RUNTIME MONITORING OF SUPERMARTINGALE CERTIFICATES

Anonymous authors
Paper under double-blind review

ABSTRACT

Supermartingale certificates formally guarantee temporal properties of discrete and continuous-time stochastic dynamical systems. However these guarantees rely and hold within a mathematical model of the world, leaving a sim-to-real gap that can render them invalid under the deployed system. We formulate the runtime monitoring problem for supermartingale certificates, in which a monitor observes a single system trajectory and raises an alarm when there is sufficient statistical evidence that the supermartingale certificate has been violated during execution. We present an algorithm to automatically construct a monitor using betting-based sequential tests that bound the probability of ever raising a false alarm at a user-specified level. Within this betting framework we formulate strategies that allow the monitor to exploit spatial regularity to improve detection power while preserving this statistical guarantee. We demonstrate the efficacy of our approach on supermartingale certificate benchmarks from the literature.

1 INTRODUCTION

Supermartingale certificates guarantee that probabilistic, potentially infinite-state systems satisfy temporal properties of their intended behaviour, making them a powerful tool for assuring sequential decision-making under uncertainty. Applications include the automated control of unmanned vehicles, manufacturing processes, and financial systems, where rigorous guarantees of safety and security are paramount. Supermartingale certificates are real-valued functions of system states whose expected values decrease along system executions in a manner determined by the specification of interest. They are the probabilistic analogues of Lyapunov and barrier functions in control systems (Kushner, 1965), and of inductive invariants and progress measures in software systems (Floyd, 1967; Chakarov & Sankaranarayanan, 2013). A supermartingale certificate reduces reasoning about a global temporal property of system behaviour—such as reachability, safety, reach-avoidance, stability, reactivity—to verifying a local expected-decrease condition at each individual step of the system.

The algorithmic construction of supermartingale certificates has traditionally relied on rigorous symbolic reasoning methods (Prajna & Jadbabaie, 2004; Chatterjee et al., 2016), with more recent work turning to neural representations and certified learning-based approaches (Abate et al., 2021; Lechner et al., 2022; Zikelic et al., 2023a; Neustroev et al., 2025). Yet, the formal guarantees provided by the existing methods are contingent on a given *model of the world*, fixed at design time. If the behaviour of the environment in which the controller is deployed deviates from this assumed model, the certificate may no longer satisfy the supermartingale conditions, invalidating its guarantees. Such deviations may arise, for example, from environmental changes, distribution shifts, or unmodelled physical conditions, and are commonly associated with what is known in control theory and robotics as the *sim-to-real gap* (Jakobi et al., 1995; Tobin et al., 2017; Zhao et al., 2020).

This raises a fundamental question: when the true environment is not fully known and can only be interacted with at runtime as a black box, how can we determine whether a certificate remains valid after deployment? This is the central question we address in this work: we propose for the first time a framework for the *runtime monitoring of supermartingale certificates* upon deployment. Given a supermartingale certificate computed at design time and a black-box deployment stochastic environment, our method observes a single stream of system states, continually tests whether the supermartingale certificate satisfies its defining expected decrease condition, and raises an alarm once it has gathered sufficient statistical evidence that the supermartingale certificate is invalid.

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

Runtime monitoring of a supermartingale certificate requires detecting violations from a single stream of observations. Because the environment is stochastic, an increase (resp. decrease) in the certificate value at any individual step does not necessarily indicate a violation (resp. satisfaction) of the expected-decrease condition. To address this challenge, our monitor accumulates statistical evidence over successive state observations, seeking to determine when the observed changes in certificate value become sufficiently inconsistent with the expected-decrease condition to signal a violation. This evidence is accumulated using a *betting-based sequential testing method* (Ramdas et al., 2023; Waudby-Smith & Ramdas, 2024). The monitor plays a betting game: it starts with an initial wealth and, for each pair of successive state observations, converts the change in certificate value into a normalised score and chooses a stake based on the observations seen so far. A positive score increases the monitor's wealth, while a negative score decreases it, with the magnitude of either change determined by the chosen stake. Crucially, while the expected-decrease condition holds, no admissible betting strategy can increase the monitor's wealth in expectation. Sustained wealth growth therefore constitutes statistical evidence against this condition. Once the wealth exceeds a threshold calibrated to control the false-alarm probability, the monitor signals a violation.

We implemented our method and evaluated betting strategies that either weight past observations uniformly or favour recent observations and those from nearby states across a wide range of benchmarks. Our results show that the framework effectively detects certificate violations under changes in deployment dynamics, and that exploiting temporal and spatial regularity in the betting strategy improves the detection of local violations.

Our contribution is threefold. First, we present the first framework for runtime verification of supermartingale certificates. Our framework is agnostic to the temporal property being certified, applies to both discrete- and continuous-time stochastic systems, and encompasses the existing supermartingale certificates (see Section 2). Second, we introduce an algorithm for automatically constructing monitors based on betting games, with a user-specified bound on the false-alarm probability that holds uniformly over an unbounded monitoring horizon. This bound allows users to control the monitor's conservativeness and avoid unnecessary intervention (see Section 3). Finally, we empirically demonstrate the effectiveness of our method in detecting certificate violations under changed deployment dynamics (see Section 4).

2 Runtime Monitoring of Supermartingale Certificates

Markov Process We consider probabilistic systems modelled as Markov processes in either discrete or continuous time, over which supermartingale certificates are defined. We generically write $\mathbb{T}$ for the time domain, with $\mathbb{T} = \mathbb{N}_0$ (the non-negative integers) in discrete time and $\mathbb{T} = \mathbb{R}_{\geq 0}$ (the non-negative reals) in continuous time, and take $t = 0$ as the initial time.

Let $\mathcal{X} = \mathbb{R}^d$ be a Euclidean space, and $(\mathcal{X}, \mathcal{F}, \mathbb{P})$ be a probability space over $\mathcal{X}$. A property holds *almost surely* if it holds with probability one. Let $(\mathcal{G}_t)_{t\in\mathbb{T}}$ be a filtration in this probability space. A *stochastic process* $\boldsymbol{X} = (X_t)_{t\in\mathbb{T}}$ with respect to $(\mathcal{G}_t)_{t\in\mathbb{T}}$ is a sequence of random variables with each X_t being $\mathcal{G}_t$-measurable. In the continuous-time setting, we also assume that $\boldsymbol{X}$ is right-continuous and that the filtration $(\mathcal{G}_t)_{t\in\mathbb{T}}$ is complete and right-continuous. A *(time-homogeneous) Markov process* $\boldsymbol{X} = (X_t)_{t\in\mathbb{T}}$ is a stochastic process such that, for every measurable function $f : \mathcal{X} \to \mathbb{R}$ and for every pair of time values $t, s \in \mathbb{T}$, we have $\mathbb{E}[f(X_{t+s}) \mid \mathcal{G}_t] = \mathbb{E}[f(X_{t+s}) \mid X_t]$.

Supermartingale Certificate A *supermartingale certificate* is a measurable function $f : \mathcal{X} \to \mathbb{R}$ which satisfies a finite number of conditions of one of the following two types:

1. *Domain bound condition.* Given a domain $\mathcal{S} \subseteq \mathcal{X}$ and a bound interval $[l, u]$ where $l, u \in \mathbb{R} \cup \{\pm\infty\}$, the $(\mathcal{S}, [l, u])$-domain bound condition specifies that, for all times $t \in \mathbb{T}$, we have $f(X_t) \in [l, u]$. Intuitively, this condition requires the value of f to lie within the interval $[l, u]$ at all times, whenever the stochastic process value is in the domain set $\mathcal{S}$.
2. *Expected decrease condition.* Given a domain $\mathcal{S} \subseteq \mathcal{X}$ and a measurable drift bound $l : \mathcal{X} \to \mathbb{R}$, zero outside $\mathcal{S}$, the $(\mathcal{S}, l)$-expected drift condition requires, for $t \in \mathbb{T}$ and $X_t \in \mathcal{S}$, a bound on
   - (*discrete-time*) the one-step expected decrease: $\mathbb{E}[f(X_{t+1}) \mid X_t] - f(X_t) \leq l(X_t)$;
   - (*continuous-time*) the instantaneous drift: $\mathcal{A}f(X_t) \coloneqq \lim_{h\downarrow 0} \frac{\mathbb{E}[f(X_{t+h})|X_t] - f(X_t)}{h} \leq l(X_t)$.

Intuitively, this bounds the expected change per step, or the instantaneous drift rate, whenever the process is in $\mathcal{S}$ by at least the amount specified by the decrease function l. The continuous time requires additional regularity assumptions specified in Appendix A.

These defining conditions subsume all classical supermartingale certificates for proving temporal properties in stochastic systems, such as ranking supermartingales (Chakarov & Sankaranarayanan, 2013) for almost-sure reachability, stochastic barrier functions (Prajna et al., 2007) for probabilistic safety, reach-avoid supermartingales (Zikelic et al., 2023a) for probabilistic reach-avoidance, or Streett supermartingales (Abate et al., 2024; 2025) or limit-deterministic Büchi supermartingales (Henzinger et al., 2025c) for general omega-regular specifications. Thus, we follow this unified presentation to design a runtime monitoring framework that applies to all these supermartingale certificates.

Example 1 (A robot climbing a hill)**.** A robot starts at distance $x_0 = 10$ from its target. Each step moves it forwards by 0.01 or makes it slip backwards by the same distance. It stops upon reaching distance $[0, 1]$ from the target, where it remains. A sloping section between positions 3 and 7 increases the probability of slipping backwards. As the robot moves from 7 towards 3, the slope becomes steeper until position 5, then gradually flattens. We describe this steepness profile by $b(x) \coloneqq (1 - |x - 5|/2)_+$. Thus, position 5 is the steepest part of the slope. Before reaching the target, the dynamics for $a \in [0, 3/4]$ and $p_a(x) \coloneqq 1/4 + a \cdot b(x)$ are

$$X_{n+1} = X_n - 0.01 + 0.02 \cdot Z_n, \qquad Z_n \mid \mathcal{G}_n \sim \text{Bernoulli}\big(p_a(X_n)\big).$$

Each draw uses fresh randomness with slip probability determined by the current state. The range of a ensures that $p_a(x) \in [0, 1]$. If we choose $f(x) = x$ and $l(x) = -0.005 \cdot \mathbf{1}\{x > 1\}$, the expected decrease before stopping is $d_a(x) \coloneqq \mathbb{E}[f(X_{n+1}) - f(X_n) \mid X_n = x] = -0.005 + 0.02 \cdot a \cdot b(x)$. If $a = 0$, f is a ranking supermartingale proving almost-sure target reachability. However, for every $a > 0$, the expected decrease condition is violated on $(3, 7)$.

Trajectory Given a Markov process $\boldsymbol{X} = (X_t)_{t \in \mathbb{T}}$ and a measurable function $f : \mathcal{X} \to \mathbb{R}$, the goal of the supermartingale certificate monitoring problem is to construct a monitor which observes states of the Markov process at different time steps and, upon each newly observed state, issues a verdict on whether the function f violates one of the supermartingale conditions or not. To that end, we define $\mathcal{Y} \coloneqq \mathbb{T} \times \mathcal{X}$ to be the set of time–state pairs, and use $\mathcal{Y}^*$ to denote the set of all finite sequences of time-state pairs. We call elements of $\mathcal{Y}^*$ *trajectories*.

Monitor A *monitor* is a tuple $\mathfrak{M} \coloneqq (\pi, \xi)$, where $\pi \colon \mathcal{Y}^* \to \mathbb{T}$ is a *sampling rule* which for each sampled trajectory specifies the next observation time, and $\xi \colon \mathcal{Y}^* \to \{0, 1\}$ is a *verdict rule* which for each sampled trajectory decides whether to raise an alarm or not. The verdict value 1 denotes an *alarm*, indicating statistical evidence of a certificate violation, while 0 means that no alarm has been raised. The semantics of a monitor are defined inductively as follows. At the beginning, the monitor starts with a single-state trajectory $\boldsymbol{Y}_{0:0} = (T_0, x_{T_0})$, with $T_0 = 0$ and x_{T_0} sampled from X_{T_0}. Then, for each n, if $\boldsymbol{Y}_{0:n} = (Y_0, \ldots, Y_n)$ is the trajectory of time-state samples that have been sampled after n steps, the next time-state sample and the monitor verdict are computed via

$$T_{n+1} \coloneqq \pi(\boldsymbol{Y}_{0:n}), \quad x_{T_{n+1}} \sim X_{T_{n+1}}, \quad \boldsymbol{Y}_{0:n+1} \coloneqq (\boldsymbol{Y}_{0:n}, (T_{n+1}, x_{T_{n+1}})), \quad V_{n+1} \coloneqq \xi(\boldsymbol{Y}_{0:n+1}).$$

Hence, the monitor determines the next sampling time T_{n+1}, samples the next state $x_{T_{n+1}}$ from $X_{T_{n+1}}$, appends the new time-state pair to the trajectory to obtain trajectory $\boldsymbol{Y}_{0:n+1}$, and finally computes the verdict V_{n+1} by applying the verdict rule to the trajectory $\boldsymbol{Y}_{0:n+1}$.

Problem Statement Our goal is to design a monitor that raises an alarm as accurately as possible, while providing *probabilistic guarantees* that its false alarm rate is below some tolerable threshold δ *over all sampling times*, and not just for each individual sampled state. The accuracy of our monitor is an empirical claim that will be evaluated in our experiments. However, the false alarm rate is a property that we want to *guarantee*. We define it as follows. Given a candidate certificate f and an error tolerance $\delta \in (0, 1)$, we want to construct a monitor $\mathfrak{M} \coloneqq (\pi, \xi)$ such that, whenever f satisfies the certificate conditions for the deployed process, we have $\mathbb{P}\left(\exists n \in \mathbb{N}_0 : V_n = 1\right) \leq \delta$.

Challenge To determine whether f is a supermartingale certificate, it suffices to design one monitor for each Domain bound condition and each Expected decrease condition of the supermartingale. But designing a monitor for Domain bound conditions is trivial – if $S \subseteq \mathcal{X}$ is the domain and $[l, u]$ is

the bound interval, the verdict rule is simply $V_n = \xi(\boldsymbol{Y}_{0:n}) = 1$ iff the last time-state pair (T_n, x_n) satisfies $x_n \in S$ but $f(x_n) \notin [l, u]$. Such monitor design ensures that the false alarm rate is 0.

Hence, the challenge in supermartingale certificate monitoring is to compute a monitor for the expected decrease condition, and in the rest of this work we focus on that. Specifically, our construction uses fixed observation times $t_n = nh$ and bounds the probability of raising an alarm before any completed observation interval has positive conditional expected corrected drift. Formally, let $L_n \coloneqq \sum_{r=t_n}^{t_{n+1}-1} l(X_r)$ in discrete time and $L_n \coloneqq \int_{t_n}^{t_{n+1}} l(X_r)\,\mathrm{d}r$ in continuous time denote the *accumulated allowed drift*. Writing τ_δ for the first alarm index and $E_n \coloneqq \mathbb{E}[f(X_{t_{n+1}}) - f(X_{t_n}) - L_n \mid \mathcal{F}_n]$ for the conditional expected corrected drift, we prove

$$\mathbb{P}\left(\tau_\delta < \infty \text{ and } E_n \le 0 \text{ for every } n < \tau_\delta\right) \le \delta.$$

3 Runtime monitoring as betting-based sequential testing

We use a betting-based sequential testing framework (Ramdas et al., 2023) as the main underlying ingredient to construct a monitor for supermartingale certificates. In this section, we first show how a sequence of observed states can be transformed into scores that provide evidence of a certificate violation. We then define a general monitoring scheme, namely an *abstract monitor*, that guarantees false-alarm control for any admissible betting strategy. Lastly, we instantiate the abstract monitor with concrete strategies designed to improve detection power while preserving monitor validity.

A supermartingale $(M_n)_{n\in\mathbb{N}_0}$ is *nonnegative* if $M_n \ge 0$ almost surely for every $n \ge 0$. In the betting-based framework, we use *Ville's inequality* to turn such a process into a monitor with a bound on the probability of ever raising a false alarm. Ville's inequality states that, for any nonnegative supermartingale $(M_n)_{n\in\mathbb{N}_0}$ with $M_0 = 1$ and for every $\delta \in (0, 1)$, the probability of the supermartingale value ever exceeding $1/\delta$ is at most δ, i.e.,

$$\mathbb{P}\left(\exists n \in \mathbb{N}_0 : M_n \ge 1/\delta\right) \le \delta. \tag{1}$$

We set the first alarm index τ_δ and the verdict V_n to

$$\tau_\delta \coloneqq \inf\{n \in \mathbb{N}_0 : M_n \ge 1/\delta\}, \quad \text{and} \quad V_n \coloneqq \mathbf{1}\{\tau_\delta \le n\} \quad \text{with } \inf \varnothing \coloneqq \infty.$$

This defines a statistical test, such that whenever the null hypothesis ensures that $(M_n)_{n\in\mathbb{N}_0}$ is a nonnegative supermartingale, the probability of ever raising an alarm is at most δ.

3.1 Scoring schemes

To apply the betting framework, we assign a numerical score to each pair of consecutive observations. A positive score indicates that the observed certificate change exceeds the allowed drift upper bound. However, a positive score can also occur by chance. We construct the scores so that, whenever the expected decrease condition holds, the expected next score, given the observations so far, is at most zero. This ensures that betting on these scores cannot increase wealth in conditional expectation while the condition holds. We also ensure that each score is at least -1 almost surely, so that admissible bets cannot make the wealth negative.

Assumptions The monitor observes the Markov process $\boldsymbol{X}$ at times $t_n = nh$, with fixed intervals $h \in \mathbb{R}_{>0}$ in continuous time and $h = 1$ in discrete time. We write $\mathcal{F}_n \coloneqq \sigma(X_{t_0}, \ldots, X_{t_n})$ for the information available to the monitor at t_n. Further in continuous time, the deployed process is absorbed at its first exit from $\mathcal{S}$, retaining its terminal state. For example, for a reach–avoid certificate, the system terminates upon reaching the target or an unsafe state.

We assume that the monitor can compute two bounds from its observations. First, after observing a state at t_{n+1}, it can compute a finite upper bound $\overline{L}_n \ge L_n$ almost surely on the accumulated allowed drift. Subtracting this bound from the observed certificate change gives the *corrected certificate change* $D_n \coloneqq f(X_{t_{n+1}}) - f(X_{t_n}) - \overline{L}_n$. Second, at t_n and before observing state $X_{t_{n+1}}$, the monitor must already be able to compute a finite *loss bound* $B_n \ge b_{\min}$, where $b_{\min} > 0$ is fixed, guaranteeing $D_n \ge -B_n$ almost surely. This ensures that the normalised score D_n/B_n is at least -1. Formally, $\overline{L}_n$ is $\mathcal{F}_{n+1}$-measurable and B_n is $\mathcal{F}_n$-measurable, and these requirements hold for every $n \in \mathbb{N}_0$. Moreover, we assume that the random variables $f(X_{t_n})$, L_n, and $\overline{L}_n$ are integrable.

Concrete conservative bounds can be obtained if we know a lower bound on the certificate value $f(x) \geq f_{\min}$ and an upper bound on the allowed drift $l(x) \leq l_{\max}$ for all states reachable. Then we get $\overline{L}_n = hl_{\max}$ and $B_n = \max\{b_{\min}, f(X_{t_n}) + hl_{\max} - f_{\min}\}$.

Scoring The monitor compares the observed certificate change with the upper bound on the allowed drift and normalises it by the loss bound. A positive score therefore indicates that the observed certificate change exceeds the drift upper bound. The score is bounded below by -1 and has nonpositive conditional mean whenever the certificate condition holds. Formally, upon observing a state at t_{n+1}, the monitor computes a raw score D_n and the normalised score W_n:

$$D_n \coloneqq f(X_{t_{n+1}}) - f(X_{t_n}) - \overline{L}_n, \qquad W_n \coloneqq D_n / B_n.$$

Let $\mu_n \coloneqq \mathbb{E}[W_n \mid \mathcal{F}_n]$ denote the expected next score given the observations so far. The following lemma shows that $\mu_n \leq 0$ whenever the supermartingale expected decrease condition holds. Together with the lower bound $W_n \geq -1$ ensured by normalisation, this makes the scores *valid* for the betting construction, where admissible bets preserve nonnegative wealth and cannot increase it in conditional expectation while the certificate condition holds. The proof is deferred to Appendix B.

Lemma 1 (Score validity)**.** *Recall that $E_n \coloneqq \mathbb{E}[f(X_{t_{n+1}}) - f(X_{t_n}) - L_n \mid \mathcal{F}_n]$ is the conditional expected corrected drift over $[t_n, t_{n+1}]$. For every $n \in \mathbb{N}_0$,*

$$\mu_n = \frac{E_n - \mathbb{E}[\overline{L}_n - L_n \mid \mathcal{F}_n]}{B_n} \leq \frac{E_n}{B_n} \qquad \textit{almost surely}. \tag{2}$$

Thus, if the expected decrease condition holds and $E_n \leq 0$, then $\mu_n \leq 0$ almost surely on $\{X_{t_n} \in \mathcal{S}\}$.

Example 2. Consider the hill robot in Example 1, with observation interval $h = 1$. Since f is nonnegative on reachable states and $l \leq 0$, valid bounds are $\overline{L}_n = 0$ and $B_n = \max\{b_{\min}, f(X_{t_n})\}$. These bounds only assume that the certificate cannot decrease by more than its current value. However, we can use the known system dynamics to derive more informative bounds. The allowed drift is exactly $l(X_{t_n})$, and a forward step decreases the certificate by only 0.01. Before stopping, the corrected increment is therefore at least $-0.01 + 0.005 = -0.005$. We can thus choose $\overline{L}_n = l(X_{t_n})$ and $B_n = b_{\min} = 0.02$, giving

$$W_n = \frac{f(X_{t_{n+1}}) - f(X_{t_n}) + 0.005}{0.02} = Z_n - \tfrac{1}{4}, \qquad \mu_n = a \cdot b(X_{t_n}).$$

Hence, a forward step results in $W_n = -1/4$, while a slip gives $W_n = 3/4$, so $W_n \geq -1$ for every amplitude, which validates the loss bound. After stopping, $W_n = 0$.

3.2 Runtime monitoring under general betting strategies

We describe how the monitor accumulates evidence from past scores and decides to raise an alarm. Our monitor combines multiple wealth accounts and alarms when its total wealth reaches $1/\delta$. The construction here is *abstract* because we do not yet specify the betting strategies in this subsection.

A single positive score may occur by chance, so the monitor accumulates evidence by betting on successive scores, following Ramdas et al. (2023). At each observation index $s \in \mathbb{N}_0$, we may launch a set of betting strategies $\mathcal{A}_s$, called *experts*. For each expert $a \in \mathcal{A}_s$, let $G^a_{s,n}$ denote its normalised wealth at index $n \geq s$, starting from $G^a_{s,s} = 1$. The capital allocated to these accounts is specified below. For every $n \geq s$, if $X_{t_n} \in \mathcal{S}$, expert a chooses a $\mathcal{F}_n$-measurable stake $\lambda^a_{s,n} \in [0, 1]$ before observing W_n, and otherwise $\lambda^a_{s,n} = 0$ if $X_{t_n} \notin \mathcal{S}$. Then the account wealth is updated via:

$$G^a_{s,n+1} \coloneqq G^a_{s,n}(1 + \lambda^a_{s,n} W_n)$$

Since $W_n \geq -1$, wealth remains nonnegative. The stake controls the gains from positive scores and the losses from negative scores. A good betting strategy can therefore help wealth grow rapidly when the certificate condition fails, allowing the monitor to raise an alarm sooner.

Mixing experts Different betting strategies may detect different violations more quickly. Since the monitor does not know in advance which strategy will work best, it combines their wealth accounts. Specifically, for each observation index s, the monitor combines the accounts in $\mathcal{A}_s$ using fixed,

nonnegative, $\mathcal{F}_s$-measurable weights $\eta_{s,a}$ satisfying $\sum_{a\in\mathcal{A}_s}\eta_{s,a}=1$. The combined wealth is $Q_{s,n}\coloneqq\sum_{a\in\mathcal{A}_s}\eta_{s,a}G^a_{s,n}$ with $Q_{s,s}=1$.

A mixture may lose wealth before a violation begins. To launch fresh mixtures without earlier losses, the monitor reserves part of its initial unit capital. The monitor starts with one unit of capital. Let c_s denote the amount allocated to the mixture launched at observation s, with $c_s=0$ if no mixture is funded. Before launching the mixture at index n, the monitor has already allocated $C_n\coloneqq\sum_{s=0}^{n-1}c_s$ to earlier mixtures, leaving $1-C_n$ in reserve. It allocates an amount $c_n\in[0,1-C_n]$ from this reserve to the new mixture. This amount may depend on the observations so far (c_n is $\mathcal{F}_n$-measurable). Launching a mixture therefore transfers existing capital rather than adding new capital. The total monitor wealth is the remaining reserve plus the current wealth of all funded mixtures:

$$M_n\coloneqq 1-C_n+\sum_{s=0}^{n-1}c_s\sum_{a\in\mathcal{A}_s}\eta_{s,a}\prod_{k=s}^{n-1}(1+\lambda^a_{s,k}W_k),\qquad M_0=1.$$

Since $Q_{n,n}=1$, funding a new mixture does not change the total wealth. For example, funding every $q\in\mathbb{N}_{\geq 1}$ observations can use $c_{jq}=6/(\pi^2(j+1)^2)$ for $j\in\mathbb{N}_0$ and $c_n=0$ otherwise. The allocations sum to one and leave reserve $6/\pi^2\sum_{i=j+2}^{\infty}i^{-2}>0$ so there always remains fresh capital.

Probabilistic guarantees If the certificate condition holds almost surely throughout execution, Lemma 1 gives $\lambda^a_{s,n}\mu_n=\lambda^a_{s,n}\mathbb{E}[W_n\mid\mathcal{F}_n]\leq 0$, and since the current wealth and the stake are determined by the observations available before W_n is observed, we obtain

$$\mathbb{E}[G^a_{s,n+1}\mid\mathcal{F}_n]=G^a_{s,n}(1+\lambda^a_{s,n}\mu_n)\leq G^a_{s,n}.$$

Thus, each expert account is a nonnegative supermartingale for any admissible stake strategy. Combining these accounts using fixed mixture weights preserves this property for $Q_{s,n}$. Funding new mixtures from the remaining reserve likewise preserves it for M_n. The threshold in (1) therefore applies to the total wealth regardless of the admissible stakes, mixture weights, and funding chosen.

Theorem 2. *For every admissible choice of stakes, expert weights, and funding, the monitor satisfies*

$$\mathbb{P}\left(\tau_\delta<\infty \text{ and } E_k\leq 0 \text{ for every } k<\tau_\delta\right)\leq\delta, \tag{3}$$

and, if the expected-decrease condition holds for the deployed process, then $\mathbb{P}(\tau_\delta<\infty)\leq\delta$.

Inequality (3) bounds the probability of raising an alarm before the expected decrease condition is violated, thus bounding the false-positive rate of the monitor, as desired. The proof is in Appendix B.

Detection power We also bound the probability that the monitor raises an alarm within a block of observations during which the supermartingale expected decrease condition is persistently violated. Consider observation intervals indexed by $k=s,\dots,s+N-1$, and suppose that there exists $\varepsilon\in(0,1]$ such that $\mathbb{E}[W_k\mid\mathcal{F}_k]\geq\varepsilon$ and $W_k\leq 1$ almost surely, for each k. Suppose that a betting strategy a, launched at index $r\leq s$, has capital $\rho\coloneqq c_r\eta_{r,a}G^a_{r,s}>0$ at observation s and uses the constant stake $\varepsilon/2$ over these intervals. Then, for any $\gamma\in(0,1)$,

$$\mathbb{P}\left(\tau_\delta\leq s+N\mid\mathcal{F}_s\right)\geq 1-\gamma\quad\text{almost surely on}\quad\left\{N\geq\frac{8}{\varepsilon^2}\log\frac{1}{\delta\rho\gamma}\right\}.$$

Thus, if the violation persists for sufficiently many observation intervals, the monitor raises an alarm by observation $s+N$ with conditional probability at least $1-\gamma$. The parameter ε is a lower bound on the expected score at each observation. The larger value of ε reduces the number of observation intervals required to detect violation. A larger starting capital ρ also reduces this number, because less wealth growth is needed to reach the alarm threshold $1/\delta$.

3.3 Weighted prediction strategies

We now give two concrete strategies for choosing the stake by predicting the next score from past scores. The better the monitor predicts the next score, the faster it can grow its capital. The prediction is based on the fact that many realistic systems exhibit regularity in space and time. Here, *spatial regularity* means that nearby states have similar conditional expected scores, whereas *temporal regularity* means that these expectations change slowly over time. We consider a mixture over several experts, where each expert guesses and weights past scores according to their spatial and temporal proximity to the current observation. If the guesses are accurate, this weighting can improve our estimate of the conditional mean of the next score.

Prediction weights For computational efficiency, we limit ourselves to a finite history, i.e., let $m \geq 1$ be the number of scores retained in the history, and define the retained indices $I_n := \{i \in \mathbb{N}_0 : \max\{0, n-m\} \leq i < n\}$. Let $a \in \mathcal{A}_s$ be an expert launched at observation index s. The expert chooses a spatial scale $r^a_{\mathcal{X},s}$ and a temporal scale $r^a_{\mathbb{T},s}$, each in $(0, \infty]$. We describe how the scales are obtained in Appendix C. The scales serve as input to a spatial kernel $k^a_{\mathcal{X},s}$ and a temporal kernel $k^a_{\mathbb{T},s}$, respectively. Both kernels are nonincreasing, take values in $[0, 1]$, and equal one at zero. The kernels and scales are fixed when the expert is launched. At each observation index $n \geq s$, a assigns each retained score W_i, $i \in I_n$, a prediction weight based on its starting time t_i and starting state X_{t_i}:

$$\omega^a_{s,n,i} := k^a_{\mathbb{T},s}\left(\frac{t_n - t_i}{r^a_{\mathbb{T},s}}\right) k^a_{\mathcal{X},s}\left(\frac{\|X_{t_n} - X_{t_i}\|}{r^a_{\mathcal{X},s}}\right), \qquad i \in I_n. \tag{4}$$

Intuitively, recent observations and observations starting near the current state therefore receive greater weight. An infinite scale removes the corresponding dependence on distance, using the convention $u/\infty = 0$, e.g., temporal-only experts use $r^a_{\mathcal{X},s} = \infty$. Setting both scales to infinity gives uniform weights over the retained history.

Betting strategies Next, we transform past scores and their prediction weights into stakes, using two concrete strategies, for when $X_{t_n} \in S$. A stake λ multiplies the wealth by $1 + \lambda W_n$, so its *log return* is $\log(1 + \lambda W_n)$. Log returns add across successive bets and therefore measure accumulated multiplicative wealth growth. *The log-return strategy* maximises the weighted historical log return:

$$\lambda^{a,\log}_{s,n} := \min \arg\max_{0 \leq \lambda \leq \overline{\lambda}} \sum_{i \in I_n} \omega^a_{s,n,i} \log(1 + \lambda W_i) \quad \text{for } 1/2 \leq \overline{\lambda} < 1 \tag{5}$$

A second strategy is *the moment strategy*, which maximises a quadratic lower bound on the weighted historical log return. Since $\log(1 + \lambda w) \geq \lambda w - \lambda^2 w^2$ for $w \geq -1$ and $\lambda \in [0, 1/2]$, this bound is $\lambda S^a_{s,n} - \lambda^2 U^a_{s,n}$, where $S^a_{s,n} := \sum_{i \in I_n} \omega^a_{s,n,i} W_i$ and $U^a_{s,n} := \sum_{i \in I_n} \omega^a_{s,n,i} W_i^2$. Its maximiser is

$$\lambda^{a,\mathrm{mom}}_{s,n} := \min\left\{\frac{1}{2}, \frac{(S^a_{s,n})_+}{2U^a_{s,n}}\right\}, \qquad (0/0 := 0). \tag{6}$$

Thus positive weighted scores favour betting, whereas large score magnitudes reduce the stake.

Both strategies choose zero when the weighted sum of the retained scores is nonpositive, including for an empty history. The moment strategy requires only two sums and bets no more than the log-return strategy for the same weights. At each launch, the monitor can include a range of experts $\mathcal{A}$ with different configurations of scales, kernels and strategies, to accommodate different violation patterns. Their accounts are then combined using the capital weights $\eta_{s,a}$ from Section 4.2.

4 Experimental Evaluation

We evaluate on a wide range of benchmarks whether our monitoring framework can effectively detect violations of supermartingale certificate conditions. We aim to address the following research questions. Can the monitors detect certificate violations under changed system dynamics (RQ1)? How does temporal and spatial scaling impact the power of monitoring (RQ2)?

Benchmarks We consider three benchmark suites. The *synthetic* suite consists of six families that isolate persistent, delayed, temporary, and recurring local violations. The *control* suite comprises three systems which are variants of Building and Tanks from PRoTECT (Wooding et al., 2026), together with an LQR system. Their expected certificate increments can be computed exactly, providing ground truth for evaluating detection. The *neural* suite uses 16 learned controller–certificate pairs from LogRASM (Badings et al., 2025) to assess monitoring under modified deployment dynamics.

Monitors We use three prediction schemes: uniform (U), which weights retained scores equally; temporal (T), which favours recent observations; and spatial (S), which favours observations from nearby states. On the *synthetic* benchmarks, we compare moment (M) and log-return (L) strategies. The *control* and *neural* experiments use log-return and equal mixtures over five scales for T and S. Each monitor uses a false-alarm tolerance of $\delta = 0.05$ and raises an alarm when its betting wealth reaches $1/\delta = 20$. Full configurations and implementation details are provided in the Appendix.

Table 1: Violation strength and detection in selected *control* settings, with 50 runs each. For each run, magnitude is the average of μ_n over violating observations, and exposure is the percentage of violating observations. Both of them report detection rates across runs. Detections entries give counts, with median delay in parentheses. Detection delay is measured in observations from violation onset or first entry into the violating region. U-L, T-L, and S-L use uniform, temporal, and spatial prediction, respectively. NR means fewer than 25 runs alarm.

			Detections (median delay)		
Benchmark & Setting	Magnitude	Exposure (%)	U-L	T-L	S-L
Building: 8% heating loss	0.0151	24.9	0 (NR)	0 (NR)	50 (3043)
Building: 12% heating loss	0.0309	37.6	0 (NR)	0 (NR)	50 (732)
Tanks: burst probability 0.35	0.00561	5.5	0 (NR)	0 (NR)	0 (NR)
Tanks: burst probability 0.45	0.00615	11.0	0 (NR)	0 (NR)	34 (28507)
LQR: 5% feedback loss	0.00955	100.0	19 (NR)	12 (NR)	16 (NR)
LQR: 20% feedback loss	0.0383	100.0	50 (2241)	50 (1349)	50 (2264)

Table 2: Median detection delay in observations on the six *synthetic* benchmark families: persistent, delayed, temporary, direct recurrence, stochastic ring, and hill. XS–XL denote extra-small through extra-large prediction scales (exact parameters given in Appendix). Temporal scales determine how far back in the history observations receive weight; spatial scales determine the neighbourhood radius around the current state. *Equal* divides capital equally among the five scale experts. *Uniform* weights all retained observations equally. NR means fewer than 25 runs alarm.

Temporal prediction							Spatial prediction						
Scale	Pers.	Del.	Temp.	Rec.	Ring	Hill	Scale	Pers.	Del.	Temp.	Rec.	Ring	Hill
Uniform	2126	3204	NR	NR	2360	588	Uniform	2126	3204	NR	NR	2360	588
Equal	2183	668	914	1102	664	492	Equal	3351	2141	NR	571	654	482
XS	NR	1565	NR	570	168	568	XS	NR	3979	NR	NR	NR	670
S	6719	756	NR	2110	1127	441	S	NR	2977	NR	NR	655	594
M	2442	575	766	NR	1713	476	M	21051	3437	NR	1578	640	552
L	2118	1174	NR	NR	2338	560	L	7406	2275	NR	1040	1652	447
XL	2127	2439	NR	NR	2347	587	XL	2190	2379	NR	1576	2153	472

RQ1: Detection of certificate violations. Our results show that the monitors detect certificate violations under changed dynamics, with spatial prediction being particularly effective when violations are localised in the state space. Table 1 illustrates how violation magnitude and exposure affect detection. In Building, S-L detects every run, and its median delay falls from 3043 to 732 observations as violations become stronger and more frequent. The weaker, less frequent violations in Tanks remain harder to detect, even with spatial prediction. For LQR, we weaken the controller by reducing the applied control input. With a 5% reduction, violations occur throughout each trajectory, yet each monitor detects fewer than half the runs. With a 20% reduction, all monitors detect every run. Overall this shows that even persistent violations can remain undetected when their effect is too small to distinguish reliably from stochastic fluctuations within the observed trajectory.

Table 3 summarises the mean eligible first-alarm percentages across a number of disturbed settings. On the *control* benchmarks, S-L achieves 83.8%, compared with 29.7% for U-L and 28.2% for T-L. Specifically in Building and Tanks, the certificate condition fails in some regions even though it holds on average along the trajectory. This suggests spatial prediction helps detect these local failures by emphasising observations from nearby states. The *neural* benchmarks show a similar advantage under expected decrease ($\epsilon = 0$), where S-L raises alarms on 43.1% of runs, compared with approximately 11% for either baseline. The higher alarm rates for positive ϵ concern stronger decrease requirements tested on the same trajectories.

RQ2: Temporal vs. spatial scaling impact Our results show that uniform scaling, while sufficient for long persistent violations, struggle on traces with transient violations. Both temporal and spatial spacing are able to detect such behaviour. Spatial spacing, in particular, benefits from recurrent

Table 3: Percentage of runs in which each monitor raises an alarm, averaged across settings (50 runs per setting). For *neural*, $\epsilon = 0$ tests expected nonincrease; larger ϵ requires faster expected decrease.

Benchmark set	No. Settings	U-L(%)	T-L(%)	S-L(%)
Synthetic	18	62.2	77.0	79.6
Control	18	29.7	28.2	83.8
Neural, $\epsilon = 0$	48	10.7	10.8	43.1
Neural, $\epsilon = 0.0005$	48	71.9	71.7	77.8
Neural, $\epsilon = 0.001$	48	76.1	76.3	81.0

observations. The correct scaling depends on the system, i.e., no uniformly best scaling is observed. However, mixing over scales can retain some of the benefits of the respective scales, being often only slightly worse than the best for the respective benchmark. Table 2 summarises the results. We can observe that the uniform monitor has a median detection delay of 2126 on persistent violations, on par with large spatial and temporal scales. By contrast, for temporary and recurrent violations (Rec. and Ring) its medium detection delay is either larger or less than 50% of the runs detect a fault. For example, on the spatial recurring ring benchmark, the medium spatial monitor has a delay of 640, while the uniform monitor has a delay of 2360; on the temporary fault benchmark only the medium temporal monitor alarmed in more than 50% of the cases.

5 Related work

Supermartingale certificates have found broad applications in probabilistic program verification, stochastic (neural) control (Badings et al., 2025; Zikelic et al., 2023b; Chatterjee et al., 2017; 2022; Takisaka et al., 2021; Majumdar & Sathiyanarayana, 2025; Kura & Unno, 2026; Prajna et al., 2007; Ansaripour et al., 2023; Xue et al., 2024; Abate et al., 2024; Henzinger et al., 2025c; Abate et al., 2026). Our work is closest to recent approaches for runtime monitoring of neural Lyapunov certificates in deterministic systems (Yu et al., 2025; Henzinger et al., 2025b). Runtime monitoring is comparatively straightforward in the deterministic setting: a violation of the required decrease condition is directly observable from a single state transition. For supermartingale certificates, however, monitoring is fundamentally harder because the required decrease holds only in expectation and therefore cannot be assessed from a single transition. We overcome this challenge using betting-based sequential testing (Ramdas et al., 2023; Waudby-Smith & Ramdas, 2024; Timans et al., 2025), which we apply for the first time to supermartingale certificates for temporal properties. We further exploit temporal and spatial regularity in stochastic systems to make informed bets and improve detection power.

Runtime monitoring is an established technique for ensuring correctness of cyber-physical systems (Bartocci et al., 2018; Bauer et al., 2011), providing a lightweight, black-box alternative to formal verification. Existing methods typically detect property violations directly from observed states, fundamentally restricting monitorable properties to safety (Alpern & Schneider, 1985; Pnueli & Zaks, 2006) and excluding properties such as reachability, stability, and general ω-regular properties. Predictive monitoring (Bortolussi et al., 2019; Lindemann et al., 2023; Henzinger et al., 2025a) extends this scope by forecasting future states, but remains limited to finite time horizons. In contrast, by monitoring supermartingale certificates, our approach can detect violations of general temporal properties over unbounded horizons. Related work (Hindy et al., 2024) uses martingale theory to detect distribution shifts in stochastic processes, but does not monitor the validity of certificates.

6 Conclusion

We introduced the first framework for runtime monitoring of supermartingale certificates for temporal properties, that is applicable to both discrete- and continuous-time systems and to a broad range of supermartingale certificates. Our framework provides rigorous probabilistic guarantees on the false positive rate being below a given threshold. Experimental evaluation demonstrates the practical effectiveness of our method for detecting supermartingale certificate violations at runtime in a range of stochastic control tasks. Interesting directions of future work include designing shielding and repair procedures that would use the alarms raised by our runtime monitoring framework.

AI USE STATEMENT

We used ChatGPT to assist with writing, literature search, theoretical development, and the design and implementation of experiments. We describe these uses below. For writing, we used these tools to draft and revise text, simplify explanations, shorten sections, and improve consistency in notation and terminology. We also used them to assist with LaTeX formatting, e.g., constructing tables. For literature search, we used these tools to identify relevant work and compare its assumptions and guarantees with ours. For theoretical development, we used these tools, to suggest proof arguments, expand derivations, and identify possible gaps and counterexamples. They also helped clarify certain measure-theoretic arguments required in the continuous-time setting. For experiments, we used these tools to write basic code and generate plots. They supported the selection of system disturbances. They were used to draft the experimental setup in the appendix. The authors take full responsibility for the final content of the paper, including all AI-assisted text, mathematical claims and proofs, references, code, and experimental results.

REFERENCES

Alessandro Abate, Mirco Giacobbe, and Diptarko Roy. Learning probabilistic termination proofs. In Alexandra Silva and K. Rustan M. Leino (eds.), *Computer Aided Verification - 33rd International Conference, CAV 2021, Virtual Event, July 20-23, 2021, Proceedings, Part II*, volume 12760 of *Lecture Notes in Computer Science*, pp. 3–26. Springer, 2021. doi: 10.1007/978-3-030-81688-9_1. URL https://doi.org/10.1007/978-3-030-81688-9_1.

Alessandro Abate, Mirco Giacobbe, and Diptarko Roy. Stochastic omega-regular verification and control with supermartingales. In Arie Gurfinkel and Vijay Ganesh (eds.), *Computer Aided Verification - 36th International Conference, CAV 2024, Montreal, QC, Canada, July 24-27, 2024, Proceedings, Part III*, volume 14683 of *Lecture Notes in Computer Science*, pp. 395–419. Springer, 2024. doi: 10.1007/978-3-031-65633-0_18. URL https://doi.org/10.1007/978-3-031-65633-0_18.

Alessandro Abate, Mirco Giacobbe, and Diptarko Roy. Quantitative supermartingale certificates. In Ruzica Piskac and Zvonimir Rakamaric (eds.), *Computer Aided Verification - 37th International Conference, CAV 2025, Zagreb, Croatia, July 23-25, 2025, Proceedings, Part II*, volume 15932 of *Lecture Notes in Computer Science*, pp. 3–28. Springer, 2025. doi: 10.1007/978-3-031-98679-6_1. URL https://doi.org/10.1007/978-3-031-98679-6_1.

Alessandro Abate, Mirco Giacobbe, Sergey Ichtchenko, and Diptarko Roy. Complete ω-regular supermartingale certificates. In Claudia Faggian and Joost-Pieter Katoen (eds.), *41st Annual Symposium on Logic in Computer Science, LICS 2026, Lisbon, Portugal, July 20-23, 2026*, volume 380 of *LIPIcs*, pp. 3:1–3:28. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2026. doi: 10.4230/LIPICS.LICS.2026.3. URL https://doi.org/10.4230/LIPIcs.LICS.2026.3.

Bowen Alpern and Fred B. Schneider. Defining liveness. *Inf. Process. Lett.*, 21(4):181–185, 1985. doi: 10.1016/0020-0190(85)90056-0. URL https://doi.org/10.1016/0020-0190(85)90056-0.

Matin Ansaripour, Krishnendu Chatterjee, Thomas A. Henzinger, Mathias Lechner, and Dorde Zikelic. Learning provably stabilizing neural controllers for discrete-time stochastic systems. In *Automated Technology for Verification and Analysis - 21st International Symposium, ATVA 2023, Singapore, October 24-27, 2023, Proceedings, Part I*, pp. 357–379, 2023. doi: 10.1007/978-3-031-45329-8_17. URL https://doi.org/10.1007/978-3-031-45329-8_17.

Thom Badings, Wietze Koops, Sebastian Junges, and Nils Jansen. Policy verification in stochastic dynamical systems using logarithmic neural certificates. In Ruzica Piskac and Zvonimir Rakamaric (eds.), *Computer Aided Verification - 37th International Conference, CAV 2025, Zagreb, Croatia, July 23-25, 2025, Proceedings, Part II*, volume 15932 of *Lecture Notes in Computer Science*, pp. 349–375. Springer, 2025. doi: 10.1007/978-3-031-98679-6_16. URL https://doi.org/10.1007/978-3-031-98679-6_16.

Ezio Bartocci, Jyotirmoy V. Deshmukh, Alexandre Donzé, Georgios Fainekos, Oded Maler, Dejan Nickovic, and Sriram Sankaranarayanan. Specification-based monitoring of cyber-physical systems:

A survey on theory, tools and applications. In Ezio Bartocci and Yliès Falcone (eds.), *Lectures on Runtime Verification - Introductory and Advanced Topics*, volume 10457 of *Lecture Notes in Computer Science*, pp. 135–175. Springer, 2018. doi: 10.1007/978-3-319-75632-5_5. URL https://doi.org/10.1007/978-3-319-75632-5_5.

Andreas Bauer, Martin Leucker, and Christian Schallhart. Runtime verification for LTL and TLTL. *ACM Trans. Softw. Eng. Methodol.*, 20(4):14:1–14:64, 2011. doi: 10.1145/2000799.2000800. URL https://doi.org/10.1145/2000799.2000800.

Luca Bortolussi, Francesca Cairoli, Nicola Paoletti, Scott A. Smolka, and Scott D. Stoller. Neural predictive monitoring. In Bernd Finkbeiner and Leonardo Mariani (eds.), *Runtime Verification - 19th International Conference, RV 2019, Porto, Portugal, October 8-11, 2019, Proceedings*, volume 11757 of *Lecture Notes in Computer Science*, pp. 129–147. Springer, 2019. doi: 10.1007/978-3-030-32079-9_8. URL https://doi.org/10.1007/978-3-030-32079-9_8.

Aleksandar Chakarov and Sriram Sankaranarayanan. Probabilistic program analysis with martingales. In Natasha Sharygina and Helmut Veith (eds.), *Computer Aided Verification - 25th International Conference, CAV 2013, Saint Petersburg, Russia, July 13-19, 2013. Proceedings*, volume 8044 of *Lecture Notes in Computer Science*, pp. 511–526. Springer, 2013. doi: 10.1007/978-3-642-39799-8_34. URL https://doi.org/10.1007/978-3-642-39799-8_34.

Krishnendu Chatterjee, Hongfei Fu, and Amir Kafshdar Goharshady. Termination analysis of probabilistic programs through positivstellensatz's. In *CAV (1)*, volume 9779 of *Lecture Notes in Computer Science*, pp. 3–22. Springer, 2016.

Krishnendu Chatterjee, Petr Novotný, and Dorde Zikelic. Stochastic invariants for probabilistic termination. In Giuseppe Castagna and Andrew D. Gordon (eds.), *Proceedings of the 44th ACM SIGPLAN Symposium on Principles of Programming Languages, POPL 2017, Paris, France, January 18-20, 2017*, pp. 145–160. ACM, 2017. doi: 10.1145/3009837.3009873. URL https://doi.org/10.1145/3009837.3009873.

Krishnendu Chatterjee, Amir Kafshdar Goharshady, Tobias Meggendorfer, and Dorde Zikelic. Sound and complete certificates for quantitative termination analysis of probabilistic programs. In Sharon Shoham and Yakir Vizel (eds.), *Computer Aided Verification - 34th International Conference, CAV 2022, Haifa, Israel, August 7-10, 2022, Proceedings, Part I*, volume 13371 of *Lecture Notes in Computer Science*, pp. 55–78. Springer, 2022. doi: 10.1007/978-3-031-13185-1_4. URL https://doi.org/10.1007/978-3-031-13185-1_4.

Robert W. Floyd. Assigning meanings to programs. *Proceedings of Symposium on Applied Mathematics*, 19:19–32, 1967. URL http://laser.cs.umass.edu/courses/cs521-621.Spr06/papers/Floyd.pdf.

Thomas A. Henzinger, Fabian Kresse, Kaushik Mallik, Emily Yu, and Djordje Zikelic. Predictive monitoring of black-box dynamical systems. In Necmiye Ozay, Laura Balzano, Dimitra Panagou, and Alessandro Abate (eds.), *7th Annual Learning for Dynamics & Control Conference, Ann Arbor, MI, USA, 4-6 June 2025*, volume 283 of *Proceedings of Machine Learning Research*, pp. 804–816. PMLR, 2025a. URL https://proceedings.mlr.press/v283/henzinger25a.html.

Thomas A. Henzinger, Konstantin Kueffner, and Emily Yu. Formal verification of neural certificates done dynamically. In Bettina Könighofer and Hazem Torfah (eds.), *Runtime Verification - 25th International Conference, RV 2025, Graz, Austria, September 15-19, 2025, Proceedings*, volume 16087 of *Lecture Notes in Computer Science*, pp. 54–72. Springer, 2025b. doi: 10.1007/978-3-032-05435-7_4. URL https://doi.org/10.1007/978-3-032-05435-7_4.

Thomas A. Henzinger, Kaushik Mallik, Pouya Sadeghi, and Dorde Zikelic. Supermartingale certificates for quantitative omega-regular verification and control. In Ruzica Piskac and Zvonimir Rakamaric (eds.), *Computer Aided Verification - 37th International Conference, CAV 2025, Zagreb, Croatia, July 23-25, 2025, Proceedings, Part II*, volume 15932 of *Lecture Notes in Computer Science*, pp. 29–55. Springer, 2025c. doi: 10.1007/978-3-031-98679-6_2. URL https://doi.org/10.1007/978-3-031-98679-6_2.

Ali Hindy, Rachel Luo, Somrita Banerjee, Jonathan Kuck, Edward Schmerling, and Marco Pavone. Diagnostic runtime monitoring with martingales. *CoRR*, abs/2407.21748, 2024. doi: 10.48550/ARXIV.2407.21748. URL https://doi.org/10.48550/arXiv.2407.21748.

Francis Hirsch and Marc Yor. On temporally completely monotone functions for Markov processes. *Probability Surveys*, 9(none):253 – 286, 2012. doi: 10.1214/11-PS179. URL https://doi.org/10.1214/11-PS179.

Nick Jakobi, Phil Husbands, and Inman Harvey. Noise and the reality gap: The use of simulation in evolutionary robotics. In Federico Morán, Alvaro Moreno, Juan Julián Merelo Guervós, and Pablo Chacón (eds.), *Advances in Artificial Life, Third European Conference on Artificial Life, Granada, Spain, June 4-6, 1995, Proceedings*, volume 929 of *Lecture Notes in Computer Science*, pp. 704–720. Springer, 1995. doi: 10.1007/3-540-59496-5_337. URL https://doi.org/10.1007/3-540-59496-5_337.

Satoshi Kura and Hiroshi Unno. A hierarchy of supermartingales for ω-regular verification. *Proc. ACM Program. Lang.*, 10(PLDI):253–276, 2026. doi: 10.1145/3808257. URL https://doi.org/10.1145/3808257.

Harold J Kushner. On the stability of stochastic dynamical systems. *Proceedings of the National Academy of Sciences*, 53(1):8–12, 1965.

Mathias Lechner, Dorde Zikelic, Krishnendu Chatterjee, and Thomas A. Henzinger. Stability verification in stochastic control systems via neural network supermartingales. In *Thirty-Sixth AAAI Conference on Artificial Intelligence, AAAI 2022, Thirty-Fourth Conference on Innovative Applications of Artificial Intelligence, IAAI 2022, The Twelveth Symposium on Educational Advances in Artificial Intelligence, EAAI 2022 Virtual Event, February 22 - March 1, 2022*, pp. 7326–7336. AAAI Press, 2022. doi: 10.1609/AAAI.V36I7.20695. URL https://doi.org/10.1609/aaai.v36i7.20695.

Lars Lindemann, Xin Qin, Jyotirmoy V. Deshmukh, and George J. Pappas. Conformal prediction for STL runtime verification. In Sayan Mitra, Nalini Venkatasubramanian, Abhishek Dubey, Lu Feng, Mahsa Ghasemi, and Jonathan Sprinkle (eds.), *Proceedings of the ACM/IEEE 14th International Conference on Cyber-Physical Systems, ICCPS 2023, (with CPS-IoT Week 2023), San Antonio, TX, USA, May 9-12, 2023*, pp. 142–153. ACM, 2023. doi: 10.1145/3576841.3585927. URL https://doi.org/10.1145/3576841.3585927.

Rupak Majumdar and V. R. Sathiyanarayana. Sound and complete proof rules for probabilistic termination. *Proc. ACM Program. Lang.*, 9(POPL):1871–1902, 2025. doi: 10.1145/3704899. URL https://doi.org/10.1145/3704899.

Grigory Neustroev, Mirco Giacobbe, and Anna Lukina. Neural continuous-time supermartingale certificates. In Toby Walsh, Julie Shah, and Zico Kolter (eds.), *Thirty-Ninth AAAI Conference on Artificial Intelligence, Thirty-Seventh Conference on Innovative Applications of Artificial Intelligence, Fifteenth Symposium on Educational Advances in Artificial Intelligence, AAAI 2025, Philadelphia, PA, USA, February 25 - March 4, 2025*, pp. 27538–27546. AAAI Press, 2025. doi: 10.1609/AAAI.V39I26.34966. URL https://doi.org/10.1609/aaai.v39i26.34966.

Amir Pnueli and Aleksandr Zaks. PSL model checking and run-time verification via testers. In Jayadev Misra, Tobias Nipkow, and Emil Sekerinski (eds.), *FM 2006: Formal Methods, 14th International Symposium on Formal Methods, Hamilton, Canada, August 21-27, 2006, Proceedings*, volume 4085 of *Lecture Notes in Computer Science*, pp. 573–586. Springer, 2006. doi: 10.1007/11813040_38. URL https://doi.org/10.1007/11813040_38.

Stephen Prajna and Ali Jadbabaie. Safety verification of hybrid systems using barrier certificates. In Rajeev Alur and George J. Pappas (eds.), *Hybrid Systems: Computation and Control, 7th International Workshop, HSCC 2004, Philadelphia, PA, USA, March 25-27, 2004, Proceedings*, volume 2993 of *Lecture Notes in Computer Science*, pp. 477–492. Springer, 2004. doi: 10.1007/978-3-540-24743-2_32. URL https://doi.org/10.1007/978-3-540-24743-2_32.

Stephen Prajna, Ali Jadbabaie, and George J. Pappas. A framework for worst-case and stochastic safety verification using barrier certificates. *IEEE Trans. Autom. Control.*, 52(8):1415–1428, 2007. doi: 10.1109/TAC.2007.902736. URL https://doi.org/10.1109/TAC.2007.902736.

Aaditya Ramdas, Peter Grünwald, Vladimir Vovk, and Glenn Shafer. Game-theoretic statistics and safe anytime-valid inference. *Statistical Science*, 38(4):576 –601, November 2023. ISSN 0883-4237. doi: 10.1214/23-STS894.

Toru Takisaka, Yuichiro Oyabu, Natsuki Urabe, and Ichiro Hasuo. Ranking and repulsing supermartingales for reachability in randomized programs. *ACM Trans. Program. Lang. Syst.*, 43(2): 5:1–5:46, 2021. doi: 10.1145/3450967. URL https://doi.org/10.1145/3450967.

Alexander Timans, Rajeev Verma, Eric T. Nalisnick, and Christian A. Naesseth. On continuous monitoring of risk violations under unknown shift. In Silvia Chiappa and Sara Magliacane (eds.), *Conference on Uncertainty in Artificial Intelligence, Rio Othon Palace, Rio de Janeiro, Brazil, 21-25 July 2025*, volume 286 of *Proceedings of Machine Learning Research*, pp. 4204–4226. PMLR, 2025. URL https://proceedings.mlr.press/v286/timans25a.html.

Josh Tobin, Rachel Fong, Alex Ray, Jonas Schneider, Wojciech Zaremba, and Pieter Abbeel. Domain randomization for transferring deep neural networks from simulation to the real world. In *2017 IEEE/RSJ International Conference on Intelligent Robots and Systems, IROS 2017, Vancouver, BC, Canada, September 24-28, 2017*, pp. 23–30. IEEE, 2017. doi: 10.1109/IROS.2017.8202133. URL https://doi.org/10.1109/IROS.2017.8202133.

Ian Waudby-Smith and Aaditya Ramdas. Estimating means of bounded random variables by betting. *Journal of the Royal Statistical Society Series B: Statistical Methodology*, 86(1):1–27, 02 2024. ISSN 1369-7412. doi: 10.1093/jrsssb/qkad009. URL https://doi.org/10.1093/jrsssb/qkad009.

Ben Wooding, Viacheslav Horbanov, and Abolfazl Lavaei. Protect: Parallelized construction of safety barrier certificates for nonlinear polynomial systems. In Zhiming Liu, Adnane Saoud, and Heike Wehrheim (eds.), *Theoretical Aspects of Computing – ICTAC 2025*, pp. 448–458, Cham, 2026. Springer Nature Switzerland. ISBN 978-3-032-11176-0.

Bai Xue, Naijun Zhan, and Martin Fränzle. Reach-avoid analysis for polynomial stochastic differential equations. *IEEE Trans. Autom. Control.*, 69(3):1882–1889, 2024. doi: 10.1109/TAC.2023.3332570. URL https://doi.org/10.1109/TAC.2023.3332570.

Emily Yu, Dorde Zikelic, and Thomas A. Henzinger. Neural control and certificate repair via runtime monitoring. In Toby Walsh, Julie Shah, and Zico Kolter (eds.), *Thirty-Ninth AAAI Conference on Artificial Intelligence, Thirty-Seventh Conference on Innovative Applications of Artificial Intelligence, Fifteenth Symposium on Educational Advances in Artificial Intelligence, AAAI 2025, Philadelphia, PA, USA, February 25 - March 4, 2025*, pp. 26409–26417. AAAI Press, 2025. doi: 10.1609/AAAI.V39I25.34840. URL https://doi.org/10.1609/aaai.v39i25.34840.

Wenshuai Zhao, Jorge Peña Queralta, and Tomi Westerlund. Sim-to-real transfer in deep reinforcement learning for robotics: a survey. In *2020 IEEE Symposium Series on Computational Intelligence, SSCI 2020, Canberra, Australia, December 1-4, 2020*, pp. 737–744. IEEE, 2020. doi: 10.1109/SSCI47803.2020.9308468. URL https://doi.org/10.1109/SSCI47803.2020.9308468.

Dorde Zikelic, Mathias Lechner, Thomas A. Henzinger, and Krishnendu Chatterjee. Learning control policies for stochastic systems with reach-avoid guarantees. In Brian Williams, Yiling Chen, and Jennifer Neville (eds.), *Thirty-Seventh AAAI Conference on Artificial Intelligence, AAAI 2023, Thirty-Fifth Conference on Innovative Applications of Artificial Intelligence, IAAI 2023, Thirteenth Symposium on Educational Advances in Artificial Intelligence, EAAI 2023, Washington, DC, USA, February 7-14, 2023*, pp. 11926–11935. AAAI Press, 2023a. doi: 10.1609/AAAI.V37I10.26407. URL https://doi.org/10.1609/aaai.v37i10.26407.

Dorde Zikelic, Mathias Lechner, Abhinav Verma, Krishnendu Chatterjee, and Thomas A. Henzinger. Compositional policy learning in stochastic control systems with formal guarantees. In Alice Oh, Tristan Naumann, Amir Globerson, Kate Saenko, Moritz Hardt, and Sergey Levine (eds.), *Advances in Neural Information Processing Systems 36: Annual Conference on Neural Information Processing Systems 2023, NeurIPS 2023, New Orleans, LA, USA, December 10 - 16, 2023*, 2023b. URL http://papers.nips.cc/paper_files/paper/2023/hash/95827e011b9e899f189a01fe2f4ef316-Abstract-Conference.html.

A CONTINUOUS TIME

Termination. For a drift condition $(\mathcal{S}, l)$, we assume that the deployed process terminates at its first exit from $\mathcal{S}$. We retain the terminal state, so $\boldsymbol{X}$ denotes the absorbed process. More precisely,

$$\sigma \coloneqq \inf\{t \geq 0 : X_t \notin \mathcal{S}\}, \qquad \inf \varnothing \coloneqq \infty,$$

is an $(\mathcal{F}_t)$-stopping time, and, on $\{\sigma < \infty\}$, $X_\sigma \notin \mathcal{S}$ and $X_t = X_\sigma$ for every $t \geq \sigma$. We do not require σ to be finite. For example, for a reach–avoid certificate, $\mathcal{S}$ is the continuation region before reaching the target or an unsafe state. This assumption concerns the actual first exit, including exits between observations. This stopping-time requirement ensures that the process does not leave $\mathcal{S}$ without our knowledge. Meaning, we are only concerned with the certificate and loss in $\mathcal{S}$. Hence, we extend the allowed drift by zero outside $\mathcal{S}$:

$$l_{\mathcal{S}}(x) \coloneqq \begin{cases} l(x), & x \in \mathcal{S}, \\ 0, & x \notin \mathcal{S}. \end{cases} \tag{7}$$

Absorption ensures that an observed state in $\mathcal{S}$ rules out an earlier exit. The proof of Lemma 3 uses this to extend the regional drift inequality to all visited states.

Continuous-time assumptions. In continuous time, the expected drift condition bounds the instantaneous rate of change of the certificate. To obtain an expected bound on the certificate change between observations, we impose the following assumptions. As in the preliminaries, $\boldsymbol{X}$ is a right-continuous, time-homogeneous Markov process, and $(\mathcal{F}_t)$ is complete and right-continuous. Right-continuity and adaptation ensure the joint measurability used in the Tonelli argument in the proof of Lemma 3.

- *Generator:* There is a finite measurable function $\mathcal{A}f$ such that, for every fixed $t \geq 0$, almost surely,

  $$\mathcal{A}f(X_t) = \lim_{u \downarrow 0} \frac{\mathbb{E}[f(X_{t+u}) \mid X_t] - f(X_t)}{u}.$$

  This is the generator drift of the absorbed process, with $\mathcal{A}f = 0$ outside $\mathcal{S}$. The identity is used in the proof of Lemma 3 to express the instantaneous certificate condition as $\mathcal{A}f(X_r) \leq l(X_r)$ on $\{X_r \in \mathcal{S}\}$.

- *Integrability:* For every finite T, we assume

  $$\mathbb{E}\left[|f(X_0)| + \int_0^T \big(|\mathcal{A}f(X_r)| + |l_{\mathcal{S}}(X_r)|\big)\,\mathrm{d}r\right] < \infty. \tag{8}$$

  Together with integrability of N_t^f below, this establishes integrability of $f(X_t)$ and L_n in the opening argument of the proof of Lemma 3. It also ensures that the conditional expectation of the integrated drift difference in the final step is finite.

- *Martingale:* The process $(N_t^f)_{t \geq 0}$ defined by

  $$N_t^f \coloneqq f(X_t) - f(X_0) - \int_0^t \mathcal{A}f(X_r)\,\mathrm{d}r \tag{9}$$

  is a right-continuous $(\mathcal{F}_t)$-martingale. Thus, subtracting the accumulated actual drift leaves a process with zero conditional expected change. The proof of Lemma 3 uses this property to obtain $\mathbb{E}[N_{t_{n+1}}^f - N_{t_n}^f \mid \mathcal{F}_{t_n}] = 0$, leaving only the integrated drift difference in the expression for E_n.

For the martingale formulation of extended generators, see Hirsch & Yor (2012, Definition 4.1 and Proposition 4.1). We additionally require the displayed infinitesimal limit to exist. The certificate condition $\mathcal{A}f \leq l$ on $\mathcal{S}$ is imposed separately. The usual filtration conditions and right-continuity of N^f are retained as regularity assumptions; the fixed-time proof of Lemma 3 does not use them separately.

Observations. We observe the absorbed process at fixed times $t_n = nh$. The observation filtration satisfies $\mathcal{F}_n \subseteq \mathcal{F}_{t_n}$, and X_{t_n} is $\mathcal{F}_n$-measurable. The proof of Lemma 3 applies the martingale identity at these fixed times and uses the filtration inclusion to condition further on $\mathcal{F}_n$. Measurability of the sampled state makes the loss bound below predictable, as required when taking $1/B_n$ outside the conditional expectation in the proof of Lemma 1.

The accumulated allowed drift is

$$L_n \coloneqq \int_{t_n}^{t_{n+1}} l_{\mathcal{S}}(X_r)\,\mathrm{d}r = \int_{t_n\wedge\sigma}^{t_{n+1}\wedge\sigma} l_{\mathcal{S}}(X_r)\,\mathrm{d}r. \tag{10}$$

Thus, the allowed drift stops accumulating at termination. The interval containing termination is scored using the retained terminal state. On later intervals, we set $\overline{L}_n = 0$, giving $D_n = W_n = 0$. Termination itself does not raise a statistical alarm.

Bounds at termination. Suppose that $l(x) \le c$ on $\mathcal{S}$ and $f(x) \ge f_{\min}$ on all reachable states, including terminal states. Write $c_+ \coloneqq \max\{c, 0\}$. If the termination time is available when it occurs, define

$$\Delta_n \coloneqq (t_{n+1} \wedge \sigma) - (t_n \wedge \sigma), \qquad \overline{L}_n \coloneqq c\Delta_n.$$

Then $L_n \le c\Delta_n$ for either sign of c. If only sampled states are available, an upper bound is

$$\overline{L}_n \coloneqq \begin{cases} ch, & X_{t_{n+1}} \in \mathcal{S}, \\ c_+h, & X_{t_n} \in \mathcal{S},\ X_{t_{n+1}} \notin \mathcal{S}, \\ 0, & X_{t_n} \notin \mathcal{S}. \end{cases} \tag{11}$$

Indeed, survival until t_{n+1} ensures that the whole interval was spent in $\mathcal{S}$, while a terminating interval has duration $0 \le \Delta_n \le h$ and $L_n \le c\Delta_n \le c_+h$. If the process has already terminated, $L_n = 0$. In both cases, the predictable bound

$$B_n \coloneqq \max\{b_{\min},\ f(X_{t_n}) + c_+h - f_{\min}\} \tag{12}$$

satisfies $D_n \ge -B_n$, because $\overline{L}_n \le c_+h$. For a negative constant drift bound, using ch on the terminating interval would generally underestimate the accumulated allowance. The piecewise bound above avoids this without requiring the exact termination time.

Lemma 3 (Continuous-time expected increments)**.** *Under the continuous-time assumptions above, $f(X_t)$ is integrable for every finite t. If the $(\mathcal{S}, l)$-expected drift condition holds, then, for every $n \in \mathbb{N}_0$,*

$$E_n \coloneqq \mathbb{E}\left[f(X_{t_{n+1}}) - f(X_{t_n}) - L_n \,\middle|\, \mathcal{F}_n\right] \le 0 \qquad \textit{almost surely}. \tag{13}$$

Proof. Fix an observation index $n \in \mathbb{N}_0$.

Step 1: We check integrability. By the martingale decomposition in (9),

$$f(X_t) = f(X_0) + N_t^f + \int_0^t \mathcal{A}f(X_r)\,\mathrm{d}r.$$

Taking absolute values and expectations gives

$$\mathbb{E}[|f(X_t)|] \le \mathbb{E}[|f(X_0)|] + \mathbb{E}[|N_t^f|] + \mathbb{E}\left[\int_0^t |\mathcal{A}f(X_r)|\,\mathrm{d}r\right] < \infty.$$

The first and third terms are finite by (8). The second is finite because N^f is a martingale. The same integrability assumption gives

$$\mathbb{E}[|L_n|] \le \mathbb{E}\left[\int_{t_n}^{t_{n+1}} |l_{\mathcal{S}}(X_r)|\,\mathrm{d}r\right] < \infty.$$

Thus, the certificate values and allowance defining E_n are integrable.

Step 2: We bound the drift difference at each fixed time. Fix $r \ge 0$. If $X_r \in \mathcal{S}$, the certificate condition gives

$$\mathcal{A}f(X_r) \le l(X_r) = l_{\mathcal{S}}(X_r) \qquad \text{almost surely.}$$

If $X_r \notin \mathcal{S}$, the process is absorbed. Its generator drift is therefore zero, and $l_{\mathcal{S}}(X_r) = 0$ by definition. Combining the two cases yields

$$\mathcal{A}f(X_r) - l_{\mathcal{S}}(X_r) \leq 0 \qquad \text{almost surely.}$$

Step 3: We pass from fixed times to the whole interval. Write

$$J_n \coloneqq \int_{t_n}^{t_{n+1}} \big(\mathcal{A}f(X_r) - l_{\mathcal{S}}(X_r)\big)\,\mathrm{d}r.$$

This integral is integrable because

$$\mathbb{E}[|J_n|] \leq \mathbb{E}\left[\int_{t_n}^{t_{n+1}} \big(|\mathcal{A}f(X_r)| + |l_{\mathcal{S}}(X_r)|\big)\,\mathrm{d}r\right] < \infty.$$

We next show that $J_n \leq 0$ almost surely. Step 2 shows that, for each fixed time r, the drift difference is nonpositive with probability one. We now apply Tonelli's theorem to deduce that its integral over the observation interval is also nonpositive with probability one. For $(z)_+ \coloneqq \max\{z, 0\}$, joint measurability and nonnegativity give

$$\mathbb{E}\left[\int_{t_n}^{t_{n+1}} \big(\mathcal{A}f(X_r) - l_{\mathcal{S}}(X_r)\big)_+\,\mathrm{d}r\right] = \int_{t_n}^{t_{n+1}} \mathbb{E}\left[\big(\mathcal{A}f(X_r) - l_{\mathcal{S}}(X_r)\big)_+\right]\,\mathrm{d}r = 0.$$

The last equality follows from Step 2 at each fixed r. A nonnegative random variable with expectation zero is zero almost surely. Consequently,

$$J_n \leq \int_{t_n}^{t_{n+1}} \big(\mathcal{A}f(X_r) - l_{\mathcal{S}}(X_r)\big)_+\,\mathrm{d}r = 0 \qquad \text{almost surely.}$$

Step 4: We decompose the corrected certificate increment. Subtracting (9) at t_n from the same identity at t_{n+1} gives

$$f(X_{t_{n+1}}) - f(X_{t_n}) = N^f_{t_{n+1}} - N^f_{t_n} + \int_{t_n}^{t_{n+1}} \mathcal{A}f(X_r)\,\mathrm{d}r.$$

Subtracting L_n then yields

$$f(X_{t_{n+1}}) - f(X_{t_n}) - L_n = N^f_{t_{n+1}} - N^f_{t_n} + J_n.$$

Thus, the corrected increment is the sum of a martingale increment and the nonpositive accumulated drift difference.

Step 5: We take conditional expectations. The martingale property gives

$$\mathbb{E}[N^f_{t_{n+1}} - N^f_{t_n} \mid \mathcal{F}_{t_n}] = N^f_{t_n} - N^f_{t_n} = 0.$$

Since $\mathcal{F}_n \subseteq \mathcal{F}_{t_n}$, the tower property gives

$$\begin{aligned}\mathbb{E}[N^f_{t_{n+1}} - N^f_{t_n} \mid \mathcal{F}_n] &= \mathbb{E}\left[\mathbb{E}[N^f_{t_{n+1}} - N^f_{t_n} \mid \mathcal{F}_{t_n}] \,\middle|\, \mathcal{F}_n\right] \\ &= 0.\end{aligned}$$

Applying conditional expectation to the decomposition in Step 4, we therefore obtain

$$\begin{aligned}E_n &= \mathbb{E}\left[f(X_{t_{n+1}}) - f(X_{t_n}) - L_n \,\middle|\, \mathcal{F}_n\right] \\ &= \mathbb{E}[N^f_{t_{n+1}} - N^f_{t_n} \mid \mathcal{F}_n] + \mathbb{E}[J_n \mid \mathcal{F}_n] \\ &= \mathbb{E}[J_n \mid \mathcal{F}_n] \leq 0.\end{aligned}$$

The final inequality follows from $J_n \leq 0$ almost surely and monotonicity of conditional expectation. □

B VALIDITY

Score decrease. We show the score decrease condition. Intuitively, the score compares the observed certificate change with an upper bound on the allowed drift. Subtracting this upper bound can only reduce the corrected increment. Normalising by a positive bound chosen before the next observation then preserves the sign of its conditional mean.

Proof of Lemma 1. The integrability assumptions imply that D_n is integrable. Since $B_n \geq b_{\min} > 0$, $|W_n| \leq |D_n|/b_{\min}$, so W_n is also integrable. The loss bound gives $W_n \geq -1$. Adding and subtracting L_n and using the $\mathcal{F}_n$-measurability of B_n gives

$$\mu_n = \frac{\mathbb{E}[D_n \mid \mathcal{F}_n]}{B_n} = \frac{E_n - \mathbb{E}[\overline{L}_n - L_n \mid \mathcal{F}_n]}{B_n} \leq \frac{E_n}{B_n}.$$

The inequality follows from $\overline{L}_n \geq L_n$. In continuous time, Lemma 3 gives $E_n \leq 0$ under the certificate condition. In discrete time, $h = 1$ gives $t_{n+1} = t_n + 1$. On $\{X_{t_n} \in \mathcal{S}\}$, we have $L_n = l(X_{t_n})$. The Markov property and the one-step expected-drift condition therefore give

$$\begin{aligned} E_n &= \mathbb{E}[f(X_{t_{n+1}}) - f(X_{t_n}) \mid \mathcal{F}_n] - l(X_{t_n}) \\ &= \mathbb{E}[f(X_{t_n+1}) - f(X_{t_n}) \mid X_{t_n}] - l(X_{t_n}) \leq 0 \end{aligned}$$

almost surely on $\{X_{t_n} \in \mathcal{S}\}$. Consequently, the certificate condition implies $\mu_n \leq 0$ for all continuous-time observations and, in discrete time, on $\{X_{t_n} \in \mathcal{S}\}$. □

Statistical guarantee. For the proof, we freeze the wealth just before the first score with a positive conditional mean. Before freezing, each bet has nonpositive expected gain; afterwards, the wealth stays constant. The frozen wealth is therefore a supermartingale, so Ville's inequality applies.

Proof of Theorem 2. We first construct an auxiliary monitor whose scores always have nonpositive conditional means. We then apply Ville's inequality to its wealth and show that every alarm covered by the historical guarantee also crosses the auxiliary wealth threshold.

Step 1: We replace scores from the first violation onwards. We define the first index at which the conditional expected corrected increment is positive, i.e., $\kappa := \inf\{n \in \mathbb{N}_0 : E_n > 0\}$, with $\inf \varnothing := \infty$ and freeze the process at that time $\widetilde{W}_n := \mathbf{1}\{n < \kappa\} W_n$. The auxiliary scores agree with the actual scores before κ and are zero from κ onwards. For every $k \leq n$, the random variable E_k is $\mathcal{F}_k$-measurable and hence $\mathcal{F}_n$-measurable. Consequently,

$$\{n < \kappa\} = \bigcap_{k=0}^{n} \{E_k \leq 0\} \in \mathcal{F}_n.$$

The indicator multiplying W_n is therefore measurable before that score is observed. We may take it outside the conditional expectation:

$$\mathbb{E}[\widetilde{W}_n \mid \mathcal{F}_n] = \mathbf{1}\{n < \kappa\} \mathbb{E}[W_n \mid \mathcal{F}_n] = \mathbf{1}\{n < \kappa\} \mu_n.$$

On $\{n < \kappa\}$, score validity gives $\mu_n \leq E_n / B_n \leq 0$. On its complement, the displayed expression is zero. Hence $\mathbb{E}[\widetilde{W}_n \mid \mathcal{F}_n] \leq 0$ almost surely. Moreover, $|\widetilde{W}_n| \leq |W_n|$ and $\widetilde{W}_n \geq -1$, so the auxiliary scores remain integrable and satisfy the same lower bound.

Step 2: Each auxiliary account is a nonnegative supermartingale. For each expert a launched at s, define

$$\widetilde{G}^a_{s,s} = 1, \qquad \widetilde{G}^a_{s,n+1} = \widetilde{G}^a_{s,n}(1 + \lambda^a_{s,n} \widetilde{W}_n).$$

We use exactly the stakes chosen by the actual monitor. These stakes remain $\mathcal{F}_n$-measurable, because they are still computed from the actual observation history.

Since $0 \leq \lambda^a_{s,n} \leq 1$ and $\widetilde{W}_n \geq -1$, it follows that $1 + \lambda^a_{s,n} \widetilde{W}_n \geq 1 - \lambda^a_{s,n} \geq 0$. Every account therefore remains nonnegative. Both its current wealth and its next stake are $\mathcal{F}_n$-measurable, so Step 1 gives

$$\mathbb{E}[\widetilde{G}^a_{s,n+1} \mid \mathcal{F}_n] = \widetilde{G}^a_{s,n} \mathbb{E}[1 + \lambda^a_{s,n} \widetilde{W}_n \mid \mathcal{F}_n] = \widetilde{G}^a_{s,n} \big(1 + \lambda^a_{s,n} \mathbb{E}[\widetilde{W}_n \mid \mathcal{F}_n]\big) \leq \widetilde{G}^a_{s,n}.$$

This calculation also establishes integrability. Because the next account value is nonnegative, its conditional expectation is defined even before integrability is established, although it could initially be infinite. Starting from $\widetilde{G}^a_{s,s} = 1$, the displayed inequality and the tower property give inductively

$$\mathbb{E}[\widetilde{G}^a_{s,n+1}] \le \mathbb{E}[\widetilde{G}^a_{s,n}] \le 1.$$

Thus every account value has finite expectation, and $(\widetilde{G}^a_{s,n})_{n\ge s}$ is a nonnegative supermartingale.

Step 3: The auxiliary mixtures are also supermartingales. Use the same mixture weights as the actual monitor and set

$$\widetilde{Q}_{s,n} \coloneqq \sum_{a\in\mathcal{A}_s} \eta_{s,a}\widetilde{G}^a_{s,n}.$$

The weights are nonnegative and sum to one, so $\widetilde{Q}_{s,n} \ge 0$ and $\widetilde{Q}_{s,s} = 1$. They are chosen at launch and are therefore $\mathcal{F}_s$-measurable, hence $\mathcal{F}_n$-measurable for $n \ge s$. Using Step 2,

$$\begin{aligned}\mathbb{E}[\widetilde{Q}_{s,n+1} \mid \mathcal{F}_n] &= \sum_{a\in\mathcal{A}_s} \eta_{s,a}\mathbb{E}[\widetilde{G}^a_{s,n+1} \mid \mathcal{F}_n] \\ &\le \sum_{a\in\mathcal{A}_s} \eta_{s,a}\widetilde{G}^a_{s,n} = \widetilde{Q}_{s,n}.\end{aligned}$$

As above, nonnegativity and the initial value $\widetilde{Q}_{s,s} = 1$ give integrability by induction. Thus each mixture is a nonnegative supermartingale. No independence between its accounts is needed.

Step 4: Funding preserves the supermartingale property. Use the actual funding amounts c_s and define

$$C_n \coloneqq \sum_{s=0}^{n-1} c_s, \qquad \widetilde{M}_n \coloneqq 1 - C_n + \sum_{s=0}^{n-1} c_s\widetilde{Q}_{s,n}.$$

The funding constraint $0 \le c_n \le 1 - C_n$ ensures $0 \le C_n \le 1$. Hence the reserve and every funded contribution are nonnegative, giving $\widetilde{M}_n \ge 0$. The sum contains finitely many terms at each n. Since $c_s \le 1$ and each mixture is integrable, $\widetilde{M}_n$ is integrable as well. Initially, no accounts have been funded, so $\widetilde{M}_0 = 1$. At index n, funding removes c_n from the reserve and assigns it to a new mixture with unit initial wealth. Thus

$$\widetilde{M}_{n+1} = 1 - C_n - c_n + \sum_{s=0}^{n-1} c_s\widetilde{Q}_{s,n+1} + c_n\widetilde{Q}_{n,n+1}.$$

All funding amounts appearing here are $\mathcal{F}_n$-measurable. Taking conditional expectations and applying Step 3 gives

$$\begin{aligned}\mathbb{E}[\widetilde{M}_{n+1} \mid \mathcal{F}_n] &= 1 - C_n - c_n + \sum_{s=0}^{n-1} c_s\mathbb{E}[\widetilde{Q}_{s,n+1} \mid \mathcal{F}_n] + c_n\mathbb{E}[\widetilde{Q}_{n,n+1} \mid \mathcal{F}_n] \\ &\le 1 - C_n - c_n + \sum_{s=0}^{n-1} c_s\widetilde{Q}_{s,n} + c_n\widetilde{Q}_{n,n} \\ &= 1 - C_n + \sum_{s=0}^{n-1} c_s\widetilde{Q}_{s,n} = \widetilde{M}_n.\end{aligned}$$

The last equality uses $\widetilde{Q}_{n,n} = 1$: the capital assigned to the new mixture exactly matches the amount removed from the reserve. Therefore $(\widetilde{M}_n)$ is a nonnegative supermartingale.

Step 5: We bound threshold crossings of the auxiliary wealth. Since $\widetilde{M}_0 = 1$, Ville's inequality gives

$$\mathbb{P}\left(\exists n \in \mathbb{N}_0 : \widetilde{M}_n \ge 1/\delta\right) \le \delta.$$

Step 6: We transfer this bound to the actual monitor. Consider an outcome on which $\tau_\delta < \infty$ and $E_k \le 0$ for every $k < \tau_\delta$. By definition of κ, this implies $\kappa \ge \tau_\delta$. Consequently,

$$\widetilde{W}_k = W_k \qquad \text{for every } k < \tau_\delta.$$

These are precisely the scores used to compute the wealth at τ_δ. In particular, equality $\kappa = \tau_\delta$ causes no problem: the score with index τ_δ has not yet contributed to that wealth.

The actual and auxiliary accounts have the same initial values and use the same stakes. Since their scores agree through index $\tau_\delta - 1$, their account values agree at τ_δ. Their mixture weights, funding amounts, and reserves also agree. It follows that

$$\widetilde{M}_{\tau_\delta} = M_{\tau_\delta} \geq 1/\delta.$$

We have therefore established the event inclusion

$$\{\tau_\delta < \infty,\ E_k \leq 0 \text{ for every } k < \tau_\delta\} \subseteq \left\{\exists n \in \mathbb{N}_0 : \widetilde{M}_n \geq 1/\delta\right\}.$$

Applying the bound from Step 5 yields

$$\mathbb{P}\left(\tau_\delta < \infty \text{ and } E_k \leq 0 \text{ for every } k < \tau_\delta\right) \leq \delta.$$

Step 7: We obtain the unconditional false-alarm bound. Suppose that the certificate condition holds. Score validity gives $\mu_n \leq 0$ almost surely on $\{X_{t_n} \in \mathcal{S}\}$. Outside $\mathcal{S}$, every expert chooses stake zero. Consequently, for every expert and every $n \geq s$,

$$\lambda^a_{s,n}\mu_n \leq 0 \qquad \text{almost surely.}$$

We can therefore apply the account argument directly to the actual scores:

$$\mathbb{E}[G^a_{s,n+1} \mid \mathcal{F}_n] = G^a_{s,n}\big(1 + \lambda^a_{s,n}\mathbb{E}[W_n \mid \mathcal{F}_n]\big) = G^a_{s,n}(1 + \lambda^a_{s,n}\mu_n) \leq G^a_{s,n}.$$

The nonnegativity and integrability arguments from Step 2 apply unchanged. The mixture and funding arguments from Steps 3 and 4 then show that the actual wealth (M_n) is a nonnegative supermartingale with $M_0 = 1$. Ville's inequality therefore gives

$$\mathbb{P}(\tau_\delta < \infty) = \mathbb{P}\left(\exists n \in \mathbb{N}_0 : M_n \geq 1/\delta\right) \leq \delta.$$

□

Monitoring several conditions. For a certificate with several drift conditions, each component monitor must satisfy the score assumptions for its own condition. The termination assumption applies separately to the region used in each continuous-time condition. In particular, one physical termination time suffices when the conditions share a continuation region. If their regions differ, stopping at the target or an unsafe state does not by itself justify every component: each component must have access to the process stopped at its own first exit, including its exit state, or use another valid interval construction.

Corollary 4 (Composition)**.** *Suppose that a certificate has J expected-drift conditions. For each condition j, run a monitor satisfying Theorem 2 at level $\delta_j > 0$, where $\sum_{j=1}^{J} \delta_j \leq \delta$. Let τ be the first index at which any component monitor or sampled domain-bound check raises an alarm. If all certificate conditions hold for the deployed process, then $\mathbb{P}(\tau < \infty) \leq \delta$.*

Proof. Write τ_j for the first alarm index of component j. A valid domain-bound condition holds at every fixed sampled time almost surely. Since there are countably many samples and finitely many such conditions, their checks raise no alarm almost surely. The component guarantees and the union bound therefore give

$$\mathbb{P}(\tau < \infty) \leq \sum_{j=1}^{J} \mathbb{P}(\tau_j < \infty) \leq \sum_{j=1}^{J} \delta_j \leq \delta.$$

Independence is not required. □

C POWER

Weighted stakes. The two stake strategies use the same weighted history to decide how strongly to bet. The log-return strategy directly maximises the historical log return. The moment strategy instead maximises a quadratic lower bound, which requires only the weighted sum of the scores and their squares. The following lemma justifies both constructions and explains how these quantities relate to the expected return of the next bet. For the following lemma we fix an expert $a \in \mathcal{A}_s$, an observation index $n \geq s$, and a cap $1/2 \leq \overline{\lambda} < 1$. Moreover, we abbreviate and and define for $0 \leq \lambda \leq \overline{\lambda}$.

$$\omega_i \coloneqq \omega^a_{s,n,i}, \quad S \coloneqq \sum_{i \in I_n} \omega_i W_i, \quad U \coloneqq \sum_{i \in I_n} \omega_i W_i^2, \quad H(\lambda) \coloneqq \sum_{i \in I_n} \omega_i \log(1 + \lambda W_i).$$

Lemma 5 (Weighted stake strategies)**.** *The following statements hold on* $\{X_{t_n} \in \mathcal{S}\}$.

1. *The log-return and moment stakes are* $\mathcal{F}_n$*-measurable, both are zero exactly when* $S \leq 0$*, and* $0 \leq \lambda^{a,\mathrm{mom}}_{s,n} \leq \lambda^{a,\log}_{s,n} \leq \overline{\lambda}$.
2. *For every* $\lambda \in [0, 1/2]$*, the moment stake maximises the lower bound* $H(\lambda) \geq \lambda S - \lambda^2 U$ *over* $[0, 1/2]$.
3. *For any* $\mathcal{F}_n$*-measurable stake* $\lambda_n \in [0, 1/2]$*, if* $q_n \coloneqq \mathbb{E}[W_n^2 \mid \mathcal{F}_n] < \infty$ *a.s. then*

$$\mathbb{E}[\log(1 + \lambda_n W_n) \mid \mathcal{F}_n] \geq \lambda_n \mu_n - \lambda_n^2 q_n \qquad \textit{almost surely}. \tag{14}$$

Proof. We first establish the quadratic lower bound. For $z \geq -1/2$, define $\psi(z) \coloneqq \log(1+z) - z + z^2$. We take its derivative $\psi'(z) = \frac{z(1+2z)}{1+z}$ and observe that $\psi'(z) \leq 0$ on $[-1/2, 0]$, because the factor z is nonpositive, while $1 + 2z$ is nonnegative and $1 + z$ is positive. Moreover, on $[0, \infty)$, all factors are nonnegative, so $\psi'(z) \geq 0$. Thus, ψ attains its minimum at zero. Since $\psi(0) = 0$, substituting $z = \lambda w$ implies that

$$\log(1 + \lambda w) \geq \lambda w - \lambda^2 w^2 \qquad \text{for } w \geq -1, \quad 0 \leq \lambda \leq \tfrac{1}{2}. \tag{15}$$

Multiplying by the nonnegative weights and summing yields $H(\lambda) \geq \lambda S - \lambda^2 U$.

The linear term rewards a positive weighted score sum, while the quadratic term penalises large stakes. If $U > 0$, the derivative of this lower bound is $S - 2\lambda U$, so its maximiser over $[0, 1/2]$ is

$$\lambda^{a,\mathrm{mom}}_{s,n} = \min\left\{\frac{1}{2}, \frac{S_+}{2U}\right\}.$$

If $U = 0$, every score with positive weight is zero. Hence, $S = 0$, the lower bound is constant, and the strategy chooses zero.

We next consider the log-return strategy. Since $W_i \geq -1$ and $\overline{\lambda} < 1$, the logarithms are well-defined throughout the optimisation interval. The objective is continuous on a compact interval, so a maximiser exists. Its derivatives are

$$H'(\lambda) = \sum_{i \in I_n} \frac{\omega_i W_i}{1 + \lambda W_i}, \qquad H''(\lambda) = -\sum_{i \in I_n} \frac{\omega_i W_i^2}{(1 + \lambda W_i)^2}.$$

Consequently, H is concave, and strictly concave whenever $-\sum_{i \in I_n} \omega_i W_i^2 > 0$. The derivative $H'(0) = \sum_{i \in I_n} \omega_i W_i$ describes whether increasing the stake from zero initially improves the historical return. If $S \leq 0$, concavity implies that zero is the smallest maximiser. If $S > 0$, the maximiser is positive. It equals $\overline{\lambda}$ when $H'(\overline{\lambda}) \geq 0$; otherwise, it is the unique root of H' in $(0, \overline{\lambda})$. This gives the endpoint checks and bisection procedure described above. All inputs are $\mathcal{F}_n$-measurable, as are the bisection iterates and their limit. The explicit moment stake is likewise measurable.

To compare the stakes, suppose $S > 0$. For $\lambda \in [0, 1/2]$, we have $1 + \lambda W_i \geq 1/2$, and therefore

$$H'(\lambda) = S - \lambda \sum_{i \in I_n} \frac{\omega_i W_i^2}{1 + \lambda W_i} \geq S - 2\lambda U.$$

At the moment stake, the final expression is nonnegative. Thus, the historical log return has not started decreasing at that stake, so its maximiser cannot lie below it. If $S \leq 0$, both stakes are zero.

Finally, the pointwise logarithmic bound also applies to the next score. Taking conditional expectations and using predictability gives

$$\mathbb{E}[\log(1 + \lambda_n W_n) \mid \mathcal{F}_n] \geq \lambda_n \mu_n - \lambda_n^2 q_n.$$

The logarithm is integrable because

$$\left|\log(1 + \lambda_n W_n)\right| \leq \log 2 + \tfrac{1}{2}(W_n)_+,$$

and W_n is integrable under the standing assumptions. □

The conditional bound explains why the moment strategy uses both the scores and their squares. A positive conditional mean favours betting, while a large second moment makes a large stake less attractive. In particular, the expected log return is positive whenever

$$\lambda_n > 0 \qquad \text{and} \qquad \mu_n > \lambda_n q_n.$$

Thus, sufficiently small positive stakes ensure positive expected log return when the conditional mean is positive and the conditional second moment is finite. The log-return strategy achieves at least as much weighted historical log return as the moment strategy, but neither is necessarily better for the next score, e.g., overcommitting to a wrong outcome may decrease wealth drastically.

Locality. We need to decide on how much weight we want to give to the similar scores. Therefore, we first translate the regularity guesses and the implicitly defined tolerated difference $\mu^a_{\star,s}$ into spatial and temporal radii, with $\theta \in [0, 1]$

$$r^a_{\mathcal{X},s} := \frac{\theta \mu^a_{\star,s}}{\alpha^a_s}, \qquad r^a_{\mathbb{T},s} := \frac{\theta \mu^a_{\star,s}}{\beta^a_s}.$$

The spatial and temporal radii can be chosen from Lipschitz bounds of conditional expected scores and the positive expected score the expert aims to detect. We quantify regularity using Lipschitz continuity, stating that the expected score changes at most linearly in space and time. For the spatial regularity argument, suppose that the expected score at time n depends only on the current state through a common measurable function $g_h \colon \mathcal{X} \to \mathbb{R}$: $\mu_n = \mathbb{E}[W_n \mid \mathcal{F}_n] = g_h(X_{t_n})$ almost surely, for every n. Here, $g_h(x)$ is the expected score over an observation interval of length h starting at state x. Fix $\theta \in (0, 1)$, let $a \in \mathcal{A}_s$ be an expert spawned at time s, which aims to detect an expected increase of $\mu^a_{\star,s} > 0$. The expert guesses a spatial *spatial Lipschitz* constant $\alpha^a_{\mathcal{X},s} > 0$ that captures how much the conditional score mean can change between nearby states, and a *temporal Lipschitz* constant $\alpha^a_{\mathbb{T},s} > 0$ for how much it can change between nearby observation times, i.e.,

$$\begin{aligned} |g_h(y) - g_h(x)| &\leq \alpha^a_{\mathcal{X},s} \|x - y\|, \\ &\qquad \text{for all } x, y \in \mathcal{X}, \\ |\mu_n - \mu_i| &\leq \alpha^a_{\mathbb{T},s} |t_n - t_i|, \\ &\qquad \text{almost surely, for all } 0 \leq i < n. \end{aligned} \tag{16}$$

We use $\theta \mu^a_{\star,s}$ as the tolerated difference between past and current conditional means. For a cutoff or linear kernel with finite radius, if the corresponding guess is correct and $\mu_n \geq \mu^a_{\star,s}$, every observation with positive weight has conditional mean $\mu_i \geq (1 - \theta)\mu^a_{\star,s} > 0$. Specifically, the expert favours observations for which the guessed bound on $|\mu_i - \mu_n|$ is within this tolerance. Giving them greater weight can help the expert choose positive stakes during violations, without affecting false-alarm control.

Lipschitz Regularity. The regularity guesses guide which historical observations are useful for predicting the next score. Fix an expert $a \in \mathcal{A}_s$. Suppose that its spatial regularity guess is valid and its spatial kernel is a cutoff or linear kernel with scale

$$r^a_{x,s} = \frac{\theta \mu^a_{\star,s}}{\alpha^a_{\mathcal{X},s}}.$$

A positive prediction weight requires the spatial kernel to be positive. For a cutoff or linear kernel, this is possible only when the distance from the historical state to the current state is at most the spatial scale. Hence,

$$\omega^a_{s,n,i} > 0 \quad \Longrightarrow \quad \|X_{t_i} - X_{t_n}\| \le r^a_{x,s}.$$

If the spatial regularity guess is valid, the conditional means at these states therefore satisfy

$$|\mu_i - \mu_n| = |g_h(X_{t_i}) - g_h(X_{t_n})| \le \alpha^a_{\mathcal{X},s}\|X_{t_i} - X_{t_n}\| \le \alpha^a_{\mathcal{X},s} r^a_{x,s} = \alpha^a_{\mathcal{X},s}\frac{\theta\mu^a_{\star,s}}{\alpha^a_{\mathcal{X},s}} = \theta\mu^a_{\star,s}.$$

Thus, the chosen radius ensures that every observation with positive weight has a conditional mean within $\theta\mu^a_{\star,s}$ of the current conditional mean. The same conclusion follows from a valid temporal guess and the corresponding temporal kernel. In particular, if $\mu_n \ge \mu^a_{\star,s}$, then

$$\omega^a_{s,n,i} > 0 \quad \Longrightarrow \quad \mu_i \ge (1-\theta)\mu^a_{\star,s} > 0.$$

Hence, when the current conditional mean is sufficiently positive, the selected observations also come from positive-mean steps. Their realised scores may nevertheless be negative, and their second moments may differ. Geometric kernels retain observations beyond the chosen scale, so they favour relevant observations without ensuring this implication. These regularity guesses guide prediction and need not hold for false-alarm control.

Accumulating evidence. A positive expected log return favours wealth growth, but an alarm requires the realised wealth to reach the prescribed threshold. The following lemma connects the evidence accumulated by one expert to the monitor's verdict.

Lemma 6 (Sufficient evidence for detection)**.** *Fix a starting index s, an expert $a \in \mathcal{A}_s$ with $c_s\eta_{s,a} > 0$, and an index $n > s$. Suppose that this expert uses stakes strictly below one. If*

$$\sum_{k=s}^{n-1} \log(1 + \lambda^a_{s,k} W_k) \ge \log \frac{1}{\delta c_s \eta_{s,a}}, \tag{17}$$

then $\tau_\delta \le n$. If its stakes are at most $1/2$, it suffices that

$$\sum_{k=s}^{n-1} \left(\lambda^a_{s,k} W_k - (\lambda^a_{s,k})^2 W_k^2\right) \ge \log \frac{1}{\delta c_s \eta_{s,a}}. \tag{18}$$

Proof. The mixture launched at index s receives c_s units of capital, of which the fraction $\eta_{s,a}$ is assigned to expert a. The account $G^a_{s,n}$ starts from one and records how this initial capital grows through betting. Hence, the expert's contribution to the total wealth at index n is $c_s\eta_{s,a}G^a_{s,n}$.

By definition, the total wealth is

$$M_n = 1 - C_n + \sum_{r=0}^{n-1} c_r \sum_{b\in\mathcal{A}_r} \eta_{r,b} G^b_{r,n}.$$

Since $n > s$, this sum includes the contribution of expert a launched at s. The reserve $1 - C_n$ and all other contributions are nonnegative. Keeping only the chosen expert's contribution therefore gives the lower bound

$$M_n \ge c_s\eta_{s,a}G^a_{s,n}.$$

Thus, it suffices to show that this expert's contribution alone reaches the alarm threshold $1/\delta$. The expert's account is the product of its successive betting factors:

$$G^a_{s,n} = \prod_{k=s}^{n-1}(1 + \lambda^a_{s,k}W_k).$$

Each factor is strictly positive because $W_k \ge -1$ and $\lambda^a_{s,k} < 1$. We can therefore express the product as the exponential of the sum of its logarithms. Under the first condition of the lemma,

$$G^a_{s,n} = \exp\left(\sum_{k=s}^{n-1}\log(1+\lambda^a_{s,k}W_k)\right) \ge \exp\left(\log\frac{1}{\delta c_s\eta_{s,a}}\right) = \frac{1}{\delta c_s\eta_{s,a}}$$

Multiplying by the expert's initial capital gives

$$M_n \geq c_s \eta_{s,a} G^a_{s,n} \geq c_s \eta_{s,a} \frac{1}{\delta c_s \eta_{s,a}} = \frac{1}{\delta}.$$

Since τ_δ is the first index at which the total wealth reaches $1/\delta$, it follows that $\tau_\delta \leq n$.

Finally, suppose that all stakes of this expert are at most $1/2$. Applying (15) to each betting factor and summing yields

$$\sum_{k=s}^{n-1} \log(1 + \lambda^a_{s,k} W_k) \geq \sum_{k=s}^{n-1} \left(\lambda^a_{s,k} W_k - (\lambda^a_{s,k})^2 W_k^2\right).$$

If the second condition holds, the right-hand side is at least $\log(1/(\delta c_s \eta_{s,a}))$. The sum of log returns is therefore also at least this threshold, which establishes the first condition. The preceding argument then gives $\tau_\delta \leq n$. □

The lemma explains the cost of dividing capital among experts. An expert with less initial capital must accumulate more log return to trigger an alarm by itself. For example, dividing a mixture equally among K experts adds $\log K$ to this sufficient threshold compared with assigning that mixture to one expert. This cost allows the monitor to support several strategies without selecting one in advance. Since other accounts also contribute wealth, the monitor may alarm before any individual expert satisfies the condition.

Monitoring several conditions. A certificate may contain several expected-decrease conditions. We can monitor them jointly by dividing the permitted false-alarm probability among their monitors. This union-bound approach allows the monitors to use the same observations, because independence is not required.

Corollary 7 (Composition)**.** *Suppose that a certificate has J expected-decrease conditions. For each condition j, run a monitor satisfying Theorem 2 at a level $\delta_j > 0$, where $\sum_{j=1}^J \delta_j \leq \delta$. Let τ be the first index at which any component monitor or sampled domain-bound check raises an alarm. If all certificate conditions hold for the deployed process, then $\mathbb{P}(\tau < \infty) \leq \delta$.*

Proof. A valid domain bound cannot be violated at an observed state. Writing τ_j for the first alarm index of component monitor j, Theorem 2 and the union bound give

$$\mathbb{P}(\tau < \infty) \leq \sum_{j=1}^J \mathbb{P}(\tau_j < \infty) \leq \sum_{j=1}^J \delta_j \leq \delta.$$

□

Monitor power. To quantify detection, we consider a block on which the conditional expected corrected drift E_k exceeds the expected slack in the observable drift bound. The difference determines the signal available to the monitor. The following result uses an additional upper score bound and an expert whose stake is constant throughout the block. For the next lemma, fix a starting index s, an expert $a \in \mathcal{A}_r$ launched at $r \leq s$, a duration $N \in \mathbb{N}_{\geq 1}$, and $\beta \in (0, 1)$. Suppose that the expert's capital at the start of the block is $\rho_{r,s} := c_r \eta_{r,a} G^a_{r,s} > 0$ almost surely. Let $d_\star$ be a lower bound on the conditional expected corrected drift E_k, and let $s_\star \geq 0$ be an upper bound on the conditional expected slack $\mathbb{E}[\overline{L}_k - L_k \mid \mathcal{F}_k]$ throughout the block. We require $d_\star > s_\star$, so a positive excess remains after accounting for the slack. Let $B_{\max} > 0$ be an upper bound on B_k, and assume $\varepsilon := (d_\star - s_\star)/B_{\max} \in (0, 1]$. Formally, for every $k = s, \ldots, s + N - 1$, we require almost surely

$$E_k \geq d_\star, \qquad \mathbb{E}[\overline{L}_k - L_k \mid \mathcal{F}_k] \leq s_\star,$$
$$B_k \leq B_{\max}, \qquad W_k \leq 1, \qquad \lambda^a_{r,k} = \varepsilon/2.$$

The standing score assumptions already give $W_k \geq -1$, so the scores lie in $[-1, 1]$.

Lemma 8 (Detection of a persistent certificate violation)**.** *On the $\mathcal{F}_s$-measurable event*

$$N \geq \frac{8 B^2_{\max}}{(d_\star - s_\star)^2} \log \frac{1}{\delta \rho_{r,s} \beta}, \tag{19}$$

the monitor satisfies $\mathbb{P}(\tau_\delta \leq s + N \mid \mathcal{F}_s) \geq 1 - \beta$ almost surely.

Proof of Lemma 8. First, we translate the certificate excess into a lower bound on the conditional score mean. Since B_k is positive and $\mathcal{F}_k$-measurable,

$$\mu_k = \frac{E_k - \mathbb{E}[\overline{L}_k - L_k \mid \mathcal{F}_k]}{B_k} \geq \frac{d_\star - s_\star}{B_k} \geq \frac{d_\star - s_\star}{B_{\max}} = \varepsilon.$$

Thus, each score in the block has conditional mean at least ε.

Write $Z_k = W_k - \mu_k$. Each Z_k has conditional mean zero and conditional range of length at most 2, because $W_k \in [-1, 1]$. The conditional Hoeffding bound gives, for every $q \geq 0$,

$$\mathbb{E}[e^{-qZ_k} \mid \mathcal{F}_k] \leq e^{q^2/2}.$$

For $j = 0, \ldots, N$, write $S_j := \sum_{k=s}^{s+j-1} Z_k$, with $S_0 = 0$. Since S_j is $\mathcal{F}_{s+j}$-measurable, the tower property and the conditional Hoeffding bound give

$$\begin{aligned}\mathbb{E}[e^{-qS_{j+1}} \mid \mathcal{F}_s] &= \mathbb{E}\big[e^{-qS_j}\mathbb{E}[e^{-qZ_{s+j}} \mid \mathcal{F}_{s+j}] \,\big|\, \mathcal{F}_s\big] \\ &\leq e^{q^2/2}\mathbb{E}[e^{-qS_j} \mid \mathcal{F}_s].\end{aligned}$$

Applying this inequality successively for $j = 0, \ldots, N-1$, starting from $e^{-qS_0} = 1$, yields

$$\mathbb{E}[e^{-qS_N} \mid \mathcal{F}_s] \leq e^{Nq^2/2}.$$

For $q > 0$, the event $S_N \leq -u$ is equivalent to $e^{-qS_N} \geq e^{qu}$. Conditional Markov's inequality therefore gives

$$\begin{aligned}\mathbb{P}(S_N \leq -u \mid \mathcal{F}_s) &= \mathbb{P}(e^{-qS_N} \geq e^{qu} \mid \mathcal{F}_s) \\ &\leq e^{-qu}\mathbb{E}[e^{-qS_N} \mid \mathcal{F}_s] \\ &\leq \exp(-qu + Nq^2/2).\end{aligned}$$

For $u > 0$, the exponent is minimised at $q = u/N$. Substituting this choice gives

$$\mathbb{P}(S_N \leq -u \mid \mathcal{F}_s) \leq \exp\left(-\frac{u^2}{2N}\right).$$

For $u = 0$, the same bound holds trivially.

Now choose $u = \sqrt{2N\log(1/\beta)}$, so the right-hand side equals β. Thus, with conditional probability at least $1 - \beta$, $S_N \geq -\sqrt{2N\log(1/\beta)}$. Since $W_k = \mu_k + Z_k$ and $\mu_k \geq \varepsilon$ throughout the block, on this event we have

$$\sum_{k=s}^{s+N-1} W_k = \sum_{k=s}^{s+N-1} \mu_k + S_N \geq N\varepsilon - \sqrt{2N\log(1/\beta)}.$$

We next translate the score bound into growth of the expert's wealth. Since $\rho_{r,s} > 0$, the account wealth $G^a_{r,s}$ is positive. During the block, the expert uses the constant stake $\lambda = \varepsilon/2 \leq 1/2$. Each betting factor satisfies $1 + \lambda W_k \geq 1/2$, so

$$\frac{G^a_{r,s+N}}{G^a_{r,s}} = \prod_{k=s}^{s+N-1} (1 + \lambda W_k) > 0.$$

Taking logarithms converts this product into a sum. For each score, the inequality $\log(1 + \lambda w) \geq \lambda w - \lambda^2 w^2$ and the bound $W_k^2 \leq 1$ give

$$\log(1 + \lambda W_k) \geq \lambda W_k - \lambda^2 W_k^2 \geq \lambda W_k - \lambda^2.$$

Summing over the N scores therefore yields

$$\log \frac{G^a_{r,s+N}}{G^a_{r,s}} \geq \lambda \sum_{k=s}^{s+N-1} W_k - N\lambda^2.$$

On the score event established above, substituting $\lambda = \varepsilon/2$ gives

$$\log \frac{G^a_{r,s+N}}{G^a_{r,s}} \geq \frac{\varepsilon}{2}\left(N\varepsilon - \sqrt{2N\log(1/\beta)}\right) - \frac{N\varepsilon^2}{4} = \frac{N\varepsilon^2}{4} - \varepsilon\sqrt{\frac{N}{2}\log(1/\beta)}.$$

To simplify the square-root term, apply $2\sqrt{uv} \leq u + v$ with $u = N\varepsilon^2/8$ and $v = \log(1/\beta)$:

$$\varepsilon\sqrt{\frac{N}{2}\log(1/\beta)} = 2\sqrt{\frac{N\varepsilon^2}{8}\log(1/\beta)} \leq \frac{N\varepsilon^2}{8} + \log(1/\beta).$$

Consequently,

$$\log \frac{G^a_{r,s+N}}{G^a_{r,s}} \geq \frac{N\varepsilon^2}{8} - \log(1/\beta).$$

It remains to show that this growth suffices to reach the alarm threshold. Since $\varepsilon = (d_\star - s_\star)/B_{\max}$, the duration condition (19) implies

$$\frac{N\varepsilon^2}{8} \geq \log \frac{1}{\delta\rho_{r,s}\beta} = \log \frac{1}{\delta\rho_{r,s}} + \log(1/\beta).$$

Combining the two bounds and exponentiating gives

$$\log \frac{G^a_{r,s+N}}{G^a_{r,s}} \geq \log \frac{1}{\delta\rho_{r,s}} \quad \Longrightarrow \quad \frac{G^a_{r,s+N}}{G^a_{r,s}} \geq \frac{1}{\delta\rho_{r,s}}.$$

The expert starts the block with capital $\rho_{r,s} = c_r\eta_{r,a}G^a_{r,s}$. Multiplying this capital by its growth factor gives its contribution to the total wealth at $s + N$. Since the reserve and all other accounts are nonnegative,

$$M_{s+N} \geq c_r\eta_{r,a}G^a_{r,s+N} = \rho_{r,s}\frac{G^a_{r,s+N}}{G^a_{r,s}} \geq \rho_{r,s}\frac{1}{\delta\rho_{r,s}} = \frac{1}{\delta}.$$

Thus, the monitor has raised an alarm by index $s + N$.

The score event has conditional probability at least $1 - \beta$ given $\mathcal{F}_s$, and the duration condition is $\mathcal{F}_s$-measurable. Hence, on that condition,

$$\mathbb{P}(\tau_\delta \leq s + N \mid \mathcal{F}_s) \geq 1 - \beta \qquad \text{almost surely.}$$

□

The sufficient duration increases quadratically as the observable excess $d_\star - s_\star$ decreases, and logarithmically as the expert's capital at the start of the block decreases. For an expert launched at s, this capital is $\rho_{s,s} = c_s\eta_{s,a}$. For an expert launched earlier, $\rho_{r,s}$ also accounts for its previous gains and losses. A conservative drift bound can conceal a violation if its expected slack is as large as the excess. With observation gap h, the N scores span physical time hN. The upper score bound is used only for this detection result; false-alarm control still requires only $W_k \geq -1$.

D Implementation

Algorithm 1 implements the monitor using a shared history of the most recent m completed scores and their starting states. Before each new observation, it chooses the funding amount, computes the loss bound, and uses the retained history to choose the experts' stakes. Upon observing the next state, it computes the score, updates the accounts, and checks whether the total wealth reaches $1/\delta$. The completed score is then added to the history.

Algorithm 1 Monitor with combined expert accounts

Require: Certificate f, allowed drift l, spacing h, level δ, history size m;
valid score-bound routines; a fixed set $\mathcal{A}$ of K experts and weights η_a;
predictable funding rule $0 \le c_n \le R_n$.

Observe X_{t_0}; initialise $R_0 \leftarrow 1$, $A_0^a \leftarrow 0$ for every $a \in \mathcal{A}$, and empty history.
for $n = 0, 1, \ldots$ while observations continue **do**
  Choose c_n and compute B_n using information available at t_n.
  Compute prediction weights from the retained history using (4).
  If $X_{t_n} \in \mathcal{S}$ choose each expert's stake λ_n^a using (6) , or (5), otherwise $\lambda_n^a = 0$.
  Observe $X_{t_{n+1}}$ and compute $\overline{L}_n$ and W_n.
  Update the combined accounts and reserve using (20).
  if $R_{n+1} + \sum_{a \in \mathcal{A}} A_{n+1}^a \ge 1/\delta$ **then**
    return alarm at index $n + 1$.
  end if
  Append (t_n, X_{t_n}, W_n) to the history and retain its most recent m entries.
end for

Combining accounts. Repeated funding need not require storing a separate account for every starting index. Suppose that every mixture uses the same finite set of experts $\mathcal{A}_s = \mathcal{A}$, with $|\mathcal{A}| = K$, and the same nonnegative weights $\eta_{s,a} = \eta_a$ satisfying $\sum_{a \in \mathcal{A}} \eta_a = 1$. Each expert keeps its kernels, scales, and stake strategy fixed and uses the shared retained history. For the moment and log-return strategies above, its stake therefore does not depend on the starting index:

$$\lambda_{s,n}^a = \lambda_n^a \qquad \text{for every } s \le n.$$

The accounts of expert a may hold different amounts of capital, but the next score multiplies each of them by the same factor $1 + \lambda_n^a W_n$. We can therefore store their combined capital:

$$A_n^a := \sum_{s=0}^{n-1} c_s \eta_a G_{s,n}^a, \qquad R_n := 1 - C_n,$$

where R_n is the remaining reserve. Funding at index n adds $c_n \eta_a$ to the combined account of expert a and removes c_n from the reserve. Applying the common betting factor gives

$$A_{n+1}^a = (A_n^a + c_n \eta_a)(1 + \lambda_n^a W_n), \qquad R_{n+1} = R_n - c_n, \tag{20}$$

with $A_0^a = 0$, $R_0 = 1$, and total wealth $M_n = R_n + \sum_{a \in \mathcal{A}} A_n^a$. Multiplication distributes over the individual account balances, so this update gives exactly the same total wealth and alarm time as maintaining those accounts separately. The optimisation requires the combined accounts to use identical stakes. Accounts whose scales or prediction histories depend on their starting indices must generally remain separate.

The history stores each score with its starting state because spatial prediction compares the states from which the corresponding transitions began. Removing an old observation from the history affects future stake choices but leaves accumulated wealth unchanged. If observations end upon reaching a target or another observed termination condition, the monitor processes the final completed interval before stopping.

Computational cost. For states in $\mathbb{R}^d$, the retained history requires $O(md)$ storage and the combined accounts require $O(K)$. Processing prediction weights one expert at a time therefore gives total storage $O(md + K)$ for the betting implementation. At each observation, distances to the retained states can be computed once and shared across experts, costing $O(md)$. Each moment strategy then computes two weighted sums in $O(m)$ time, giving $O(md + Km)$ work per observation. For log-return stakes, each derivative evaluation scans the retained history. Using J bisection iterations therefore gives $O(md + KmJ)$ work. Finite bisection approximates the maximiser, but any returned stake in $[0, \overline{\lambda}]$ preserves false-alarm control. These bounds exclude the resources required by the certificate, score-bound, and funding routines.

Recursive updates for unlimited history. For uniform or purely temporal geometric weights, an expert using the moment strategy can use the entire history without storing individual observations.

Since its temporal scale is fixed, write r_t^a for this scale and set $\gamma^a = 2^{-h/r_t^a}$ for geometric weighting, or $\gamma^a = 1$ for uniform weighting. Rescaling the weights to $(\gamma^a)^{n-1-i}$ leaves the stake unchanged, since the common factor cancels in the ratio of weighted sums. Starting from $S_0^a = U_0^a = H_0^a = 0$, we maintain the resulting sums through

$$S_{n+1}^a = \gamma^a S_n^a + W_n, \qquad U_{n+1}^a = \gamma^a U_n^a + W_n^2, \qquad H_{n+1}^a = \gamma^a H_n^a + 1.$$

For $n \geq 1$, the weighted mean and second moment are S_n^a/H_n^a and U_n^a/H_n^a. The moment stake uses S_n^a and U_n^a directly, so maintaining H_n^a is optional. When $\gamma^a = 1$, these recursions give the ordinary sums and $H_n^a = n$; when $\gamma^a < 1$, each update discounts earlier observations. For K such experts, maintaining these sums requires $O(K)$ work per observation and $O(K)$ storage, independently of the history length. This optimisation does not generally extend to spatial weights, which change with the current state, or to exact log-return optimisation, whose objective cannot be recovered from these sums alone.

E Experimental setup

Prediction, stakes and funding. Uniform prediction (U) assigns equal weight to all retained scores. Temporal prediction (T) favours recent scores, while spatial prediction (S) favours scores whose starting states are close to the current state. Temporal and spatial prediction use the linear kernel $k(u) = (1-u)_+$ with fixed scales.

For the synthetic benchmarks, we retain 16384 observations. We use five scales, labelled XS–XL, calibrated to reference effective sample sizes $32, 128, 512, 2048, 8192$, where

$$\mathrm{ESS}(w) = \frac{(\sum_i w_i)^2}{\sum_i w_i^2}.$$

The temporal cutoffs are approximately 43.166, 171.167, 683.167, 2731.167 and 10923.167 observations. We evaluate the individual scales, an equal mixture, a centred mixture with weights $(0.1, 0.2, 0.4, 0.2, 0.1)$, and five equal mixtures that each omit one scale. For the control and neural benchmarks, we use U-L, T-L and S-L with a history of 8192 observations. T-L equally mixes temporal cutoffs $64, 128, 512, 2048, 8192$, while S-L equally mixes spatial radii $0.00625, 0.025, 0.1, 0.4, 1.6$. Log-return optimisation uses endpoint checks followed, when necessary, by 32 bisection steps. We launch accounts every 256 observations. Launch $j \geq 0$ receives $6/[\pi^2(j+1)^2]$ of the initial capital in the synthetic benchmarks and $1/[(j+1)(j+2)]$ in the control and neural benchmarks.

E.1 Synthetic benchmarks

The synthetic benchmarks isolate different sources of detection difficulty (Figures 1–4). We consider six families of processes.

Temporal: The scalar process starts at $X_0 = 32769$ and follows

$$X_{n+1} = X_n - 0.001 + g_n + \xi_n, \qquad \xi_n \overset{\text{i.i.d.}}{\sim} \mathrm{Unif}[-0.6, 0.6].$$

With certificate $f(x) = x$ and allowance -0.001, the conditional expected corrected increment is $\mathbb{E}[D_n \mid \mathcal{F}_n] = g_n$. Persistent violations use $g_n \in \{0.01, 0.02, 0.05\}$ throughout. Delayed violations use $g_n = -0.02$ until onset 2048, 4096 or 8192, followed by $g_n = 0.05$. Temporary violations use $g_n = 0.05$ for 256, 1024 or 4096 observations from onset 8192, and $g_n = -0.02$ otherwise. Each run lasts 32768 observations.

Spatial: The first cyclic system follows a deterministic circular path with directly prescribed scores. It does not use a controller–certificate pair, and its conditional score mean jumps at the sector boundaries, so it is not globally Lipschitz. Each period starts with 64 observations of mean 0.10 in a fixed sector, followed by observations of mean -0.02. Independent symmetric score noise has amplitude 0.3. We use periods $P \in \{512, 2048, 4096\}$ and horizons $8P$. Each run therefore contains exactly 512 positive-mean observations; increasing P lengthens the gaps between visits.

The second system is a stochastic ring with noise in both radial position and angular speed. The certificate is the squared radial error, and the allowance is its exact nominal expected increment. The

radial contraction is 0.2, radial noise is uniform on $[-0.01, 0.01]$, and angular speed has independent uniform modulation of $\pm 20\%$. A triangular angular envelope with half-width 0.2 reduces the radial feedback by up to 75%. We use the same periods and horizons, with a nominal sector visit of 256 observations. Stochastic return times make the number of observations under disturbance vary across runs.

Hill: We use the robot from the running example. It starts at $X_0 = 10$ and stops upon reaching $[0, 1]$. Before stopping,

$$X_{n+1} = X_n - 0.01 + 0.02Z_n, \qquad Z_n \mid \mathcal{G}_n \sim \text{Bernoulli}\big(1/4 + a(1 - |X_n - 5|/2)_+\big).$$

With certificate $f(x) = x$, allowance -0.005 and loss bound 0.02, the conditional expected score is $a(1 - |X_n - 5|/2)_+$. Amplitudes $a \in \{0.0625, 0.125, 0.25\}$ reduce expected progress at the hill centre by one quarter, one half or entirely. Stronger disturbances also keep the robot on the hill for longer.

For the cyclic systems, increasing the period changes the horizon, the observations available in the retained history and the number of funding opportunities, even though positive exposure is fixed for direct recurrence. Including the scalar, ring and hill references gives 21 settings. Direct recurrence has no separate saved reference. The six example trajectories use seed 970000003 and settings with weak violations, late onset, short duration or long gaps between visits. These figures illustrate individual runs; the heatmaps summarise all seeds.

E.2 Control benchmarks

Building and Two Tanks adapt benchmarks from PRoTECT (Wooding et al., 2026). The stochastic double-integrator LQR uses a standard quadratic value function (**?**). Each system has one reference setting and two disturbances, each evaluated at three strengths, giving 21 settings in total. All disturbances act from the start of the run. The certificates and allowances remain fixed.

Building: We reduce heating effectiveness by 8%, 10% or 12%, or make the controller underestimate the temperature by 1, 2 or 3 degrees. In the latter case, the monitor observes the true temperature.

Tanks: We add inlet flow of amplitude 0.2, 0.3 or 0.4, which fades as the downstream level rises from 2.15 to 2.25. Alternatively, we add independent inlet bursts of size 0.5 with probability 0.35, 0.40 or 0.45 per step.

Building and Tanks use bounded triangular noise with the same means and variances as their source models. Their newly synthesised sum-of-squares certificates have degrees two and four, respectively. Independent checks using exact arithmetic validate the nominal allowances 0.04 and 0.07 under these noise distributions. Initial states are uniform on $[19.5, 20]$ for Building and on $[4.45, 4.55] \times [1.45, 1.55]$ for Tanks.

LQR: The initial state lies in $[1, 2] \times [-0.1, 0.1]$. The nominal acceleration diffusion is 0.1. We sample every 0.1 time units using the exact Gaussian transition and use the exact nominal sampled certificate increment as the allowance. Disturbances either reduce feedback by 5%, 10% or 20%, or multiply the noise amplitude by 1.05, 1.10 or 1.20.

All control runs have a limit of 32768 observations. Building and Tanks also stop upon entering an unsafe set or leaving the domain; LQR stops upon leaving $[-10, 10]^2$. We compute conditional moments exactly along each recorded trajectory. These determine the fraction of runs encountering a violation and the expected-score summaries (Figures 12–14).

E.3 Neural controller–certificate pairs

We retain 16 original controller–certificate pairs from LogRASM: four CollisionAvoidance pairs, six Linear layout-0 pairs, two Linear layout-1 pairs, two Pendulum pairs and two Triple-integrator pairs. The controller and certificate parameters remain fixed. We consider three disturbance magnitude per setting. Initial states are sampled from fixed pools away from the target, unsafe sets and domain boundaries, with at least 20% headroom below the certificate threshold. Starts are shared across disturbance strengths and pairs within the same layout.

To obtain longer trajectories, we multiply the source drift increments by $1/20$ and the direct plant-noise amplitudes by $1/\sqrt{20}$. The baseline controller effectiveness is 0.1 for CollisionAvoidance, 0.5

and 0.25 for the two Linear layouts, 1 and 0.5 for the ordinary and logarithmic Pendulum pairs, and 1 for Triple integrator.

For CollisionAvoidance, we add terrain-dependent slips with peak probabilities 0.25, 0.35 and 0.50. For the Linear systems, we add fluctuating sensor offsets with amplitudes 0.30, 0.35 and 0.40. For Pendulum, random torque pushes the system away from the upright position, with amplitudes 0.075, 0.10 and 0.125. For Triple integrator, we add a positive actuator bias with amplitudes 0.06, 0.08 and 0.10. Paths stop upon reaching the target, entering an unsafe set, leaving the domain, reaching the certificate threshold or completing 20480 observations.

Let $C = 1/(1-\rho)$. For an ordinary certificate output V, we use $f(x) = \min\{V(x), C\}$. For a logarithmic output L, we use $f(x) = \exp(\min\{L(x), \log C\})$. We evaluate each recorded path under three decrease requirements, $\epsilon \in \{0, 0.0005, 0.001\}$. The allowance is $-\epsilon$ for ordinary certificates and $(e^{-\epsilon} - 1)f(x)$ for exponentiated logarithmic certificates. Thus, $\epsilon = 0$ tests expected nonincrease; positive values require additive or multiplicative decrease, respectively. All three conditions use the same predictable loss bound and the same physical trajectories (Figures 15 and 16).

F SYNTHETIC

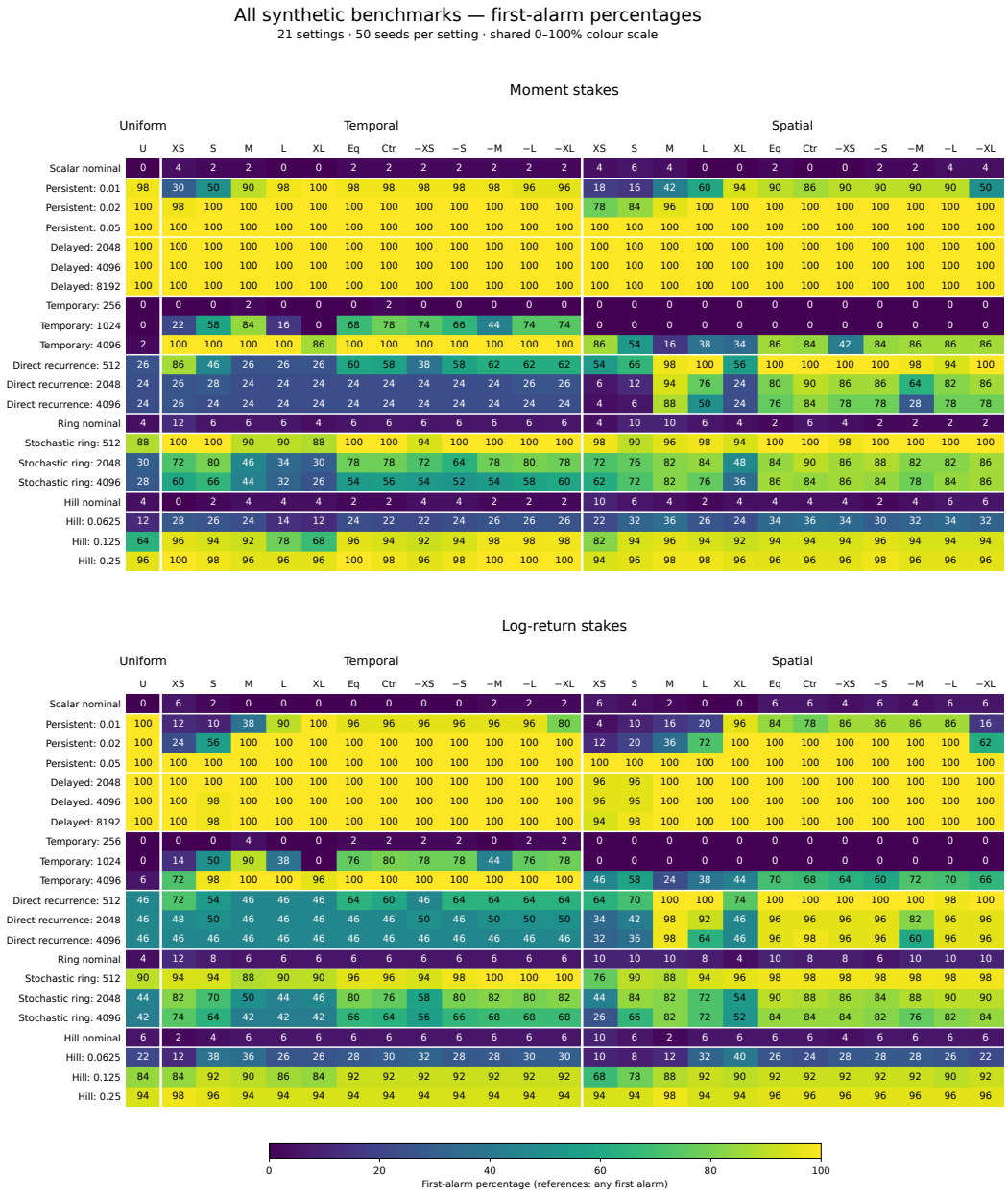

Figure 1: Detection rates for the synthetic benchmarks, with 21 settings, 50 monitors and 50 paired runs per setting. Cells show the percentage of runs with an eligible first alarm; reference rows count any first alarm. The panels separate moment (M) and log-return (L) stakes. U uses uniform prediction, while T and S use temporal and spatial prediction. XS–XL denote individual scales; Eq is their equal mixture; Ctr uses weights $(0.1, 0.2, 0.4, 0.2, 0.1)$. A minus label indicates that the corresponding scale is omitted from an otherwise equal mixture. All monitors use wealth threshold 20. The panels share a colour scale from 0 to 100%. NA denotes incomplete data.

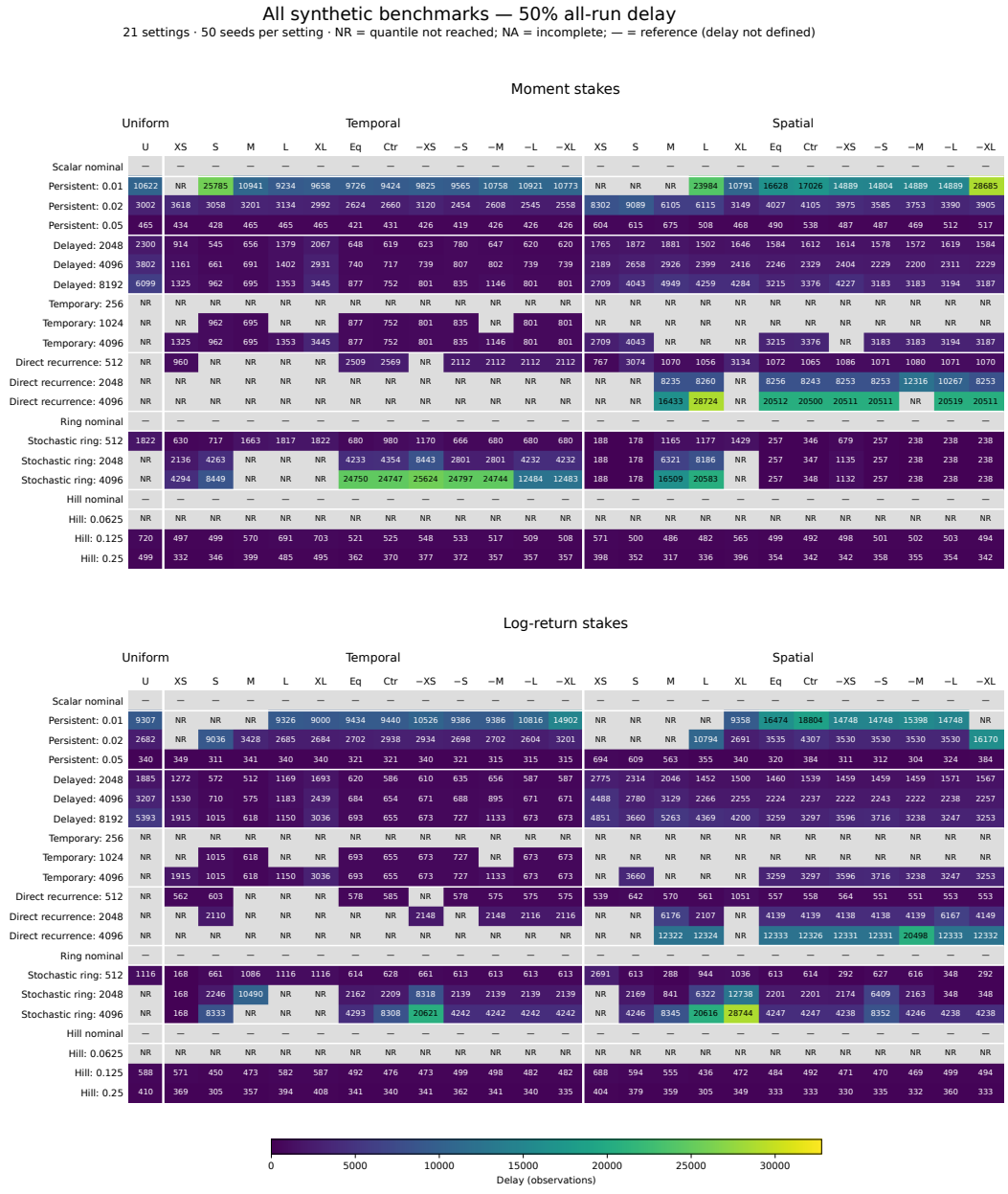

Figure 2: The 50% detection delays for the synthetic monitors in Figure 1. Each value is the first delay by which 25 of all 50 runs have eligible first alarms, measured in observations since the declared onset or first entry into the region with positive conditional mean. Nondetections, short paths and early first alarms remain in the denominator. A later crossing does not replace an early first alarm. NR denotes a median that was not reached, NA incomplete data, and a dash an undefined reference delay. Both panels use the same colour scale.

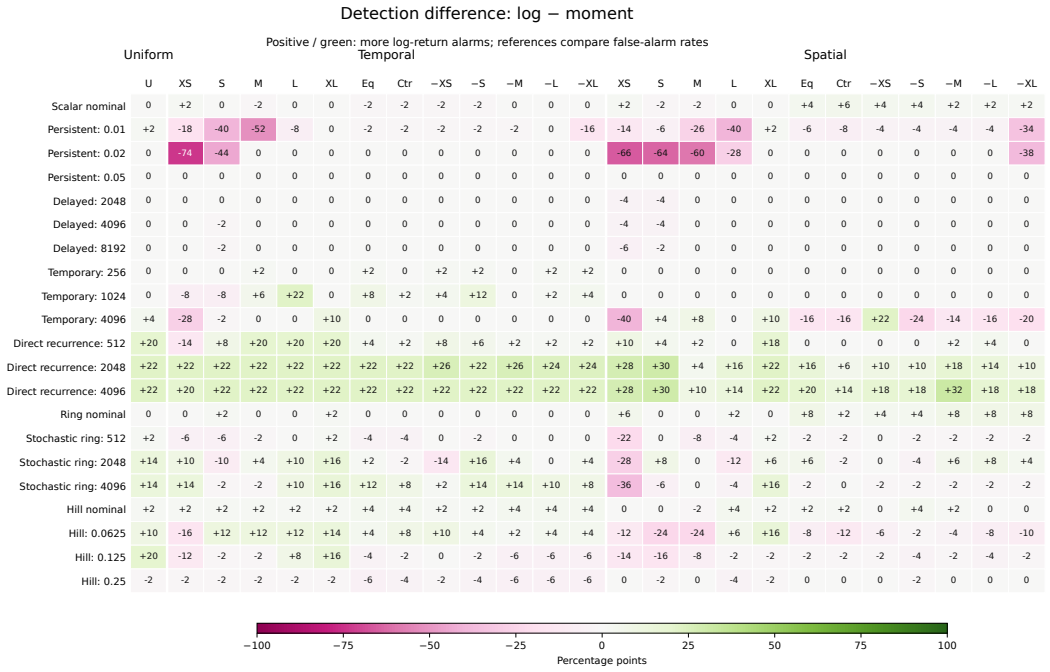

Figure 3: Differences in synthetic detection rates between log-return and moment stakes, computed as L−M in percentage points. Each comparison uses the same prediction rule and the same 50 paths. Positive green cells indicate more alarms with log-return stakes; negative cells indicate more alarms with moment stakes. For reference rows, more alarms mean a higher false-alarm rate. Column labels follow Figure 1. The comparison includes the different stake caps, 0.99 for L and 0.5 for M. These differences are descriptive; no significance test is applied.

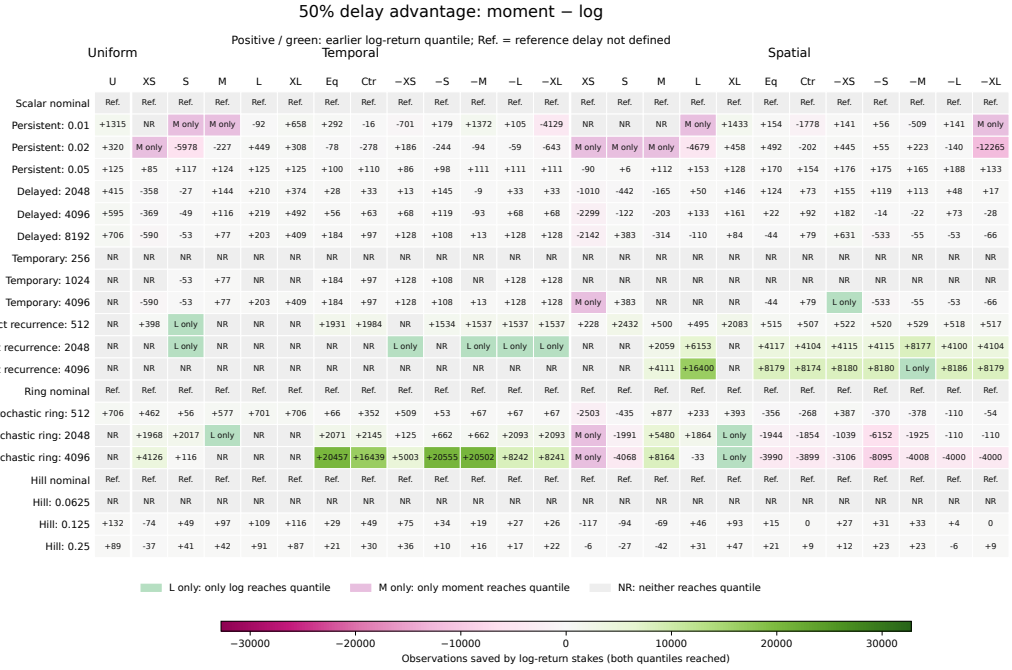

Figure 4: Differences in synthetic median detection delays, computed as $Q^{\mathrm{M}}_{0.5} - Q^{\mathrm{L}}_{0.5}$ in observations. Positive green values indicate earlier detection with log-return stakes when both strategies reach 25 eligible first alarms among all 50 runs. L only and M only indicate that only the named strategy reaches this fraction; NR means neither does. Reference delays are undefined (Ref.), and NA denotes incomplete data. No artificial delays are assigned to non-numeric cells. The comparisons use the same prediction rules and paths, with the respective stake caps of each strategy.

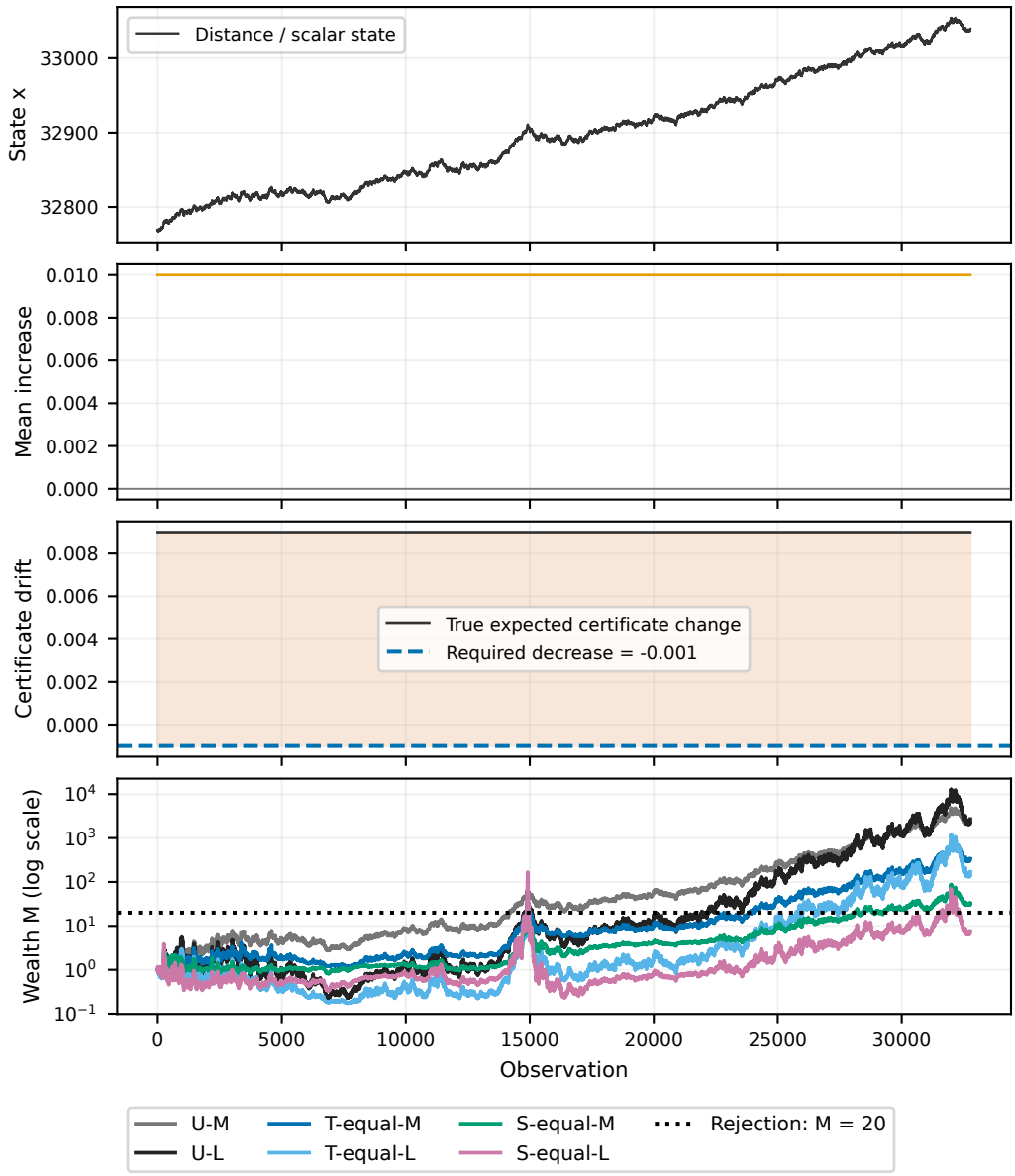

Figure 5: Persistent violation with $g = 0.01$ and seed 970000003. From top to bottom: the scalar state; mean increase g; the exact expected certificate increment $g - 0.001 = 0.009$ and allowance -0.001; and monitor wealth on a logarithmic scale. Uniform noise on $[-0.6, 0.6]$ makes individual increments noisy despite the positive conditional mean of the corrected increment. U uses uniform prediction, while T-equal and S-equal mix the five XS–XL experts equally. M and L denote moment and log-return stakes. The dotted line marks wealth threshold 20. All 32768 observations are shown, including those after alarms.

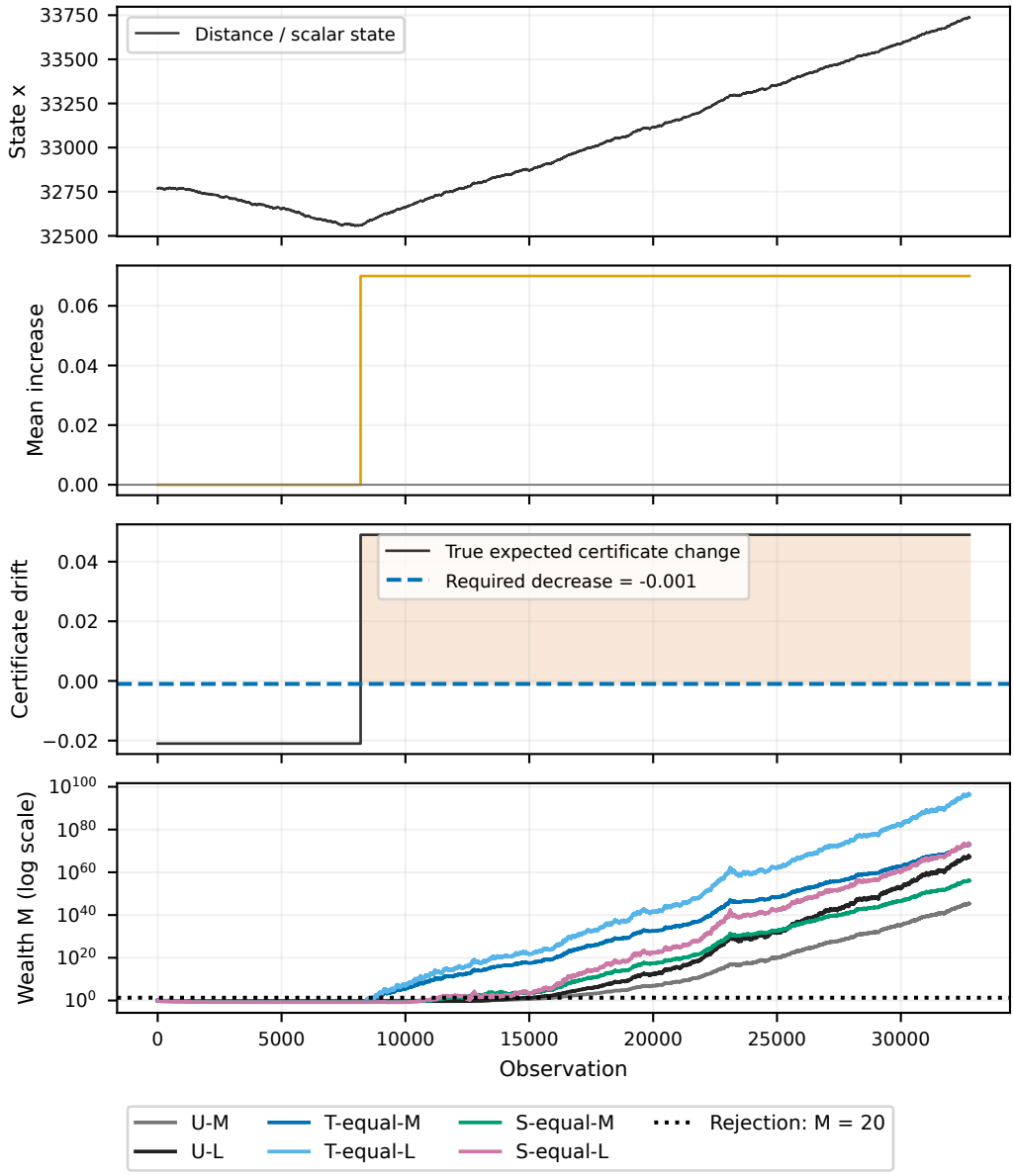

Figure 6: Delayed violation with onset 8192 and seed 970000003. The conditional mean of the corrected increment changes from -0.02 to 0.05 and remains positive. From top to bottom: the state; mean increase 0.07; the expected certificate increment, changing from -0.021 to 0.049, and allowance -0.001; and logarithmic monitor wealth. Temporal prediction reduces the influence of observations collected before onset as new scores arrive. U, T-equal and S-equal are shown with both M and L stakes, as in Figure 5. The wealth threshold is 20. Neither the history nor wealth is reset at onset.

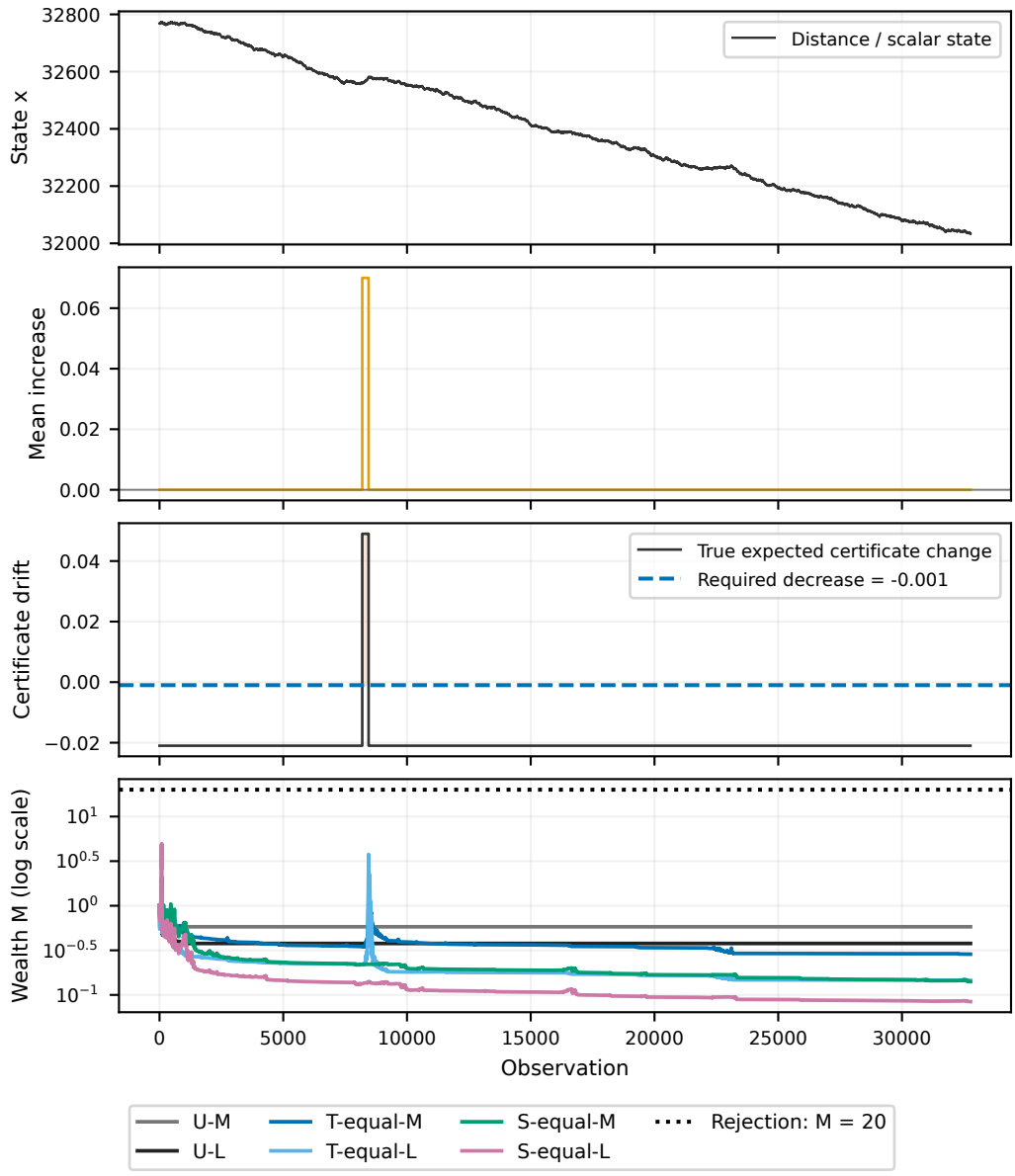

Figure 7: Temporary violation lasting 256 observations from onset 8192, with seed 970000003. The conditional mean of the corrected increment is 0.05 during the disturbance and -0.02 before and after it. From top to bottom: the state; temporary mean increase 0.07; the expected certificate increment and allowance -0.001; and logarithmic wealth for U and the equal T/S mixtures with M and L stakes. None of the six displayed monitors reaches the dotted threshold 20. The path is shown through observation 32768, including recovery. The aggregate detection summaries also count eligible first alarms occurring after recovery.

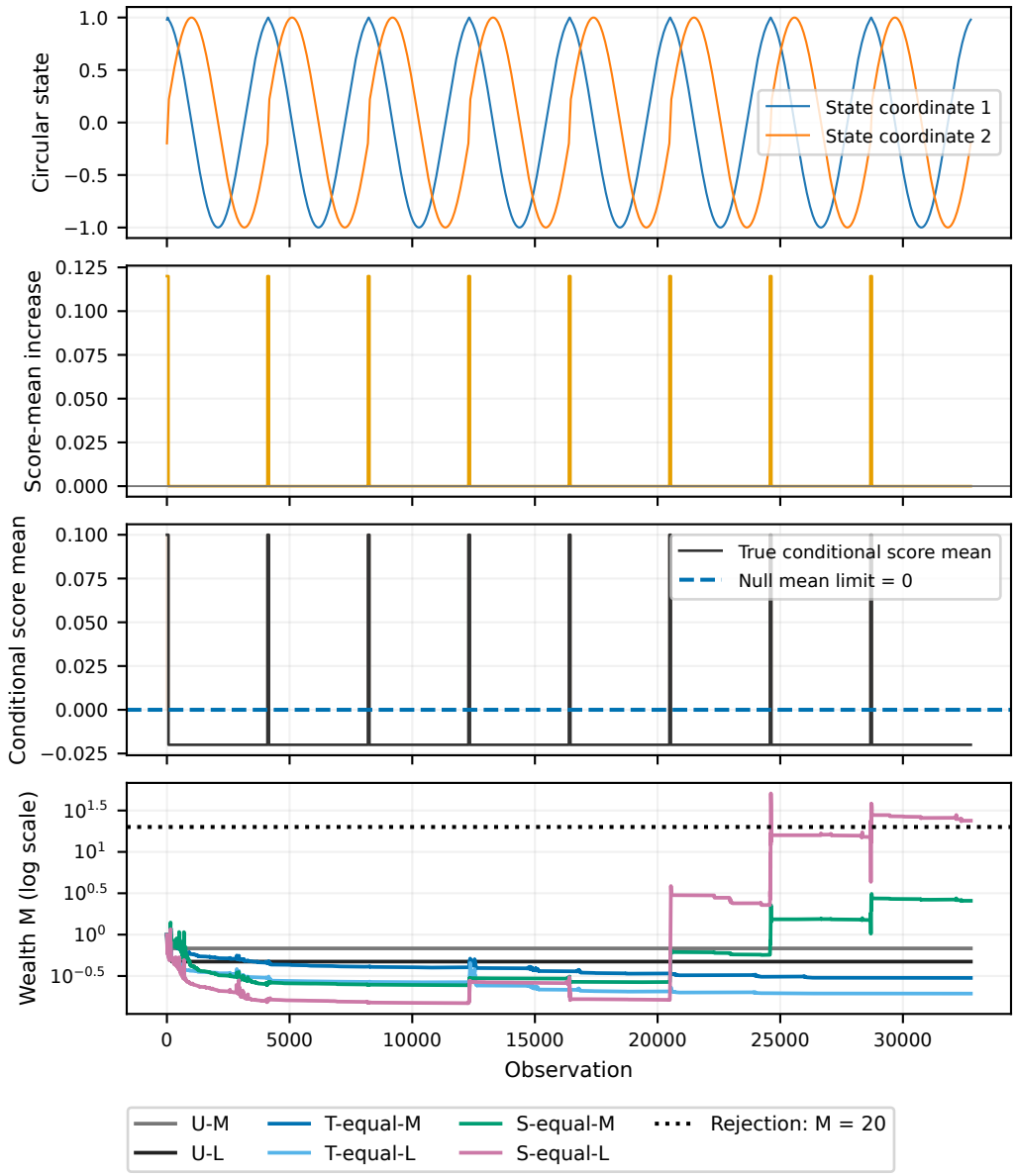

Figure 8: Direct recurrence with period $P = 4096$ and seed 970000003. Each of eight visits to a fixed circular sector contains 64 observations with score mean 0.10. Between visits, the mean is -0.02. Independent symmetric score noise has amplitude 0.3. From top to bottom: the two state coordinates; local mean increase 0.12; the prescribed conditional score mean relative to zero; and logarithmic wealth for U and the equal T/S mixtures with M and L stakes. The 32768 observations include exactly 512 with positive conditional mean. Spatial prediction can reuse observations from earlier visits despite the long gaps between them. The dotted line marks wealth threshold 20.

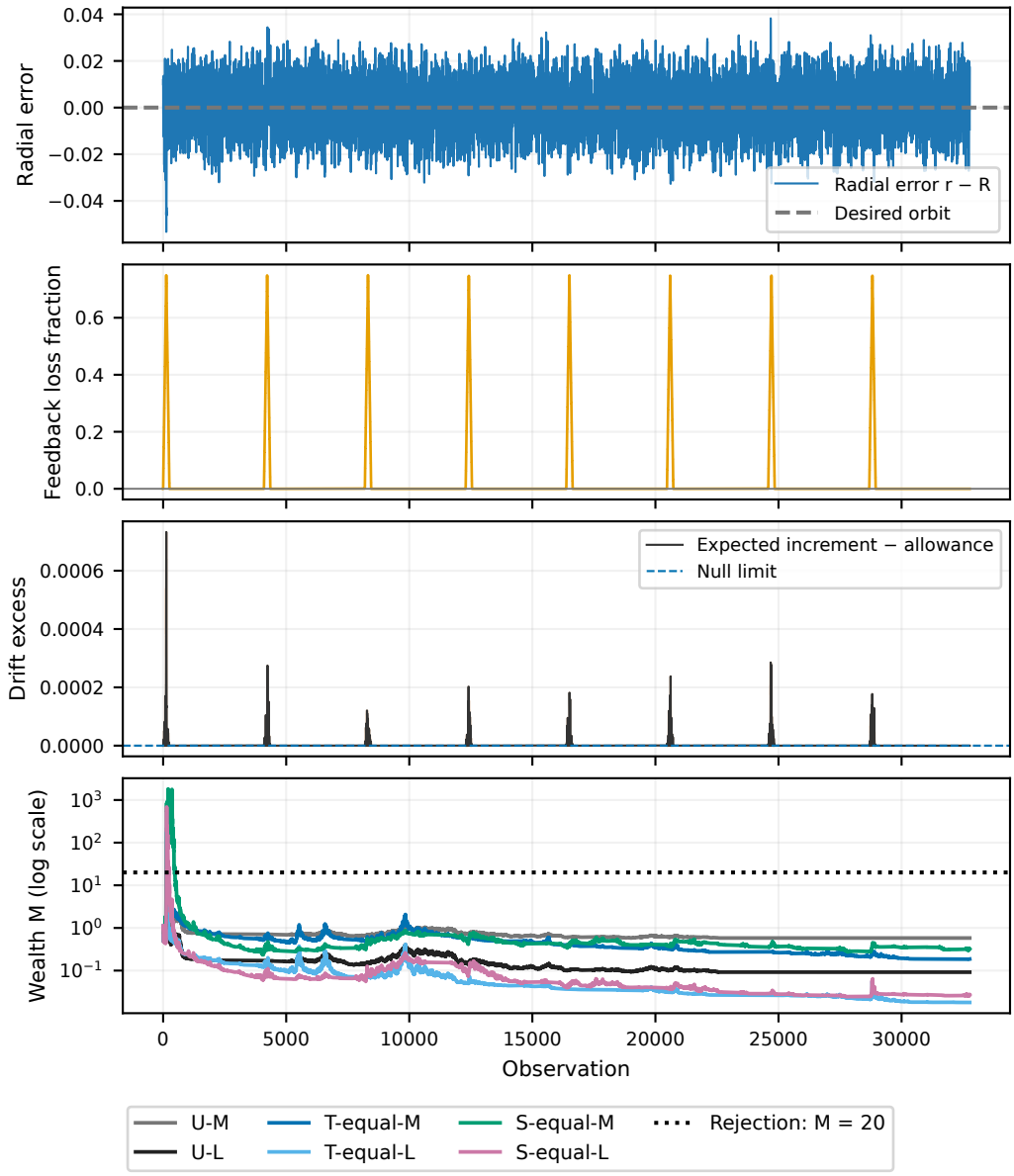

Figure 9: Stochastic ring with nominal period $P = 4096$ and seed 970000003. From top to bottom: radial error relative to the unit circle; the state-dependent reduction in restoring feedback, reaching 0.75 in the affected sector; the exact expected certificate increment minus its nominal allowance; and logarithmic monitor wealth. The certificate is the squared radial error. Radial and angular noise make return times and violation exposure vary across runs. U and the equal T/S mixtures are shown with M and L stakes. The dotted line marks wealth threshold 20. The expected-increment condition can be violated even while the process remains close to the circle. All observations are retained, including those after alarms.

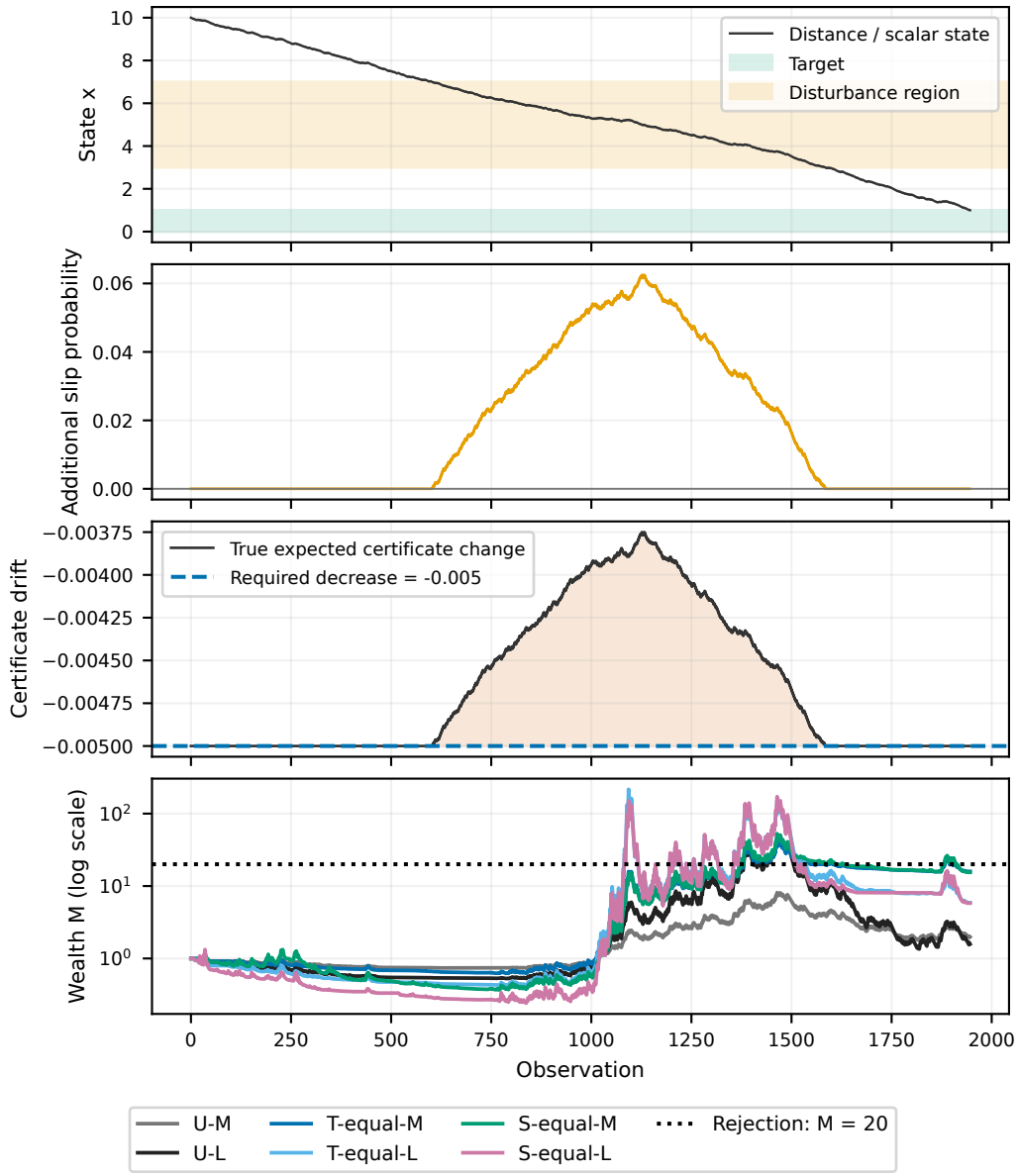

Figure 10: Hill disturbance with amplitude $a = 0.0625$ and seed 970000003. From top to bottom: distance to the target $[0, 1]$, with the hill region $(3, 7)$ shaded; additional slip probability $a(1 - |x - 5|/2)_+$; the exact expected certificate increment and required allowance -0.005; and logarithmic wealth for U and the equal T/S mixtures with M and L stakes. At $x = 5$, the expected increment is -0.00375, a 25% reduction in progress. The robot therefore still approaches the target in expectation while violating the required rate of decrease. The dotted line marks wealth threshold 20. The run ends when the robot reaches the target, not when a monitor raises an alarm.

Synthetic processes, ring and hill
Normalised expected-score summaries

	Positive observations	Signed mean	Mean when positive	Mean positive part
Scalar				
Scalar nominal	0.0%	0	—	0
Persistent: 0.01	100.0%	0.00999	0.00999	0.00999
Persistent: 0.02	100.0%	0.02	0.02	0.02
Persistent: 0.05	100.0%	0.05	0.05	0.05
Delayed: 2048	100.0%	0.05	0.05	0.05
Delayed: 4096	100.0%	0.05	0.05	0.05
Delayed: 8192	100.0%	0.05	0.05	0.05
Temporary: 256	1.0%	-0.0193	0.05	0.00052
Temporary: 1024	4.2%	-0.0171	0.05	0.00208
Temporary: 4096	16.7%	-0.00833	0.05	0.00833
Direct recurrence				
Direct recurrence: 512	12.5%	-0.005	0.1	0.0125
Direct recurrence: 2048	3.1%	-0.0163	0.1	0.00313
Direct recurrence: 4096	1.6%	-0.0181	0.1	0.00156
Ring				
Ring nominal	0.0%	0	—	0
Stochastic ring: 512	49.9%	0.0552	0.111	0.0552
Stochastic ring: 2048	12.5%	0.0136	0.109	0.0136
Stochastic ring: 4096	6.3%	0.00682	0.109	0.00682
Hill				
Hill nominal	0.0%	0	—	0
Hill: 0.0625	69.5%	0.0229	0.0329	0.0229
Hill: 0.125	73.8%	0.051	0.0695	0.051
Hill: 0.25	86.7%	0.157	0.18	0.157

Cell = median of 50 per-run summaries. IQRs: expected_scores_summary.csv.
Interval: onset/entry to stop; references: full path. Recovery is included.
NA: incomplete setting. Hatched dash: no positive observations.
Shared column scales across these maps; signed colours centre on zero.

Figure 11: Normalised expected-score summaries for the synthetic benchmarks. Each cell is the median of 50 quantities computed separately for each run: the fraction of observations with $\mu_n > 0$, the signed mean of μ_n, its mean over positive-mean observations, and the mean of $(\mu_n)_+$ over all observations. Summaries start at onset or first entry and include recovery and later excursions. References use the full path. Each run receives equal weight. A hatched dash marks an undefined mean over positive observations; NA denotes incomplete data. Column colour scales match the control heatmap, with the signed scale centred at zero.

G CONTROL

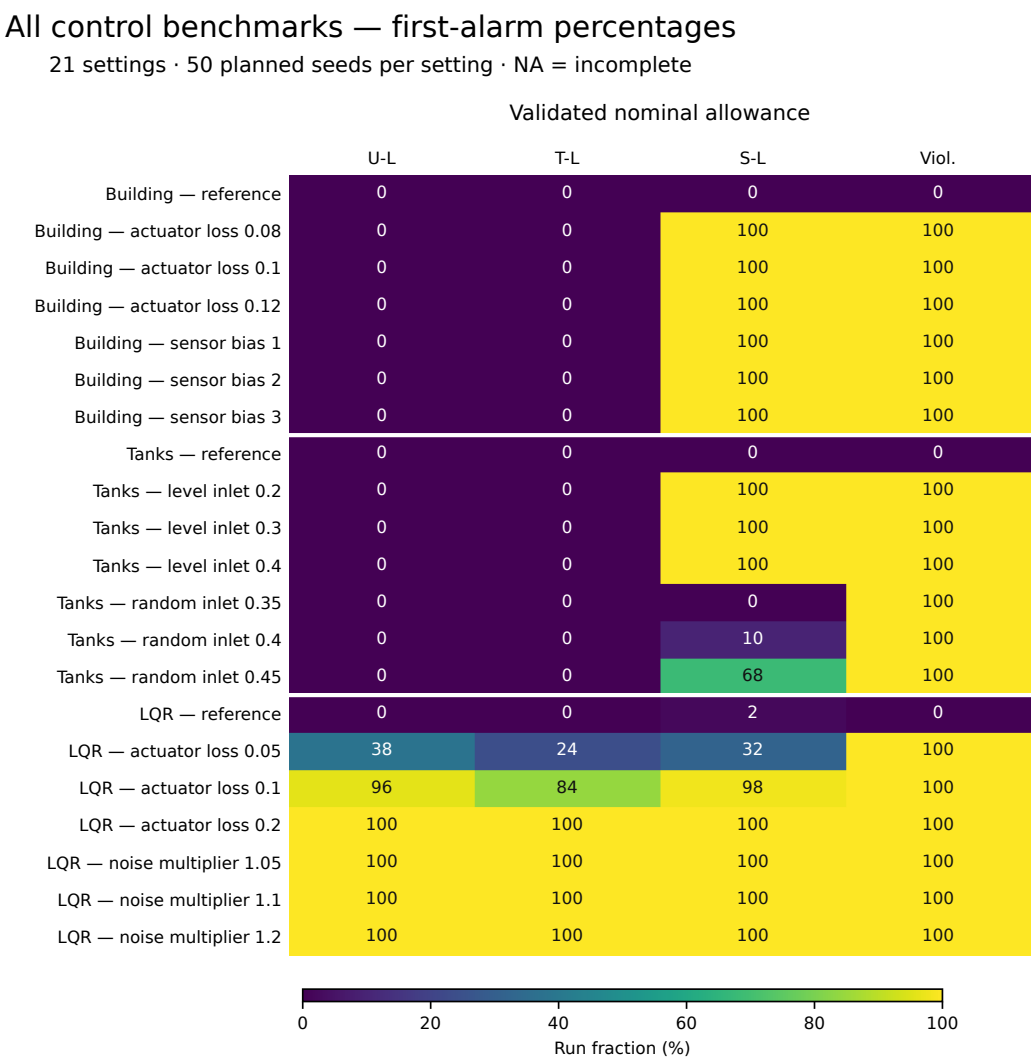

Figure 12: First-alarm percentages for Building, Two Tanks and LQR. Each system has a reference setting and two disturbances, each evaluated at three strengths. U-L, T-L and S-L use uniform, temporal and spatial prediction with log-return stakes and wealth threshold 20. Every cell uses all 50 paired runs. The Viol. column gives the percentage of trajectories encountering at least one positive conditional excess, computed from exact model moments with tolerance 10^{-12} in raw units. This differs from both the alarm rate and the fraction of observations with positive conditional excess. Reference alarms measure false alarms under the validated nominal conditions.

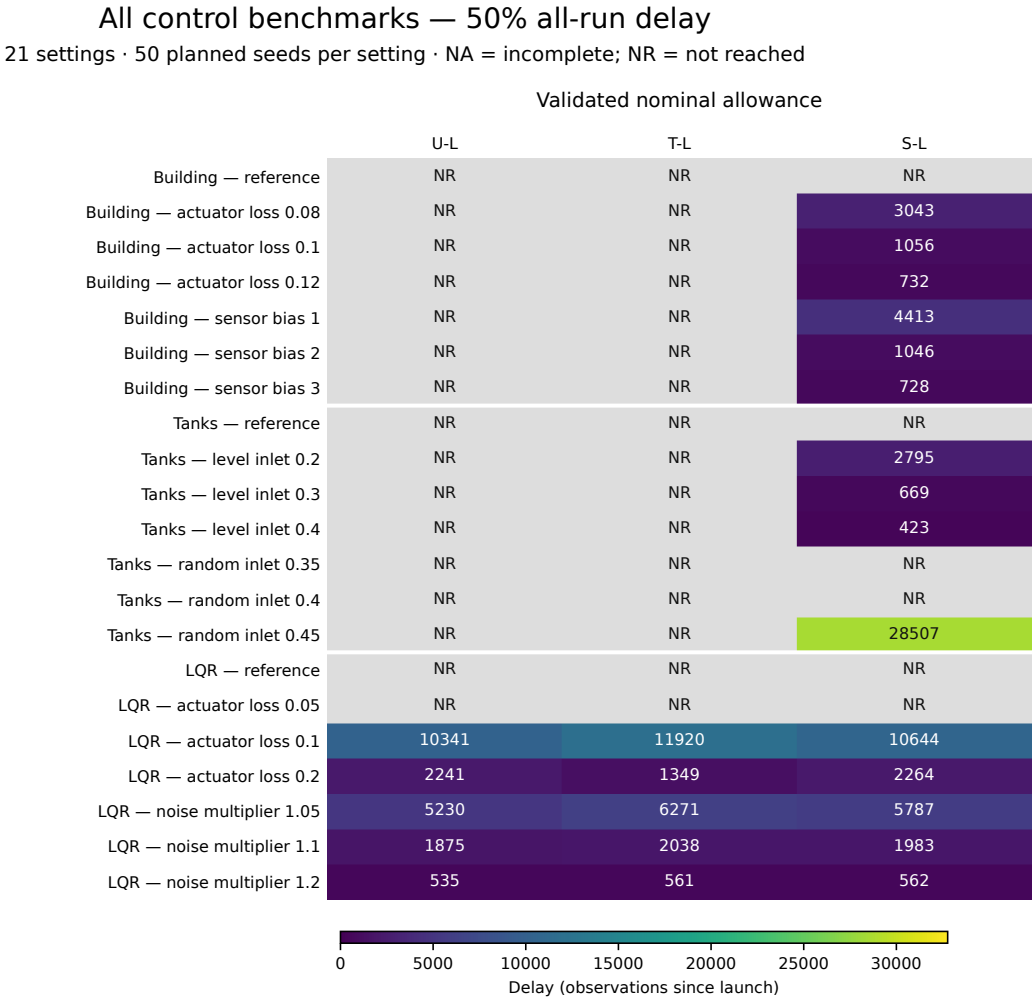

Figure 13: The 50% detection delays for the control settings and log-return monitors in Figure 12. Disturbances act from the start, so each entry is the number of observations until 25 of all 50 runs have raised their first alarm. NR means this fraction was not reached, and NA denotes incomplete data. Nondetections remain in the denominator. Reference delays also start at the beginning of the run and concern false alarms. LQR observations are spaced by 0.1 physical time units; all entries are reported in observations.

Control systems
Normalised expected-score summaries

	Positive observations	Signed mean	Mean when positive	Mean positive part
Building				
Nominal	0.0%	-0.051	—	0
Loss d=0.08	24.9%	-0.0293	0.0151	0.00377
Loss d=0.1	33.6%	-0.0166	0.0285	0.00956
Loss d=0.12	37.6%	-0.00971	0.0309	0.0116
Bias β=1	23.2%	-0.0318	0.0132	0.00306
Bias β=2	32.7%	-0.0186	0.0288	0.00944
Bias β=3	37.0%	-0.0113	0.0327	0.0121
Tanks				
Nominal	0.0%	-0.0642	—	0
Inlet a=0.2	16.1%	-0.0363	0.0141	0.00227
Inlet a=0.3	25.1%	-0.0351	0.0368	0.00923
Inlet a=0.4	24.4%	-0.0338	0.0629	0.0154
Burst p=0.35	5.5%	-0.0172	0.00561	0.00031
Burst p=0.4	8.2%	-0.0149	0.00595	0.000482
Burst p=0.45	11.0%	-0.013	0.00615	0.000677
LQR				
Nominal	0.0%	0	—	0
Loss d=0.05	100.0%	0.00955	0.00955	0.00955
Loss d=0.1	100.0%	0.0191	0.0191	0.0191
Loss d=0.2	100.0%	0.0383	0.0383	0.0383
Noise ν=1.05	100.0%	0.0298	0.0298	0.0298
Noise ν=1.1	100.0%	0.0581	0.0581	0.0581
Noise ν=1.2	100.0%	0.111	0.111	0.111

Cell = median of 50 per-run summaries. IQRs: expected_scores_summary.csv.
Interval: onset/entry to stop; references: full path. Recovery is included.
NA: incomplete setting. Hatched dash: no positive observations.
Shared column scales across these maps; signed colours centre on zero.

Figure 14: Normalised expected-score summaries for the control benchmarks, computed from exact conditional moments along each recorded path. Cells give medians of 50 equally weighted per-run summaries: the fraction of positive-mean observations, the signed mean, the mean over positive-mean observations, and the mean positive part. The last quantity includes all observations in its denominator. A negative signed mean can therefore coexist with local violations. Positivity requires a raw conditional excess above 10^{-12}. Reference and disturbed runs use their full paths, including observations after alarms. A hatched dash marks an undefined mean over positive observations; NA denotes incomplete data. Column colour scales match Figure 11.

H NEURAL

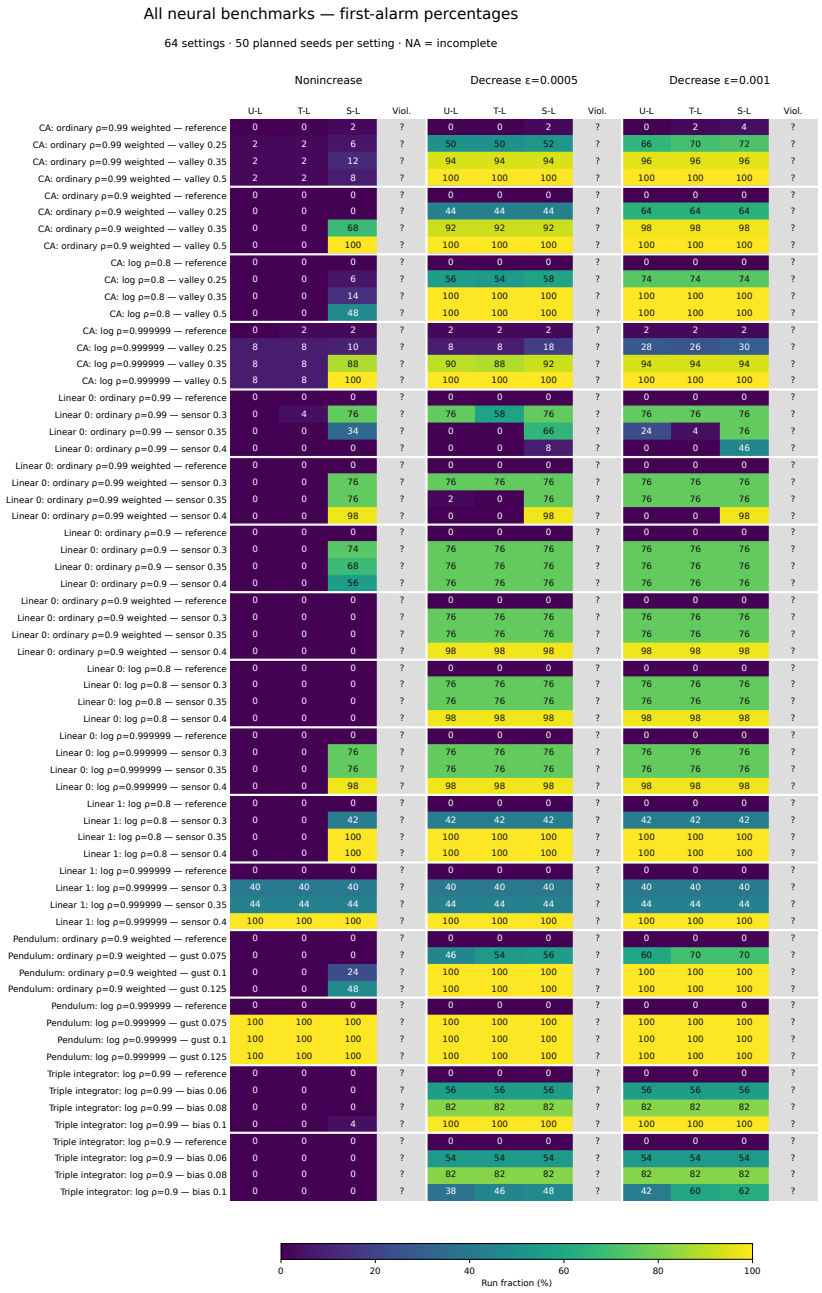

Figure 15: First-alarm percentages for the neural benchmarks. Each of the 16 original controller–certificate pairs has a reference setting and three disturbance strengths. Each row uses 50 trajectories shared by U-L, T-L and S-L and by the conditions $\epsilon = 0, 0.0005, 0.001$. The allowance is $-\epsilon$ for ordinary certificates and $(e^{-\epsilon} - 1)f(x)$ for exponentiated logarithmic certificates. All monitors use wealth threshold 20. Question marks in Viol. indicate that the conditional mean estimates at selected states do not determine the exact fraction of runs encountering a violation. All three conditions remain unverified under the adapted dynamics, including in the reference settings. Reference alarms therefore do not measure false-alarm rates under a verified null.

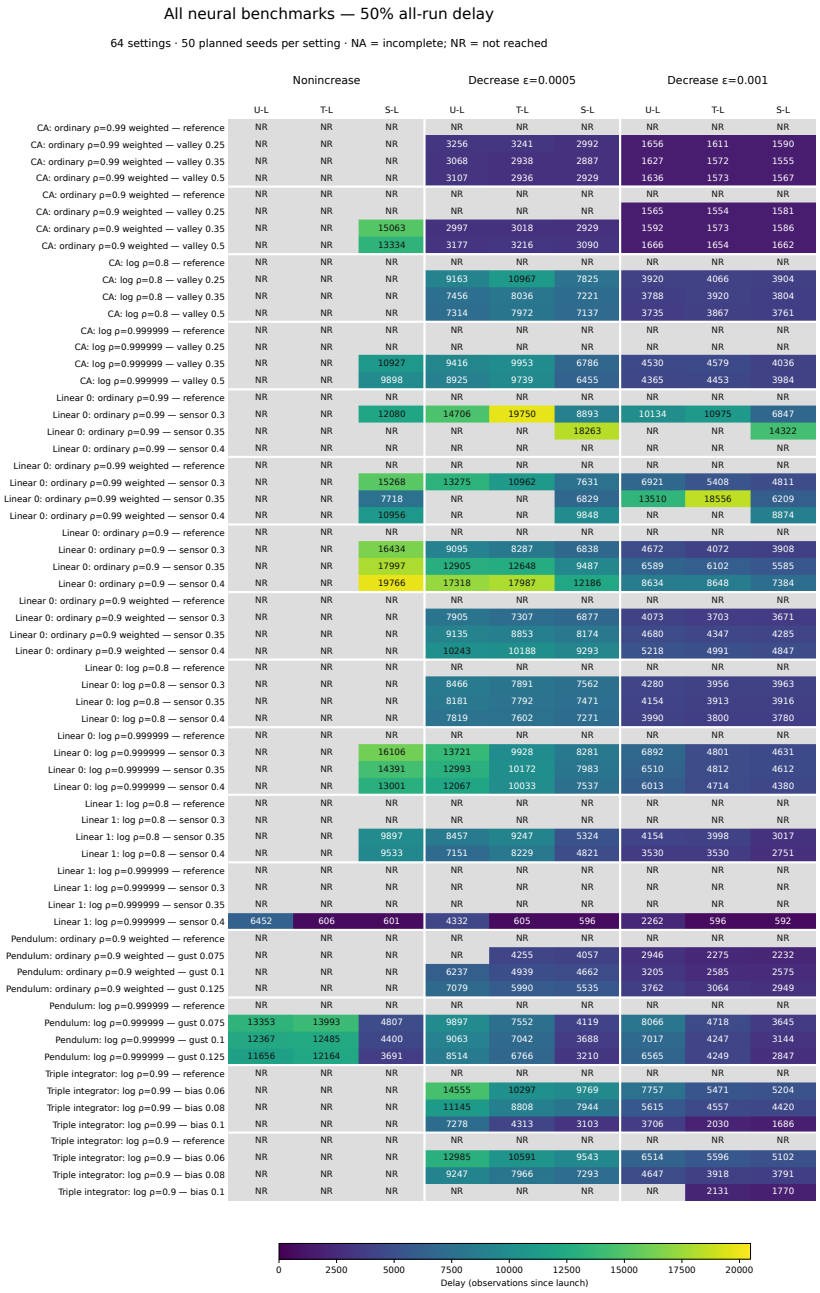

Figure 16: The 50% detection delays for the neural pairs, disturbances and conditions in Figure 15, measured in observations from the start of each run. Each finite entry requires 25 first alarms among all 50 paired paths. Short paths and nondetections remain in the denominator. NR means this fraction was not reached, and NA denotes incomplete data. Paths end at the target, an unsafe set, domain exit, certificate threshold or 20480 observations. Detection therefore depends on both the evidence available and the length of the trajectory. All condition blocks share a colour scale. The different conditions change the allowance on the same physical trajectories.