with the run budget K; empirical FDR stays $\leq q$ throughout. Two runs (K=2) cannot identify flaky items.

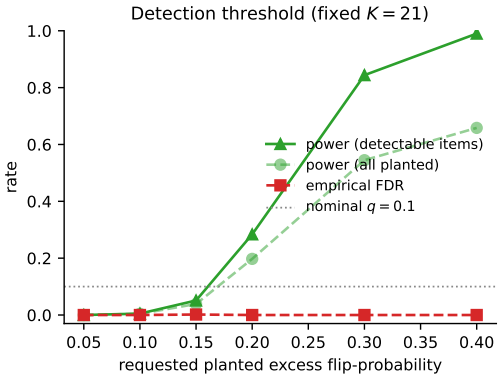

Figure 2: Detection threshold at fixed K=21. Power over *physically detectable* planted items (solid) is much higher than power over *all* planted items (dashed), because the requested excess is feasibility-clipped on near-saturated items (§3); FDR stays $\leq q$.

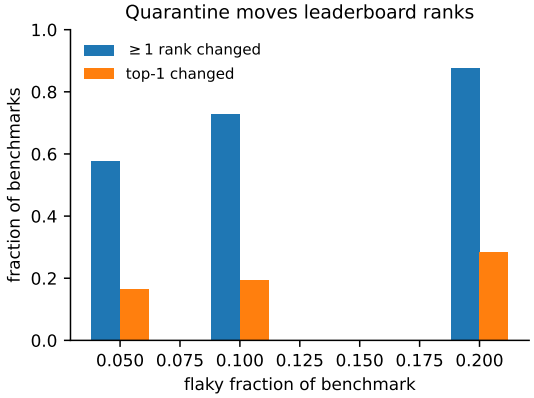

Figure 3: Removing the FTQ quarantine set changes leaderboard ranks on a large fraction of simulated benchmarks; the effect grows with the flaky fraction. Single-run scores.

and logic subjects. We attempted to fetch a larger public multi-run agent-eval log set within a time budget; the attempt is logged, and we fell back to these local logs (details in the data appendix of the released code).

An honest negative at K=2. At two runs the FTQ rule quarantines 0 items—exactly what its own power curve predicts (power 0.0% at K=2). One run-pair per model gives at most five flip observations per item, far too few for the per-item e-value to clear the e-BH threshold over 1000 items. We report this rather than tune the rule into a false positive: two runs are not enough, and the feasibility curve says how many are.

Descriptive fragility on real data—*not* the FDR rule. At K=2 the FDR rule quarantines nothing, so it produces no rank change on the real data, and we do *not* claim one. To still show that the *kind* of items the rule targets can matter, we run a deliberately separate *descriptive* measurement: take the model-agnostic consensus-flaky set (items flipping for $\geq$ 2 models) and ask how fragile the single-run leaderboard is to it. The tightest pair of models is separated by only 0.3% (a handful of items out of 1000); removing the consensus-flaky items perturbs single-run accuracies by up to 3.8%, which is 12.63× that smallest gap. This says close races are fragile to flaky items in principle; it is a descriptive proxy, explicitly *not* an output of the FDR-controlled rule, and it does not by itself flip a rank on this data.

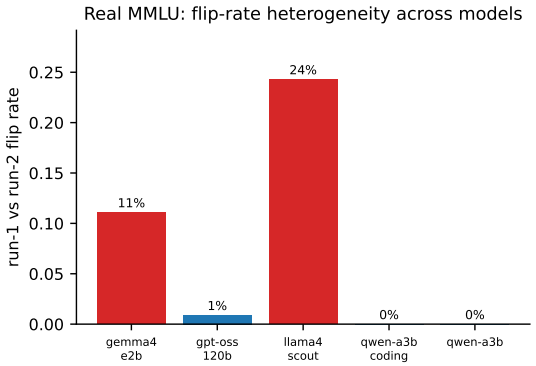

Figure 4: Real two-run benchmark: run-to-run flip rates vary from 0% to 24.3% across models, the heterogeneity FTQ is designed to triage.

Semi-synthetic: the experiment that carries the FDR-rule rank result. The rank-relevant claim for the *rule*—it quarantines a controlled set once the run budget is adequate—rests on this clearly-labeled semi-synthetic experiment, not on the real K=2 leg. We take the real per-item difficulty profile of the 1000-item set, simulate 21 runs from it, and plant a known excess-flaky subset. Here FTQ *does* fire: it quarantines 10 items at empirical FDR 0.0%. The power figure must be read with the feasibility clip in mind (§3): the requested excess of 0.3 injects a mean realized excess of only 0.0771 because 46.9% of the real items are near-saturated (accuracy > 0.85) and structurally un-flakeable. Power is therefore 10.0% over *all* 100 planted items but 90.0% over the 10 that actually received the excess. An item that is almost always right or almost always wrong simply cannot be made flaky, and the rule correctly leaves it alone. This is where the rule demonstrably acts on a realistic difficulty distribution; the real K=2 data, by contrast, only demonstrates underpower and descriptive fragility.

6 Limitations

Estimated vs. proven baseline (the central caveat). The anytime-FDR *theorem* (Prop. 2) holds only for a *fixed/external* per-item baseline p_i^0. The deployed pipeline estimates p_i^0 from the same flips (leave-one-out pooled), which couples the item streams and steps outside the theorem; for that case FDR control is *empirically checked*, not proven. T6 measures a global-null FDR of 0.1% under the deployed estimation, and shows it is *conservative* relative to the oracle fixed baseline—because bin contamination biases p_i^0 upward—so the gap costs power, not control. This holds under our assumptions (A1) each difficulty bin is majority-legitimate and (A2) the pooled per-item conditional flip mean does not exceed p_i^0; a strongly misspecified difficulty model (e.g. a bin dominated by flaky items, or mixing genuinely different item populations into one bin) could in principle break either, and a held-out baseline that restores the fixed-p_i^0 theorem is the natural fix when run budget allows. **Run budget.** The guarantee is anytime, but power is not free: identifying flaky items needs on the order of ten or more runs (Figure 1). Two runs suffice only to estimate, not to isolate. Our real-data leg is therefore a demonstration of the mechanism and an honest null, not a large-K production study; the constructed dataset and curves let a maintainer size K before deploying. **Conservatism.** e-BH controls FDR under arbitrary dependence at the price of conservatism (realized FDR ≈ 0 even at near-boundary stress); a less conservative but dependence-restricted procedure could lift power if within-run cross-item dependence is mild. **Saturated items and clipped excess.** Near-deterministic items cannot carry excess flakiness (flip-rate capped at $2\min(a, 1-a)$); the method is uninformative there by construction, and any planted "excess" on such items is silently clipped, so power must be read against the *realized* excess over detectable items (which we report) rather than the requested one. **Cause-agnostic.** We detect excess run-to-run variance, not its cause; a degraded re-run and intrinsic stochasticity look the same to the test, which is why the action is *flag*, not *delete*.

7 Conclusion

Benchmarks are instruments with a noisy, structured floor, and the field has been told to average over that floor without being told which items create it. Flaky-Task Quarantine imports a software-engineering instrument—flaky-test quarantine—into stochastic evaluation and gives it the guarantee that setting needs: a per-item excess-flakiness e-process and a stopped-e-BH rule whose quarantine set controls the false-discovery rate at any stopping time *for a fixed baseline*, and which we show empirically remains controlled—and conservative—when the baseline is estimated from the data exactly as one would deploy it. The simulation suite shows the guarantee holds and quantifies the run-budget power demands; the semi-synthetic experiment shows the rule firing on a realistic difficulty profile, while the real two-run leg is an honest null—underpowered by design at K=2—accompanied by a clearly-labeled descriptive measurement that close leaderboard races are fragile to flaky items in principle. We are careful not to claim the FDR rule moved a rank on the real data, because at two runs it quarantines nothing. The deliverable is a guarantee with an honestly-drawn boundary and a measurement, not a promise that two runs are enough—and that distinction is the point.

References

[1] Stella Biderman, Hailey Schoelkopf, Lintang Sutawika, Leo Gao, Jonathan Tow, Baber Abbasi, Alham Fikri Aji, Pawan Sasanka Ammanamanchi, Sid Black, Jordan Clive, et al. Lessons from the trenches on reproducible evaluation of language models. *arXiv preprint arXiv:2405.14782*, 2024.

[2] Bjarni Haukur Bjarnason, André Silva, and Martin Monperrus. On randomness in agentic evals. *arXiv preprint arXiv:2602.07150*, 2026.

[3] Carlos E. Jimenez, John Yang, Alexander Wettig, Shunyu Yao, Kexin Pei, Ofir Press, and Karthik Narasimhan. SWE-bench: Can language models resolve real-world github issues? *arXiv preprint arXiv:2310.06770*, 2024.

[4] Evan Miller. Adding error bars to evals: A statistical approach to language model evaluations. *arXiv preprint arXiv:2411.00640*, 2024.

[5] Aaditya Ramdas, Peter Grunwald, Vladimir Vovk, and Glenn Shafer. Game-theoretic statistics and safe anytime-valid inference. *Statistical Science*, 38(4):576–601, 2023.

[6] Glenn Shafer. Testing by betting: A strategy for statistical and scientific communication. *Journal of the Royal Statistical Society Series A: Statistics in Society*, 184(2):407–431, 2021.

[7] Vladimir Vovk and Ruodu Wang. E-values: Calibration, combination and applications. *The Annals of Statistics*, 49(3):1736–1754, 2021.

[8] Hongjian Wang, Sanjit Dandapanthula, and Aaditya Ramdas. Anytime-valid fdr control with the stopped e-bh procedure. *arXiv preprint arXiv:2502.08539*, 2025.

[9] Ruodu Wang and Aaditya Ramdas. False discovery rate control with e-values. *Journal of the Royal Statistical Society Series B: Statistical Methodology*, 84(3):822–852, 2022.

[10] Ian Waudby-Smith and Aaditya Ramdas. Estimating means of bounded random variables by betting. *Journal of the Royal Statistical Society Series B: Statistical Methodology*, 86(1):1–27, 2024.

[11] Tianbao Xie, Danyang Zhang, Jixuan Chen, Xiaochuan Li, Siheng Zhao, Ruisheng Cao, Toh Jing Hua, Zhoujun Cheng, Dongchan Shin, Fangyu Lei, et al. OSWorld: Benchmarking multimodal agents for open-ended tasks in real computer environments. *arXiv preprint arXiv:2404.07972*, 2024.