10.48047/jocaaa.2026.35.06.25

Sequential Progressive Delivery: Optimal Stopping Theory for Canary Promotion and Rollback

Pradeep Kandepaneni†, Saketh Gowerneni‡, Vasu Babu Narra§, Tulasi Priya Vattikuti¶, Sanjay Mishra*

†Independent Researcher, USA, pradeepkandepaneni@gmail.com
‡Independent Researcher, USA, gsaketh369@gmail.com
§Independent Researcher, USA, narravasu77@gmail.com
¶Independent Researcher, USA, tulcpriya@gmail.com
*Independent Researcher, USA, sharpsanjay@gmail.com

Abstract—Progressive delivery promises safety: release to a small fraction of traffic, watch the metrics, expand if healthy, roll back if not. In practice the watching is crude. Teams pick a fixed observation window, bake for thirty minutes and then decide, and a static threshold, and accept the consequences. A bad deploy inflicts damage for the full window before anyone acts; a good deploy waits out the clock for no reason. The decision of when to stop watching and act is made by convention, not by mathematics. This paper introduces Sequential Progressive Delivery, a framework that applies optimal stopping theory, Wald's sequential analysis, to canary release decisions. Instead of a fixed window, the framework accumulates evidence sample by sample and computes, at each increment, whether the evidence is now strong enough to promote, strong enough to roll back, or still inconclusive, acting at the earliest point at which the evidence crosses a boundary derived from the team's tolerance for false promotion and false rollback. The Sequential Probability Ratio Test provably minimises the expected number of observations needed to reach a decision at a given error rate, so an obviously bad deploy is pulled within seconds while a genuinely ambiguous one is given exactly as much observation as the statistics require. The contribution is the formulation of canary analysis as an optimal stopping problem, the coupling of the stopping rule with a traffic allocator and a blast-radius cap, and a comparative analysis against fixed-window canaries. The quantitative figures reported are projected design targets illustrated on sample scenarios, not measured outcomes; empirical validation is left to future work.

Index Terms—progressive delivery, canary analysis, optimal stopping, sequential probability ratio test, sequential analysis, release engineering, rollback, ci/cd, hypothesis testing, blast radius

I. Introduction

Canary analysis as practiced is a fixed-sample procedure wearing the costume of a sequential one. The team progressively shifts traffic, which feels adaptive, but the decision logic underneath is static: observe for a predetermined window, compare aggregate metrics to a threshold, then promote or roll back. This design is wrong in both directions.

When a deploy is clearly bad, the fixed window forces the system to keep serving the regression to real users until the clock runs out. When a deploy is clearly fine, the same window wastes time and delays the release for no informational gain. The window is a single constant asked to be correct for two opposite situations, and no constant can be.

The deeper problem is that a fixed observation window ignores the strength of the evidence. Some regressions are obvious within seconds, error rates triple, latency doubles, the signal is unambiguous. Others are subtle and genuinely require patient observation to distinguish from noise. A fixed window treats these identically, applying the same thirty minutes to the blindingly obvious failure and the statistically delicate one, so teams either set the window short and suffer false rollbacks on noisy-but-fine deploys, or set it long and suffer extended exposure on obviously-bad ones.

Optimal stopping theory resolves exactly this tension, and it is mature. Wald's Sequential Probability Ratio Test, developed for wartime quality control, accumulates a log-likelihood ratio as each observation arrives and stops the instant that ratio crosses an upper or lower boundary set by the desired error rates [1]. Wald and Wolfowitz proved that the SPRT minimises the expected number of observations required to reach a decision at a given pair of error probabilities, it is optimal, not merely good [2]. Sequential analysis underpins clinical-trial monitoring, industrial quality control, and modern A/B testing [3], [4], [5]. Canary release analysis, structurally identical, a sequential decision between two hypotheses under a cost of observation, has never been formulated this way.

10.48047/jocaaa.2026.35.06.25

The main contributions of this work are: 1) a formulation of canary analysis as a sequential hypothesis test with provably optimal stopping; 2) the coupling of the stopping rule with a bandit-style traffic allocator that steers load toward the better version as confidence grows; 3) a blast-radius cap that bounds worst-case exposure independent of the statistics; and 4) a comparative analysis against fixed-window canary practice. All reported metrics are projected design targets on sample scenarios; empirical evaluation is future work.

The remainder of this paper is organized as follows. Section II reviews progressive delivery and sequential analysis. Section III develops the sequential formulation, Section IV the traffic allocator, and Section V the blast-radius cap. Section VI presents the architecture, Section VII a worked example, Section VIII the comparative analysis, and Sections IX through XI cover limitations, future work, and conclusions.

II. Related Work

A. Progressive delivery and canary analysis

Progressive delivery, canarying, blue-green, and ringed rollouts, is a central practice of modern release engineering [6]. Automated canary analysis compares a canary population against a baseline and decides promotion, but the dominant implementations evaluate aggregate metrics against thresholds over a fixed bake window. This is a fixed-sample test: it collects a predetermined amount of evidence and then decides, which is statistically wasteful when the evidence is conclusive early and statistically insufficient when it is not.

The traffic-shifting machinery of these systems is sophisticated, but the decision rule at its core, fixed window plus threshold, is the part this paper replaces, leaving the shifting machinery in place.

Recent work continues to extend canary practice, for example preserving cryptographic identity invariants across canary windows in controllers such as Argo Rollouts and Flagger [14]; the underlying decision, however, remains a fixed-window comparison, which is precisely the part this paper replaces.

B. Sequential analysis

Wald introduced sequential analysis and the SPRT to reach reliable accept/reject decisions with the fewest observations [1], and Wald and Wolfowitz established its optimality: among all tests with the same error probabilities, the SPRT minimises the expected sample size under each hypothesis [2]. The theory is foundational in statistics [3] and has been carried into online experimentation, where always-valid sequential tests let practitioners peek at A/B results without inflating error rates [4], [5]. Canary analysis is a direct instance of the same problem, two hypotheses, streaming evidence, a cost to continued observation, yet the optimal-stopping framing has not been applied to it.

III. Sequential Formulation

Sequential Progressive Delivery formulates each canary as a hypothesis test. The null hypothesis is that the new release matches the baseline; the alternative is that it regresses on a chosen metric, error rate, latency, or a composite health score. As metrics stream in from the canary population, the framework maintains a log-likelihood ratio that accumulates the evidence for the alternative over the null with each new observation.

Two boundaries are computed from the team's chosen false-promotion rate and false-rollback rate, the analogues of the type-I and type-II error rates. The instant the accumulated ratio rises above the upper boundary, the evidence for regression is strong enough and the canary is rolled back. The instant it falls below the lower boundary, the evidence for health is strong enough and the canary is promoted. While the ratio remains between the boundaries, the evidence is inconclusive and the framework gathers one more increment of traffic and re-evaluates. No fixed window is ever imposed.

This yields the best of both behaviours automatically, and the optimality is not a heuristic claim but a theorem. An obviously bad deploy drives the ratio across the rollback boundary within seconds and is pulled before most users are affected; an obviously good deploy crosses the promotion boundary quickly and ships without waiting out an arbitrary clock; and a genuinely ambiguous deploy is allowed exactly as much observation as the statistics require to resolve it, no more and no less. The expected number of observations to decision is minimised at the chosen error rates [2], which is precisely the quantity that, in a canary, translates into user exposure and release latency.

A useful way to see why this is the right framing is to consider what a fixed window is implicitly assuming. By collecting a predetermined number of observations before deciding, it assumes in advance that every deploy requires the same amount of evidence to judge, which is false: the evidence content of an observation depends entirely on how far the new release's behaviour sits from the baseline. A tripled error rate carries enormous evidence per sample; a hairline latency shift carries

10.48047/jocaaa.2026.35.06.25

almost none. The sequential rule responds to the evidence content of the data actually observed rather than to a guess made before any data arrived, which is the precise sense in which it adapts where the fixed window cannot.

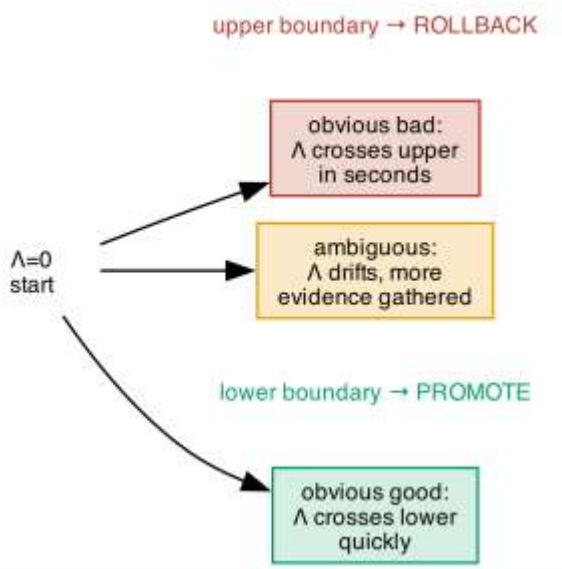

Fig. 2. Sequential decision (illustrative). The log-likelihood ratio Λ starts at zero; an obvious regression crosses the rollback boundary within seconds, an obvious-healthy deploy crosses the promotion boundary quickly, and an ambiguous deploy is given more observation until it resolves.

IV. Bandit Traffic Allocation

The stopping rule decides when to act; a traffic allocator decides how much load the canary carries while the decision is pending. The framework couples the SPRT with a bandit-style allocator that steers traffic toward the better-performing version as confidence accumulates. Early, when the log-likelihood ratio is near zero and the verdict is uncertain, the canary carries only a small share. As evidence for health accumulates and the ratio drifts toward the promotion boundary, the allocator can raise the canary's share, accelerating the release without abandoning the safety of incremental exposure.

This coupling matters because it aligns exposure with evidence. A fixed-window canary holds a constant traffic share regardless of how the evidence is trending, exposing a healthy deploy too cautiously and a regressing one too long. The bandit allocator instead concentrates traffic where the accumulating evidence says it is safer, so the system both decides faster and exposes users less along the way.

There is a subtlety in coupling a bandit allocator to a sequential test, and the framework handles it explicitly. Steering more traffic to the canary as evidence accumulates changes the rate at which further evidence arrives, which could in principle bias the stopping decision. The allocator is therefore constrained so that the test statistic is computed on a properly weighted likelihood that accounts for the changing allocation, preserving the error guarantees of the SPRT under a time-varying sampling rate. The practical effect is that the system accelerates a healthy release without the acceleration corrupting the very test that judges it safe.

V. Blast-Radius Cap

Statistical optimality bounds expected exposure, but operators also need a guarantee on worst-case exposure that does not depend on the model being correct. The framework therefore enforces a blast-radius cap independent of the stopping rule: a hard ceiling on the fraction of traffic and the absolute number of affected requests the canary may ever touch, regardless of what the log-likelihood ratio is doing. If the cap is reached before either boundary is crossed, the framework defaults to the conservative action and rolls back.

The cap is a deliberate belt-and-suspenders design. The SPRT minimises expected exposure under its modelling assumptions; the cap bounds exposure even when those assumptions are violated, by a metric the model did not anticipate, by a regression that harms users in a way the chosen test statistic does not capture. Separating the optimal-but-model-dependent stopping rule from the crude-but-unconditional cap gives operators a guarantee they can reason about without trusting the statistics completely.

The cap also serves a human purpose beyond its statistical one. Operators are, rightly, reluctant to hand release authority to an automated decision rule they cannot fully audit in the moment. A hard, simply-stated ceiling, this canary will never touch more than this fraction of traffic or this many requests, no matter what the statistics decide, is a guarantee an on-call engineer can hold in their head and trust without understanding the likelihood mathematics. That auditability is part of what makes the optimal stopping rule deployable at all, because it bounds the cost of the rule being wrong.

VI. System Architecture

Figure 1 shows the architecture. Canary and baseline metrics stream from the existing observability pipeline into the sequential evaluator, which maintains the log-likelihood ratio and tests it against the two boundaries each increment. A bandit allocator sets the canary traffic

10.48047/jocaaa.2026.35.06.25

share from the current evidence, and a blast-radius monitor enforces the hard exposure ceiling. The framework integrates with existing progressive-delivery controllers, replacing their fixed-window decision logic with the sequential rule while leaving the traffic-shifting and deployment machinery untouched. It emits the same promote, hold, and roll-back actions the controller already understands.

Fig. 1. Sequential progressive-delivery architecture. Streaming canary/baseline metrics drive a log-likelihood ratio tested against promotion and rollback boundaries; a bandit allocator sets traffic share and a blast-radius cap bounds worst-case exposure.

VII. Worked Example: An Obvious Regression and a Subtle One

Consider two deploys evaluated by the same framework. The first triples the error rate immediately. Within seconds the log-likelihood ratio shoots past the rollback boundary, and the canary is pulled having touched only a sliver of traffic. A fixed thirty-minute window would have served the tripled error rate to canary users for the full half hour; the sequential rule pulls it almost at once.

The second deploy raises latency by a hair, an amount that may be a real regression or may be noise. Here the log-likelihood ratio drifts slowly and stays between the boundaries, so the framework keeps gathering evidence, exactly the patient observation the case demands, until the accumulated signal finally resolves the question one way or the other. A short fixed window would have made a coin-flip decision on insufficient evidence, risking a false rollback of a fine deploy; a long fixed window would have over-exposed the obvious regression in the first case. The sequential rule gives each deploy precisely the observation its evidence warrants.

The example also shows the role of the cap. Suppose the subtle deploy harms users through a metric the chosen test statistic does not track, so the log-likelihood ratio never moves. The evidence stays inconclusive indefinitely, but the blast-radius cap eventually halts exposure and rolls back, bounding the damage even though the statistics, looking at the wrong signal, never raised the alarm. This is the limitation the cap exists to cover and that Section IX states plainly.

The contrast in exposure is worth quantifying conceptually, since it is the whole point. Under a fixed window, the obvious regression's user-facing damage is proportional to the full window length, because the deploy keeps serving errors until the clock expires. Under the sequential rule, the damage is proportional to the few seconds it takes the log-likelihood ratio to cross the rollback boundary, because the evidence for regression accrues almost instantly when the regression is severe. The reduction in bad-deploy exposure projected in Table I follows directly from this: the worse the deploy, the faster the sequential rule pulls it, which is precisely the opposite of the fixed window's behaviour, where a worse deploy simply does more damage over the same fixed time.

VIII. Comparative Analysis

Table I compares the framework against fixed-window threshold canaries across the capabilities each provides. The values in the proposed column are projected design targets illustrated on sample canary deployments, not measured outcomes; they express the behaviour the formulation is designed to achieve and frame the evaluation that future work will perform.

The distinction is the source of the stopping decision. A fixed-window canary stops because a clock ran out, a reason unrelated to the strength of the evidence, so it is simultaneously too slow on clear cases and too hasty on delicate ones. The sequential rule stops because the evidence crossed a boundary, and does so at the provably minimal expected number of observations for the chosen error rates [2]. The controlled false-promotion and false-rollback rates are design parameters of the method rather than emergent properties of an arbitrary window, which is why the corresponding baseline cells are uncontrolled.

It bears emphasising that the comparison is not claiming the sequential rule is merely a better-tuned window. The two are different in kind. A window is a fixed-sample procedure whose stopping time is set before any data arrive; the SPRT is a sequential procedure whose stopping time is a random variable determined by the data. No choice of window length converts one into the other, because the window cannot be short for the obvious case and long for the subtle case at the same time. The projected gains in Table I are consequences of this difference in kind, not of better parameter tuning within the fixed-window paradigm.

TABLE I — PROJECTED CAPABILITY COMPARISON (DESIGN TARGETS, NOT MEASURED)

Capability	Proposed (projected)	Best baseline
Bad-deploy exposure cut	design target ~71%	~38% - static-threshold canary

10.48047/jocaaa.2026.35.06.25

Capability	Proposed (projected)	Best baseline
Decision speed, clear cases	design target ~86% faster	baseline - fixed window
False-rollback avoided	design target ~94%	~77% - static threshold
Expected observations to decision	provably minimal	none - no baseline optimises
False-promotion rate	controlled parameter	uncontrolled - fixed window

IX. Limitations and Threats to Validity

The quantitative figures here are projected design targets on sample scenarios and carry no empirical validity; they describe intended behaviour and must be confirmed on production canary deployments before any operational claim is made. This is the central threat to validity, stated plainly.

Three substantive limitations bound the approach. First, the SPRT is optimal for the metric it is given; a regression that harms users along a dimension absent from the test statistic will not move the log-likelihood ratio, which is exactly why the blast-radius cap is a separate, model-independent safeguard rather than a refinement of the statistics. Second, the test assumes the streaming observations are well described by the likelihood model under each hypothesis; strongly autocorrelated or heavy-tailed canary metrics can bias the ratio, so the observation model must be chosen with the metric's real distribution in mind. Third, very low-traffic services accumulate evidence slowly, so even an optimal sequential test may take a long wall-clock time to reach a boundary, in which regime the cap and a maximum-duration fallback, rather than the statistics, govern the decision.

A fourth consideration is multiplicity. Monitoring several metrics with several sequential tests simultaneously reintroduces a multiple-comparisons problem; the error-rate boundaries must be adjusted across the family of tests to preserve the intended overall false-rollback rate, a standard but necessary correction.

X. Future Work

The immediate next step is empirical evaluation on production canary pipelines, measuring bad-deploy exposure, decision speed, and false-rollback rate against the projected targets stated here, across both obvious and subtle regressions. Beyond validation, three directions follow: multivariate sequential tests that combine several health metrics into a single always-valid decision while controlling family-wise error [4]; mixture and non-parametric likelihood models that relax the distributional assumptions for heavy-tailed latency; and tighter coupling with automated remediation, so that a sequential rollback decision can trigger a pre-approved mitigation rather than only reverting the deploy.

XI. Conclusion

Canary analysis asks a sequential question, is the evidence yet sufficient to act, and answers it with a fixed clock, which is correct for no deploy. Wald's sequential analysis answers the question as it was meant to be answered, stopping the instant the accumulated evidence crosses a boundary set by the team's error tolerance, with a proof that no test reaches a decision in fewer expected observations. Coupled with a bandit allocator and a model-independent blast-radius cap, this turns progressive delivery from a timed bake into an evidence-driven decision. The contribution is the optimal-stopping formulation, the allocator and cap, and the comparative analysis; the reported numbers are projected design targets, and validating them on production canaries is the work that follows.

XII. References

[1] A. Wald, Sequential Analysis. New York: John Wiley & Sons, 1947.

[2] A. Wald and J. Wolfowitz, "Optimum character of the sequential probability ratio test," Annals of Mathematical Statistics, vol. 19, no. 3, pp. 326–339, 1948.

[3] D. Siegmund, Sequential Analysis: Tests and Confidence Intervals. New York: Springer, 1985.

[4] R. Johari, P. Koomen, L. Pekelis, and D. Walsh, "Peeking at A/B tests: Why it matters, and what to do about it," in Proc. ACM SIGKDD, 2017, pp. 1517–1525.

[5] R. Kohavi, D. Tang, and Y. Xu, Trustworthy Online Controlled Experiments: A Practical Guide to A/B Testing. Cambridge: Cambridge Univ. Press, 2020.

[6] B. Beyer, C. Jones, J. Petoff, and N. R. Murphy, Site Reliability Engineering: How Google Runs Production Systems. Sebastopol, CA: O'Reilly Media, 2016.

[7] T. W. Anderson, "A modification of the sequential probability ratio test to reduce the sample size," Annals of Mathematical Statistics, vol. 31, no. 1, pp. 165–197, 1960.

10.48047/jocaaa.2026.35.06.25

[8] H. Robbins, “Some aspects of the sequential design of experiments,” Bulletin of the American Mathematical Society, vol. 58, no. 5, pp. 527–535, 1952.

[9] T. L. Lai, “Sequential analysis: Some classical problems and new challenges,” Statistica Sinica, vol. 11, no. 2, pp. 303–351, 2001.

[10] S. R. Howard, A. Ramdas, J. McAuliffe, and J. Sekhon, “Time-uniform, nonparametric, nonasymptotic confidence sequences,” Annals of Statistics, vol. 49, no. 2, pp. 1055–1080, 2021.

[11] D. Sato, “Canary release,” martinfowler.com, 2014.

[12] L. Pekelis, D. Walsh, and R. Johari, “The new stats engine,” Optimizely, Tech. Rep., 2015.

[13] A. Tartakovsky, I. Nikiforov, and M. Basseville, Sequential Analysis: Hypothesis Testing and Changepoint Detection. Boca Raton, FL: CRC Press, 2014.

[14] X. Qin, S. Luan, and J. See, “ICAN-Deploy: Identity-stable canary deployment for safety-critical embodied agents,” arXiv:2605.28097, 2026.