Active Inference Power Suite: Conditional Statistical Power under Controlled Generative Settings

Connecting generative-model formalisms to executable action-in-the-loop evidence

Daniel Ari Friedman

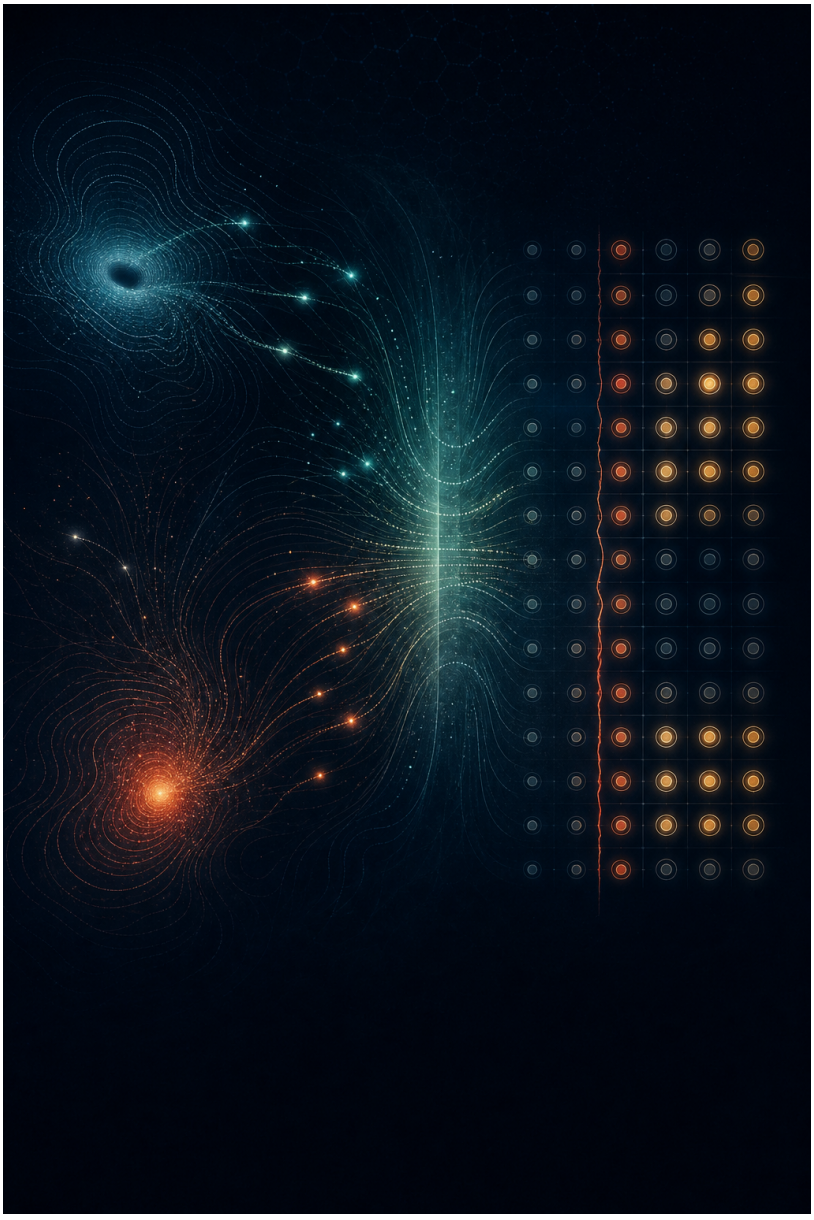
Active Inference Institute

daniel@activeinference.institute

ORCID: 0000-0001-6232-9096

DOI: 10.5281/zenodo.21695160

2026-07-29



Contents

1 Abstract

Statistical power is an investigator-facing operating characteristic of an adaptive-study design. Before simulation, the investigator fixes an agent-side model, evaluator-side process, testing setting, policy, and replication plan. Each embedded agent acts only on visible history; simulated hidden truth is retained for scoring. `active_inference_power` makes that conditional estimand inspectable. The suite combines fixed-horizon procedures, analytic references, dependence/calibration experiments, and a discrete-state active-inference agent checked against a binary oracle. It extends this to action loops with sensing reliability, latent context, target choice, cost, and stopping.

The study distinguishes model-relative posterior belief from calibrated p-value and likelihood-ratio e-process evidence. It compares Benjamini–Hochberg (BH) false discovery rate (FDR) procedures with family-wise error rate (FWER) alternatives, and separates either evidence object from online FDR procedure-specific accounting. Results are scenario-indexed finite-simulation estimates with Monte Carlo standard error (MCSE) and declared error, dependence, filtration, and optional-stopping boundaries; they do not assign a universal power value to an agent, task environment, or active inference. Instead, they support auditable comparisons among explicitly declared adaptive-study designs. Contracts, seed schedules, certificates, figures, claim ledger, and rendered manuscript form a linked evidence chain, allowing readers to trace each claim to its design and artifact. Source and release materials are available at the verified GitHub repository `ActiveInferenceInstitute/active_inference_power`.

2 Introduction

2.1 A concrete adaptive-design problem

Consider the controlled action loop used in this suite. A policy faces two candidate targets with binary truth and a latent context. On each round, it chooses a target and a sensing quality, receives an observation, and either pays to continue or stops. Buying a more reliable observation after a promising signal can change the chance of identifying a true alternative, the data that will be available later, and the resources that the run consumes. Repeatedly looking at a fixed-horizon statistic can also change the inferential question. The study therefore needs more than a headline power value: it needs a predeclared account of the process, decision rule, and stopping rule [Chernoff, 1959, Lindley, 1956].

There are two roles in that study. An embedded agent uses only its visible history to form beliefs and choose an action. An investigator specifies the synthetic task environment, testing contract, policy, and replication plan, then uses evaluator-only hidden truth to score completed traces. Thus the agent faces uncertainty *within* a specified process, whereas power is estimated *across* replications of an investigator-declared design [Neyman and Pearson, 1933, Morris et al., 2019]. In this manuscript, "simulated niche" is only informal shorthand for that evaluator-side process; it is not a claim about an empirical ecological, social, or deployment population.

The suite consequently has two related study lanes. The fixed-horizon reference is an investigator simulation of declared observation streams, calibrated evidence, and correction procedures; it does not by itself instantiate an action-controlled agent–environment loop. The action-loop lane adds an embedded agent whose policy chooses actions under uncertainty within the fixed synthetic process. In both lanes, reported power is an operating characteristic of the investigator's declared design, not a general measure of agent competence, fitness, or task-environment quality.

Active inference supplies a language for the data-collection part of that problem. An agent represents hidden states, updates model-relative beliefs from observations, and selects actions or policies using stated preferences and epistemic value [Friston et al., 2016, 2017, Parr and Friston, 2019]. In a discrete state-space model, those ingredients can be written as explicit likelihoods, transitions, observations, and policy evaluations [Da Costa et al., 2020, Heins et al., 2022]. Multiple testing supplies the complementary decision contract: a family procedure controls a named error criterion only for evidence objects and dependence conditions in its stated scope [Benjamini and Hochberg, 1995, Benjamini and Yekutieli, 2001].

The practical question is consequently not whether active inference has one universal power. It is **how an investigator-estimated operating characteristic changes when a generative model, generative process, testing setting, or policy changes**. A posterior belief can help a policy choose an observation, but it is not thereby a calibrated p-value or e-value. A likelihood-ratio e-process can support a stopping-time statement only under its declared null and filtration conditions. An online false discovery rate (FDR) procedure uses a different accounting contract again. This manuscript makes those distinctions visible before it compares procedures or policies.

2.2 Reader's guide to the conditional-power question

Readers can use the manuscript in three passes. First, identify the scenario: what process creates observations, what policy can see, and what it costs to collect another observation. Next, identify the evidence interface and error criterion: a posterior, p-value, e-process, and online alpha-accounting balance are related computational objects but do not license the same decision claim. Finally, read the uncertainty and validity label with every result. The label states whether a row is a formal conditional statement, a scenario-specific operating characteristic, a calibration diagnostic, a stress test, or a deliberately invalid contrast. The public application programming interface (API) names what downstream code can call; JavaScript Object Notation (JSON) names the result, trace, manifest, and dashboard payloads through which the same declared objects can be inspected.

2.3 Study aims and evidence boundary

The reader guide yields four linked, executable questions. The fixed-horizon reference asks whether a posterior-threshold decision and a family correction applied to calibrated evidence behave differently on the same observation streams. The action-loop studies ask how sensing quality, target choice, environment-changing actions, stopping, and resource cost alter conditional power and error summaries. The adaptive-selection studies ask whether the information used to select a target is separated from its confirmation evidence. The sequential studies ask which conclusion remains available when the evidence object and filtration support continued sampling.

Each answer has a bounded evidence class. The package supplies a tested comparison surface for established correction procedures, not a new FDR procedure or a general active-inference model. Its practical contribution is an inspectable, reproducible path [Sandve et al., 2013, Wilkinson et al., 2016] from scenario contract to rendered artifact; provenance cannot make an uncalibrated object valid. Independent-confirmation, e-process, confidence-sequence, and online-FDR tracks retain separate assumptions, and common-random-number simulations estimate declared-scenario operating characteristics and paired contrasts rather than proofs [Morris et al., 2019].

2.3.1 Notation and terminology: a reader guide

2.3.1.1 Design and information contract

Reader term	Meaning in this study	Why it matters
Scenario s	The scientific model/process/setting components of one `PowerScenario`: a fixed agent-side generative model, evaluator-side generative process, and testing setting.	Its serialized hash also binds scenario identity and claim metadata; a separately supplied surface/action policy is not hidden in that hash.
Evaluated design $d = (s, \pi)$	One scenario paired with an executed agent policy. The replication plan B controls the estimator's precision, not the definition of d.	This is the investigator-facing object whose operating characteristics are estimated and compared.
Generative model	The agent-side representation of hidden states, observations, priors, likelihoods, transitions, preferences, and available actions.	It determines how the agent forms beliefs and evaluates actions.
Generative process P_N	The evaluator-side mechanism that generates hidden truth, context, action-dependent observations, missingness, reliability, and costs. "Simulated niche" is informal shorthand only.	It determines the data and truth against which an operating characteristic is scored; it is not a sampled population of real-world environments.
Testing setting	The hypothesis family, null law, evidence statistic, correction, nominal level, stopping rule, cost budget (a cap in the declared C units), and stated validity assumptions.	It fixes what a rejection and an error criterion mean without turning a cost ceiling into alpha accounting.
Policy π and visible filtration $\mathcal{F}_t$	A policy maps only the history visible before time t to an action, target choice, or stopping decision. Hidden evaluator truth and future confirmation observations are excluded.	This information boundary determines whether adaptive selection or stopping has the claimed interpretation.
Resource cost C	Cumulative simulated sampling cost in the scenario's hash-bound `cost_unit` and `cost_scale_id`. The current scenarios do not identify those units with money, time, energy, or physical resource use; cross-scenario cost comparisons require matching identifiers.	It reports what a trace consumed; it is neither a probability, evidence score, nor an accounting balance.
Source-work proxy Q_{comp}	A deterministic, version-specific proxy assembled from named array passes, history scans, and declared execution paths; the ledger reports retained-draw elements separately as retention accounting.	It makes the code path and input shape auditable; it is not simulated cost C, elapsed time, a memory measurement, statistical evidence, or alpha accounting.

Reader term	Meaning in this study	Why it matters
Local elapsed time τ_{wall}	Repeated wall-clock samples from one named local runtime configuration, retained only in an isolated diagnostic.	It can describe that host's observed timing protocol but cannot supply a cross-hardware ranking, a release gate, or a statistical guarantee.
Conditional power / mean true-positive rate (TPR) $\theta_{\text{TPR}}(d)$	The expected replication-level true-positive rate under the fixed scenario and policy. Independent replications estimate it with MCSE; the different "any true target rejected" event is reported separately.	It makes the conditioning explicit instead of treating power as a property of an algorithm, agent, or task environment alone.

This separation is substantive. The model can be misspecified relative to the process. A policy can alter the process by choosing a sensing channel or a target. A testing setting can require evidence that a posterior does not provide. The suite records the scenario components and their hashes so a comparison remains attached to the design that produced it.

2.3.1.2 Belief, calibrated evidence, and accounting are not interchangeable For a visible history $\mathcal{F}_t$, a posterior such as $q_t = \Pr(H_1 \mid \mathcal{F}_t)$ answers a model-relative question: given the prior, likelihood, and observations, how much probability does the agent assign to a hidden state? It may be useful for action selection, but it has no family-level error interpretation by itself. The fixed-horizon reference turns the same observation stream into a p-value only by specifying a null distribution, then passes that calibrated evidence to a family procedure.

Reader term	Meaning in this study	Not the same thing as
Posterior belief q_t	Model-relative probability assigned to a hidden state after the visible observations.	A calibrated p-value, e-value, FDR guarantee, or resource budget.
P-value p_i	A statistic whose null-distribution behavior is declared for hypothesis i; the fixed-horizon reference uses it as the input to a family correction.	A posterior probability or a license to keep peeking after the stated horizon.
Named online procedures	Levels based On Recent Discovery (LORD) is an online p-value family; the implemented LORD++ rule is one member. Serial estimate of the Alpha Fraction that is Futilely Rationed On true null hypotheses (SAFFRON) uses candidate information under its stated assumptions. Levels based On Number of Discoveries (LOND) underlies the distinct predictable e-value procedure e-LOND described below.	A substitute for an e-process or a proof about an undeclared arrival process.
Likelihood-ratio e-process E_t	A nonnegative sequential evidence process that is a supermartingale under its declared null (a martingale for a correctly specified likelihood ratio); each predictable multiplicative e-factor has conditional mean at most one. Its stopping-time interpretation is conditional on the predictable update rule and model/process likelihood agreement [Shafer et al., 2011, Grünwald et al., 2024, Howard et al., 2020].	Online alpha accounting, a posterior, or a simulated resource cost.
Online alpha-wealth W_t^{α} (LORD++/SAFFRON)	A bookkeeping balance that determines which future p-value testing levels an online procedure may spend or replenish under its stated rule. Its update depends on the declared history and assumptions [Foster and Stine, 2008, Ramdas et al., 2017, 2018].	The e-process value E_t, an effect size, expected utility, or a sampling budget.

Reader term	Meaning in this study	Not the same thing as
e-LOND nominal levels and cumulative allocation	e-LOND emits predictable nominal test levels α_t for e-values and rejects when $E_t \geq 1/\alpha_t$. Its accounting trace is the cumulative emitted level $\sum_{i \leq t} \alpha_i$; recycled levels after rejections can make that trace exceed the nominal α [Xu and Ramdas, 2024].	Alpha-wealth, a remaining alpha budget, the e-process value E_t, or a resource cost.
Resource cost C	The cost accumulated by chosen actions in the declared generative process.	Either form of statistical evidence or alpha accounting.

To avoid ambiguity, this guide qualifies the accounting quantities. A plotted e-process path is called the e-process value E_t; LORD++/SAFFRON panels call their balance alpha-wealth W_t^α; and e-LOND panels call their path the cumulative nominal-level allocation $\sum_{i \leq t} \alpha_i$. Captions also state the e-LOND decision rule $E_t \geq 1/\alpha_t$ where it matters. Neither quantity is the resource cost C. This naming convention prevents a numerical display label from suggesting a statistical guarantee that belongs to another object.

Computational feasibility has two further, deliberately separate labels. The release ledger uses Q_{comp} for a dimensionless source-work proxy derived from the current implementation and the declared input shape. An optional local diagnostic uses τ_{wall} for repeated elapsed time on one host. Neither is simulated resource cost C, e-process evidence E_t, LORD++/SAFFRON alpha-wealth W_t^α, or e-LOND cumulative nominal-level allocation; none changes a power, FDR, FWER, or validity label. This distinction lets a reader ask whether a design is computationally feasible without mistaking a machine observation for statistical evidence.

2.3.1.3 Error rates, uncertainty, and evidence labels

Reader term	Working definition and boundary
True nulls, alternatives, and rejections	$\mathcal{H}_0$ and $\mathcal{H}_1$ are the declared true-null and true-alternative sets; $\mathcal{R}$ is the rejection set, with total rejections R, false rejections V, and true rejections S.
False discovery proportion (FDP)	The realized ratio $V/\max(R,1)$ for one replication. It is a path-level outcome, not a guarantee.
False discovery rate (FDR)	The expectation of FDP under the declared model, process, evidence, and correction. Independent replications estimate it with MCSE.
Family-wise error rate (FWER)	The probability of at least one false rejection under the declared setting. It is a different target from FDR and typically has a different power trade-off.
Calibration	Agreement of an evidence object's behavior with its declared null-law requirement. A model-relative posterior, even if assessed as calibrated under its own generative model, is not automatically a calibrated p-value or e-value.
Positive regression dependence on a subset (PRDS)	A dependence condition relevant to selected FDR guarantees. The suite's positive-factor simulations are stress evidence under a declared regime; they do not establish arbitrary-dependence behavior [Benjamini and Yekutieli, 2001].
Monte Carlo standard error (MCSE)	The simulation standard error of a replication-based estimate. It describes finite-simulation precision, not a theorem about a broader population of processes [Morris et al., 2019].
Split-stream confirmation	Selection uses one visible data stream and the selected target is tested on a fresh independent confirmation stream. Its conditional-null contract must still be stated; a split alone cannot repair a misspecified process or stopping rule [Fithian et al., 2014].
Confidence sequence	A sequence of intervals with a stated time-uniform interpretation under its declared process and filtration assumptions [Howard et al., 2021].

Reader term	Working definition and boundary
Validity label	A compact evidence class carried with results and figures: *formal conditional* when a stated theorem contract and its implementation checks apply, *conditional* for a fully specified operating characteristic, *trace diagnostic* for one retained path, *diagnostic* for a calibration or contrast check, *stress* for a broader perturbation, and *invalid contrast* for a deliberately unsuitable decision path.

3 Methodology

The package separates deterministic domain functions from orchestration. The public API is specified in the package contract; the experiment script only loads configuration, calls those functions, and writes artifacts. Every stochastic path accepts or records a seed.

3.1 Estimands, error criteria, and evidence layers

The suite connects formal statements to executable objects through a versioned traceability registry. Each entry identifies a formal object, public symbol, source module, real-value test, assumptions, evidence class, artifact, and non-claim. The registry is validated during result generation and carried into the result artifact with hash `36d0ab9e47ca5be60575703145cbda4592d3d18e3bbd65d2feaba290d3a2b2e9`.

The table uses word-spaced display labels so long identifiers remain readable in narrow publication columns; the exact importable symbols, source paths, and tests remain available in the hashed registry artifact.

Table 1: Formalism-to-code traceability rows generated from the validated registry; the table is a lineage contract, not an additional numerical result.

Layer	Formal object	Public anchor / evidence class
estimand	Conditional power functional	scenario / PowerScenario (+2 symbols); conditional operating characteristic
generative model	Generative model, process, and testing-setting composition	scenario / GenerativeModelSpec (+3 symbols); formal conditional
inference	Bayesian belief update and executable oracle	active inference / run AI test (+3 symbols); formal conditional
testing setting	Fixed-horizon multiple-testing procedures	corrections / benjamini hochberg (+3 symbols); formal conditional
testing setting	Global-null BH rejection count and FDR distinction	theory / expected false rejections global null (+2 symbols); analytic reference
testing setting	Conservative fixed-lambda adaptive FDR	corrections / storey qvalue conservative (+1 symbols); calibration diagnostic
testing setting	Independent split-stream Storey confirmation	corrections / storey qvalue split confirmation (+1 symbols); calibration diagnostic
generative process	Dependence-aware Gaussian process generators	dependence / correlation matrix (+2 symbols); calibration diagnostic
generative process	Imported two-sided non-PRDS stress protocol	protocols / load dobriban BH protocol (+2 symbols); stress test
evidence	Likelihood-ratio e-process and Ville boundary	sequential / binary e process (+2 symbols); formal conditional
evidence	Action-loop e-process channel contract	action loop / e process contract valid (+1 symbols); formal conditional

Layer	Formal object	Public anchor / evidence class
decision	LORD++, SAFFRON, and e-LOND online evidence accounting	online FDR / default gamma (+3 symbols); formal conditional
decision	Action-in-the-loop filtration and policy trace	action loop / run action loop (+2 symbols); conditional operating characteristic
decision	Independent split-stream selection and confirmation	selection / SelectionPolicySpec (+3 symbols); formal conditional
evidence	Bernoulli mixture confidence sequence	sequential / bernoulli confidence sequence (+1 symbols); formal conditional
decision	Frozen learned-policy evaluation	learned policy / LearnedPolicyArtifact (+2 symbols); conditional operating characteristic
testing setting	Testing-setting sensitivity surface	testing setting sensitivity / TestingSettingSensitivityPlan (+1 symbols); conditional operating characteristic
decision	Stratified independent split-stream calibration audit	scenario / make static null binary e process scenario (+2 symbols); calibration diagnostic
generative process	Paired dynamic-process stress controls	dynamic process sensitivity / DynamicProcessSensitivity-Plan (+2 symbols); stress test
evidence	Sequential e-process comparison atlas	sequential atlas / SequentialEvidenceAtlasConfig (+1 symbols); calibration diagnostic
decision	Repeated train/freeze/evaluate policy partitions	repeated policy evaluation / RepeatedLearnedPolicyPlan (+1 symbols); conditional operating characteristic
generative process	Multi-target arrival and independent-confirmation environment	multi target environment / MultiTargetEnvironmentSpec (+2 symbols); stress test
artifact	Exact terminal and diagnostic certificate partition	certificate gate / normalize certificate gate policy (+2 symbols); provenance contract
evidence	Frozen three-way Storey method review	storey method review / StoreyMethodReviewPlan (+2 symbols); calibration diagnostic
artifact	Offline checksum and provenance benchmark preflight	benchmark preflight / OfflineBenchmarkPreflight (+1 symbols); provenance contract
uncertainty	Replication-level uncertainty and common-random-number contrasts	uncertainty / wilson interval (+3 symbols); calibration diagnostic
artifact	Schema, hash, and claim-ledger lineage	schema / validate results (+3 symbols); provenance contract

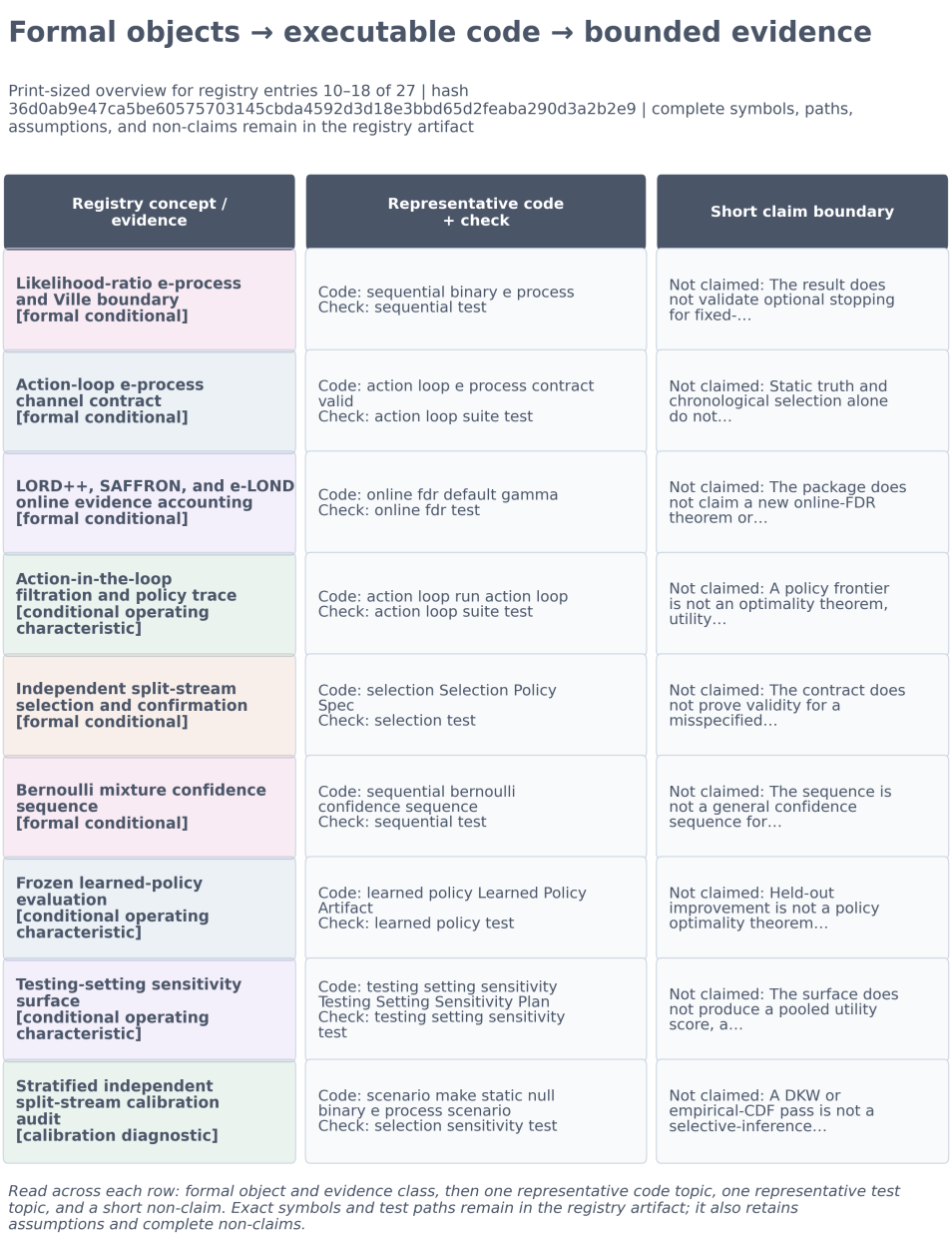
Formal objects → executable code → bounded evidence

Print-sized overview for registry entries 1–9 of 27 | hash 36d0ab9e47ca5be60575703145cbda4592d3d18e3bbd65d2feaba290d3a2b2e9 | complete symbols, paths, assumptions, and non-claims remain in the registry artifact

Registry concept / evidence	Representative code + check	Short claim boundary
Conditional power functional [conditional operating characteristic]	Code: scenario Power Scenario Check: action loop suite test	Not claimed: No unconditional or agent-invariant power claim, niche…
Generative model, process, and testing-setting composition [formal conditional]	Code: scenario Generative Model Spec Check: action loop suite test	Not claimed: The contract does not prove that an arbitrary user-…
Bayesian belief update and executable oracle [formal conditional]	Code: active inference run ai test Check: active inference test	Not claimed: Oracle agreement is not evidence that the binary channel…
Fixed-horizon multiple-testing procedures [formal conditional]	Code: corrections benjamini hochberg Check: corrections test	Not claimed: Finite simulations do not establish a new theorem or…
Global-null BH rejection count and FDR distinction [analytic reference]	Code: theory expected false rejections global null Check: theory test	Not claimed: This identity and finite expectation calculation do not…
Conservative fixed-lambda adaptive FDR [calibration diagnostic]	Code: corrections storey qvalue conservative Check: corrections test	Not claimed: Applying that bound to the same p-values is an…
Independent split-stream Storey confirmation [calibration diagnostic]	Code: corrections storey qvalue split confirmation Check: corrections test	Not claimed: The finite certificate is not a new universal Storey FDR…
Dependence-aware Gaussian process generators [calibration diagnostic]	Code: dependence correlation matrix Check: dependence test	Not claimed: Negative, block, and factor regimes do not constitute…
Imported two-sided non-PRDS stress protocol [stress test]	Code: protocols load dobriban bh protocol Check: dobriban protocol test	Not claimed: The adapter does not reproduce the external outward-…

Read across each row: formal object and evidence class, then one representative code topic, one representative test topic, and a short non-claim. Exact symbols and test paths remain in the registry artifact; it also retains assumptions and complete non-claims.

Figure 1: How does the first registry panel connect formal objects to executable symbols, tests, and limits of interpretation? First formalism-to-code traceability matrix panel for registry entries 1–9; each print-sized row shows a formal object, evidence class, representative code and test topics, and an explicit non-claim boundary. The hashed registry retains complete symbols, test paths, assumptions, and non-claims; two separately rendered panels continue the overview. Traceability supports auditability but does not convert implemented code into a proof. Traceability makes assumptions auditable; it does not prove theorems or broaden statistical validity.



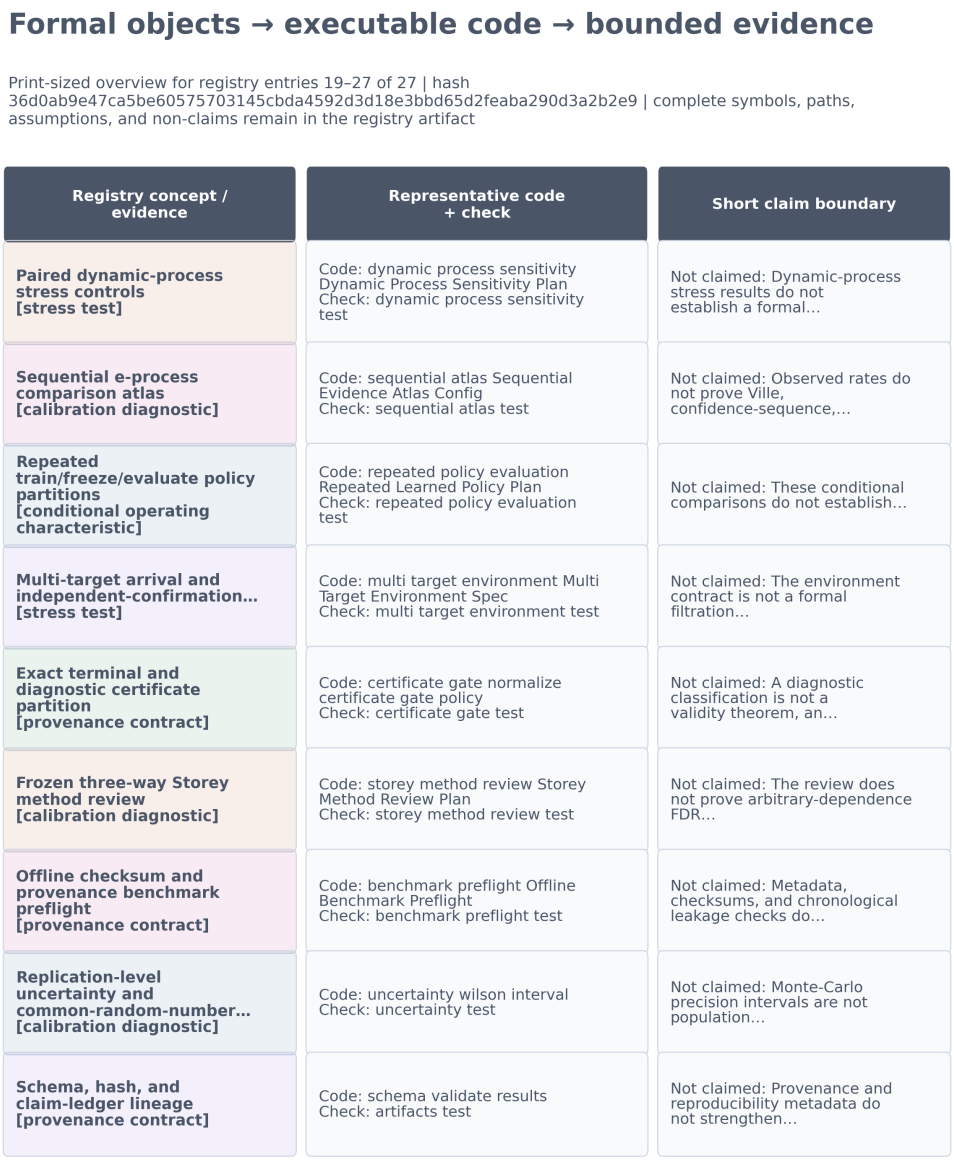
Formal objects → executable code → bounded evidence

Print-sized overview for registry entries 10–18 of 27 | hash 36d0ab9e47ca5be60575703145cbda4592d3d18e3bbd65d2feaba290d3a2b2e9 | complete symbols, paths, assumptions, and non-claims remain in the registry artifact

Registry concept / evidence	Representative code + check	Short claim boundary
Likelihood-ratio e-process and Ville boundary [formal conditional]	Code: sequential binary e process Check: sequential test	Not claimed: The result does not validate optional stopping for fixed-…
Action-loop e-process channel contract [formal conditional]	Code: action loop e process contract valid Check: action loop suite test	Not claimed: Static truth and chronological selection alone do not…
LORD++, SAFFRON, and e-LOND online evidence accounting [formal conditional]	Code: online fdr default gamma Check: online fdr test	Not claimed: The package does not claim a new online-FDR theorem or…
Action-in-the-loop filtration and policy trace [conditional operating characteristic]	Code: action loop run action loop Check: action loop suite test	Not claimed: A policy frontier is not an optimality theorem, utility…
Independent split-stream selection and confirmation [formal conditional]	Code: selection Selection Policy Spec Check: selection test	Not claimed: The contract does not prove validity for a misspecified…
Bernoulli mixture confidence sequence [formal conditional]	Code: sequential bernoulli confidence sequence Check: sequential test	Not claimed: The sequence is not a general confidence sequence for…
Frozen learned-policy evaluation [conditional operating characteristic]	Code: learned policy Learned Policy Artifact Check: learned policy test	Not claimed: Held-out improvement is not a policy optimality theorem…
Testing-setting sensitivity surface [conditional operating characteristic]	Code: testing setting sensitivity Testing Setting Sensitivity Plan Check: testing setting sensitivity test	Not claimed: The surface does not produce a pooled utility score, a…
Stratified independent split-stream calibration audit [calibration diagnostic]	Code: scenario make static null binary e process scenario Check: selection sensitivity test	Not claimed: A DKW or empirical-CDF pass is not a selective-inference…

Read across each row: formal object and evidence class, then one representative code topic, one representative test topic, and a short non-claim. Exact symbols and test paths remain in the registry artifact; it also retains assumptions and complete non-claims.

Figure 2: How do the next formalism entries connect to executable symbols, tests, and claim boundaries? Second formalism-to-code traceability matrix panel for registry entries 10–18; each print-sized row shows evidence class, representative code/test topics, and an explicit non-claim boundary; the hashed registry retains the complete record. The continuation remains a print-sized overview; the complete symbols, tests, assumptions, and non-claims remain in the registry artifact. Traceability makes assumptions auditable; it does not prove theorems or broaden statistical validity.



Formal objects → executable code → bounded evidence

Print-sized overview for registry entries 19–27 of 27 | hash 36d0ab9e47ca5be60575703145cbda4592d3d18e3bbd65d2feaba290d3a2b2e9 | complete symbols, paths, assumptions, and non-claims remain in the registry artifact

Registry concept / evidence	Representative code + check	Short claim boundary
Paired dynamic-process stress controls [stress test]	Code: dynamic process sensitivity Dynamic Process Sensitivity Plan Check: dynamic process sensitivity test	Not claimed: Dynamic-process stress results do not establish a formal…
Sequential e-process comparison atlas [calibration diagnostic]	Code: sequential atlas Sequential Evidence Atlas Config Check: sequential atlas test	Not claimed: Observed rates do not prove Ville, confidence-sequence,…
Repeated train/freeze/evaluate policy partitions [conditional operating characteristic]	Code: repeated policy evaluation Repeated Learned Policy Plan Check: repeated policy evaluation test	Not claimed: These conditional comparisons do not establish…
Multi-target arrival and independent-confirmation… [stress test]	Code: multi target environment Multi Target Environment Spec Check: multi target environment test	Not claimed: The environment contract is not a formal filtration…
Exact terminal and diagnostic certificate partition [provenance contract]	Code: certificate gate normalize certificate gate policy Check: certificate gate test	Not claimed: A diagnostic classification is not a validity theorem, an…
Frozen three-way Storey method review [calibration diagnostic]	Code: storey method review Storey Method Review Plan Check: storey method review test	Not claimed: The review does not prove arbitrary-dependence FDR…
Offline checksum and provenance benchmark preflight [provenance contract]	Code: benchmark preflight Offline Benchmark Preflight Check: benchmark preflight test	Not claimed: Metadata, checksums, and chronological leakage checks do…
Replication-level uncertainty and common-random-number… [calibration diagnostic]	Code: uncertainty wilson interval Check: uncertainty test	Not claimed: Monte-Carlo precision intervals are not population…
Schema, hash, and claim-ledger lineage [provenance contract]	Code: schema validate results Check: artifacts test	Not claimed: Provenance and reproducibility metadata do not strengthen…

Read across each row: formal object and evidence class, then one representative code topic, one representative test topic, and a short non-claim. Exact symbols and test paths remain in the registry artifact; it also retains assumptions and complete non-claims.

Figure 3: How do the final formalism entries complete the trace from formal object to bounded evidence? Final formalism-to-code traceability matrix panel for registry entries 19–27; it completes the print-sized overview of evidence class, representative code/test topics, and explicit non-claim boundaries; the hashed registry retains the complete record. The final panel closes the registry without implying that traceability alone proves the listed claims. Traceability makes assumptions auditable; it does not prove theorems or broaden statistical validity.

The testing setting is a separate executable object. It fixes the hypothesis family, null law, evidence interface, accounting rule, default stopping rule, cost contract, reported estimands, and validity boundary before replications begin. The testing-setting contract in fig. 4 makes that preregistration surface inspectable; its boxes describe inputs and contracts rather than measured outcomes.

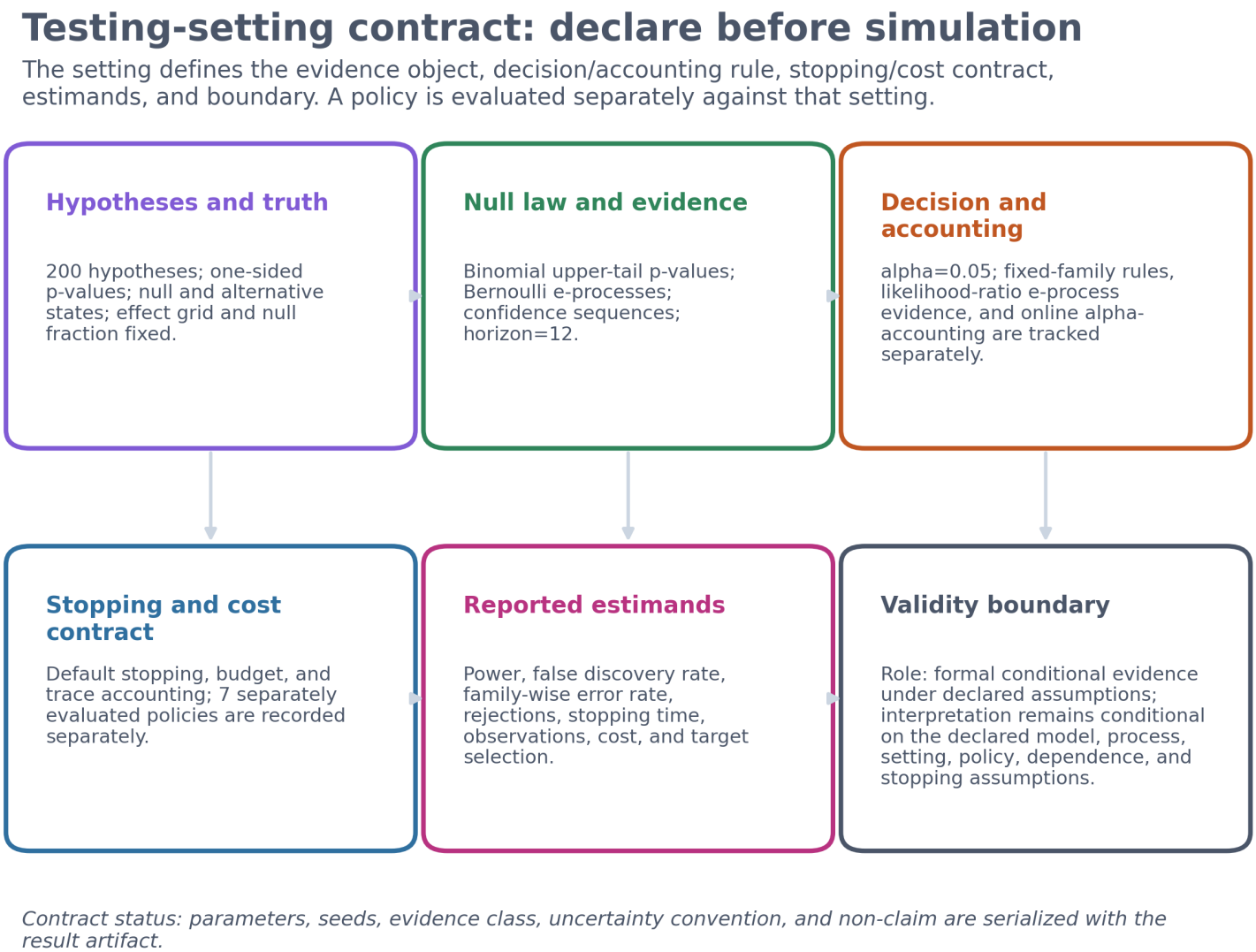


Figure 4: What must be declared before simulated power or error summaries can be compared? Testing-setting contract for the declared release: hypothesis family, null law and evidence interface, accounting procedure, default stopping rule and cost contract, estimands, and validity boundary are fixed before simulation. This visualization records the interface contract and does not add an operating-characteristic estimate. The contract turns otherwise implicit hypothesis, evidence, accounting, default stopping, cost, and validity choices into auditable inputs. A declared setting makes assumptions auditable; it does not establish validity outside its null, filtration, dependence, and stopping contracts.

3.1.1 Three contracts, one policy, and two roles

The implementation keeps three probability objects distinct. The agent model defines the likelihood and transition quantities used for belief updating. The process defines the evaluator-side distribution from which truth and observations are drawn, including action-dependent changes. The testing setting defines the null law, evidence statistic, rejection rule, and error criterion. The policy is a measurable rule that maps visible history to actions, target choices, and stopping. A policy can be aligned with the agent model yet be evaluated under a different generative process; that mismatch is part of the declared evaluated design, not an implementation detail.

This separation also fixes the role of simulation. The investigator declares a scenario, policy, and replication plan; the embedded agent sees only visible history, while the evaluator retains hidden truth only to score a completed trace. A random seed controls a reproducible approximation to an operating characteristic; it is not an additional scientific condition. Seed schedules, worker counts, and checkpoint cursors are therefore recorded as computational provenance, while the model, process, setting, and policy define the scientific estimand.

3.2 The estimand: power as a conditional suite functional

The suite makes the conditioning set explicit. A `GenerativeModelSpec` describes the agent's state, observation, prior, and action-conditioned model; a `GenerativeProcessSpec` describes the evaluator-side hidden-state transitions and action-conditioned observations; and a `SettingSpec` declares the hypothesis family, alpha, correction, default stopping rule, cost budget, and validity boundary. These three objects are the scientific components $s = (M_A, P_N, T)$ of the hashable `PowerScenario`; the serialized hash also binds scenario identity and claim metadata. An action policy π is recorded per result row. When no policy is supplied, execution uses the hashed default stopping rule in T; action-surface comparisons explicitly record their executed policy. The investigator therefore evaluates $d = (s, \pi)$ rather than hiding a separately supplied surface policy inside the scenario hash.

For the action-loop surface, the reported power estimand is the scenario- and-policy-indexed expected replication-level true-positive rate

$$\theta_{\mathrm{TPR}}(d) = \mathbb{E}_{P_N, U_\pi}\left[\frac{|\mathcal{R}_d \cap \mathcal{H}_1(P_N)|}{\max\{1, |\mathcal{H}_1(P_N)|\}}\right], \qquad d = (s, \pi). \tag{1}$$

Here P_N supplies the declared process randomness, while U_π denotes any policy and trace-level randomization used by π. The expectation is over independent replications conditional on the fixed model, setting, and policy contract. Finite independent replications produce its sample-mean estimator, with Monte-Carlo standard error (MCSE) describing simulation precision. The distinct event "at least one true target is rejected" is recorded separately as `any_true_rejection_rate`; it is not silently substituted for mean TPR. Changing any conditioning object changes the estimand. The suite therefore reports scenario-indexed power, observed mean FDP (the finite-simulation FDR estimate), FWER, stopping time, observation count, target-selection rate, and cost rather than treating power as an invariant property of an agent or task environment.

For m hypotheses, let $\mathcal{H}_0$ and $\mathcal{H}_1$ denote the true null and alternative sets. If $\mathcal{R}$ is the rejection set, then $R = |\mathcal{R}|$, $V = |\mathcal{R} \cap \mathcal{H}_0|$, and $S = |\mathcal{R} \cap \mathcal{H}_1|$. The true-positive rate (TPR) and other realized quantities are

$$\mathrm{FDP} = \frac{V}{\max(R, 1)}, \qquad \mathrm{TPR} = \frac{S}{\max(|\mathcal{H}_1|, 1)}, \qquad \text{FWER event} = \mathbf{1}\{V \geq 1\}. \tag{2}$$

Their Monte-Carlo averages estimate FDR, conditional mean-TPR power, and FWER respectively. The denominator conventions make empty-rejection and global-null cases explicit rather than silently dropping them. They are replication-level estimands: a ratio of expected counts is not substituted for the expected FDP, and a mean stopping time is not substituted for a validity guarantee.

3.3 Fixed-horizon correction formalisms

All procedures accept a one-dimensional p-value vector, validate its range, and return adjusted p-values in input order. The rejection mask is always defined by adjusted p-value $\leq \alpha$.

- **Bonferroni** uses $p_i \leq \alpha/m$ and controls FWER under valid marginal p-values without a dependence assumption.
- **Šidák** uses $p_i \leq 1 - (1-\alpha)^{1/m}$ under independence.
- **Holm** is the step-down FWER procedure with critical values $\alpha/(m-i+1)$.
- **Hochberg** is the step-up FWER procedure under its stated dependence conditions [Hochberg, 1988].
- **BH** rejects through the largest k satisfying $p_{(k)} \leq (k/m)\alpha$ and reports reverse-cumulative-minimum q-values [Benjamini and Hochberg, 1995]. Under the independent continuous global null, its FDR is the event probability $P(R > 0) = \alpha$; the expected false-rejection count $E[R]$ is a separate quantity computed from the finite ordered-threshold distribution.
- **Benjamini–Yekutieli (BY)** scales BH by $c(m) = \sum_{i=1}^{m} 1/i$ for an arbitrary-dependence FDR guarantee under the procedure's validity assumptions [Benjamini and Yekutieli, 2001].
- **Storey q-values** plug in $\hat{\pi}_0 = \min\{1, \#\{p_i > \lambda\}/[m(1-\lambda)]\}$ [Storey, 2002]. The finite-sample estimate is reported, not treated as an exact guarantee.
- **Adaptive BH** implements the two-stage Benjamini–Krieger–Yekutieli construction [Benjamini et al., 2006].
- **Weighted BH** applies fixed, positive, normalized hypothesis weights, as in the weighted multiple-testing extension of Benjamini and Hochberg [Benjamini and Hochberg, 1997]. Its interpretation is conditional on the declared fixed weights and the usual independence or PRDS assumptions; weights are not estimated from the same p-values in this release track. A weight reallocates the testing budget; it does not create information or make a misspecified p-value valid.

The implementation also exposes a dispatcher and a structural staircase check. Cross-checks include the BH q-values against `scipy.stats.false_discovery_control`, permutation equivariance, and the expected ordering of conservative procedures.

3.3.1 Adaptive-FDR audit: separate calibration from power

The adaptive comparison is intentionally decomposed into distinct estimands. The Storey power comparison uses common random-number replication streams and reports the paired difference in average TPR with its own Monte-Carlo standard error. The BH and Storey FDR checks instead use the replication-level FDP samples, with separate standard errors and separate finite-band ceilings. The certificate passes only when the power comparison and both FDR checks satisfy their declared inequalities. Thus a power gain cannot compensate for an FDR shortfall.

This distinction is important because the same-vector Storey plug-in estimates the null proportion from the same p-value vector used for rejection. That construction is retained as a diagnostic comparison, not treated as a universal finite-sample guarantee. The conservative fixed-lambda and independent split-confirmation implementations provide stricter alternatives whose assumptions are recorded separately. The audit is designed to expose a small but reproducible calibration discrepancy when increased replication precision resolves it; it does not convert a finite simulation into a proof.

The finite band is a Monte-Carlo consistency rule around the declared target, not an inferential interval for an unknown universal FDR. A point estimate may lie above the nominal target while remaining inside a wider low-replication band; the publication artifact therefore retains both the point estimate and the band ceiling. The full metric decomposition is generated in the supplementary certificate table.

3.4 Analytic operating-characteristic references

For a one-sample z-test with standardized effect d and noncentrality $\delta = \sqrt{n}\, d$,

$$\text{Power}_1 = 1 - \Phi(z_{1-\alpha} - \delta), \tag{3}$$

and the two-sided calculation adds the lower tail. The t-test uses the noncentral t distribution with $\delta = d\sqrt{n}$ and one fewer degree of freedom than n for a one-sample test, or $\delta = d\sqrt{n/2}$ and degrees of freedom two times n less two for equal-sized groups. `required_sample_size` uses a monotone integer search. A stable normal-tail fallback is used only when SciPy returns a non-finite noncentral-t tail at very large noncentrality.

For independent two-groups p-values, the asymptotic BH threshold t^* is the largest solution of

$$t^* = \alpha\left[\pi_0 + (1-\pi_0)G(t^*)\right], \qquad G(t) = \Phi(z_t + \text{effect}), \quad \Phi(z_t) = t, \tag{4}$$

and asymptotic average power is $G(t^*)$ [Genovese and Wasserman, 2002]. This is a model-based asymptotic comparison, not a replacement for the finite-sample simulation.

3.5 Executable generative-process regimes

Each replication draws

$$z_i = \sqrt{\rho}F + \sqrt{1-\rho}\, e_i + \mu_i, \qquad F, e_i \sim \mathcal{N}(0,1), \tag{5}$$

where μ_i is zero for true nulls and the configured effect for alternatives. One- or two-sided Gaussian p-values are then corrected and scored against the known alternative mask. The covariance generator validates positive semidefiniteness and audits marginal calibration separately for independent, negative-equicorrelation, block, and factor regimes. The positive-factor construction is the declared positive-dependence setting for the one-sided certificate; it is not evidence for arbitrary dependence or for two-sided BH control. Negative and block regimes are stress comparisons, not theorem extensions. The Dobriban/BH result is cited for that boundary rather than re-claimed here [Dobriban, 2026]. The publicly documented finite protocol is imported by an offline adapter in `active_inference_power.protocols`; its local Monte Carlo retains the stated model and stratification but remains a stress track. The external outward-rounded Arb certificate is a separate theorem artifact and is not reproduced by this package [Dobriban, 2026].

3.6 Belief updates, calibrated evidence, and the p-value boundary

The hidden state has levels H_0 and H_1, observations are `low` and `high`, and $r = \Pr(\text{high} \mid H_1) = \Pr(\text{low} \mid H_0)$ is the channel reliability. The real `pymdp.agent.Agent` uses an identity transition, so the posterior at one step becomes the prior at the next. With n_h high and n_l low observations, posterior odds are

$$\frac{P(H_1 \mid y)}{P(H_0 \mid y)} = \frac{P(H_1)}{P(H_0)} \left(\frac{r}{1-r}\right)^{n_h - n_l}. \tag{6}$$

The belief threshold is intentionally separated from multiplicity control. Under H_0, $n_h \sim \text{Binomial}(T, 1-r)$. For $r > 0.5$, `evidence_pvalue` computes the upper-tail Binomial p-value; for the supported anti-correlated channel $r < 0.5$, it uses the corresponding lower tail. The experiment compares the posterior-threshold decision with BH applied to those calibrated p-values on identical evidence streams. This is a fixed-horizon calibration step, not a test-martingale, e-process, or optional-stopping construction [Shafer et al., 2011, Grünwald et al., 2024]. The active-inference literature motivates the generative-model interpretation, but the statistical validity claim comes only from the stated Binomial null and the downstream multiple-testing procedure [Friston et al., 2017, Da Costa et al., 2020].

3.7 Action-in-the-loop filtration and policy evaluation

The reference action environment has two targets, binary truth, latent contexts, action-conditioned sensing quality, a context-changing action, unequal action costs, and a stop action. Policies include fixed horizon, posterior threshold, expected information gain, cost-normalized information gain, posterior sampling, adaptive target selection, and e-process stopping. The real `pymdp` posterior is checked against a closed-form Bayesian update at every observed step.

The policy receives only its visible history. Hidden process states are retained in a separate evaluation trace for scoring and never enter action selection, target choice, or stopping. Target-selection p-values are fed chronologically to LORD++ or SAFFRON only as conditional diagnostics unless the declared predictable-selection and conditional-super-uniformity assumptions hold. This separation prevents a posterior decision, an environment shift, or an evaluator-only truth label from being mislabeled as family-level evidence. Each replication retains true and false rejection counts, target-specific rejection rates, cost and stopping burdens, power per unit cost, and policy-paired contrasts. Power remains the mean replication-level true-positive rate; FDR remains the mean replication-level FDP.

The interface contract is summarized in fig. 5. Reading from left to right, the model, process, and setting define the hashed scenario; the separately recorded policy completes the evaluated design, and evaluator- only truth scores replicated traces. This decomposition is the organizing object for the roadmap, not an assertion that every downstream class is formally valid.

3.7.1 Split-stream confirmation and sequential evidence

The formal adaptive-selection protocol is deliberately narrow. The selection policy is a measurable function of a visible action/observation filtration and does not receive hidden states or confirmation observations. It selects one target from the family. A fresh, independent confirmation stream then evaluates only that target with the implemented fixed-horizon Binomial p-value. Under a conditional-null law that is super-uniform given the selection history, this separates discovery information from confirmation information. An e-value confirmation variant would require its own implemented conditionally valid e-factor and filtration contract; it is not supplied by this protocol. Split streams alone do not rescue a misspecified likelihood, truth-changing process, or invalid stopping rule; they make the required conditional law inspectable. They also do not make chronological same-stream selection or post-hoc minimum-p selection valid.

The static confirmation channel is declared separately from the static e-process optional-stopping contrast. The former tests one selected target on fresh confirmation data; the latter concerns time-uniform evidence along a single declared trace. Their shared use of a static process does not make their filtrations, estimands, or validity labels interchangeable.

The sequential module exposes binary likelihood-ratio e-processes, predictable alternative schedules, Beta-mixture confidence-sequence inversion, and Levels based On Number of Discoveries (LOND) e-value accounting. Each path is serialized with either likelihood-ratio e-process evidence E_t, LORD++/SAFFRON alpha-wealth accounting W_t^α, or e-LOND's cumulative nominal-level allocation $\sum_{i\le t}\alpha_i$, together with the applicable test levels, e-value thresholds, crossing or stopping time, and filtration assumptions. For e-LOND, the nominal level is α_t, the corresponding e-value condition is $E_t \ge 1/\alpha_t$, and recycled levels can make the cumulative allocation exceed α; it is neither alpha-wealth nor a remaining budget. LORD++ and SAFFRON are online p-value procedures whose levels depend on prior history [Javanmard and Montanari, 2018, Ramdas et al., 2017, 2018]; e-LOND uses predictable e-value accounting [Xu and Ramdas, 2024]. They are reported as finite traces with uncertainty accounting. Their theorem-level guarantees remain conditional on the published independence, conditional p-value super-uniformity, or conditional e-factor mean-at-most-one assumptions and on the arrival filtration actually used by the policy.

Dynamic process schedules are explicit contracts rather than implicit array indexing: each schedule declares `hold_last`, `cycle`, or `strict` behavior outside its stated time support. This removes a source of silent simulation variation while leaving nonstationary and action-dependent regimes in the diagnostic or stress-test evidence classes unless their inferential assumptions are separately established.

3.7.2 Held-out policy learning

The learned-policy implementation is a finite tabular search over visible expected information gain, target uncertainty, and action cost. Candidate parameters are selected using child streams reserved for training, then frozen and hashed before evaluation on disjoint child streams. The artifact also hashes the search configuration and records the candidate count and parameter dimension, so the finite search burden is part of the provenance rather than an unstated degree of freedom. This makes leakage auditable and keeps the estimand explicit. The search is intentionally not described as optimization of a universal utility function: a held-out policy comparison is conditional design evidence for the declared scenario.

3.8 Replication accounting and uncertainty

For every bounded replication-level estimand, the result artifact retains the replication count, standard error, a Student-t precision interval, and a distribution-free Hoeffding radius. The implementation uses Wilson or exact binomial intervals for indicated event rates, paired intervals from within-replication differences, a configured Hoeffding planning floor, and a configured Dvoretzky–Kiefer–Wolfowitz (DKW) band for its independent global-null cumulative distribution function (CDF) audit. These are declared finite-simulation reporting conventions [Morris et al., 2019], not claims that those displays widen a theorem's assumptions, repair a misspecified process, or create arbitrary-dependence evidence.

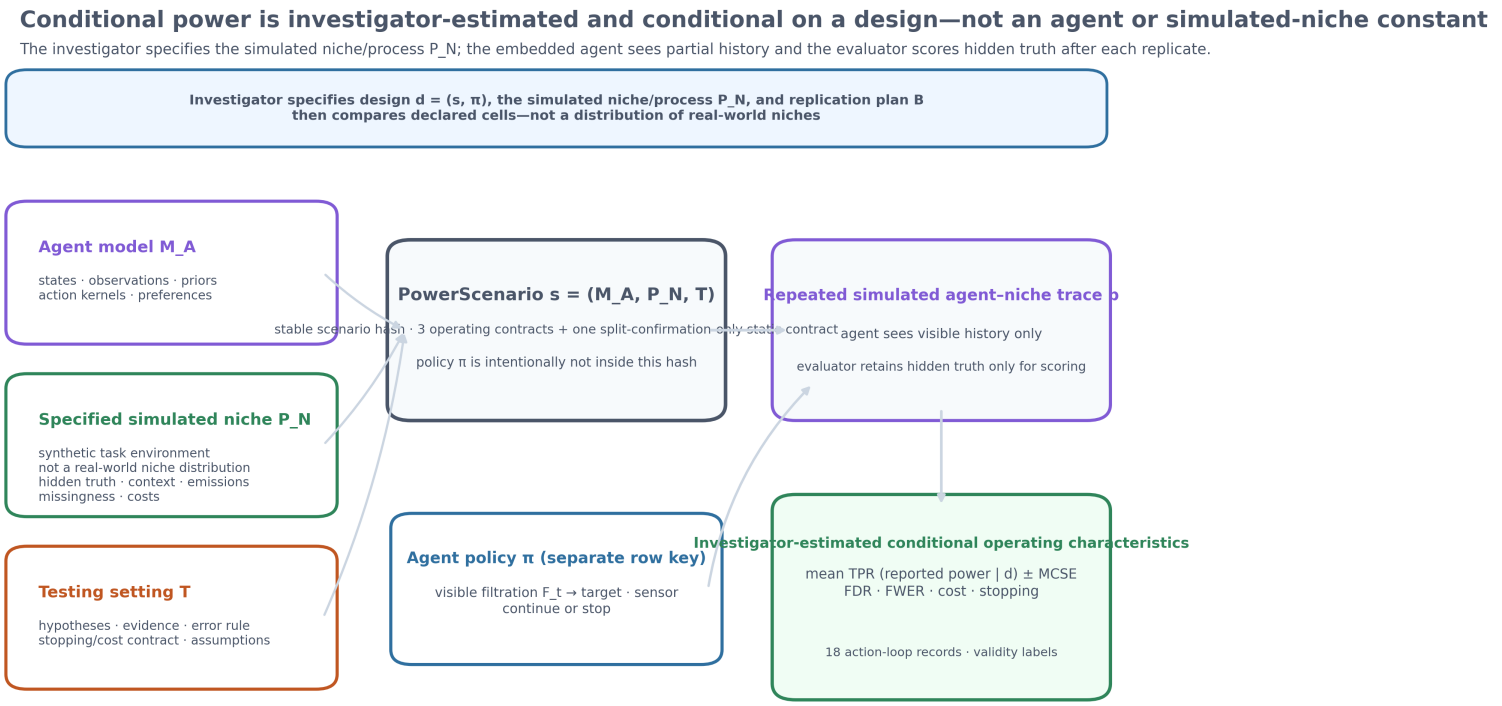


Figure 5: Who specifies the design and simulated niche, what can the agent observe, and what does the investigator estimate? Investigator-facing design decomposition: a model, specified simulated niche/process, and testing setting form a hashable scenario; a separate agent policy completes it. Evaluator-only truth scores conditional replication-mean true-positive rate, error, cost, and stopping. Scenario plus policy defines the estimand; neither the agent nor the simulated niche has intrinsic power. The diagram makes conditioning explicit; it does not establish validity for any unimplemented scenario.

3.9 Source-level workload accounting and isolated timing diagnostics

The suite reports computational feasibility on a separate evidence layer. For a declared input shape, Q_{comp} is a deterministic source-work proxy: it counts dominant passes, history scans, and declared action-loop calls in the current code path. The same ledger separately records retained draw elements as implementation-specific retention accounting. For example, the single-pass correction proxy is $q_{\text{linear}}(m) = m$, while the sort-dominant proxy is $q_{\text{rank}}(m) = m \max\{1, \lceil \log_2(m) \rceil\}$. For online traces, the proxy covers both the recurrence and the package's serialized-trace validation replay; it does not pretend that a single procedure update is the whole implementation path. The binary e-process and e-LOND have coincident displayed linear proxies, so the figure uses one combined curve label while the analytical rows retain their different evidence and accounting roles. For named action-loop categories j, the ledger records the permitted step bound $Q_{\text{action}} = \sum_j N_j H_j$, where N_j is the declared trace-call count and H_j is that category's maximum permitted horizon. These expressions are accounting choices tied to this source version; they are not lower bounds, processor-instruction counts, or complexity statements about a statistical method in the abstract.

The compact ledger notation is local to this accounting: m is family size, B is the main independent-replication count, K is the number of compared procedures, D is the number of declared dependence regimes, T is a sequential or online horizon, and R is the number of prior rejections in an online history. Let $w_{\text{corr}}(m)$ denote the source-work cost of the particular selected correction implementation, rather than an undefined universal operation. The action row uses the already defined category-specific N_j and H_j, not a second set of opaque symbols.

This layer does not reuse any statistical quantity. Simulated resource cost C remains the scenario-defined burden accumulated by actions; E_t remains likelihood-ratio e-process evidence; W_t^{α} remains LORD++/SAFFRON procedural alpha-wealth; and e-LOND's $\sum_{i \le t} \alpha_i$ remains a cumulative nominal-level allocation. Q_{comp} is dimensionless and does not measure any of them. The release artifact contains the deterministic ledger and explicitly excludes elapsed-time fields, so replication accounting continues to describe statistical precision rather than computational speed.

Host-specific elapsed time is inspected only in a separate local diagnostic. It records configured warmups, repeated raw wall and process time samples, and the runtime environment while keeping input construction outside the timed region. The use of declared warmups and repeated samples follows benchmark reporting discipline narrowly [Georges et al., 2007]. The resulting τ_{wall} is an observation of one host and runtime configuration, not a cross-machine benchmark, a release criterion, or evidence about power, calibration, or validity.

3.10 Finite-simulation certificates and claim boundaries

Each certificate records an estimate, the analytic or procedural bound, a Monte-Carlo standard error, the seed, configuration, and a Boolean pass flag. The predeclared pass rule allows 3 Monte-Carlo standard errors for sampling noise. A passing certificate therefore means "consistent with the stated bound under this finite simulation," not "proved by simulation." The release finalizer is fail-closed: if a predeclared certificate fails, the run and manuscript may still be hydrated for review, but the manifest cannot be marked complete. This makes a negative result a preserved method-review object rather than an operational failure that disappears before publication.

3.11 Reporting hierarchy and interpretation rules

The analysis uses a four-level reporting hierarchy. A *formal* result is one whose stated theorem assumptions are part of the declared scenario-and-policy (design) contract and whose implementation has a direct semantic test. A *conditional* result is an operating characteristic for a fully specified model, process, setting, and policy, but not a theorem over a broader class. A *diagnostic* result probes calibration, filtration behavior, or a deliberate contrast within a declared simulation. A *stress* result changes dependence, missingness, dynamics, or external data conditions to locate where the conditional interpretation may fail. These labels are evidence metadata, not prose decoration: they are serialized with result blocks, figure registries, captions, and claim-ledger entries.

The hierarchy prevents three common category errors. A high posterior is not a family-level rejection; a low Monte-Carlo error is not a validity theorem; and an action policy that performs well in one process is not an optimal or transportable policy. The same discipline applies to the manuscript: every reported number is attached to an estimand, replication unit, uncertainty rule, scenario hash, and claim boundary. The findable, accessible, interoperable, and reusable (FAIR) research-object framework motivates the artifact lineage, while the statistical contract determines what the lineage can and cannot justify [Wilkinson et al., 2016].

3.12 Why the action loop changes the estimand

In a fixed design, the process distribution can be written before the data are collected. In an action loop, the policy is part of the data-generating map: the next action changes the observation kernel, may change the latent context, and can determine whether another observation exists. Consequently, two policies evaluated under the same initial state do not generally generate paired observations. Common random numbers pair the exogenous replication stream for a contrast, but they do not make the resulting trajectories identical or erase the policy-induced process difference. The reported paired quantity is therefore a policy contrast under a shared seed schedule, not a causal effect without additional design assumptions.

The suite reports this distinction directly. Power is the mean of the declared replication-level true-positive estimand; FDR is the mean FDP; stopping burden and cost are means over completed traces; and power per cost is the descriptive mean of replication-level TPR/C scores (recorded as zero for a zero-cost trace), not a ratio of scenario summaries. No ratio of pooled counts is substituted

for mean FDP, and no cost-normalized score is presented as a utility function. This separation keeps inferential validity, resource efficiency, and agent belief quality as related but non-interchangeable outcomes.

4 Results

The results below are generated by the result generator from the experiment configuration. The script writes the JSON result artifact, the figure set, a lightweight dashboard, and a manifest containing the configuration hash. Manuscript variables are hydrated only after those artifacts exist.

4.1 Reading the evidence

This section reports the declared experiment; it is not a ranking of agents in the abstract. The generated bundle is `release candidate bound to the current source`: the run ledger is `passed` and the predeclared certificate gate is `passed`. A review-only bundle can contain complete finite-simulation tables and figures, but it cannot be represented as a passed release. This distinction matters for adaptive procedures, where a larger run can expose a finite-simulation shortfall that a smaller run did not resolve. The reader-facing status is deliberately fixed when renderer inputs are hydrated; terminal publication completion is instead declared by the exact manifest and its passing review receipt after rendering, never by a prose token that could be stale across that finalization boundary.

For each result, read the estimand first, then the uncertainty convention, then the validity label. Point estimates describe replication means under one scenario. Precision intervals describe Monte-Carlo uncertainty under the replication schedule. Neither changes the null, dependence, filtration, or process assumptions that define the evidence class.

Unless otherwise stated, the comparison uses $m = 200$, $\pi_0 = 0.8$, one-sided tests at $\alpha = 0.05$, effect 3.0, and 10000 replications. The configured comparison contains 9 procedures.

Tables report point estimates from the seeded replications. The figures add error bars equal to the configured ± 3 MC SE band, computed from the replication-level outcomes where available. These intervals describe simulation precision; they are not confidence intervals for a universal error rate guarantee and do not repair a misspecified null or dependence model.

4.2 Error-control comparisons under the declared design

The FWER-oriented procedures reach power from 0.315 to 0.321 and reject at least 12.6 hypotheses on average in the configured design. The FDR-oriented procedures reach power from 0.430 to 0.749. This is the expected power/error trade-off, not a claim that one family dominates for every data-generating process.

Table 2: Point estimates from the paired Monte-Carlo comparison; the corresponding figure shows ± 3 MC SE.

Procedure	FDR	Power	FWER	Mean rejections
Bonferroni	0.0031	0.315	0.039	12.6
Šidák	0.0031	0.317	0.040	12.7
Holm	0.0032	0.321	0.041	12.9
Hochberg	0.0032	0.321	0.041	12.9
Benjamini–Hochberg	0.0401	0.715	0.690	29.8
Benjamini–Yekutieli	0.0070	0.430	0.122	17.3
Storey q-value	0.0508	0.749	0.782	31.6
Adaptive BH (BKY)	0.0472	0.738	0.754	31.0
Weighted BH	0.0401	0.707	0.686	29.5

The BY procedure is deliberately conservative under the positive-factor setting: its FDR is 0.007 and its power is 0.430. Storey's finite-sample FDR is 0.051 in the same comparison; its dedicated certificate reports the separate power gain of 0.073. The preregistered independent split-stream Storey calibration uses $\lambda = 0.50$ and a 99% one-sided null-count confidence level; its finite diagnostic FDR is 0.042. Adaptive procedures should therefore be read with their calibration assumptions, not as free power. Under the canonical release policy, every declared certificate, including the same-vector Storey power-gain record, is terminal. If that all-terminal gate is not passed, the observed adaptive comparison remains a method-review result rather than a release claim. The isolated v2 campaign has a separate frozen policy: it keeps independent split confirmation terminal and retains the same-vector Storey record as a visible diagnostic. That campaign-specific completion does not revise the canonical all-terminal release policy or create a publication claim.

4.2.1 Adaptive-FDR precision audit: power gain does not certify calibration

The dedicated Storey audit uses 5,000 replications in the current generated bundle. It separates the paired power comparison from the FDR calibration checks. Storey power is 0.90891 with standard error (SE) 0.000472, and its paired gain over BH is 0.07339 with SE 0.000436. The power component is therefore a comparison of operating characteristics, not a validity certificate.

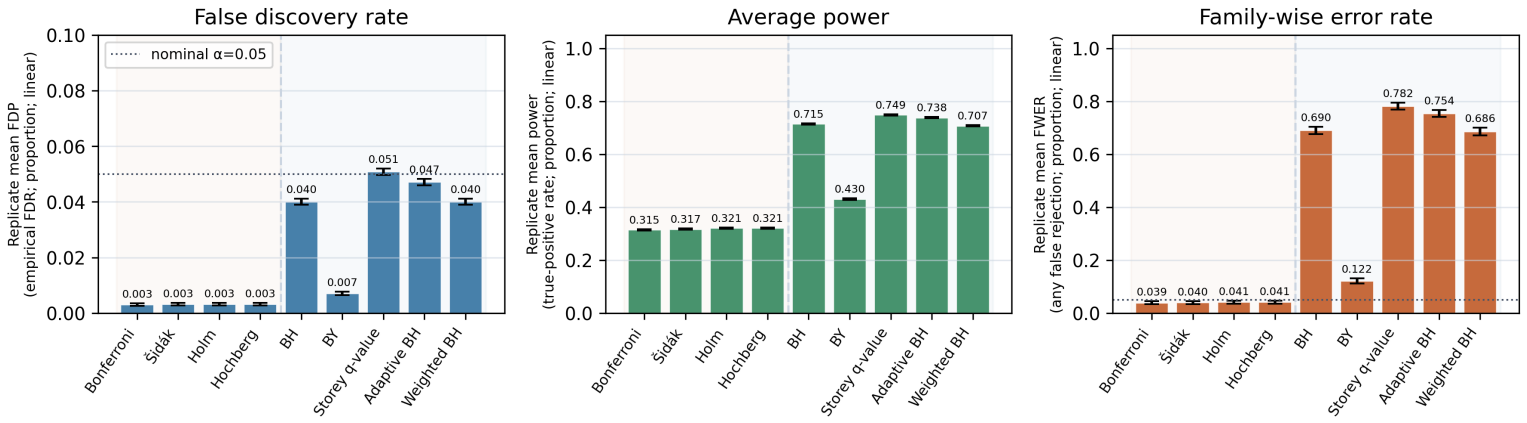


Figure 6: Within the declared two-groups scenario, how do correction procedures trade false discoveries, true positives, and any false rejection? Empirical FDR, average power, and FWER for 9 configured correction procedures under the seeded two-groups model; pale orange shading left of the dashed divider identifies FWER-oriented methods and pale blue shading right identifies FDR-oriented methods. Bars show means with ±3 MC SE across 10,000 replications. The dotted line marks the nominal alpha level (0.05). The panels make the configured error-control trade-offs inspectable within one finite simulation, not universally ranked. One seeded two-groups design; guarantees remain assumption-specific.

The FDR component reports Storey FDR 0.05092, with SE 0.000337 and finite-band ceiling 0.05101. Its point excess over the nominal target is 0.00092 and its standardized excess is 2.74 Monte-Carlo SE. The decomposed Storey FDR gate is passed. BH's corresponding FDR estimate is 0.02520, with SE 0.000242 and ceiling 0.02573.

Table 3: The adaptive-FDR audit separates FDR precision from power precision. The finite-band decision is not a theorem. In any bundle with a failed Storey row, the row is retained as method evidence and blocks promotion under that bundle's declared policy rather than being removed from the narrative. The canonical release classifies the same-vector record as terminal; only the separately frozen v2 campaign classifies it as diagnostic.

Metric	Estimate	MC SE	Finite-band ceiling / reference	Gate
BH FDR	0.0252	0.000242	0.0257	passed
Storey FDR	0.0509	0.000337	0.0510	passed
BH power	0.8355	0.000608	reference	reference
Storey power	0.9089	0.000472	0.8355	passed
Paired power gain	0.0734	0.000436	0.0000	passed

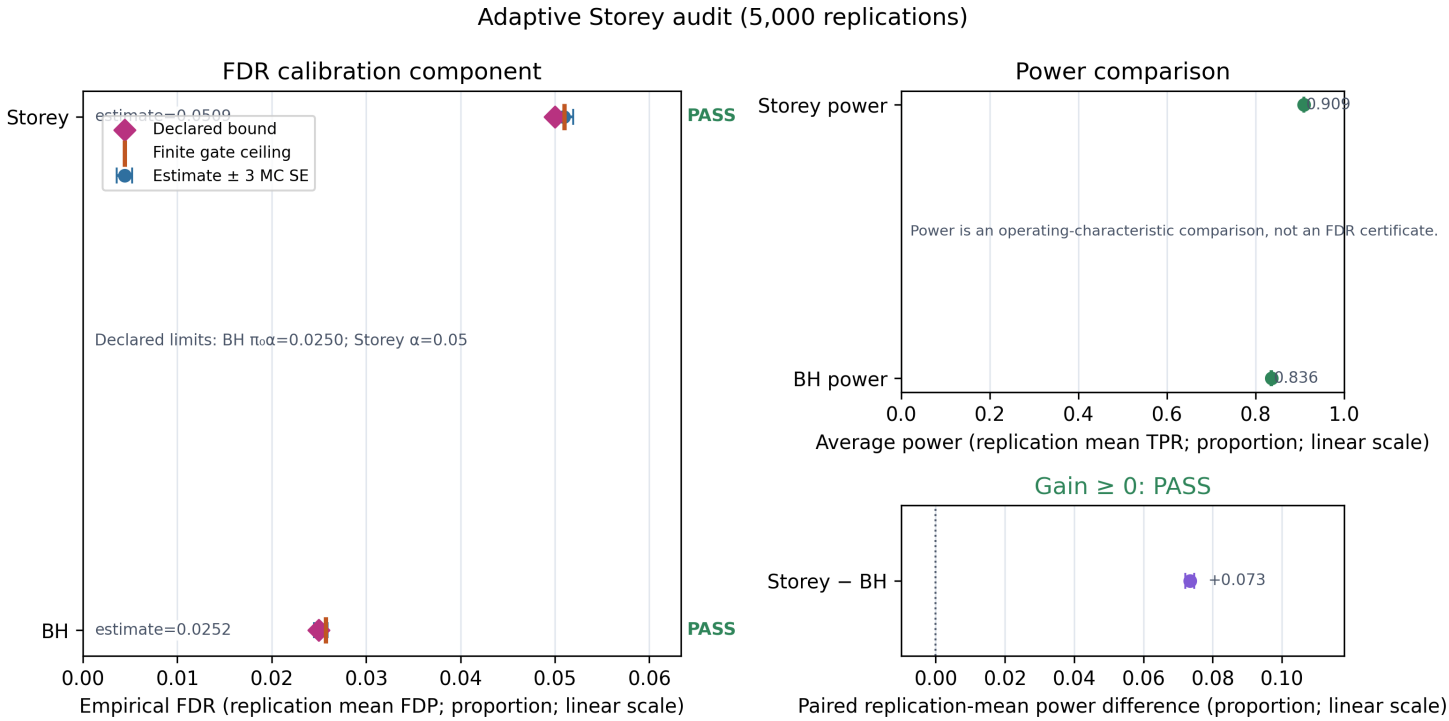


Figure 7: How do the adaptive FDR and paired power components compare with their separate declared gate criteria? Metric-separated adaptive-FDR audit from 5,000 certificate replications. The FDR panel shows BH and Storey estimates with metric-specific Monte-Carlo intervals, declared bounds, finite gate ceilings, and gate status; the power panel shows BH and Storey power and the paired Storey-minus-BH gain. Separating FDR control checks from power gain prevents an apparent power improvement from being mistaken for a validity certificate. Finite-simulation audit of the declared design; a passing or failing gate is not a theorem, and same-stream adaptive calibration remains assumption-dependent.

The audit is grounded in the distinction between Storey's null-proportion estimation and adaptive linear step-up procedures [Storey, 2002, Benjamini et al., 2006]. It also follows the proof-oriented discipline that FDR claims must expose the self-consistency and dependence conditions connecting an algorithm to its guarantee [Blanchard and Roquain, 2008].

This table makes the profile-specific policy boundary inspectable. In a lower-replication canonical bundle with an all-terminal policy, a point estimate slightly above the nominal FDR can remain inside its wider finite band; in an isolated precision or campaign receipt, the same operating behavior can become a failed gate when the band narrows. The correct interpretation is not that replication changes the procedure, but that more replications resolve whether the observed behavior is consistent with the predeclared finite check. The split-confirmation result remains the primary adaptive route because it separates selection information from confirmation evidence. The canonical release requires the same-vector Storey record to pass, whereas only the frozen v2 campaign retains it as diagnostic; neither policy supplies a theorem.

4.3 Effect-size operating characteristics

BH average power rises to 0.952 at the largest configured effect while its realized FDR remains the quantity monitored by the certificate and the table below. The curve is a finite-sample operating characteristic; the analytic Genovese–Wasserman calculation in sec. 3.4 is an asymptotic reference.

Table 4: Point estimates of BH power and realized FDR across the configured effect grid; the figure shows ±3 MC SE.

Effect	BH power	BH FDR
0.0	0.000	0.039
0.5	0.002	0.039
1.0	0.009	0.040
1.5	0.049	0.038
2.0	0.202	0.039
2.5	0.467	0.040
3.0	0.714	0.040
3.5	0.872	0.040
4.0	0.952	0.040

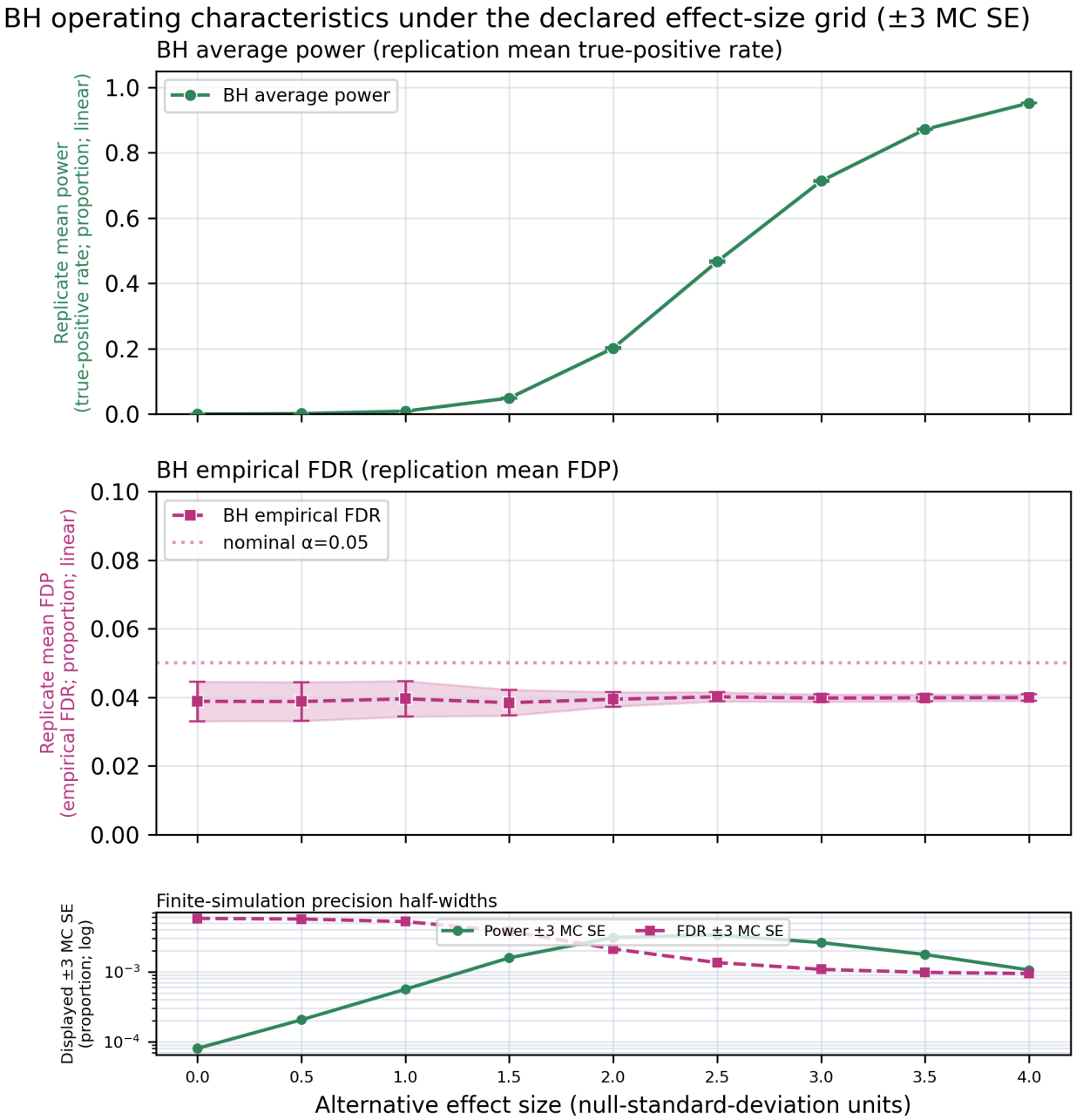


Figure 8: For the declared fixed-horizon design, how do BH power and false-discovery summaries change across the effect-size grid? Three aligned BH panels share the configured effect-size axis: power, empirical FDR, and precision half-widths. Shaded ribbons, on-top capped error marks, and the lower log-scale half-width panel show ±3 MC SE across 10,000 replications. The panels support a design-specific comparison, not a population curve or universal monotonicity. Finite-simulation operating characteristic; no universal monotonicity or guarantee.

4.4 Dependence regimes: observed behavior and validity boundary

Across the configured dependence grid, including the negative-equicorrelation stress point and the positive-factor range ending at $\rho = 0.75$, the empirical BH estimates remain inside the configured finite-simulation band while BY remains more conservative. This is consistent with the stated positive-dependence regime; it is not a new PRDS proof. The experiment does not test the arbitrary-dependence guarantee of BY by simulation. The imported Dobriban/BH protocol adapter provides a separate two-sided non-PRDS stress lane; it does not reproduce the external Arb theorem and therefore cannot upgrade this package's claims [Dobriban, 2026].

The dependence panel should be read as a map of conditional behavior. The negative and block cells probe sensitivity to covariance structure, while the factor cells are the declared calibration setting for the one-sided BH certificate. Similar curves across cells would not establish exchangeability, PRDS, or arbitrary-dependence validity; different curves would instead be evidence that the scenario is part of the estimand.

Table 5: Point estimates under the declared dependence regimes; the figure shows ±3 MC SE and does not establish behavior under arbitrary dependence.

Regime	ρ	BH FDR	BH power	BY FDR	BY power
negative equicorrelation	-0.00	0.0398	0.715	0.0068	0.432
independent	0.00	0.0398	0.715	0.0067	0.430
positive factor	0.25	0.0375	0.691	0.0065	0.421
positive factor	0.50	0.0335	0.673	0.0061	0.417
positive factor	0.75	0.0277	0.661	0.0050	0.420
block (size=10)	0.50	0.0387	0.707	0.0064	0.426

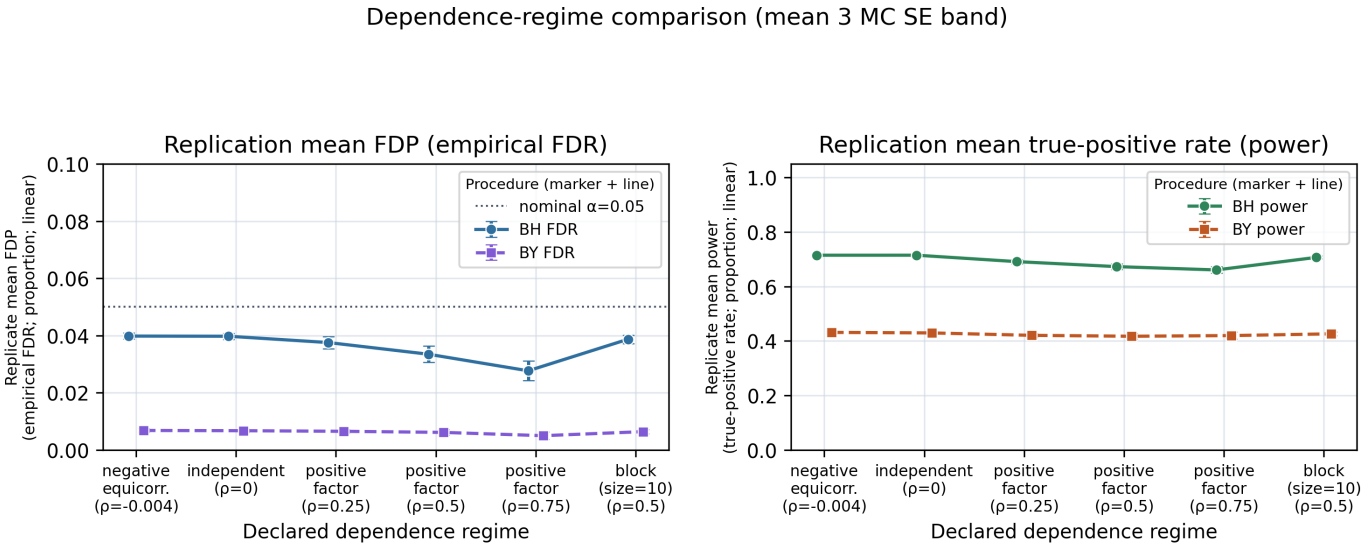


Figure 9: How do BH and BY behave across the named covariance stress regimes? BH circles/solid lines and BY squares/dashed lines show empirical FDR and average power across the 6 declared covariance regimes (including ρ=-0.004 and ρ=0.75); error bars show ±3 MC SE. This is a finite-simulation stress comparison, not arbitrary-dependence evidence. The comparison makes the declared dependence sensitivity visible but does not generalize to arbitrary dependence. Declared covariance stress does not establish arbitrary-dependence behavior.

4.5 Seeded certificates: finite consistency checks

The certificate table reports a point estimate, its comparison bound, the direction of that bound, and a pass/fail decision with the configured ±3 MC SE band. Under the canonical release policy, all configured certificates must pass before the manifest is considered publishable. That all-terminal rule is distinct from the frozen v2 campaign's campaign-only diagnostic partition and cannot be relaxed after observing a result. A pass means consistency with the selected finite simulation and declared assumptions; it is not a substitute for the theorem or for a proof under broader conditions.

Table 6: Seeded certificate estimates and explicit finite-simulation bounds; upper bounds are shown with ≤ and the Storey power comparison with >=.

Certificate	Estimate	Bound	Passes
BH FDR control	0.0400	≤ 0.0400 ($\pi_0\alpha$)	✓

Certificate	Estimate	Bound	Passes
Bonferroni FWER	0.0498	≤ 0.0500 (α)	✓
BH under PRDS	0.0346	≤ 0.0400	✓
Storey power gain	0.9089	>= 0.8355 (BH power)	✓
Split-stream Storey calibration	0.0424	≤ 0.0500 (α)	✓

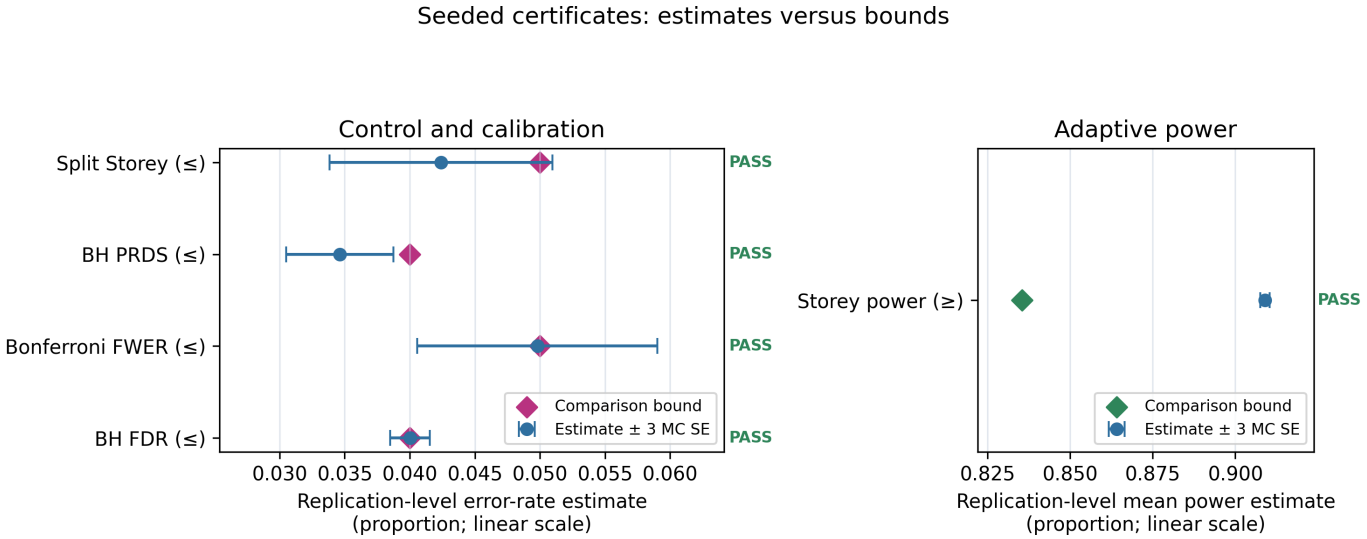


Figure 10: Which seeded certificates meet their declared replication-level error-rate or power comparison bounds? Seeded certificate estimates from 5,000 replications per check with ±3 MC SE intervals; diamonds mark comparison bounds. Control checks, the same-vector Storey power comparison, and the independent split-stream Storey calibration are separated by panel and label. Intervals quantify finite-simulation precision and are not proof certificates by themselves. The forest plot separates finite-simulation agreement with a bound from theorem-level validity. Pass means finite-simulation consistency with a declared bound, not a theorem.

4.6 Belief decisions versus family-level evidence

The active-inference comparison uses 1000 replications per evidence regime and identical streams for the posterior-threshold decision and the BH-calibrated evidence rule. Under weak evidence, the posterior-threshold policy has FDR 0.153 and FWER 0.366; the calibrated policy has FDR 0.000, power 0.000, and FWER 0.000. Strong evidence yields posterior-threshold FDR 0.002 and power 0.997. These values illustrate a calibration gap; they do not establish a universal threshold for active-inference agents.

The comparison is diagnostic of the interface between belief and evidence. The posterior rule answers whether the agent assigns high probability to a state; the BH rule answers whether a calibrated evidence statistic supports family-level rejection. Agreement or disagreement is scientifically useful only after the likelihood, horizon, null law, and family procedure are held fixed and named.

Table 7: Point estimates for posterior-threshold decisions versus BH on calibrated evidence p-values; the figure shows ±3 MC SE.

Regime	Policy	FDR	Power	FWER
Strong (r=0.80, T=20)	Posterior threshold ≥ 0.9	0.002	0.997	0.091
Strong	BH on evidence p-values	0.036	1.000	0.784
Weak (r=0.62, T=6)	Posterior threshold ≥ 0.9	**0.153**	0.056	0.366
Weak	BH on evidence p-values	0.000	0.000	0.000

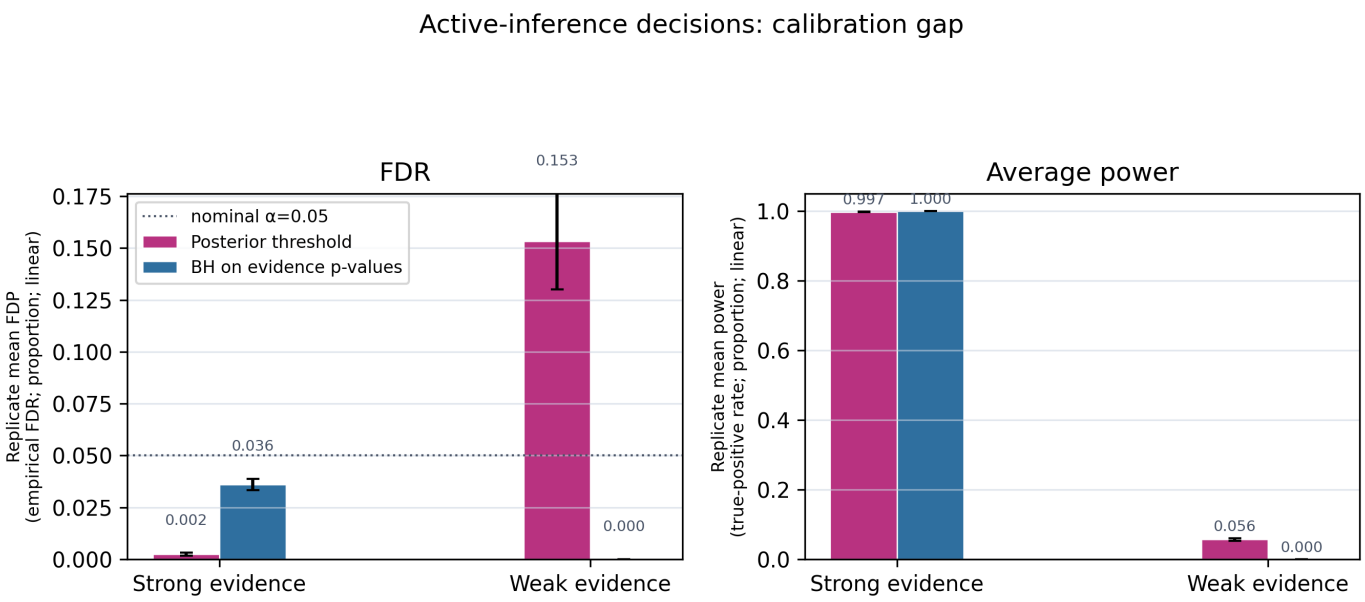


Figure 11: What changes when posterior-threshold decisions are compared with BH applied to calibrated evidence p-values? Empirical FDR and average power for posterior-threshold decisions versus BH on calibrated evidence p-values; bars show ±3 MC SE across the configured active-inference replications (1,000 per regime). Belief thresholds and family-level evidence are intentionally shown as separate decision layers; posterior probability is not treated as an error-rate quantity. Belief-based decisions and family-level evidence remain distinct objects even when shown side by side. Fixed binary channel and horizon; posterior probability is not an error-rate quantity.

4.7 Conditional action-loop operating characteristics

The reference action-loop scenario is `reference_action_loop__r0p65__h12__a0p05` with scenario hash `6b1dcec912f49219c999513d3bcfb9b88f95d7c30f7fc8ebb212000badee9c91`. Its first policy summary uses 1000 replications and reports power 0.000, observed mean FDP (the reported finite-simulation FDR estimand) 0.000, mean simulated cost C 19.50 in the scenario-defined units, and mean stopping time 12.00. Its evidence label is `diagnostic finite-simulation contrast`. These are conditional finite-simulation operating characteristics of the declared model, process, setting, and policy; they are not universal active-inference power claims.

To keep the ten-column table readable, its compact policy labels name the underlying decision rule: Fixed-T (fixed horizon), Posterior cutoff (posterior-threshold stopping), Info gain (expected-information-gain policy), Cost rule (cost-aware policy), Posterior draw (posterior-sampling policy), and E_t rule (likelihood-ratio e-process stopping).

Table 8: Which declared policy trades off power, error, cost, and stopping across reliability cells? Rel. denotes reliability; Mean R mean rejections; Stop. time mean stopping time; and Mean C the scenario-defined simulated-cost units. Rows are scenario-indexed finite-simulation operating characteristics; Evid. `diag.` marks diagnostic evidence. The comparison supports choosing among declared designs, not a universal policy ranking or utility claim. It does not make posterior or fixed-horizon decisions valid under optional stopping.

Rel.	Policy	Power	FDR	FWER	Mean R	Power / C	Mean C	Stop. time	Evid.
r=0.65	Fixed-T	0.000	0.000	0.000	0.00	0.000	19.50	12.00	diag.
r=0.65	Posterior cutoff	1.000	0.000	0.000	1.00	0.500	2.00	1.00	diag.
r=0.65	Info gain	0.985	0.000	0.000	0.98	0.040	24.71	12.00	diag.
r=0.65	Cost rule	0.801	0.000	0.000	0.80	0.067	12.00	12.00	diag.
r=0.65	Posterior draw	0.494	0.000	0.000	0.49	0.021	27.04	12.00	diag.
r=0.65	E_t rule	0.875	0.000	0.000	0.88	0.080	14.03	6.75	diag.
r=0.80	Fixed-T	0.000	0.000	0.000	0.00	0.000	19.50	12.00	diag.
r=0.80	Posterior cutoff	1.000	0.000	0.000	1.00	0.500	2.00	1.00	diag.
r=0.80	Info gain	0.893	0.000	0.000	0.89	0.037	24.14	12.00	diag.
r=0.80	Cost rule	0.953	0.000	0.000	0.95	0.079	12.00	12.00	diag.
r=0.80	Posterior draw	0.512	0.000	0.000	0.51	0.021	26.93	12.00	diag.
r=0.80	E_t rule	1.000	0.000	0.000	1.00	0.218	5.28	2.63	diag.
r=0.92	Fixed-T	0.000	0.000	0.000	0.00	0.000	19.50	12.00	diag.
r=0.92	Posterior cutoff	1.000	0.000	0.000	1.00	1.000	1.00	1.00	diag.
r=0.92	Info gain	0.983	0.000	0.000	0.98	0.082	12.00	12.00	diag.
r=0.92	Cost rule	0.986	0.000	0.000	0.99	0.082	12.00	12.00	diag.
r=0.92	Posterior draw	0.498	0.000	0.000	0.50	0.042	12.00	12.00	diag.
r=0.92	E_t rule	1.000	0.000	0.000	1.00	0.458	2.39	2.39	diag.

Adaptive target selection uses LORD++; its finite any-rejection rate is 0.000, mean FDP is 0.000, mean power is 0.000, FWER is 0.000, and mean selected tests per trace are 12.00. This is a conditional online-FDR diagnostic requiring predictable selection and conditionally super-uniform evidence, not a replacement for the fixed-family BH comparison.

The separate all-null selected-target calibration diagnostic contains 12000 selected p-values in its pooled view. Its maximum empirical CDF excess is 0.481 against a DKW radius of 0.012; the declared finite check is flagged. This diagnostic is conditioned on the explicit all-null process and adaptive-target policy. It does not establish selective-inference validity, conditional super-uniformity for another process, or an online-FDR theorem.

The split-stream protocol is the primary gated result because its confirmation sample is not available to the selection policy. Its claim is correspondingly conditional: the confirmation statistic must retain the declared null law after conditioning on the visible selection history. The chronological diagnostic shows what the current policy does; it does not turn chronological adaptation into a proof.

The release also includes a deliberately invalid future-evidence contrast: a target is selected after inspecting the complete fixed-horizon trace using the minimum observed p-value. It selects 1000 p-values, with empirical CDF excess 0.742 against DKW radius 0.043 (flagged). This contrast makes the filtration failure visible; it is not a valid testing policy and cannot support selective-inference or online-FDR claims.

4.7.1 Gated split-stream confirmation

The primary adaptive-selection lane separates target choice from evidence. A visible selection stream chooses one target, after which an independent child stream measures only that target. The release records separate stream seeds and hashes, the selected target, its confirmation observations, and the declared conditional-null assumptions. This is a separately declared static split-confirmation scenario, `reference_static_split_confirmation` (`5a13b1a434661de593eb828c81beb966af0c5864cf6bf77bac7af5836a945590`), rather than another policy-by-scenario operating cell. Let T denote that the selected target is truly non-null and R that its independent confirmation rejects. Across 1,000 replications, the selected-true rate $P(T)$ is 1.000 (1000 true selected targets), the joint

rate $P(R \cap T)$ is 0.596, and the conditional confirmation rate $P(R \mid T)$ is 0.596. The false-rejection rate $P(R \cap \neg T)$ is 0.000; because exactly one target is confirmed per replication, this is the FWER for this one-confirmed-target estimand. Its evidence label is `formal_conditional_split_confirmation`.

Protocol	Reps	$P(R)$	$P(T)$	$P(R \cap T)$	$P(R \mid T)$	$P(R \cap \neg T)$	Evidence class
Independent split-stream confirmation	1,000	0.596	1.000	0.596	0.596	0.000	formal conditional

This table answers a narrower reader question than an unconditional power comparison: conditional confirmation performance is shown separately from how often the adaptive selection rule lands on a true target. The hashes are part of the filtration audit: selection stream `b05435b110e9935e0c1120d08e944dee2a751f8a7dc47e31cdcdfb4cd6674a11` and confirmation stream `4d8e9ee1da811e34599372991684d0a126160d960df10020f4c91cdd4e16ba61`. Independence of the streams is a design contract, not something that can be inferred from a favorable result. The formal label is therefore withheld whenever process schedules, missingness, truth-changing transitions, or non-constant within-cell emission rates break the declared confirmation law.

4.7.2 Held-out learned-policy evaluation

The learned-policy track searches a finite, predeclared candidate set on independent training streams, freezes the selected parameter vector, and evaluates it on fresh streams. The held-out evaluation uses 1,000 replications, with power 0.762 and mean simulated cost C 12.00 in the scenario-defined units. The search contains 4 candidate parameter vectors of dimension 3. Policy hash `f37c09a53cf47e05bab09e4d0fcb9fb001a49c9951abd2571af291f214f57ff1` identifies the frozen parameters, while search configuration hash `bc29cbc99eaad619d075f799cb3e4c499efaf0d7dd43eb30fbc4cc71de70be37` identifies the candidate space, training budget, and cost penalty used to select them. This is a conditional operating characteristic for one scenario and one train/evaluation partition—not an optimality, utility, causal, or universal active-inference claim.

The held-out split answers a design question—how this frozen rule behaves under this evaluation process—not an optimization question. Candidate count, parameter dimension, training seeds, evaluation seeds, and policy hash are therefore part of the reported evaluated design and evaluation provenance. A higher power-cost point is not evidence that the policy is optimal, utility-maximizing, or transferable to a new process.

Policy track	Evaluation replications	Power	Mean C	Mean stopping time	Frozen policy hash
Held-out learned policy	1,000	0.762	12.00	12.00	f37c09a53cf4

Common-random-number policy contrasts are reported separately from the scenario surface. The first listed policy is the reference; differences are policy minus reference, with Bonferroni-adjusted finite-simulation intervals across the policy-by-metric family. These comparisons pair replication seeds, not observed trajectories, because changing an action changes the process kernel.

Policy contrast	Metric	Difference	Simultaneous interval
Cost aware vs Fixed horizon	Power	0.807	[0.767, 0.847]
Cost aware vs Fixed horizon	FDR	0.000	[0.000, 0.000]
Cost aware vs Fixed horizon	FWER	0.000	[0.000, 0.000]
Cost aware vs Fixed horizon	Mean cost	-7.500	[-7.500, -7.500]
Cost aware vs Fixed horizon	Stopping time	0.000	[0.000, 0.000]
Cost aware vs Fixed horizon	Power / cost	0.067	[0.064, 0.071]
E-process vs Fixed horizon	Power	0.866	[0.831, 0.901]
E-process vs Fixed horizon	FDR	0.000	[0.000, 0.000]

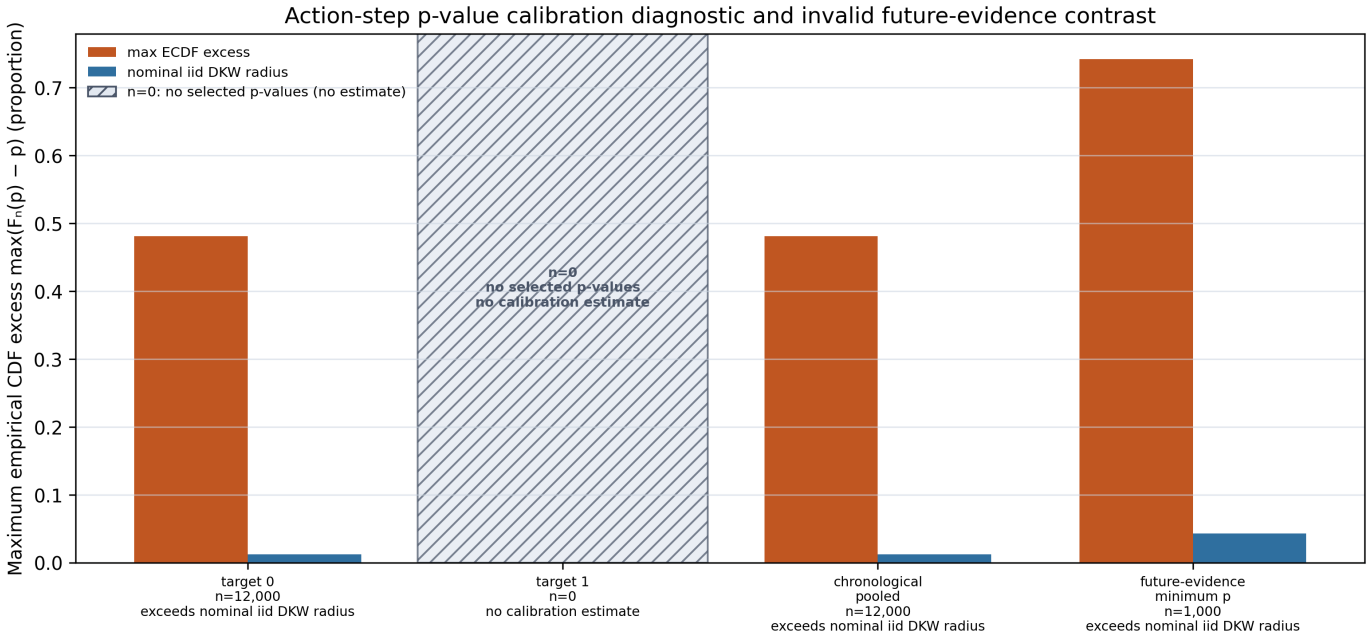
Policy contrast	Metric	Difference	Simultaneous interval
E-process vs Fixed horizon	FWER	0.000	[0.000, 0.000]
E-process vs Fixed horizon	Mean cost	-5.338	[-5.998, -4.678]
E-process vs Fixed horizon	Stopping time	-5.178	[-5.498, -4.858]
E-process vs Fixed horizon	Power / cost	0.079	[0.075, 0.083]
Information gain vs Fixed horizon	Power	0.990	[0.980, 1.000]
Information gain vs Fixed horizon	FDR	0.000	[0.000, 0.000]
Information gain vs Fixed horizon	FWER	0.000	[0.000, 0.000]
Information gain vs Fixed horizon	Mean cost	5.226	[5.174, 5.278]
Information gain vs Fixed horizon	Stopping time	0.000	[0.000, 0.000]
Information gain vs Fixed horizon	Power / cost	0.040	[0.040, 0.040]
Posterior sampling vs Fixed horizon	Power	0.502	[0.451, 0.553]
Posterior sampling vs Fixed horizon	FDR	0.000	[0.000, 0.000]
Posterior sampling vs Fixed horizon	FWER	0.000	[0.000, 0.000]
Posterior sampling vs Fixed horizon	Mean cost	7.488	[7.181, 7.795]
Posterior sampling vs Fixed horizon	Stopping time	0.000	[0.000, 0.000]
Posterior sampling vs Fixed horizon	Power / cost	0.021	[0.019, 0.023]
Posterior threshold vs Fixed horizon	Power	1.000	[1.000, 1.000]
Posterior threshold vs Fixed horizon	FDR	0.000	[0.000, 0.000]
Posterior threshold vs Fixed horizon	FWER	0.000	[0.000, 0.000]
Posterior threshold vs Fixed horizon	Mean cost	-17.500	[-17.500, -17.500]
Posterior threshold vs Fixed horizon	Stopping time	-11.000	[-11.000, -11.000]
Posterior threshold vs Fixed horizon	Power / cost	0.500	[0.500, 0.500]

The paired policy analysis uses 1000 replications and is a conditional decision-design comparison. It does not establish a universal ranking, optimal policy, or utility theorem.

4.8 Null calibration and replication precision

The raw p-value layer is audited separately under an independent global null: the configured mixture is replaced by $\pi_0 = 1$, effect zero, and $\rho = 0$. Pooling 1,000,000 p-values from 5,000 replications gives a Kolmogorov–Smirnov distance of 0.0007 from the Uniform$(0, 1)$ CDF. The distance is compared with a 95% Dvoretzky–Kiefer–Wolfowitz radius of 0.0014. Passing this finite diagnostic supports the marginal calibration of this declared independent null regime; it is not a proof under dependence or for unexamined test statistics.

The configured Monte-Carlo target is a 0.95 confidence precision summary with target half-width 0.020 for bounded replication means. The conservative Hoeffding planning floor is 4,612 independent replications; BH's observed FDR t-based half-width is 0.0007, with a distribution-free radius of 0.0136. The accounting table distinguishes independent replication units from the number of hypothesis or observation draws, so larger families are not mistaken for more independent replications.



Chronological action-step values = process/policy diagnostic; future-evidence minimum-p = invalid contrast. A gray hatched n=0 cell means no selected p-values, not zero ECDF excess. Dedicated split-stream confirmation uses a separately declared static channel and is reported outside this diagnostic.

Figure 12: How does chronological target selection compare with a deliberately invalid future-evidence selector under the all-null diagnostic? Bars compare maximum empirical CDF excess for 12,000 chronological action-step p-values and 1,000 post-hoc selected p-values against a nominal iid DKW radius. 1 target row(s) are gray-hatched n=0 cells: no selected p-values and no calibration estimate, not zero excess. The contrast exposes the filtration boundary; split-stream confirmation is reported separately. Finite all-null diagnostic for the declared model, process, setting, and policy; not selective-inference validity.

Table 9: Replication and raw-draw accounting from the configured full run. The Hoeffding floor is a conservative planning reference, not a claim that every estimand has the same variance.

Block	Independent units	Raw draws / simulations
Method comparison	90,000	18,000,000
Power curve	90,000	18,000,000
Dependence sweep	120,000	24,000,000
Paired method comparisons	10,000	240,000
Active inference	2,000	5,200,000
Sequential evidence	1,000	20,000
Online FDR	1,000	500,000
Agent sampling traces	64	7,680
Conditional action loop	1,000	216,000
Null calibration	5,000	1,000,000
Certificates	5	25,000

4.9 Computational workload: source-bound feasibility accounting

The computational ledger makes the declared release workload inspectable without converting it into a hardware benchmark. Per fixed-family procedure, the current code retains 2,000,000 p-value elements and the same number of truth elements before scoring. Across the named action-loop paths, it records 42,064 trace calls and a maximum of 504,768 permitted action or confirmation-observation slots. These are source-level counts for the configured run, not independent replication units, elapsed time, allocated bytes, or simulated resource cost C.

Table 10: Deterministic, implementation-scoped work accounting for the declared release inputs. The entries distinguish data-shape and trace-path workload from Monte-Carlo precision, action cost, evidence, and online alpha accounting. They do not predict processor instructions, peak memory, throughput, or another machine's elapsed time.

Component	Declared scale	Source-work accounting	Boundary
Single-step corrections	m	$\Theta(m)$; declared proxy 200	Not a processor-instruction or cross-hardware claim.
Rank-based corrections	m	$\Theta(m \log_2 m)$; declared proxy 1,600	Does not expand NumPy or SciPy internals.

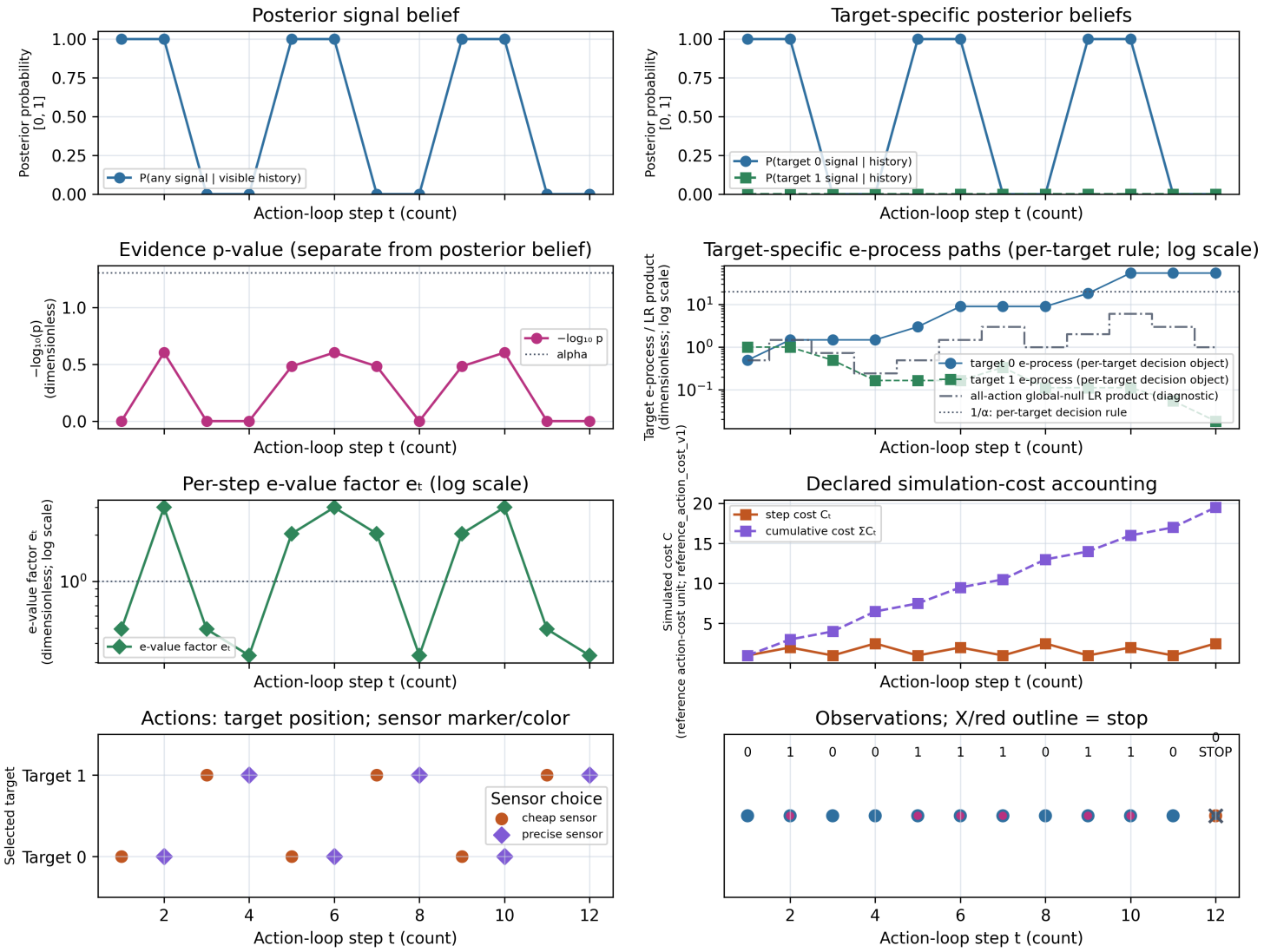


Figure 13: What can an agent see on a simulated trace, and what is scored later? One trace from 1,000 declared replications aligns actions, observations, posterior belief, p-value evidence, target e-process paths, an all-action global-null product, declared simulation cost C, and stopping. Target paths—not the product—carry decisions; X outline marks stopping. Posterior belief, p-values, target e-processes, the diagnostic all-action product, and simulated cost C are distinct; only target paths carry optional-stopping decisions. Hidden state is evaluation-only. Decisions are per target, not an FWER or FDR-control claim; fixed-horizon contrasts are invalid if used for optional stopping.

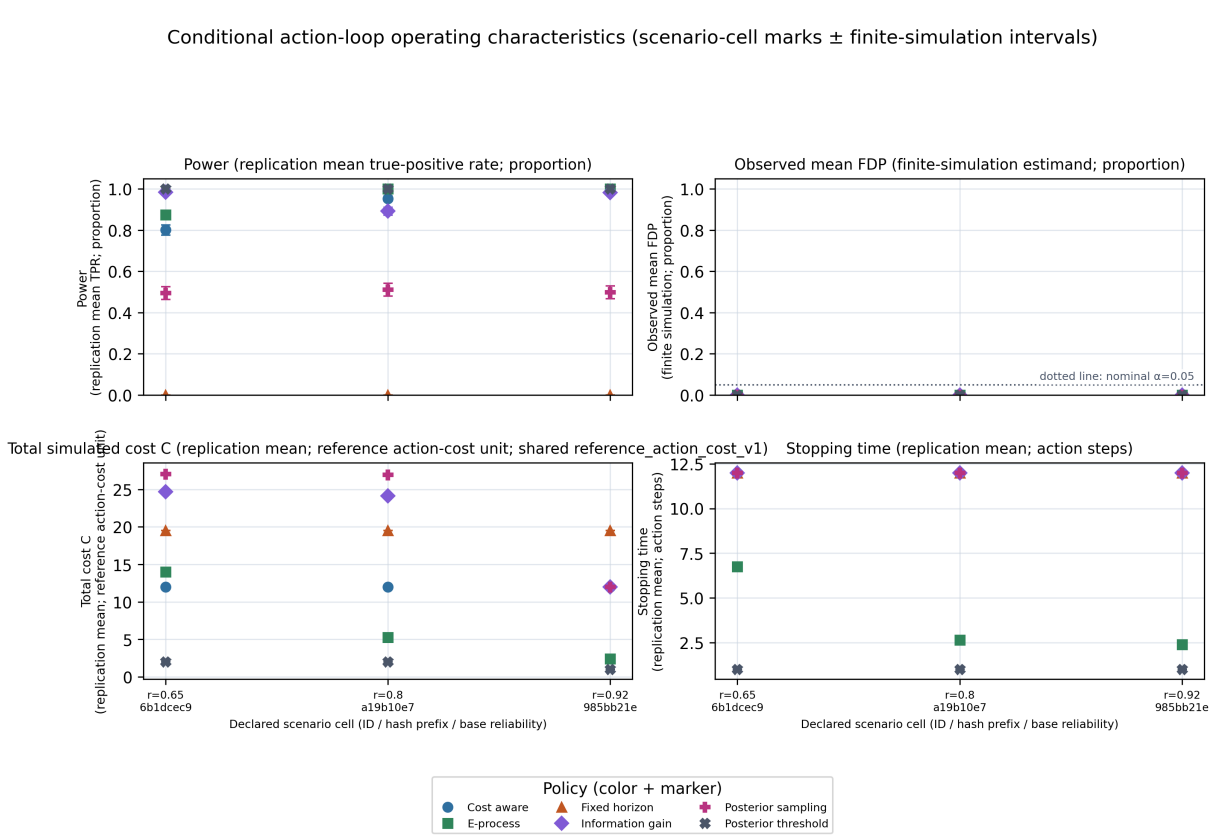


Figure 14: How do policy-specific power, mean FDP, cost, and stopping compare? Across 3 investigator-declared scenario/policy cells, panels show replication-mean power, observed mean FDP, simulated cost C, and stopping from 1,000 replications. Student-t/Wilson intervals are shown; the observed-mean-FDP panel directly labels its dotted nominal-alpha reference. Declared agent-process designs—not an agent's intrinsic power or a distribution over environments—are compared; mean FDP is a finite-simulation estimand. Observed mean FDP is a finite-simulation estimand, not an FDR-control claim for correction-none or per-target e-process rows; fixed-horizon comparisons are invalid if used for optional stopping.

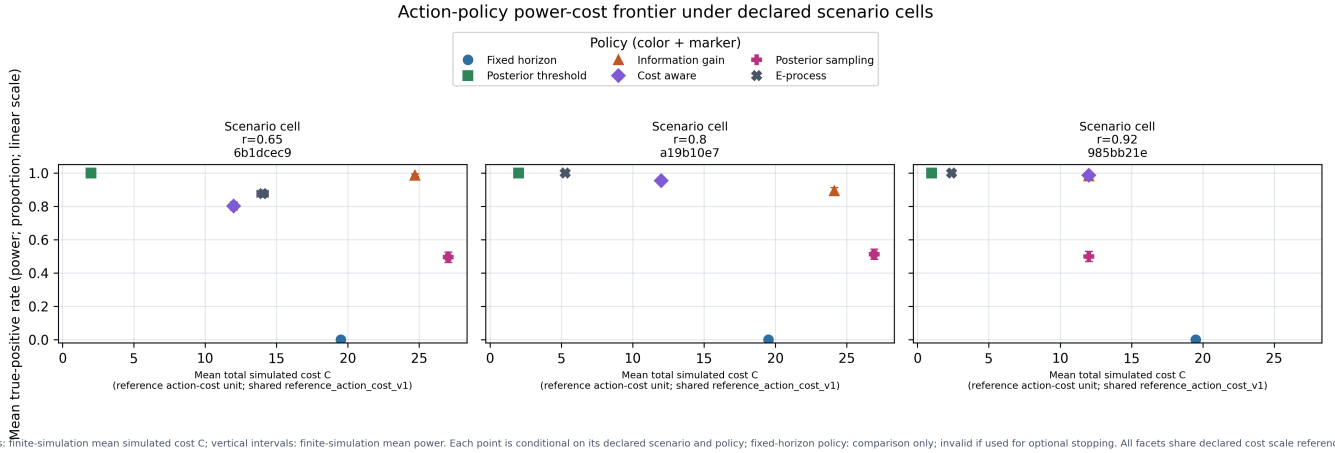


Figure 15: Within each declared scenario, what power-cost trade-offs are visible among policies? Across 3 investigator-declared scenario cells, facets compare replication-mean power with simulated cost C from 1,000 action-loop replications; horizontal and vertical intervals show finite-simulation uncertainty. The frontier supports conditional design choice, not an agent-optimality, utility, or deployment claim. Each facet fixes a declared scenario and cost scale; finite simulations do not establish policy optimality, and fixed-horizon comparisons are invalid if used for optional stopping.

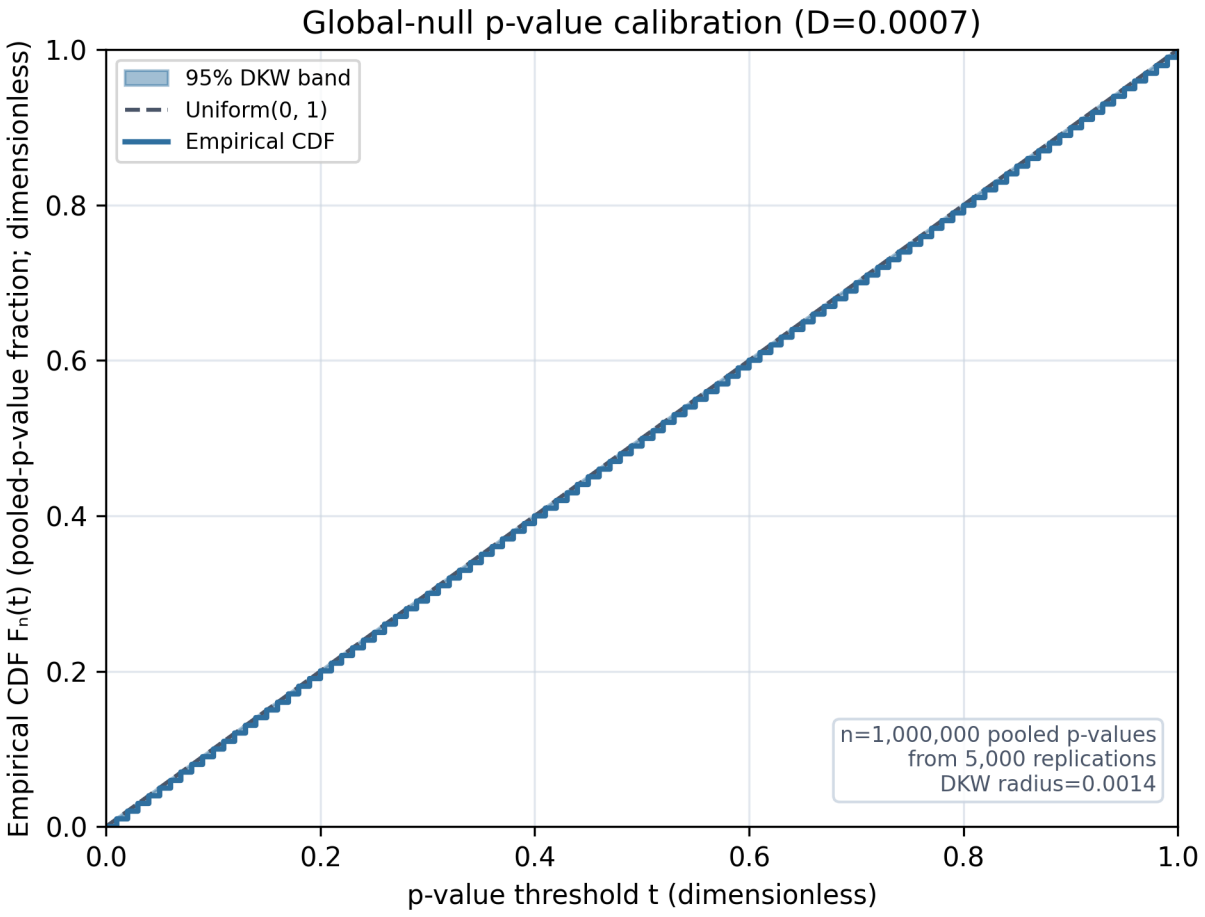


Figure 16: Are global-null p-values visually consistent with the uniform reference in the declared independent calibration regime? Global-null p-value calibration in the independent regime; the empirical CDF is shown with the uniform reference and a 95% DKW band over 1,000,000 pooled p-values from 5,000 independent calibration replications. The ECDF is a regime-specific calibration diagnostic, not a general guarantee of p-value validity. Marginal calibration diagnostic for this null regime only.

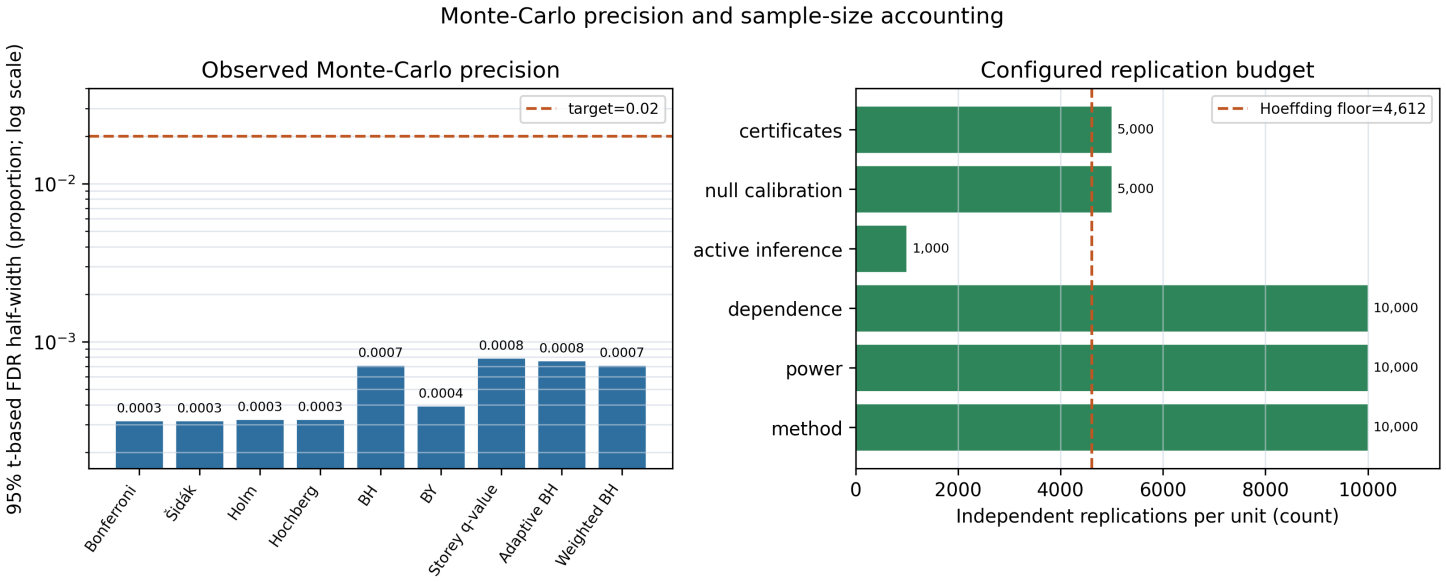


Figure 17: How precise are the simulated FDR summaries, and how is the replication budget allocated? Monte-Carlo precision and replication-budget accounting; observed half-widths from the 10,000 main replications are compared with the configured target and conservative planning floor for the 95% summaries. Observed precision and conservative planning are shown as separate accounting quantities rather than interchangeable guarantees. Planning and accounting aid; estimands need not share a variance.

In tbl. 10, m is family size, B is the main replication count, K is the procedure count, D is the dependence-regime count, T is a stream horizon, and R is prior rejections in an online history. The action row uses N_j for named trace calls and H_j for their permitted horizon; $w_{\mathrm{corr}}(m)$ means the named implementation's source work at the displayed family size.

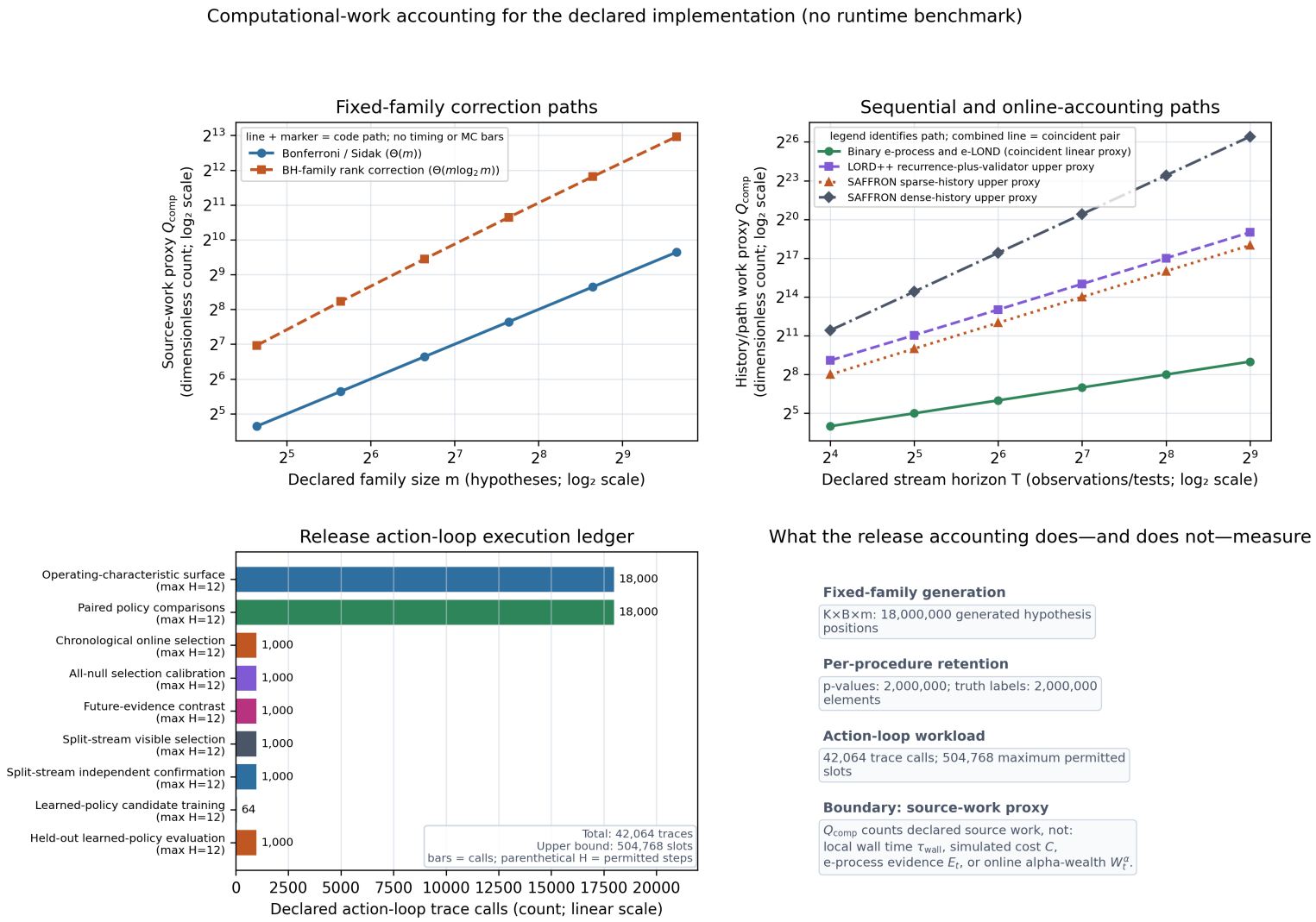


Figure 18: What computational work must be declared before adaptive-study designs are compared? Log-scaled source-work accounting curves compare vector, rank, and history-scan work; one combined line marks the coincident binary-e-process/e-LOND proxies. The action-loop ledger reports 42,064 trace calls and at most 504,768 permitted slots. Marks are deterministic, not timing measurements or statistical uncertainty. The curves and ledger keep workload separate from likelihood-ratio e-process evidence, online alpha-wealth, simulated resource cost C, and host-specific runtime. Source-work accounting excludes dependency internals and supports no cross-hardware runtime or statistical-validity claim.

The curves and trace ledger in fig. 18 make the implementation scope visible: linear, rank-dominant, and history-scan proxies are compared only for the named code paths and declared grids. The figure does not display τ_{wall}. Repeated local timing remains in a separate, non-promotional diagnostic because its value depends on the host and runtime configuration rather than on the statistical design alone.

4.10 Paired method contrasts and sequential evidence

The method contrasts reuse the same replication-level random draws. Their method-minus-BH intervals are Bonferroni-adjusted across the configured method-by-metric grid, so they describe a simultaneous finite-simulation band rather than a set of independent comparisons. The paired contrasts are shown in fig. 19.

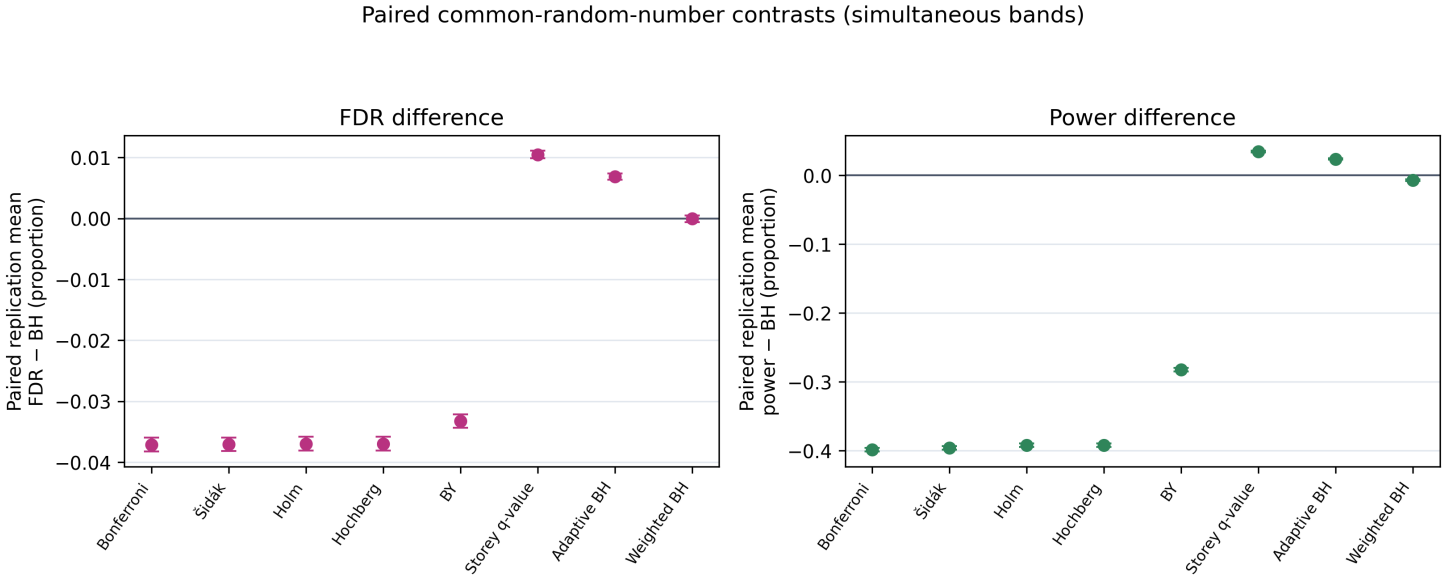


Figure 19: Relative to BH on shared random streams, which configured procedures differ in FDR or power? Paired method-minus-BH differences from common random numbers; point intervals show Bonferroni-simultaneous finite-simulation confidence bands over the 9 configured procedures and 10,000 paired replications. Paired intervals reveal design-specific contrasts without establishing a global method ranking. Finite-simulation contrast, not a universal method ranking.

The sequential track keeps two decisions separate. The likelihood-ratio e-process has a Ville boundary under the declared Bernoulli null, while the fixed-horizon p-value path is included as a deliberately invalid optional-stopping contrast. The online-FDR trace similarly records LORD++ and SAFFRON levels, candidates, rejections, and procedural alpha-wealth states alongside e-LOND nominal levels, e-value thresholds, and cumulative nominal-level allocation; its null simulation is a calibration diagnostic, not a new proof of the procedures. The visual separation is substantive: an e-process can be inspected at a stopping time under its contract, whereas repeatedly checking a fixed-horizon p-value is not made valid by drawing the path in the same panel.

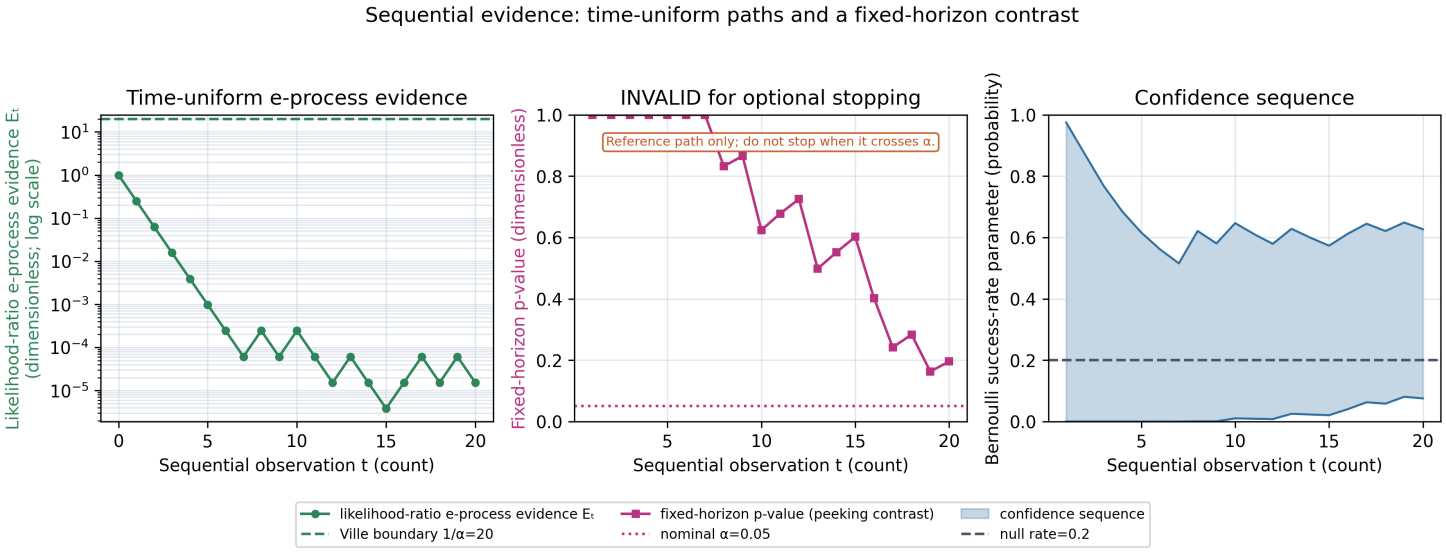


Figure 20: Which sequential summaries remain interpretable under optional stopping, and which do not? Over 20 observations from 1,000 replications, panels show representative null likelihood-ratio e-process evidence E_t, fixed-horizon p-values, and an inverted Bernoulli confidence sequence; the E_t axis is logarithmic. The e-process route supports sequential evidence interpretation; the fixed-horizon p-value is invalid if used for optional stopping. Time-uniform scope is limited to the declared Bernoulli e-process null and confidence-sequence inversion; the p-value path is fixed-horizon.

The e-process experiment uses 1,000 replications over a configured horizon of 20 observations. The table reports the observed crossing/rejection rates and Wilson intervals; only the e-process row is labeled valid under the declared optional-stopping model.

Table 11: Sequential stopping diagnostics with explicit validity labels.

Sequential object	Rate	Wilson interval	Validity boundary
E-process (Ville-valid)	0.010	[0.005, 0.018]	valid under declared Bernoulli null
Fixed-horizon p-value stopping (contrast)	0.107	[0.089, 0.128]	invalid under optional stopping

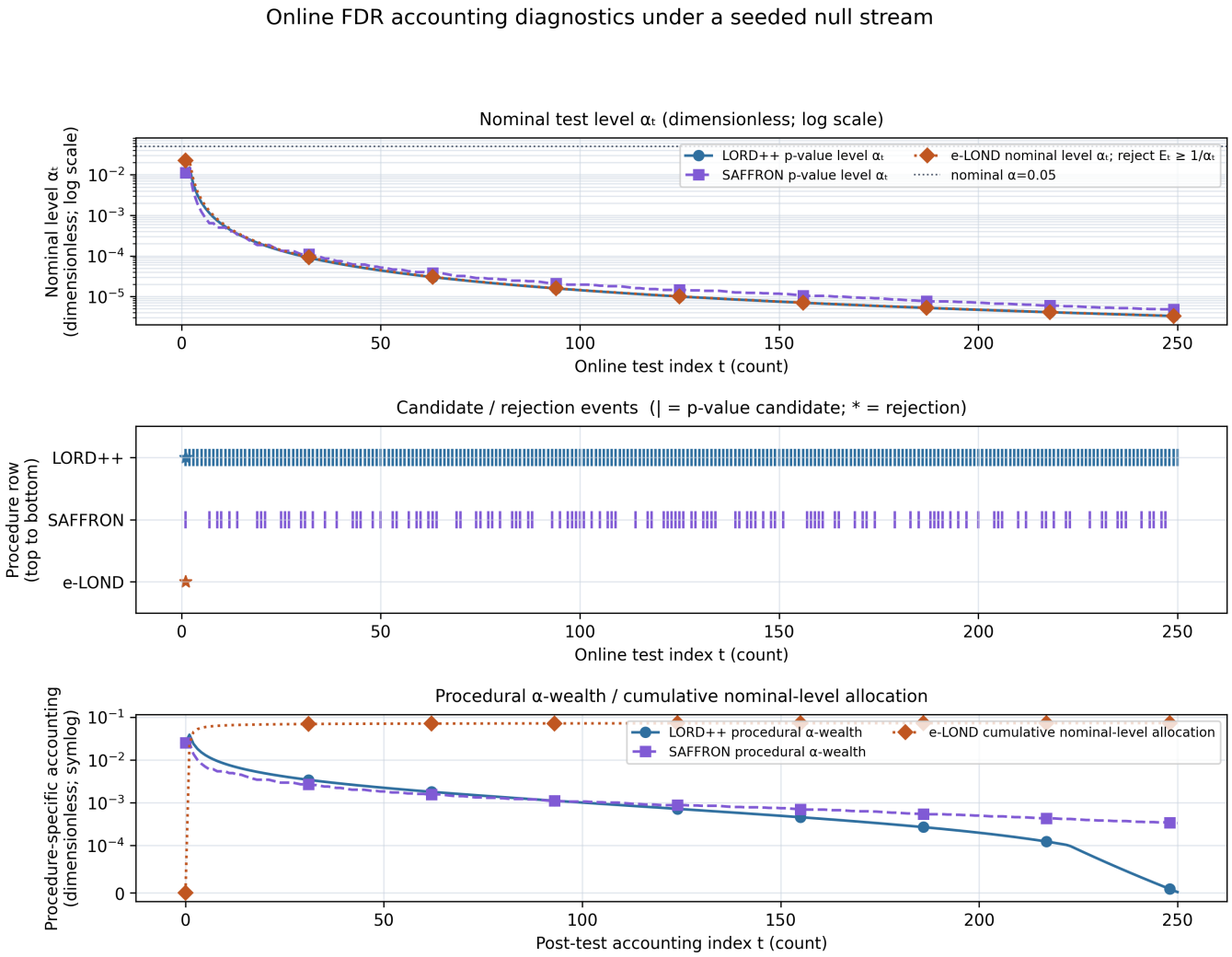


Figure 21: How do online procedures allocate accounting on a null stream? Across 250 null tests from 1,000 replications, panels show LORD++/SAFFRON nominal levels, events, and procedural alpha-wealth; e-LOND nominal levels, decisions, and cumulative allocation. Rows/ticks run top-to-bottom LORD++/SAFFRON/e-LOND; shapes and lines distinguish procedures without color. LORD++/SAFFRON procedural alpha-wealth and e-LOND cumulative allocation are accounting, not likelihood-ratio e-process evidence; e-LOND recycles levels. For e-LOND, E_t >= 1/alpha_t is reciprocal; cumulative allocation is not alpha-wealth or likelihood-ratio e-process evidence. Scope requires declared arrival and evidence assumptions.

The online-FDR null track uses 1,000 replications and a configured horizon of 250 tests. LORD++ and SAFFRON levels, candidates, rejections, and procedure-specific alpha-wealth accounting states are retained in the result artifact. e-LOND instead records nominal levels α_t, the corresponding e-value rule $E_t \geq 1/\alpha_t$, and the cumulative nominal-level allocation $\sum_{i \leq t} \alpha_i$, which can exceed α after recycled levels; it is not a remaining budget. The finite-simulation intervals below are calibration evidence only.

Table 12: Online-FDR null calibration summary; finite-simulation uncertainty does not replace the procedures' assumptions.

Procedure	Mean rejections	Any-rejection rate	Wilson interval
e-LOND	0.024	0.024	[0.016, 0.035]
LORD++	0.024	0.024	[0.016, 0.035]

Procedure	Mean rejections	Any-rejection rate	Wilson interval
SAFFRON	0.058	0.057	[0.044, 0.073]

4.11 Agent-controlled sampling: traces and stopping burden

Agent-controlled sampling is evaluated separately from family-level evidence. The fixed-horizon, posterior-threshold, and e-process policies each retain action, observation, belief, e-process evidence, decision, and stopping traces. There are 64 replications across the configured regimes. Only the e-process policy carries the optional-stopping validity label; posterior decisions are not treated as multiplicity-controlled evidence.

Table 13: Agent policy operating characteristics and validity boundary.

Regime	Policy	Decision rate	Mean stopping time	Optional-stopping valid
Strong ($r = 0.80$, $T = 20$)	E-process	0.984	5.27	Yes
Strong ($r = 0.80$, $T = 20$)	Fixed horizon	1.000	20.00	No
Strong ($r = 0.80$, $T = 20$)	Posterior threshold	1.000	3.59	No
Weak ($r = 0.62$, $T = 6$)	E-process	0.000	6.00	Yes
Weak ($r = 0.62$, $T = 6$)	Fixed horizon	0.078	6.00	No
Weak ($r = 0.62$, $T = 6$)	Posterior threshold	0.172	5.83	No

4.12 Offline benchmark stress track

The benchmark adapter is offline-only. The committed staged agent trajectory is checksum verified and the Golub source is recorded with a uniform resource locator (URL), digital object identifier (DOI), license, retrieval date, schema, preprocessing, and checksum metadata, but external archives are not downloaded during tests or release generation. These fixtures are stress tests and cannot upgrade the synthetic validity claims.

The benchmark is consequently a transportability probe, not an external certificate. Its value is to expose preprocessing, chronology, missingness, and evidence-interface assumptions on a fixed fixture. Any discrepancy between synthetic and benchmark behavior is a prompt to refine the scenario or claim boundary, not evidence that the synthetic guarantee was universal.

The staged trajectory contains 8 rows from 2 subjects. The benchmark adapter keeps these observations separate from the synthetic certificates and records whether the external fixture is available locally.

Table 14: Offline benchmark provenance and stress-test summary.

Track	Sample	Terminal e-process value E_T	Claim role
Synthetic reference	8	0.949	synthetic calibration context only
Staged agent trajectory	8 (2 subjects)	0.949	sequential stress test only
Golub microarray fixture	available (16 rows)	not estimated	external stress test only

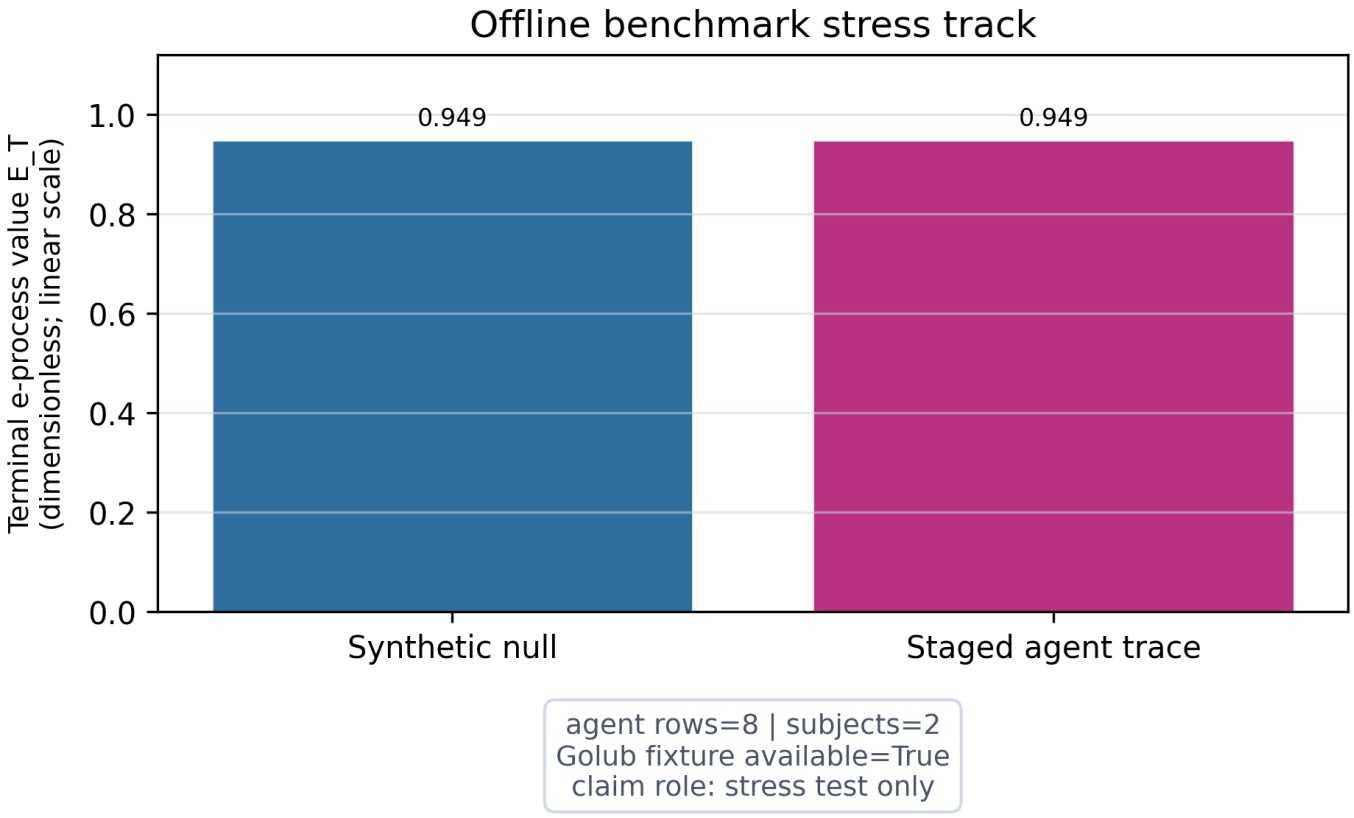


Figure 22: What does the staged external trajectory contribute beyond the synthetic reference context? Offline benchmark stress comparison of the synthetic-null and staged-agent terminal e-process values E_T (8 staged rows). The figure supplies an external stress comparison without promoting the fixture to a validity claim. Offline benchmark stress cannot upgrade synthetic validity claims.

4.13 Evidence-class synthesis

The generated results support a layered conclusion. The fixed-horizon binary reference is the strongest calibration anchor because its null law and family procedure are explicit. Split-stream confirmation is the primary adaptive selection result because the confirmation stream is independent of the visible selection history under the declared protocol. The action-loop surfaces then show how reliability, persistence, cost, and stopping alter operating characteristics for named policies. Sequential, learned-policy, and benchmark tracks add important realism and transport checks, but their labels remain conditional, diagnostic, or stress unless the corresponding filtration, training/evaluation, and provenance contracts are satisfied.

This ordering is substantive. It prevents the most visually attractive power surface from becoming the headline claim when its process changes the evidence law. It also makes negative findings useful: a calibration shortfall identifies which contract needs refinement, while a policy-cost frontier describes a design trade-off only inside its declared scenario.

5 Conclusion

When a policy buys a more reliable observation after a promising signal, returns to a selected target, or decides to look again, it no longer runs the fixed sampling plan. The data path, resource cost, and evidence contract change together. Do not carry a power number into a new design: **Declare the new scenario**—its model/process/setting contract—along with the policy and replication plan, then compare the resulting operating characteristics.

This suite estimates statistical power for that declared design. In each replicate, the embedded agent acts under uncertainty from its visible history; the evaluator uses simulated hidden truth only after the trace to score mean true-positive rate, FDR, and related quantities. It separates an agent's posterior belief, calibrated p-value, likelihood-ratio e-process, online accounting, and resource cost: a common trace does not make them exchangeable. LORD++/SAFFRON alpha-wealth W_t^{α} is distinct from e-LOND's cumulative nominal-level allocation $\sum_{i \le t} \alpha_i$, which can exceed α after recycled levels and is not a remaining budget. It likewise separates computational proxies. Q_{comp} describes a code path and τ_{wall} one host; neither is simulated cost C, e-process evidence E_t, or either online accounting quantity, and neither turns an operating characteristic into a validity claim. The resulting comparison makes observations, target revisits, stopping, error criteria, and resource burden auditable. It is a design-based operating characteristic, not a universal power claim for an algorithm, agent, or task environment [Neyman and Pearson, 1933, Genovese and Wasserman, 2002].

The workflow is actionable. Before comparing policies, declare the observation process, hypothesis family, evidence object, correction, stopping rule, and visible filtration, and cost convention. Keep model-relative belief separate from the calibrated evidence used by a false discovery rate (FDR) or family-wise error rate (FWER) procedure [Benjamini and Hochberg, 1995, Benjamini and Yekutieli, 2001]. For selection, separate selection information from fresh confirmation evidence or state the required conditional-null law [Fithian et al., 2014]. For continued sampling, use an e-process or confidence sequence only under the null, likelihood, and filtration assumptions supporting its time-uniform interpretation [Shafer et al., 2011, Howard et al., 2021].

Declare computational feasibility with the same discipline: record Q_{comp} with its input shape and code scope, report C separately, and keep repeated τ_{wall} samples with runtime context in an isolated diagnostic—not as a design ranking or inferential certificate.

These steps clarify what the results do not show. A finite simulation estimates operating behavior for a stated synthetic scenario with recorded Monte Carlo uncertainty; it cannot prove a theorem, certify arbitrary dependence from a positive-dependence stress regime, or make a posterior threshold family-level calibration. Chronological same-stream selection, future-evidence selection, environment-changing action loops, learned-policy comparisons, and external fixtures therefore retain conditional, diagnostic, stress, or invalid-contrast labels until stronger contracts are established [Morris et al., 2019].

Nor does the scenario grid estimate a distribution of real-world niches or a deployment effect. It is a sensitivity comparison over investigator-declared synthetic environments. A niche-averaged or field-valid claim would require a new sampling frame, environment distribution, and external validation rather than another simulation cell.

The deliverable is a traceable conditional evidence chain from configuration and scenario to evidence, uncertainty, workload accounting, tests, results, and the rendered artifact, enabling revision without silently strengthening a claim [Sandve et al., 2013, Wilkinson et al., 2016].

6 Experimental Setup

A release candidate is generated by a configured experiment with a reference action-loop scenario. Its horizon is 12, with 7 configured policies. The action-loop reliability grid is 0.65, 0.80, 0.92. An investigator uses these settings to define a conditional power surface; they are not universal properties of an Active Inference agent or task environment.

6.1 Configuration as a scientific contract

The canonical experiment is declared in the experiment configuration. The default run uses 200 simultaneous hypotheses, null fraction 0.8, alternative effect 3.0, significance level 0.05, seed 0, profile `release`, and stable SeedSequence child-stream reduction across the listed correction procedures, the configured effect grid, and the negative/positive-factor dependence grid. The result artifact records

a Secure Hash Algorithm digest (SHA-256) of this file so a result cannot be mistaken for a run under another design. The requested worker count is operational provenance; stable child-stream reduction is the scientific reproducibility contract.

The configuration is intentionally declarative. It fixes the hypothesis family, alternative mask, effect and dependence grids, active-inference regimes, action-loop policies, checkpoint policy, uncertainty convention, and artifact schema. A change to a model/process/setting contract changes the scenario; a change to a separately executed action policy changes the evaluated design; and a change to an evidence or execution setting can change its evidence class or provenance. Each requires a new generated result, caption, and claim review.

6.2 Model, process, setting, and policy

The agent-side generative model contains a binary hidden-state factor and a binary observation channel with the declared prior, reliability, transition, and preference semantics. The evaluator-side process supplies the true state, observations, context schedules, action-conditioned kernels, missingness, and costs; it is a synthetic task environment, not an empirical niche. The testing setting supplies the hypotheses, alpha, correction, evidence statistic, stopping rule, and validity assumptions. These three contracts make the hashable scenario; the action policy is evaluated separately against it. The policy sees only the visible history; the evaluator retains hidden truth in a separate diagnostic channel for post-trace scoring. This is the experimental form of the decomposition in sec. 3.2.

6.3 Fixed-horizon data-generating process

Each replication constructs a known true-alternative mask with $(1 - \pi_0)m$ alternatives, draws a common standard-normal factor and independent residuals, adds the effect to alternatives, converts statistics to one-sided Gaussian p-values, and randomly permutes hypothesis positions. The same child stream is reused for paired procedure comparisons and effect sweeps, while stage, scenario, policy, and replication streams remain disjoint. The negative-equicorrelation and block regimes are explicit stress settings; the positive-factor lane is the declared one-sided dependence certificate setting.

6.4 Active-inference calibration

The agent uses the real `inferactively-pymdp==1.0.3` pinned NumPy application programming interface (API). The strong regime uses reliability 0.80, horizon 20, prior signal 0.50, and posterior threshold 0.90; the weak regime uses reliability 0.62, horizon 6, the same configured prior signal, and threshold 0.90. Under the null, the high-observation count has a Binomial distribution, which supplies the calibrated evidence p-value used by BH. The closed-form posterior oracle is evaluated at every observed step; agreement tests semantic correspondence for this channel, not general active-inference correctness.

6.5 Action-in-the-loop setting

The reference action environment separates agent-visible actions and observations from hidden evaluation states. Actions may select a target and sensing quality, transition latent context, incur cost, and stop sampling. The action-loop result records scenario hashes, policy names, stopping times, costs, target decisions, trace visibility, and finite-simulation intervals. Adaptive target selection is passed to the configured online-FDR diagnostic only in chronological order; it is not merged into the fixed-family BH result. The split-confirmation lane uses independent child streams, and the learned-policy lane freezes parameters before fresh evaluation.

6.6 Reproducibility and accounting controls

The analysis records the Python version `3.13.11`, platform, configuration hash, seed, method list, and schema version in the selected bundle's `results.json`. Figures are generated at a configured resolution of 300 dots per inch (DPI) from that JSON, with 23 registered figures and ± 3 MC SE bands. A manifest records expected figure, registry, dashboard, benchmark, checkpoint, and manuscript paths; quality metrics are generated separately so test and coverage counts remain measured rather than hand-coded. The canonical `output/` path is only the default development location; it does not identify a release candidate or the bundle used to render this manuscript.

The full configured run uses 10000 replications per method/effect unit, 1000 active-inference families per regime, and 5,000 global-null calibration replications. The result artifact also records raw hypothesis and observation draws, independent replication units, seed schedules, and checkpoint cursors. These quantities must not be conflated: more observations within a replication do not equal more independent replications.

The configuration also declares the computational-complexity input grids, named correction and online procedures, action policies, warmup policy, and repetition policy. It supplies deterministic source-work accounting for the release and a separate host-specific timing protocol; their distinct evidence roles and reproducibility boundary are described in sec. 7.

7 Reproducibility

Reproducibility is treated here as a scientific control: a reader must be able to identify the scenario, rerun the declared computation, distinguish deterministic evidence from operational telemetry, and see when a release gate has failed. The suite therefore binds code, seeds, results, figures, captions, manuscript variables, claims, and rendered outputs without treating hashes as statistical evidence.

7.1 Release identity

The cover cites the persistent concept DOI 10.5281/zenodo.21695160. The immutable v1 archive is 10.5281/zenodo.21695161, and the exact source release is v1.0.0 in ActiveInferenceInstitute/active_inference_power. These links identify the released implementation and archived version; they do not broaden the finite, scenario-indexed statistical claims made below.

7.2 Verification chain

1. The canonical package imports as `active_inference_power` without test-only path mutation, including from an isolated wheel installation.
2. Corrections validate p-values and alpha, and BH q-values are cross-checked against SciPy on real vectors.
3. The active-inference posterior is checked against its closed-form likelihood ratio and its evidence p-value is derived from the null Binomial law.
4. Seeded simulations produce comparison, paired, power, dependence, policy, sequential, online-FDR, benchmark, action-loop, and calibration blocks in the selected bundle's `results.json`.
5. `scripts/generate_quality_report.py` measures the test count, line coverage, and 95% branch gate; it writes quality evidence to the selected bundle.
6. `scripts/z_generate_manuscript_variables.py` fails closed when results or quality evidence is missing, then writes the resolved manuscript tree.
7. The renderer consumes the resolved tree; the terminal release finalizer then confirms the hashed result, figure, dashboard, benchmark, checkpoint, quality, claim, manuscript, Portable Document Format (PDF), and HyperText Markup Language (HTML) artifacts.
8. The scholarship map and citation-integrity tests keep literature roles, implementation status, evidence artifacts, and validity boundaries linked; this follows the research-object emphasis on findable, accessible, interoperable, and reusable outputs [Wilkinson et al., 2016].
9. The result artifact carries configuration hash `f08204a9b94b6f60b3e634c59144fc6b0288d2ffee4a5f0d88adc732af0887e2` and claim-ledger hash `01f1a4f320f638b3d600c43412ea998590f05929f67dd201d59e281d29e6e366`; strict hydration rejects source drift before it writes renderer input, and the manifest binds both witnesses to the run. The complete manuscript, variable, registry, and figure input tree is bound by render-input hash `1a5f268de8c0a64232f53ab2483b8ba72796c4e02a9902d3a246ae2d8ea9bf24`, which terminal review requires to appear in each reader-facing rendered format.
10. The conditional-power suite hashes each generative model, generative process, and testing-setting scenario; every action-loop operating row carries that scenario hash together with its separately recorded policy. Action-loop hidden states are retained only in the evaluation trace, while the demo consumes a separate deterministic payload.
11. The release result carries a deterministic computational-complexity ledger with its declared input plan and source-work boundary. The optional local timing runner writes a separate hash-bound diagnostic bundle whose frozen profile is checked against the exact current transitive source closure; its elapsed time samples are excluded from release results, certificates, and publication status.

The selected generated bundle is `release candidate bound to the current source`: the run ledger reports `passed` and the certificate gate reports `passed`. If a gate fails, the structured run log records the exception and the manifest remains incomplete. This is deliberate: a review artifact can be useful for diagnosing a method, but it is not silently reclassified as a passed release. The rendered status is an evidence label fixed during hydration. Only the post-render manifest together with its current passing review receipt can declare terminal publication completion, so a reader-facing token never races the fail-closed finalization step.

The measured quality snapshot records 952 real no-mock tests, 95.02% branch coverage, and 96.90% line coverage. The line gate is 95% and the branch gate is 95%; counts and percentages are generated, not asserted in prose by hand.

The generated result schema also records a null p-value calibration audit, Student-t Monte-Carlo precision summaries, conservative Hoeffding planning radii, and the selected bundle's `run_log.json` with stage durations and seeds. These records make sample-size and execution accounting inspectable without treating runtime logging as scientific evidence. Scientific artifacts exclude timestamps and worker timing from their content hashes wherever possible, so operational variation cannot create unexplained numerical differences.

The computational ledger makes a different kind of execution evidence reproducible. The configuration declares its input grids, named correction and online procedures, action policies, warmup policy, and repetition policy; Q_{comp} is then regenerated deterministically from that declared input shape and closed source-work contract. It records correction-vector and stream-horizon proxies, named action-loop trace counts, and separate retained-draw accounting after the scientific result blocks are constructed. It adds neither a simulation stage nor a statistical estimand.

By contrast, the isolated timing profile builds inputs before timing, excludes configured warmups, and retains repeated local τ_{wall} and process-time samples, a canonical plan digest alongside the frozen configuration digest, runtime context, a transitive local source-closure witness, and a self-digest. It rejects a source or frozen-plan change observed across the measurement interval, and frozen-profile review requires the exact current closure rather than a self-consistent subset. That profile can be audited or repeated on the same host, but it is not merged into a selected bundle's `results.json`, a manifest completion decision, or a statistical claim. Simulated resource cost C, e-process evidence E_t, and online alpha-wealth W_t^{α} retain their separate meanings; e-LOND's cumulative nominal-level allocation trace $\sum_{i \le t} \alpha_i$ is a fourth, separate accounting quantity, not alpha-wealth or a remaining budget.

7.3 Reproduction commands

The following is a **development-only** smoke sequence. It writes a new, disposable bundle rather than mutating the canonical `output/` tree, and it is not a source-bound release-candidate recipe:

```
repro_parent="$(mktemp -d "${TMPDIR:-/tmp}/active_inference_power_repro.XXXXXX")"
repro_root="$repro_parent/results"
uv run pytest tests/ --cov=active_inference_power --cov-branch --cov-fail-under=95
uv run python scripts/generate_results.py --profile fast --output-root "$repro_root"
uv run python scripts/generate_quality_report.py --output-root "$repro_root"
uv run python scripts/z_generate_manuscript_variables.py --output-root "$repro_root"
```

Any abbreviated producer command that omits `--output-root` mutates the canonical `output/` development tree. Such commands are development-only and cannot create, repair, or identify a release candidate.

For a source-bound candidate, use the complete explicit-root sequence in `docs/rendering_pipeline.md`: results, quality, strict hydration, audits, rendering, rendered-alt synchronization, renderer provenance, unskipped review, and finalization all bind one candidate root. Do not substitute this development-only root, `output/`, `output/campaign/`, `output/precision/`, or `output/campaign-v2/` for a fresh candidate root. For a larger development campaign, use an explicit profile, seed, worker count, checkpoint interval, and output root. Resume accepts a checkpoint only when the scenario, configuration, schema, and seed-schedule hashes match; this prevents a partial run from being mistaken for a different experiment.

The publication PDF/HTML is produced in a separately provisioned controlled release environment, rather than by a renderer bundled with this package. The public source reproduces results, figures, and the hydrated manuscript; the immutable PDF and accessible HTML are shipped with the GitHub and Zenodo release. The renderer-provenance receipt binds the final bytes to that controlled environment. Inspect both the Portable Document Format (PDF) and HyperText Markup Language (HTML) outputs: figures, tables, captions, alt text, dashboard links, and manuscript variables are part of the release, not post-processing decoration.

7.4 Honest limitations

The following map separates what a released artifact checks from what a reader may infer from it.

Component	Release evidence	Not established
Certificate gate	Configured finite Monte-Carlo band checks.	A formal proof.
Adaptive FDR	Estimate and uncertainty under the declared model; a larger run can expose finite-sample behavior different from BH.	Universal adaptive-FDR control.
Dependence and theorem protocol	Bounded PRDS-style and stress simulations; an offline checksum-pinned adapter permits controlled reruns. The external outward-rounded Arb certificate remains the theorem artifact.	Arbitrary-dependence control or a replacement for the external theorem.
Active-inference model	Binary observation channel, static truth, and fixed-horizon action-loop contract.	A general agent evaluation.
Sequential and online path	E-process interpretation under the declared Bernoulli null, predictable filtration, and update rule; finite online-FDR diagnostics.	Validity of posterior or fixed-horizon p-value stopping, or universal sequential-FDR control.
Computational workload	Deterministic, source-version-specific Q_{comp} accounting for named code paths and declared inputs; separately hash-bound local timing diagnostics.	Processor-instruction counts, peak memory, cross-hardware performance, a release gate, or statistical evidence.
Benchmark fixture	Offline transport and stress workflow checks.	Ground truth for synthetic claims or implicit external retrieval.

7.4.1 Claim-admission checklist

A stronger label requires a declared estimand, primary scholarship, code contract, real-value test, uncertainty calculation, inspectable artifact, and explicit non-claim. A favorable finite result can motivate the next scenario or theorem check; it cannot silently promote itself to one.

8 Scope, Scholarship, and Formal-to-Code Positioning

This project sits at an interface between statistical design and active-inference implementation. The two literatures retain distinct objects and responsibilities. Statistical theory supplies estimands, null laws, error criteria, and validity conditions. Active-inference theory supplies a language for generative models, belief updating, action, and policy selection. The suite connects them through an executable boundary: beliefs remain model-relative states, evidence is calibrated under a named testing setting, and family-level decisions are made by a declared procedure.

The same division assigns two roles. The investigator declares and repeatedly simulates a testing design; the embedded agent performs within-trace inference and action from visible history. Evaluator-only hidden truth scores outcomes after the trace. This is a design-based operating-characteristic study in the sense of the testing and simulation-reporting literatures [Neyman and Pearson, 1933, Morris et al., 2019], not evidence that a synthetic task environment represents an empirical niche or that an agent possesses an intrinsic power value.

The fixed-horizon comparison lanes are investigator simulations of declared streams, evidence interfaces, and correction procedures; they do not by themselves instantiate an action-controlled agent–environment loop. The action-loop lanes add an embedded policy acting under uncertainty within a specified synthetic process. In both cases, the reported power belongs to the investigator's declared design and does not benchmark an agent's general competence, fitness, or ecological adaptation.

8.1 Design-based power and statistical error criteria

The classical testing tradition makes power meaningful relative to a specified null, alternative, test, and sampling distribution [Neyman and Pearson, 1933]. In this suite, that design dependence is extended to action: the policy can alter which target is observed, which channel is used, how the process evolves, and when sampling stops. The resulting quantity is therefore a conditional operating characteristic indexed by the complete design tuple.

Sequential experimental design gives this extension an important historical precedent. Chernoff's work formalizes information acquisition when later sampling decisions depend on earlier observations [Chernoff, 1959], while Lindley provides an information-based account of experiment choice [Lindley, 1956]. The action-loop policies here use finite, scenario-specific information and cost summaries inspired by that design tradition. They provide operating-characteristic comparisons for declared policies; the suite does not claim asymptotic optimality or a universal value of information.

Bonferroni, Holm, and Hochberg provide FWER-oriented procedures [Holm, 1979, Hochberg, 1988]. Benjamini and Hochberg introduced the FDR criterion [Benjamini and Hochberg, 1995], while Benjamini and Yekutieli established a dependence-sensitive boundary and the PRDS context [Benjamini and Yekutieli, 2001]. Storey and Benjamini–Krieger–Yekutieli provide adaptive alternatives [Storey, 2002, Benjamini et al., 2006]. Genovese and Wasserman connect the BH threshold to operating characteristics and asymptotic power [Genovese and Wasserman, 2002].

Blanchard and Roquain provide a proof-oriented decomposition of FDR control into self-consistency and dependency-control conditions [Blanchard and Roquain, 2008]. This is the natural scholarly template for extending the suite beyond finite calibration diagnostics: an implementation should name the rejection rule, the filtration, and the dependence contract before a simulation result is promoted to a formal claim.

The weighted extension is a separate design choice rather than a cosmetic rescaling: fixed positive weights encode priorities that must be specified before inspecting the p-values [Benjamini and Hochberg, 1997]. The implementation validates positivity and normalization, tests permutation behavior under the declared weighted rule, and labels the independence/PRDS boundary in its registry. No data-adaptive weight selection is folded into the release comparison.

The project implements these methods as a tested comparison surface; it does not introduce a new correction. The fixed-point calculation is an asymptotic reference, while two-groups simulations expose finite behavior for explicit factor, block, and negative-correlation designs. A procedure's guarantee is conditional on its assumptions and target error criterion, not on the visual appearance of a power curve.

8.2 Sequential selection, online testing, and adaptive sensing

Online testing changes the information structure: hypotheses arrive in an ordered stream and the testing level may depend on past decisions. Javanmard and Montanari established online FDR control for generalized alpha-investing rules under explicit dependence conditions [Javanmard and Montanari, 2018]. Ramdas et al. developed LORD++-style online control with decaying memory [Ramdas et al., 2017], and their SAFFRON work introduced candidate-indexed adaptation [Ramdas et al., 2018]. Xu and Ramdas develop an e-value formulation that makes predictable e-value evidence and a separate cumulative nominal-level allocation path central [Xu and Ramdas, 2024].

These methods are natural comparators for an agent that chooses what to sample next, but chronological order alone is not enough. The policy must use a declared filtration: a null p-value needs conditional super-uniformity, while an e-factor needs conditional mean at most one. The suite therefore implements finite LORD++, SAFFRON, and e-LOND traces while labeling their simulation summaries as calibration diagnostics unless the assumptions are verified for the scenario.

The broader sequential-testing literature also includes alpha-investing and selection procedures that make the error budget evolve with prior discoveries [Foster and Stine, 2008, G'Sell et al., 2016]. Bandit-testing work connects online false discovery control to adaptive allocation across arms [Yang et al., 2017]. These references clarify why target selection, arrival order, and procedure-specific alpha accounting belong to the testing setting. LORD++ and SAFFRON use alpha-wealth; e-LOND instead records the cumulative nominal levels it has emitted. Its e-value decision rule is $E_t \geq 1/\alpha_t$, and recycled levels can make the cumulative trace exceed α, so it is neither alpha-wealth nor a remaining budget. The suite implements a controlled binary target environment and records those contracts; its finite policy traces do not establish validity for arbitrary adaptive allocation or an unexamined arrival process.

Adaptive target selection is also adjacent to selective inference, where the selection event is part of the inferential contract [Fithian et al., 2014]. A target chosen because it looked promising is not automatically a target with a super-uniform confirmatory p-value. The suite separates three cases: independent split-stream confirmation, chronological same-stream selection as a diagnostic, and post-hoc future-evidence minimum-p selection as an invalid contrast. This is a distinction between information design and statistical validity, not a terminological preference.

8.3 Active inference as a generative-model formalism

Active inference is a family of generative-model formulations in which perception, learning, and action are expressed through variational inference and policy evaluation. The process-theory account derives belief-propagation dynamics from a Markov decision-process generative model [Friston et al., 2017]. The learning account emphasizes the interaction of epistemic information-seeking and pragmatic preference-driven behavior [Friston et al., 2016]. Generalised free energy extends the formal vocabulary for inference over future states and policies [Parr and Friston, 2019]. Da Costa et al. provide a mathematical synthesis for discrete state-space models, which is the closest conceptual neighbor of the binary `pymdp` construction used here [Da Costa et al., 2020]. Smith, Friston, and Whyte provide a step-by-step POMDP-oriented tutorial for this modelling context [Smith et al., 2022]; it is cited for exposition, not as evidence that the binary environment here represents the broader active-inference field.

The present package instantiates only a small, auditable slice of that space: the reference agent has a binary hidden-state factor, a binary observation modality, an identity transition, configured preferences, and a fixed-horizon calibration path. The action-loop extension adds environment-level control, but it remains a controlled synthetic process rather than a claim to represent all active-inference architectures or applications. The real `pymdp` agent and closed-form oracle establish semantic agreement for the declared likelihood and transition objects; they do not establish that every active-inference implementation has the same statistical behavior. The software context is documented by the `pymdp` library paper [Heins et al., 2022].

8.4 From belief states to executable evidence interfaces

A posterior belief answers a model-relative question: how much probability the agent assigns to a hidden state under its prior, likelihood, and observations. A p-value answers a different question: how surprising an evidence statistic would be under a null distribution. An e-value is different again: it is a nonnegative evidence object whose expectation is controlled under the null. The bridge in this project is therefore not an identification of these quantities. It is the explicit map

$$\text{observation stream} \longrightarrow n_{\text{high}} \longrightarrow \Pr_{H_0}\{\text{Binomial}(T, 1-r) \geq n_{\text{high}}\} \longrightarrow \text{BH or another family procedure.} \tag{7}$$

The null Binomial model is exact for the configured channel and fixed horizon. It becomes a valid testing interface only because the null law is stated and the evidence statistic is calibrated under that law. This separation makes the formalism-to-code connection inspectable without treating posterior probability as FDR, FWER, p-value, or e-value control.

The same boundary applies to power. In the suite, power is conditional on the agent's generative model, the underlying generative process, the testing setting, and the action policy. Action-dependent sensing, target selection, environment transitions, costs, and stopping therefore produce scenario-indexed operating characteristics. They do not establish a universal power value for Active Inference as a field.

8.5 Sequential validity and time-uniform evidence

Test martingales connect nonnegative evidence processes to Bayes factors and p-values, including under stopping rules [Shafer et al., 2011]. Safe testing develops e-values whose type-I guarantees are designed for optional continuation [Grünwald et al., 2024]. Time-uniform concentration and confidence sequences provide a complementary way to state uncertainty over a continuing filtration [Howard et al., 2020, 2021].

The package implements a binary likelihood-ratio e-process, its Ville-bound crossing diagnostic, and a Bernoulli confidence-sequence inversion. The reference evidence p-value remains a fixed-horizon Binomial upper tail, and posterior or fixed-p-value stopping are deliberately invalid contrasts. The e-process guarantee is consequently limited to its declared Bernoulli null, predictable update rule, and filtration; it is not a general active-inference or sequential multiple-testing theorem.

8.6 Reproducible research objects and formal-to-code traceability

The suite treats a method claim as a chain: **formal object → public symbol → test → artifact → evidence class → non-claim.**

This design follows the research-object emphasis on findability, accessibility, interoperability, and reuse [Wilkinson et al., 2016], but adds a statistical discipline: provenance cannot make an uncalibrated evidence object valid. The machine-readable registry, claim ledger, scenario hashes, figure metadata, generated captions, manuscript variables, and release finalizer make that boundary inspectable.

The simulation workflow also follows reporting principles for simulation studies: identify the estimand, data-generating mechanism, method parameters, replication unit, uncertainty calculation, and interpretation before reading the result [Morris et al., 2019]. Reproducible-computation guidance motivates the separation of deterministic scientific artifacts from operational telemetry and the preservation of executable provenance [Sandve et al., 2013]. These practices improve inspection and reuse; they do not add statistical information to a generated sample.

Layer	Established reference frame	Implemented contribution here	Explicit boundary
Design and power	Design-based operating characteristics	Scenario-indexed power and cost/stopping estimands	No invariant agent-level power
Family error control	FWER/FDR procedures and dependence conditions	Tested correction dispatcher, invariants, simulations, and certificates	No new multiple-testing procedure
Generative inference	Active-inference process and discrete-state formulations	Real `pymdp` binary agent with closed-form oracle	No general active-inference benchmark
Evidence calibration	Null-distribution p-values; martingale/e-value alternatives	Fixed-horizon Binomial p-value, Bernoulli e-process, and confidence sequence	Each object retains its own assumptions
Adaptive selection	Selective inference and online-FDR filtration	Split confirmation plus chronological and future-selection contrasts	No post-selection theorem without conditional calibration
Policy evaluation	Held-out simulation and adaptive design	Frozen learned policy, paired common-random-number contrasts, cost frontier	No optimality, causal, or utility theorem
Provenance	FAIR research objects	Hash-linked results, figures, claims, manuscript, and render	Provenance does not strengthen validity

The boundary map in fig. 23 makes these distinctions visible. Each row connects a literature family to the concrete object implemented in the package, the artifact that can be inspected, and the inference that remains out of scope. Its purpose is scholarly provenance and claim discipline, not the display of an additional effect estimate.

9 Supplement: Adaptive-FDR certificate audit

This supplement gives the complete accounting behind the adaptive-FDR result in sec. 4.2.1. It is included to keep the main narrative readable while preserving the distinction between a power comparison, a calibration check, and a release gate.

9.1 S1. Audit contract

The audit fixes the declared two-groups model, one-sided p-values, hypothesis family, nominal alpha, Storey tuning rule, and replication seed schedule. Each replication applies BH and the same-vector Storey plug-in q-value to the same generated p-value vector. This common-random-number construction is appropriate for a paired power contrast, but it does not make Storey FDR valid. FDR is computed as the mean replication-level FDP, not as a ratio of pooled false and total discoveries.

The reported certificate has three logically separate conditions: Storey power must meet the BH reference, BH FDR must remain inside its declared bound, and Storey FDR must remain inside the nominal target's finite Monte-Carlo band. Only their conjunction produces a pass. The metric-specific uncertainty is serialized because the power SE cannot be used as an FDR SE.

9.2 S2. Generated audit table

Table 15: Complete adaptive-FDR audit generated from the certificate artifact. The values are conditional on the current model, process, setting, correction, seed schedule, and replication profile.

Metric	Estimate	MC SE	Finite-band ceiling / reference	Gate
BH FDR	0.0252	0.000242	0.0257	passed
Storey FDR	0.0509	0.000337	0.0510	passed

Metric	Estimate	MC SE	Finite-band ceiling / reference	Gate
BH power	0.8355	0.000608	reference	reference
Storey power	0.9089	0.000472	0.8355	passed
Paired power gain	0.0734	0.000436	0.0000	passed

The Storey FDR estimate is 0.05092 and its finite-band ceiling is 0.05101. The excess over the nominal target is 0.00092, corresponding to 2.74 Monte-Carlo SE. The bundle's Storey FDR gate is passed, while its power-gain gate is passed. These labels describe the declared finite check; they do not establish or refute a universal theorem.

9.3 S3. Interpretation and diagnostic ablations

The same-vector Storey plug-in estimates the null fraction from the same p-values used to form the rejection set. A lower estimated null fraction makes the adjusted q-values more permissive. The current audit therefore motivates, but does not by itself prove, a same-vector plug-in calibration mechanism as the source of the observed overrun. The next diagnostic should record the distribution of the estimated null fraction, its underestimation frequency, and its replication-level association with FDP.

The method comparison retains three separate lanes. First, the same-vector Storey plug-in is a transparent sensitivity diagnostic. Second, a conservative fixed-lambda null-count upper bound is a more cautious estimator-based comparator. Third, independent split-stream confirmation is the primary conditional adaptive protocol: selection information estimates the tuning quantity, while fresh confirmation evidence determines rejection. The latter still requires its conditional null law and independence contract.

9.4 S4. Reporting rule

The publication surface reports both passed and failed finite checks. A failed certificate leaves the underlying results, figures, and manuscript variables available for review but prevents the bundle from being marked complete. This is a scientific result: higher replication can narrow uncertainty and reveal a shortfall that a smaller run could not resolve. It is not a reason to increase the threshold, remove the method, or relabel a diagnostic as formal evidence.

9.5 S5. Action-loop estimands and uncertainty accounting

The same accounting rule applies to the action-in-the-loop results. Let $Z_{b,k}$ denote the replication-level outcome for policy k in seed block b, where the outcome may be a true-positive rate, FDP, stopping time, cost, or an indicator of any rejection. The reported operating characteristic is

$$\hat{\theta}_k = \frac{1}{B}\sum_{b=1}^{B} Z_{b,k}, \qquad \widehat{\mathrm{SE}}(\hat{\theta}_k) = \sqrt{\frac{1}{B(B-1)}\sum_{b=1}^{B}(Z_{b,k}-\hat{\theta}_k)^2}. \tag{8}$$

For a paired policy contrast, the unit is the seed block rather than an individual observation. The contrast is computed first, $D_b = Z_{b,k} - Z_{b,\mathrm{ref}}$, and its uncertainty is estimated from the empirical variance of D_b. This preserves the common-random-number design without pretending that action-dependent trajectories share the same observed history. Simultaneous bands apply the declared finite-simulation adjustment across the prespecified policy-by-metric family. They quantify precision for the generated scenario; they do not provide a confidence set over policies, processes, or unobserved environments.

For bounded event rates, Wilson or exact binomial intervals are retained when the estimand is an event probability. For continuous bounded summaries, the finite band uses the configured Monte-Carlo standard-error multiplier, while a Hoeffding radius supplies a distribution-free planning reference. The DKW calibration radius applies to the pooled independent null p-values only. These choices keep interval meaning tied to the sampling unit and avoid using a single generic error bar for every object. The interval conventions are therefore part of the result schema and figure metadata, not a rendering choice.

9.6 S6. Filtration, evidence, and action chronology

The selection trace contains only actions and observations available at the time a target is selected. Hidden truth, confirmation observations, and future outcomes are evaluation fields and are not admissible policy inputs. This is the executable version of the filtration condition: a selection rule is predictable only with respect to the history it is allowed to see. Independent split confirmation makes the next evidence stream fresh, but it does not repair a misspecified null or an action-dependent emission law that violates conditional super-uniformity.

The three adaptive lanes consequently answer different questions. Split confirmation asks whether a selected target can be tested on fresh evidence under its conditional null. Chronological same-stream selection asks how the current policy behaves when discovery and confirmation share a stream. The future-evidence minimum-p lane demonstrates the failure mode created by using information that arrives after the selection decision. The latter is included as a negative control for the filtration audit, not as a candidate method.

Scholarship, implementation, and validity boundaries

Read each row from established theory → coded object → observed evidence → explicit non-claim.

Literature layer	Implemented object	Evidence shown	Boundary
FDR and dependence	BH · BY · weighted BH	Finite fixed-horizon simulations and certificates	Declared p-value and dependence assumptions; no universal ranking
Active-inference generative models	Binary pymdp agent + closed-form oracle	Posterior/evidence calibration under one binary channel	Narrow two-state instantiation; not active inference in general
Sequential evidence	Bernoulli likelihood-ratio e-process	Likelihood-ratio e-process E_t and Ville-bound crossing diagnostic	Optional-stopping validity only under the declared null
Online multiple testing	LORD++ · SAFFRON finite traces	Levels, candidates, rejections, procedural α-accounting, Wilson uncertainty	Calibration evidence, not a new universal online-FDR theorem
Selective inference	Predictable selection + held-out controls	Filtration contract and roadmap entry criteria	Chronology alone does not validate post-selection p-values
Time-uniform concentration	Confidence sequences + Ville-bound vocabulary	Bernoulli e-process path and bounded uncertainty tools	Only the declared Bernoulli e-process is implemented

Primary source families: Benjamini–Hochberg / Benjamini–Yekutieli; Friston / Da Costa; Shafer / safe testing; Ramdas / SAFFRON.

Release evidence context: 10,000 seeded synthetic replications. This map is a provenance aid, not a measured outcome.

Figure 23: Which literature families motivate each implemented object, and what evidence or non-claim remains? Scholarly boundary map linking FDR and dependence theory, active-inference generative modeling, sequential evidence, and online FDR to implemented objects, evidence classes, and explicit non-claims. This is a provenance aid, not a quantitative result or a simulation-derived estimate. The map keeps scholarly context, implementation, generated evidence, and unmade claims visibly separate. Scholarship and implementation context must not be read as measured evidence.

This interpretation aligns the implementation with selective-inference and online-testing scholarship: selection is part of the inferential object, and online validity depends on the history-conditioned null law rather than on chronological ordering alone [Fithian et al., 2014, Javanmard and Montanari, 2018].

9.7 S7. What a larger sample can and cannot resolve

Increasing the number of independent replications narrows Monte-Carlo uncertainty and can separate a small finite-simulation excess from sampling noise. It does not change the generative model, process, testing setting, policy, or theorem assumptions. In particular, a larger run can turn a previously inconclusive finite gate into a failure without making the underlying procedure less or more valid in a mathematical sense. The precision campaign is therefore retained as an auditable review bundle, while the release manifest remains fail-closed whenever a predeclared certificate does not pass.

This rule is central to the suite's scholarship. The simulation is an empirical study of a specified operating characteristic, not a substitute for a proof. The proof-relevant conditions remain the null distribution, dependence or conditional-superuniformity contract, filtration, and stopping rule. The artifact chain supplies reproducibility and inspection; it cannot promote diagnostic evidence to a formal theorem.

9.8 S8. Reader checklist for adaptive claims

Before interpreting an adaptive result, a reader should identify: (i) the target population and true-null definition; (ii) the visible filtration at selection and stopping; (iii) whether confirmation evidence is independent; (iv) the p-value or e-value null law; (v) the family-level accounting rule; (vi) the independent replication unit and interval convention; and (vii) the scenario hash and separately recorded policy that bind the prose to the generated artifact (a frozen learned policy may additionally carry its own parameter hash). If any item is absent, the result should be read as exploratory or diagnostic until the missing contract is supplied. This checklist is deliberately more restrictive than a plot-level comparison because adaptive sensing can change both what is observed and the distribution against which it must be judged.

10 References

The bibliography is stored in the bibliography file and is resolved by the template renderer. Every citation key used in the manuscript has a corresponding BibTeX entry. DOI links are retained whenever a stable DOI exists; non-DOI URLs are retained for primary preprints, archival records, software or data artifacts, and provenance-critical implementation bundles. A link records where and in what status a source can be inspected; it does not turn a local reproduction, finite simulation, or artifact into an external theorem.

Yoav Benjamini and Yosef Hochberg. Controlling the false discovery rate: a practical and powerful approach to multiple testing. *Journal of the Royal Statistical Society: Series B (Methodological)*, 57(1):289–300, 1995. doi: 10.1111/j.2517-6161.1995.tb02031.x.

Yoav Benjamini and Yosef Hochberg. Multiple hypotheses testing with weights. *Scandinavian Journal of Statistics*, 24(3):407–418, 1997. doi: 10.1111/1467-9469.00072.

Yoav Benjamini and Daniel Yekutieli. The control of the false discovery rate in multiple testing under dependency. *The Annals of Statistics*, 29(4):1165–1188, 2001. doi: 10.1214/aos/1013699998.

Yoav Benjamini, Abba M. Krieger, and Daniel Yekutieli. Adaptive linear step-up procedures that control the false discovery rate. *Biometrika*, 93(3):491–507, 2006. doi: 10.1093/biomet/93.3.491.

Gilles Blanchard and Etienne Roquain. Two simple sufficient conditions for false discovery rate control. *Electronic Journal of Statistics*, 2:963–992, 2008. doi: 10.1214/08-EJS180.

Herman Chernoff. Sequential design of experiments. *The Annals of Mathematical Statistics*, 30(3):755–770, 1959. doi: 10.1214/aoms/1177706205.

Lancelot Da Costa, Thomas Parr, Noor Sajid, Sebastijan Veselic, Victorita Neacsu, and Karl Friston. Active inference on discrete state-spaces: A synthesis. *Journal of Mathematical Psychology*, 99:102447, 2020. doi: 10.1016/j.jmp.2020.102447.

Edgar Dobriban. The benjamini–hochberg procedure can fail to control the fdr for correlated two-sided gaussian tests, July 2026. URL https://arxiv.org/abs/2607.12208. Preprint submitted to arXiv on 13 Jul 2026. Reproducibility bundle: https://github.com/dobriban/BH.

William Fithian, Dennis Sun, and Jonathan Taylor. Optimal inference after model selection, 2014. URL https://arxiv.org/abs/1410.2597.

Dean P. Foster and Robert A. Stine. α-investing: a procedure for sequential control of expected false discoveries. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 70(2):429–444, 2008. doi: 10.1111/j.1467-9868.2007.00643.x.

Karl Friston, Thomas FitzGerald, Francesco Rigoli, Philipp Schwartenbeck, and Giovanni Pezzulo. Active inference: A process theory. *Neural Computation*, 29(1):1–49, 2017. doi: 10.1162/NECO_a_00912.

Karl J. Friston, Thomas FitzGerald, Francesco Rigoli, Philipp Schwartenbeck, John O'Doherty, and Giovanni Pezzulo. Active inference and learning. *Neuroscience and Biobehavioral Reviews*, 68:862–879, 2016. doi: 10.1016/j.neubiorev.2016.06.022.

Christopher Genovese and Larry Wasserman. Operating characteristics and extensions of the false discovery rate procedure. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 64(3):499–517, 2002. doi: 10.1111/1467-9868.00347.

Andy Georges, Dries Buytaert, and Lieven Eeckhout. Statistically rigorous java performance evaluation. In *Proceedings of the 22nd Annual ACM SIGPLAN Conference on Object-Oriented Programming Systems and Applications*, OOPSLA '07, pages 57–76. Association for Computing Machinery, 2007. doi: 10.1145/1297105.1297033.

Peter Grünwald, Rianne de Heide, and Wouter Koolen. Safe testing. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 86(5):1091–1128, 2024. doi: 10.1093/jrsssb/qkae011.

Maximilian G'Sell, Stefan Wager, Alexandra Chouldechova, and Robert Tibshirani. Sequential selection procedures and false discovery rate control. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 78(2):423–444, 2016. doi: 10.1111/rssb.12122.

Conor Heins, Beren Millidge, Daphne Demekas, Brennan Klein, Karl Friston, Iain D. Couzin, and Alexander Tschantz. pymdp: A python library for active inference in discrete state spaces. *Journal of Open Source Software*, 7(73):4098, 2022. doi: 10.21105/joss.04098.

Yosef Hochberg. A sharper bonferroni procedure for multiple tests of significance. *Biometrika*, 75(4):800–802, 1988. doi: 10.1093/biomet/75.4.800.

Sture Holm. A simple sequentially rejective multiple test procedure. *Scandinavian Journal of Statistics*, 6(2):65–70, 1979. doi: 10.2307/4615733.

Steven R. Howard, Aaditya Ramdas, Jon McAuliffe, and Jasjeet Sekhon. Time-uniform chernoff bounds via nonnegative supermartingales. *Probability Surveys*, 17:257–317, 2020. doi: 10.1214/18-PS321.

Steven R. Howard, Aaditya Ramdas, Jon McAuliffe, and Jasjeet Sekhon. Time-uniform, nonparametric, nonasymptotic confidence sequences. *The Annals of Statistics*, 49(2):1055–1080, 2021. doi: 10.1214/20-AOS1991.

Adel Javanmard and Andrea Montanari. Online rules for control of false discovery rate and false discovery exceedance. *The Annals of Statistics*, 46(2):526–554, 2018. doi: 10.1214/17-AOS1559.

Dennis V. Lindley. On a measure of the information provided by an experiment. *The Annals of Mathematical Statistics*, 27(4):986–1005, 1956. doi: 10.1214/aoms/1177728069.

Tim P. Morris, Ian R. White, and Michael J. Crowther. Using simulation studies to evaluate statistical methods. *Statistics in Medicine*, 38(11):2074–2102, 2019. doi: 10.1002/sim.8086.

Jerzy Neyman and Egon S. Pearson. On the problem of the most efficient tests of statistical hypotheses. *Philosophical Transactions of the Royal Society of London. Series A, Containing Papers of a Mathematical or Physical Character*, 231(694–706):289–337, 1933. doi: 10.1098/rsta.1933.0009.

Thomas Parr and Karl J. Friston. Generalised free energy and active inference. *Biological Cybernetics*, 113(5–6):495–513, 2019. doi: 10.1007/s00422-019-00805-w.

Aaditya Ramdas, Fanny Yang, Martin J. Wainwright, and Michael I. Jordan. Online control of the false discovery rate with decaying memory. In *Advances in Neural Information Processing Systems*, volume 30, 2017. URL https://papers.nips.cc/paper_files/paper/2017/hash/7f018eb7b301a66658931cb8a93fd6e8-Abstract.html.

Aaditya Ramdas, Tijana Zrnic, Martin J. Wainwright, and Michael I. Jordan. Saffron: An adaptive algorithm for online control of the false discovery rate. In *Proceedings of the 35th International Conference on Machine Learning*, volume 80, pages 4286–4294, 2018. URL https://proceedings.mlr.press/v80/ramdas18a.html.

G. K. Sandve, Anton Nekrutenko, James Taylor, and Eivind Hovig. Ten simple rules for reproducible computational research. *PLoS Computational Biology*, 9(10):e1003285, 2013. doi: 10.1371/journal.pcbi.1003285.

Glenn Shafer, Alexander Shen, Nikolai Vereshchagin, and Vladimir Vovk. Test martingales, bayes factors and p-values. *Statistical Science*, 26(1):84–101, 2011. doi: 10.1214/10-STS347.

Ryan Smith, Karl J. Friston, and Christopher J. Whyte. A step-by-step tutorial on active inference and its application to empirical data. *Journal of Mathematical Psychology*, 107:102632, 2022. doi: 10.1016/j.jmp.2021.102632.

John D. Storey. A direct approach to false discovery rates. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 64(3):479–498, 2002. doi: 10.1111/1467-9868.00346.

Mark D. Wilkinson, Michel Dumontier, IJsbrand Jan Aalbersberg, Gabrielle Appleton, Myles Axton, Arie Baak, Niklas Blomberg, Jan-Willem Boiten, Luiz Bonino da Silva Santos, Philip E. Bourne, Jildau Bouwman, Anthony J. Brookes, Timothy Clark, Mercè Crosas, Ingrid Dillo, Olivier Dumon, Scott Edmunds, Chris T. Evelo, Richard Finkers, Alejandra Gonzalez-Beltran, Alasdair J. G. Gray, Paul Groth, Carole Goble, Jeffrey S. Grethe, Jaap Heringa, Peter A. C. 't Hoen, Rob Hooft, Tobias Kuhn, Ruben Kok,

Joost Kok, Scott J. Lusher, Maryann E. Martone, Barend Mons, Erik van Mulligen, Jan Velterop, Peter Wittenburg, Katherine Wolstencroft, and Jun Zhao. The fair guiding principles for scientific data management and stewardship. *Scientific Data*, 3:160018, 2016. doi: 10.1038/sdata.2016.18.

Ziyu Xu and Aaditya Ramdas. Online multiple testing with e-values. In *Proceedings of the 27th International Conference on Artificial Intelligence and Statistics*, volume 238 of *Proceedings of Machine Learning Research*, pages 3997–4005, 2024. URL https://proceedings.mlr.press/v238/xu24a.html.

Fanny Yang, Aaditya Ramdas, Kevin G. Jamieson, and Martin J. Wainwright. A framework for multi-armed bandit testing with online false discovery rate control. In *Advances in Neural Information Processing Systems*, volume 30, pages 5959–5968, 2017. URL https://papers.nips.cc/paper/7177-a-framework-for-multi-armedbandit-testing-with-online-fdr-control.