Maintaining the Means of Knowing: Recurrent Diagnostic Capacity with Exact and Erasure-Prone Calibration

Brian Theory
AI-assisted research draft

13 November 2026

Classification: Technical note with a research prospectus. The viability algorithm is standard. The maintenance thresholds are proved for explicit finite families; external novelty is unresolved. Version 3 adds erasure-prone calibration, a failure theorem and an exact feedback comparison. Internal AI review is not external peer review.

Abstract

An information-gathering action can consume the resources needed for later observation. We specify a finite system containing a degrading sensor, a degrading actuator, a bounded resource store, and a monitor requiring recurrent diagnostic observations. Sensor service requires an actuator; actuator service requires a sensor; both consume resources replenished only by harvesting. A greatest-fixed-point computation characterizes the states from which diagnostic activity can be maintained indefinitely. For equal service lifetimes and diagnostic deadline four, and equal service and observation cost k, the epistemic viability kernel is nonempty exactly when the lesser of harvest inflow and storage capacity is at least $3k$. A four-action cycle attains this threshold. In the same scalable family, immediate information-gain maximization selects a one-bit observation that destroys every safe continuation, while a maintenance schedule supports one fresh bit every four actions indefinitely. We then introduce hidden sensor orientation and an exact reference action that consumes time and resources. For service lives and deadline five, when the reliability bound permits only a one-tick-old reference, the reliable recurrent kernel is nonempty exactly when the lesser of inflow and capacity is at least $4k$. A common-interface example separates recurrent physical operation from reliable recurrent reporting. An exact age quotient avoids belief discretization in this restricted model. With recognized calibration erasures, every policy eventually loses the hard reliable-recurrence contract with probability one. A six-tick example proves survival probability $1 - \eta^2$ with feedback versus $1 - \eta$ for every fixed schedule. These are model-specific results; hidden physical health, silent calibration errors, historical novelty and empirical validation remain unresolved.

1 What must be maintained?

A functioning sensor is more than a channel in a static inference diagram. Its calibration, power, and actuator support may degrade or be exhausted. The question is whether a controller can preserve the conditions under which further informative observations remain executable. It differs from asking how much information can be extracted before a fixed apparatus stops functioning.

The existing B-Theory corpus already separates restoration of current function from preservation of an original fault label. The companion *Diagnosing While Repairing*, version 0.2.5 of 8 November

2026, treats information loss under silent resets and adaptively ordered repair [3]. *Causal Self-Diagnosis*, version 0.3, already discusses resource depletion and gives standard controlled-invariant constructions in Sections 10–11 and 14 [2]. The additional object here is therefore *recurrent* diagnostic capacity under coupled service and resource dynamics, not the fixed-point method itself.

Viability theory supplies established control tools. Doyen and De Lara formulate finite-horizon stochastic viability through dynamic programming, with an explicit state/control/disturbance contract [6, Section 3 and Appendix A]. Our deterministic finite-state greatest fixed point is a simpler safety construction, re-proved below for completeness. Montévil and Mossio distinguish open material/energy flows from dependence among constraints [11, Sections 3–5]. That conceptual distinction motivates specifying both maintenance dependence and resource inflow. It is not evidence that the following automaton constitutes biological organization.

The contribution is a small complete example with sharp resource thresholds, a proof that greedy present information can eliminate future information, and an exactly solvable coupling of hidden orientation reliability to physical maintenance. This result may support an application note or a section in a larger maintenance paper. Its mathematics does not establish historical originality in reliability scheduling or preventive maintenance.

Budgeted inspection and maintenance are also established operational-control topics. Vora and colleagues specify inspection, replacement, and deterioration actions for multiple partially observed components, with observable expenditure included in the state [15, Sections 3.1 and 4.1]. Their finite-budget, finite-horizon model couples otherwise independent components through a shared budget. The present example instead uses mutually required service actions, replenishable resources, and an infinite recurring-observation requirement. These differences define the model being checked; they do not establish a new general maintenance framework or superiority to a budgeted POMDP controller.

2 A fully declared finite system

The state at the start of a tick is

$$x = (r, s, a, b) \in \{0, \dots, R\} \times \{0, \dots, S\} \times \{0, \dots, A\} \times \{0, \dots, W\}.$$

Here r is the stored resource, s and a are sensor and actuator service lives, and b counts consecutive ticks since the latest diagnostic observation. The positive integers R, S, A, W are fixed. Resource inflow per harvesting action is $g \in \mathbb{N}$, and each service or observation costs $k \in \mathbb{N}$, $k \geq 1$.

The controller observes these counters exactly. They are bookkeeping variables with known deterministic dynamics, not hidden true component health secretly revealed to an inference algorithm. The model does not yet solve diagnosis of unknown degradation. Actuator service uses an operational sensor to align the repair; sensor service uses an operational actuator to execute it. These dependencies are stipulated parts of the action interface.

There are exactly four actions. Their successor functions are

$$H(r, s, a, b) = (\min\{R, r + g\}, s - 1, a - 1, b + 1), \tag{1}$$
$$S_r(r, s, a, b) = (r - k, S, a - 1, b + 1), \tag{2}$$
$$A_r(r, s, a, b) = (r - k, s - 1, A, b + 1), \tag{3}$$
$$P(r, s, a, b) = (r - k, s - 1, a - 1, 0). \tag{4}$$

Harvesting requires an operational actuator. Sensor service requires $a \geq 1$; actuator service requires $s \geq 1$. The last three actions require $r \geq k$. Service restores the serviced component to its stated

life without aging it again in the same tick; the other component ages by one. Every tick consumes one unit of life from every component not serviced. There is no free waiting action, no simultaneous service, and no harvesting during service. The probe requires $s \geq 1$ and reveals the current external target bit X_t without noise. It occurs before the stated end-of-tick degradation. Its resource and tick costs are charged even if the resulting state is no longer viable.

A fresh external fair bit X_t is generated at every tick before action choice, independently of the counters, private policy coins, and the entire past. It is revealed only if P is selected; otherwise it is discarded unobserved. The other actions return only their already predictable counter changes and reveal nothing about this same pre-action target. A probe therefore has immediate information gain exactly one bit; the other actions have zero. Only the most recent observed bit need be retained. The accumulating information count below means distinct bits observed over time, not all bits stored permanently. Memory integrity, code storage, and computation are held reliable and unpriced in this first model.

The physical safe set is

$$G = \{(r, s, a, b) : 0 \leq r \leq R,\ 1 \leq s \leq S,\ 1 \leq a \leq A,\ 0 \leq b < W\}. \tag{5}$$

The monitor constraint means at most $W - 1$ consecutive nonprobe actions. Thus every infinite safe trajectory has a probe in every block of W actions. Requiring current health alone would be weaker: a machine might remain powered without making another observation. Requiring $r \geq k$ on every tick would be stronger and is *not* imposed, since the machine may harvest before its next deadline.

Policies choose legal actions from the observed history and private randomness independent of the targets. All results concern the above controller information and the same action durations and costs. The resource is an abstract consumable unit; no energy, heat, work, or thermodynamic entropy is inferred from it. The inflow is external. Nothing here claims material or energetic self-sufficiency.

3 Epistemic viability as a finite safety problem

Definition 3.1. The epistemic viability kernel K is the set of states in G from which a policy keeps every subsequent state in G indefinitely. The adjective refers solely to the monitor's specified recurring observation requirement.

For a set $Z \subseteq G$, let

$$\mathrm{Pre}(Z) = \{x \in G : \text{some legal action } u \text{ has } T(x, u) \in Z\}.$$

Set $K_0 = G$ and $K_{n+1} = K_n \cap \mathrm{Pre}(K_n)$.

Proposition 3.2 (Standard finite viability construction)**.** *The sequence stabilizes after at most $|G|$ strict removals. Its final set is exactly K. At every state in K, choosing a single action whose successor lies in K defines a deterministic stationary viable policy. Randomization and additional retained history do not enlarge K.*

Proof. The sets decrease. If the next set differs, at least one state is removed; finiteness gives stabilization. At a fixed point, every state has a legal successor in the set. Choosing one such action for each state and inducting on time keeps the state there forever.

Conversely, let V be the set of states admitting an infinite safe policy. Clearly $V \subseteq K_0$. If $V \subseteq K_n$, any viable policy's first action at $x \in V$ must lead to a state in V and hence in K_n. For a randomized policy with probability-one safety, at least one positive-probability first action has a

viable continuation; in this deterministic finite transition system, any selected nonviable successor has zero probability in a viable policy. Thus $x \in K_{n+1}$. Induction gives $V \subseteq K$, while the stationary construction proves the reverse inclusion. □

The proof also distinguishes safety from reachability. A path that reaches a healthy state once need not permit indefinite maintenance. A cycle in a dependency diagram likewise does not prove that the resource balance and deadlines permit an executable cycle. If component state were partly hidden, the controller would need an information-state formulation and a common feasible action for all relevant possibilities; the pointwise predecessor above would generally be inadequate.

4 A sharp resource threshold

Specialize to equal maximum service lives and diagnostic deadline

$$S = A = W = 4. \tag{6}$$

The sensor, actuator, and observation obligations each need at least one of every four actions in the long run. The remaining action must replenish the resources used by those three obligations.

Theorem 4.1 (Exact nonemptiness threshold for the four-tick family)**.** *For the integer model* (1)–(6)*, with common service and observation cost $k \geq 1$, the epistemic viability kernel is nonempty if and only if*

$$\min\{g, R\} \geq 3k. \tag{7}$$

When this holds, the action cycle H, S_r, A_r, P is viable from $(0, 2, 3, 0)$*.*

Proof. Consider any safe trajectory of T actions. Let N_S, N_A, N_P, N_H count its four action types and let $N = N_S + N_A + N_P$. Since the sensor cannot survive four consecutive actions without service, $N_S \geq T/4 - 1$. The same holds for N_A. The monitor gives $N_P \geq T/4 - 1$. Hence

$$N \geq 3T/4 - 3. \tag{8}$$

Each harvest increases the store by at most $h := \min\{g, R\}$, and each nonharvest action decreases it by k. Because $r_0 \leq R$ and $r_T \geq 0$,

$$kN \leq R + hN_H = R + h(T - N).$$

Combining with (8) yields

$$\frac{3k - h}{4} T \leq R + 3(k + h).$$

If $h < 3k$, the left side is unbounded as T increases, excluding every infinite safe trajectory, independently of policy or randomness.

For sufficiency, start at $(0, 2, 3, 0)$. Harvest gives resource at least $3k$ and component lives $(1, 2)$, with monitor one. Service the sensor, then the actuator, then probe. The lives and monitor follow

$$(2, 3, 0) \xrightarrow{H} (1, 2, 1) \xrightarrow{S_r} (4, 1, 2) \xrightarrow{A_r} (3, 4, 3) \xrightarrow{P} (2, 3, 0).$$

The three charged actions are affordable. At the next cycle start resources are nonnegative; the next harvest again brings them to at least $3k$ because both R and g are at least $3k$. Induction proves indefinite safety and one probe every four actions. □

The threshold concerns the existence of *some* viable initial state. It is not a claim that every physically safe state is viable. For example, service-life and resource deadlines can conflict before the machine reaches the cycle. Nor does the theorem extend unchanged to simultaneous service, free waiting, different component lives, imperfect repairs, or a weaker observation schedule. These are alternative models, not small implementation details.

The count proof supplies a useful design check: storage capacity can bind just as strongly as inflow. A large external resource flow does not help if the store cannot retain the charge needed between necessary harvesting opportunities. This is an abstract scheduling conclusion, not a thermodynamic law.

5 Why immediate information gain can be destructive

Theorem 5.1 (Scalable conflict between present information and future capacity)**.** *For every integer $k \geq 1$, take $R = g = 3k$, $S = A = W = 4$, and initial state*

$$x_0 = (k, 2, 3, 0). \tag{9}$$

The unique action maximizing immediate information about the current fresh target is P, with one bit gained. Its successor is physically safe but has no safe continuation. In contrast, the cycle H, S_r, A_r, P is viable indefinitely and obtains one fresh target bit per four actions.

Proof. At x_0 a probe is affordable and the sensor is operational, so it reveals one fair bit exactly. Every other action has a deterministic readout independent of that bit and gives zero information. The probe yields

$$x_1 = (0, 1, 2, 0).$$

The service and probe actions require $k > 0$ and are unavailable at resource zero. Harvesting is the only legal action, but it yields sensor life zero and leaves G. Therefore no safe continuation exists.

For the alternative, the first harvest fills the store to $3k$ and starts the life/monitor cycle in the proof of Theorem 4.1. All later cycles are affordable. Each probe reveals a new independent target bit, so after $4n$ actions the schedule has obtained n bits through n distinct probes while remaining in G. □

The comparison uses identical initial state, physical transitions, prices, information, and time. It does not compare a one-step reward with a longer-horizon optimum and then declare a new control principle: the point is precisely that the one-step objective omits the persistence requirement. A controller maximizing total future diagnostic reward, or enforcing the viability kernel as a constraint, can anticipate the loss. The theorem refutes the proposed universal implication "more information now preserves the ability to know later." It does not show that information acquisition is generally harmful.

A receding-horizon controller can also exploit terminal resources in ways that violate a requirement extending beyond its planning horizon. A safe terminal set contained in K prevents that particular truncation problem. Proving useful short-horizon approximations for larger partially observed systems remains an open control problem for this programme.

6 Exact computations and fair comparators

The script `code/reproduce.py` computes the full finite kernel by state elimination. It tests the theorem for $k = 1, 2$ and every pair $(R, g) \in \{1, \ldots, 8\}^2$, totaling 128 parameter settings. It saves

kernel sizes, elimination counts, a selected viable action at every surviving state in the reference case, and complete policy traces. No Monte Carlo sampling is used and no seed is needed.

Three comparators share the same initial state $(1, 2, 3, 0)$, $R = g = 3$, and all four actions: the periodic maintenance cycle; a stationary action selected from the computed kernel; and an immediate-information policy that probes whenever affordable and otherwise harvests. The last exits the safe set on its second transition. The periodic policy executes twenty safe actions and completes five probes. Its indefinite continuation follows from the proof, not from observing a twenty-tick trace.

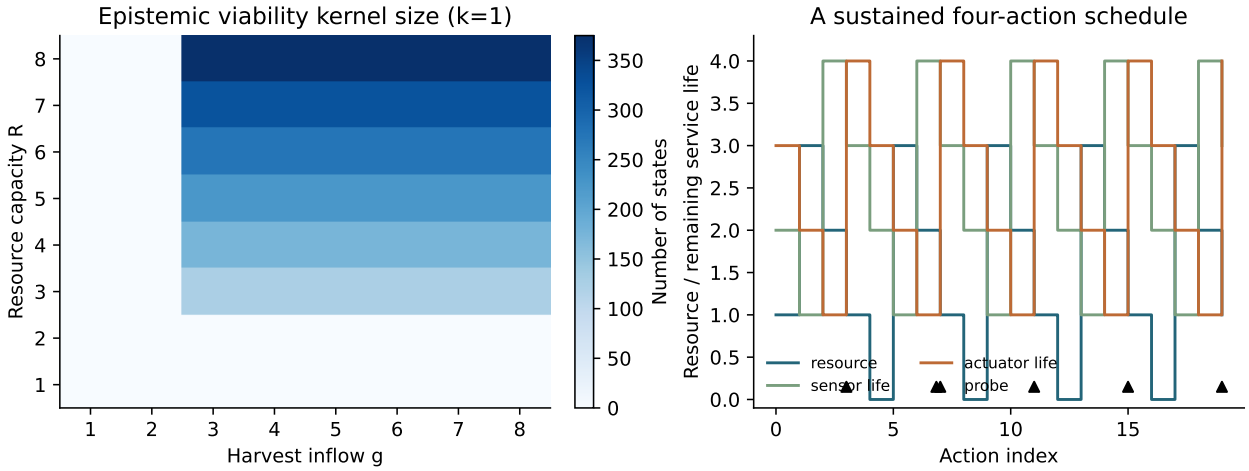


Figure 1: Exact finite-model calculations. Left: kernel sizes for all capacities and harvest inflows one through eight, with $k = 1$ and $S = A = W = 4$. Nonemptiness begins at the analytically proved threshold. Right: the periodic schedule's resource and service-life counters; markers indicate probes. The units are abstract resources and ticks, not joules or seconds. SVG and generating Python are supplied.

The periodic comparator is intentionally strong here: it attains the threshold because the dynamics are known and deterministic. The kernel feedback policy need not improve its long-run probe rate. An adaptive superiority claim would require uncertainty, perturbations, or variable task urgency and a new comparison. Purely reactive repair fails in some states because the resource necessary to repair may already be gone; this observation does not establish that every reactive policy is inferior.

For uncertainty, two separate extensions are available. With adversarial successor sets $F(x, u)$, replace the successor requirement by $\varnothing \neq F(x, u) \subseteq Z$, which is a robust safety game. With stochastic degradation and a finite horizon, use a probability-valued dynamic program, as in the standard viability literature. A positive one-step survival probability does not establish almost-sure infinite survival when unavoidable independent failure continues indefinitely.

7 Coupling maintenance to partially observed reliability

We now solve one restricted part of the D03–D05 integration proposal. Component lives remain observed deterministic counters. The new hidden variable is sensor *orientation*, not physical health. The independent calibration reference remains exact. This distinction is essential: the following result is not a solution of general partially observed maintenance or unreliable-reference control.

7.1 A costed, timed reference contract

Retain the four physical actions and their costs, but replace the probe's perfect readout by

$$Y_t = X_t \oplus C_t, \qquad C_{t+1} = C_t \oplus F_{t+1},$$

where the fresh fair targets X_t remain independent over ticks, C_0 is fair, and the flips F_t are independent Bernoulli(q) with known $0 < q < 1/2$. Targets, orientation increments and policy coins are mutually independent. No subsequent target labels are disclosed. The orientation flip occurs *after every action*, including service and calibration. Service does not change or reveal orientation.

Add a fifth action Q, which uses an exact external reference to reveal C_t before that tick's flip. It requires $r \geq k$, operational sensor and actuator, costs k, consumes one entire tick and has physical successor

$$Q(r, s, a, b) = (r - k, s - 1, a - 1, b + 1).$$

It neither probes X_t nor repairs a component. All actions remain exclusive. The controller retains the last observed orientation c and its age $d = t - \tau$, where τ was the last Q tick. This bit and the age counter are reliable and unpriced; no claim of memory self-repair is made. Immediately after Q the next decision has age one, not zero. Before the first calibration the orientation prior is fair. An initial record, if supplied, must have the stated exact-reference provenance and age; arbitrary priors or free repeated references are excluded.

Every probe must produce a binary report. Abstaining or discarding a probe does not reset the diagnostic monitor. A feasible policy must keep the physical counters in G and ensure that at every probe the conditional error, given the available history and the current reading, is at most $\varepsilon < 1/2$. This is a per-report posterior-risk requirement, not a guarantee that no report is ever wrong. Calibration outcomes and previous readings may affect action choices; unobserved targets and future flips may not.

Proposition 7.1 (Exact reliability reduction)**.** *After a latest exact calibration with value c and age $d \geq 1$, the optimal report is $\widehat{X}_t = Y_t \oplus c$. For each possible current reading, its conditional error is*

$$\rho_d = \frac{1 - (1 - 2q)^d}{2}. \tag{10}$$

Before any calibration that error is $1/2$. In particular, if

$$q \leq \varepsilon < 2q(1 - q), \tag{11}$$

a probe is feasible exactly at age one: except for a possible initial fresh record, its immediately preceding action must be Q.

Proof. For any orientation value the current masked target Y_t is fair, so its likelihood is the same for both orientations. Successive fresh targets make the routine readings uninformative about the entire orientation trajectory. Physical counter readouts are deterministic and also uninformative about orientation. After an exact reference, future flips are independent of that reference, older records, and policy coins. Up to the next reference, adaptive decisions based on these records and masked targets do not reveal or select these unobserved flips. Thus, conditional on each admissible past before a new reference, the parity since the latest reference has probability ρ_d of being one; this follows from $\mathbb{E}(-1)^{F_1 + \cdots + F_d} = (1 - 2q)^d$. The current fair masked reading leaves that probability unchanged. Since $q < 1/2$, the more likely orientation is c, proving the report and risk. A fair unanchored orientation stays fair. Finally $\rho_1 = q$, $\rho_2 = 2q(1 - q)$, and ρ_d is strictly increasing, so (11) permits only age one. □

The binary aging identity is established prior art, explicitly present in Sun–Cyr [14, Section III-B, equations (11)–(12)], and is the noiseless specialization of D03. The increment here is coupling it to a reference action charged in both the resource and service clocks. D03's instantaneous reference convention is *not* imported. Unlike D03's two noisy readings, our single noiseless masked reading has the same posterior risk for either readout; this is why a per-report guarantee is justified here.

7.2 A sharp coupled threshold and a strict viability gap

Write K_{rel} for the coupled reliable recurrent kernel. Let K_{op} impose the same actions, hidden orientation, resource dynamics, service counters and recurrent probe deadline, but omit the report-error restriction. The latter maintains recurring *opportunities* for observation, not correct diagnoses. It is a relaxed feasibility comparator, not an equal-objective performance competitor.

Theorem 7.2 (Five-tick reliable recurrence)**.** *Assume* (11)*, common paid-action cost* $k \geq 1$*, and* $S = A = W = 5$*. Then*

$$K_{\mathrm{rel}} \neq \varnothing \quad \Longleftrightarrow \quad \min\{g, R\} \geq 4k.$$

When this holds, H, S_r, A_r, Q, P *is an indefinitely viable cycle from* $(r, s, a, b) = (0, 2, 3, 0)$ *with an uncalibrated or stale reference. At* $R = g = 3k$*,* K_{op} *is nonempty but* K_{rel} *is empty.*

Proof. In a safe prefix of T actions each component service and the probe occurs at least $T/5 - 1$ times. Every probe except a possible first one has a distinct immediately preceding calibration, so $N_Q \geq N_P - 1$. Hence the total paid count satisfies

$$N = N_S + N_A + N_Q + N_P \geq 4T/5 - 5.$$

With $h = \min\{g, R\}$, resource balance gives $kN \leq R + h(T - N)$ and therefore

$$\frac{4k - h}{5} T \leq R + 5(k + h).$$

If $h < 4k$ no indefinitely safe reliable path exists. This excludes randomized and history-dependent policies as well.

For sufficiency the service and monitor counters follow

$$(2, 3, 0) \xrightarrow{H} (1, 2, 1) \xrightarrow{S_r} (5, 1, 2) \xrightarrow{A_r} (4, 5, 3)$$
$$\xrightarrow{Q} (3, 4, 4) \xrightarrow{P} (2, 3, 0).$$

Harvest supplies at least $4k$; the next four actions consume exactly $4k$. The probe immediately follows Q, has age one and risk $q \leq \varepsilon$. The cycle repeats with nonnegative resources and the same physical counters, independently of orientation outcomes. It can start without an initial reference because its first probe follows a paid calibration.

For the strict gap at $R = g = 3k$, the relaxed schedule H, S_r, A_r, P is safe from $(0, 2, 3, 0)$ with lives five. Its first cycle ends with lives $(3, 4)$ and later cycles repeat those lives. Each harvest pays for three charged actions and the four-tick probe spacing respects $W = 5$. The reliable kernel is empty by the necessary bound. Both objectives therefore concern the same physical and observational interface; only the reliability constraint differs. □

This proves the concrete separation requested by the portfolio dependency proposal: recurrent physical operation, even with recurrent readouts, need not sustain a declared reliability level. It is an elementary special-case scheduling theorem. It does not establish historical novelty of maintenance-of-evidence or an advantage over an optimal POMDP controller.

7.3 Exact finite abstraction and computational evidence

Under (11), use the age classes $z = 1$ and $z = 2$, with 2 meaning stale (any age at least two) or not yet calibrated. Q maps every class to 1 and every other action maps it to 2. Permit P only from 1. The last calibration bit is still required for the actual report, but neither its value nor the exact stale age affects action admissibility or physical transitions. Therefore applying Proposition 3.2 to (r, s, a, b, z) computes K_{rel} exactly for this contract. It is a quotient preserving feasible actions and viability, *not* a posterior-preserving approximation. The two possible calibration values have identical viable actions. They can be omitted from the control graph while still retained by the reporting rule.

To justify exactness against arbitrary history-dependent policies, every viable history at a reduced state must choose a feasible action and have a viable continuation for each positive-probability reference/readout outcome. All such outcomes have the same reduced successor. Induction through the finite predecessor iteration excludes every removed state, regardless of observation-dependent choices. Conversely a stationary selector from the final reduced kernel, combined with $\widehat{X} = Y \oplus c$, gives an executable reliable policy. No numerical belief discretization or approximation error is involved.

The standard-library script `code/reliability_extension.py` checks 192 exact rational posterior calculations and all 162 settings $k \in \{1, 2\}$, $(R, g) \in \{1, \ldots, 9\}^2$ for Theorem 7.2. At $k = 1, R = g = 4$ the reduced safe graph has 1,250 states and the kernel has 557; elimination sizes are $1250, 960, 783, 658, 557$. At $R = g = 3$ the reliable kernel is empty, while the relaxed graph has 648 viable states (including redundant age classes). Full selectors and witness traces are saved. Two independent internal AI reviewers implemented strongly connected components plus reverse reachability, without importing the author algorithm; both obtained the same threshold checks and reference kernel count. These computations corroborate the proofs and are not external peer review or empirical validation.

7.4 Boundaries and a counterexample to overgeneralization

For general $q \le \varepsilon < 1/2$, define $D = \max\{d \ge 1 : \rho_d \le \varepsilon\}$. A finite safety quotient retains ages $1, \ldots, D$ and one overflow class. The proof of exactness above applies because stale ages forbid probes until the next exact reference. It does not make the five-tick $4k$ threshold valid outside (11). If $\varepsilon < q$, no steady recurrent report is feasible with this separate-action timing. At the upper endpoint $\varepsilon = 2q(1 - q)$, age two is allowed and the immediate-predecessor counting proof no longer applies.

It is false that a four-tick physical deadline rules out every finite freshness requirement regardless of resources. With $S = A = W = 4$, maximum admissible age $D = 5$, and $R = g = 7k$, the word

$$P, S_r, A_r, H, P, S_r, A_r, Q$$

is viable and repeats from $(r, s, a, b, d) = (3k, 2, 3, 3, 1)$. The two probes use ages one and five. Seven paid actions occur between harvests. The reproduction code checks the complete state trace; this is a counterexample to an overbroad impossibility statement, not a claim of optimality for $D = 5$. A 112-case sensitivity sweep is retained separately as finite exploratory evidence.

All ages $d \ge 1$ have orientation support $\{0, 1\}$, while their probabilities and risks differ. Thus a support-only safety construction cannot replace the quantitative reliability calculation. Unknown physical health, imperfect references, persistent targets and failures that reveal health would require richer beliefs. Exact calibration is what discards older orientation information here; with imperfect anchors a latest-reading-only rule need not be optimal.

8 Imperfect calibration as recognized erasure

The perfect-reference theorem is not robust to every form of imperfection. We now change Q alone: with probability $1-\eta$ it returns the exact pre-action C_t; with probability η it returns a recognized erasure $\perp$. Both outcomes cost k, consume a whole tick and age both components. The outcome indicator is trusted and independent of orientation, target, past and other action outcomes. A failure therefore supplies no orientation information. There is no free fallback reference, no within-tick retry and no false successful value. This is an *availability failure* model, not unobserved readout corruption or arbitrary hidden physical health.

Let $0 < q < 1/2$ and $q \le \varepsilon < 1/2$, and let $D = \max\{d \ge 1 : \rho_d \le \varepsilon\}$ be finite. A successful Q sets next-tick age to one; failure increments the previous age. Ages above D, including the uncalibrated case, can be grouped into $D+1$. Exactness follows as before, now using the independent erasure outcome: the controller knows whether the latest anchor was renewed. The anchor value remains necessary for reports and its storage stays reliable. Probes are permitted only at ages at most D.

Theorem 8.1 (Eventual failure under unbounded erasure runs)**.** *Suppose $0 < \eta \le 1$, the recurring-probe deadline W is finite, and the initial reference age is at least one or uncalibrated. For every admissible observation-dependent policy, the probability of maintaining physical counters, recurrence and admissible report risk forever is zero. More precisely, if τ is the first failed constraint and $B = D + W$, then*

$$\Pr(\tau > T) \le (1-\eta^B)^{\lfloor T/B \rfloor}.$$

The exponent-zero value is one. This bound is conservative and holds even with ample resources and service capability.

Proof. Couple the process to independent potential erasure coins at every tick, using a coin only when Q is selected. For any block of B ticks, the event that all these coins erase has probability η^B, conditionally on the past, under every adaptive policy. On this event there is no successful calibration in that block. After its first D actions every pre-existing reference is stale. The next W actions therefore cannot contain a reliable probe; they must violate the monitor deadline if no earlier constraint has already failed. Thus, conditional on reaching each new block safely, the probability of surviving it is at most $1-\eta^B$. Iteration gives the bound and its limit zero. The argument uses finite blocks; it never assigns positive probability to an infinite all-erasure sequence. □

Consequently the infinite-horizon feasibility region can collapse discontinuously at $\eta = 0$. Finite-horizon probabilities do not exhibit that discontinuity: for a finite state space and horizon they are maxima of finitely many policy-tree polynomials in η, hence continuous. The deterministic $4k$ theorem remains correct at $\eta = 0$; it is not a robustness guarantee at positive erasure rate. Similarly, this negative result does not refute average-age or average-error guarantees, which impose different quantifiers.

8.1 An exact finite-horizon safety objective

For $x = (r, s, a, b, d)$ with truncated age, define $V_n(x)$ as the greatest probability of maintaining safe counters and only admissible probes during the next n actions. A state or action violating any guard leads to an absorbing failure state with value zero. Stopping early also counts as failure; no successful termination before n is supplied. Put $V_0(x) = 1$ on safe states. Let $T_u x$ denote the successor for a nonreference action, $T_Q^+ x$ the successor on successful calibration and $T_Q^- x$ that on

erasure. Then

$$V_{n+1}(x) = \max\left\{\max_{u\in\{H,S_r,A_r,P\}} V_n(T_u x), (1-\eta)V_n(T_Q^+ x) + \eta V_n(T_Q^- x)\right\}, \tag{12}$$

where invalid actions have value zero. Backward induction proves optimality: each first action has the displayed conditional expectation, and the observed successor permits a separately optimal continuation. Randomization cannot improve a maximum. This is standard stochastic-safety dynamic programming, not a new algorithm [6, 5]. Its finite age state is exact for this erasure interface, without belief-grid approximation.

The event being optimized is *survival of the operational contract*, not the event that every target estimate is actually correct. A permitted probe can still err with probability up to ε. The terminal convention $V_0 = 1$ also permits depletion near the horizon; these values do not certify indefinite continuation. Horizon and terminal convention are identical for every comparator below.

8.2 A sharp value of feedback and coupled resource slack

Theorem 8.2 (Six-tick retry advantage)**.** *Fix $k = 1$, $D = 1$, $S = A = W = L$, $R = g = h$, initial state $x_0 = (0, 2, 3, 0, 2)$, and $0 \le \eta \le 1$. The optimal six-action survival probabilities are*

(L, h)	$(5,4)$	$(5,5)$	$(6,4)$	$(6,5)$
$V_6(x_0)$	$1-\eta$	$1-\eta$	$1-\eta$	$1-\eta^2$.

In the last model, the best survival probability over all *schedules selected in advance, including randomized ones, is*

$$V_T^{\mathrm{open}}(x_0) = (1-\eta)^{\lfloor T/6\rfloor}$$

for every integer horizon $T \ge 0$, with the exponent-zero value interpreted as one. In particular, at $\eta = 1/5$ feedback gives $V_6 = 24/25$, whereas $V_6^{\mathrm{open}} = 4/5$.

Proof. The first three safe actions are forced to be H, S_r, A_r: resources initially vanish, the first harvest leaves sensor life one, and sensor service then leaves actuator life one. In the $(6, 5)$ model only three actions remain, so at most two independent calibration attempts can occur before the required first probe. A successful six-action path needs at least one of them to succeed, giving upper bound $1 - \eta^2$.

Attain it by attempting Q fourth. On success, probe fifth and use a safe Q sixth; on erasure, retry Q fifth and probe sixth if successful. A second erasure leaves no permitted final probe and is counted as failure. Five paid actions at most use the five available units. The physical lives and monitor remain safe on both successful branches. This proves the last table entry without relying on numerical optimization.

For $L = 5$ a first probe is required within five actions, leaving only one preceding Q after the forced prefix. For $L = 6, h = 4$, two services, two calibration attempts and a probe would cost five units. Affording them requires a second harvest, and together these require seven actions. Thus the other three models have upper bound $1 - \eta$, attained by H, S_r, A_r, Q, P, H. More resource alone or more service/deadline slack alone is insufficient in this example.

For an action sequence fixed before observing outcomes, a safe T-action path requires at least $\lfloor T/6\rfloor$ prescribed probes. Each must be immediately preceded by a distinct successful Q because $D = 1$. Their independent successes give upper bound $(1-\eta)^{\lfloor T/6\rfloor}$. Early safety stopping when a prescribed probe is unreliable is counted as failure, not as an unreliable report. Mixing fixed schedules cannot improve the bound. Repeating H, S_r, A_r, Q, Q, P attains it: resources and service lives repeat after six actions, the probe lies at each deadline, and only the final Q before each probe needs to succeed. This proves the all-horizon open-loop value. □

For instance, $q = 1/10$ and $\varepsilon = 3/20$ give $D = 1$. Both classes receive the same success indicator and pay the same prices. The open-loop comparator deliberately cannot adapt its selected action sequence to that indicator; it can only stop safely on a failing branch. The theorem quantifies the benefit of feedback for this declared comparison, not superiority over other feedback controllers or real maintenance systems.

Exact rational dynamic programming also gives the following *finite* values in the $(6, 5)$ example with $\eta = 1/5$:

T	6	12	30
V_T	24/25	576/625	7962624/9765625
V_T^{open}	4/5	16/25	1024/3125.

The values for $T = 12, 30$ are computational optima of the specified finite recursion, not an asserted closed form for all horizons. The all-policy erasure theorem still forces every policy's infinite survival probability to zero.

8.3 Validation, prior art and D03 coordination

The new standard-library program checks 150 exact parameter/horizon cases, including $\eta = 0, 1$, monotonicity in horizon and erasure rate, the universal block bound and the analytic open-loop values. A specialist independently computes finite values without importing author routines. A hostile reviewer independently propagates probability mass through all 15,625 fixed six-action words at five erasure rates, and through the explicit adaptive policy, using untruncated age. This verifies the comparator over all fixed words rather than over a weak hand-picked periodic policy. These are synthetic finite-model checks, not measurements from a sensor or maintenance facility.

Intermittent observations, paid retries and energy-harvesting updates are established subjects. Feng–Yang explicitly study update erasures and perfect feedback with retransmission [7, Sections II and IV]; their infinite battery, instantaneous updates and average-age objective differ from our finite store, full action ticks, service clocks and hard report deadline. Sinopoli et al. study intermittent measurements through expected estimation-error covariance [13, Introduction and model]; that expectation criterion is not our pathwise recurrence event. We do not claim the erasure-run argument, Bellman recursion or feedback-retry idea is new. The contribution is the supplied coupled example, exact boundary, and reproducible comparison. Historical originality of even this special case remains unverified.

The accessible *current* D03 revision now also relaxes perfect restoration, but differently: it resets reference polarity with an *unobserved* binary error, conditions on the full report record, and proves a conservative positive infinite-recurrence regime plus a conflicting-readout counterexample. D05's failure is recognized and uninformative, while a successful reading is exact. Thus D03's positive result and our almost-sure eventual failure are compatible. They concern different channels and risk conditions. A coordinated study should separate exact calibration, recognized erasure and silent corruption in an interface table, share the physical preliminaries, and present the broad reliability-versus-operation point once. Neither continuation solves hidden physical degradation or an untrusted failure indicator; neither's threshold transfers to the other model merely by substituting an error parameter.

9 Prior art, portfolio ownership and unresolved work

The resource-clock construction is closely related to pinwheel scheduling: service and probe obligations must recur within bounded windows [9, Section 1]. The later journal version [10] has a

correction restoring missing supporting material and updating the software and data availability statement [12]. We use the recurring-window model as background, not Kawamura's density theorem. Utilization counts provide necessity; sufficiency here comes from explicit cyclic words, not a general implication from utilization to schedulability. Partial observability, resource replenishment, and safety shields already coexist in the control literature. Brázdil et al. [4] study energy-constrained POMDPs, and Ajdarów et al. [1] study resource-constrained goal POMDP shielding. Their objectives and resource semantics differ from our infinite recurrent posterior-risk contract; those differences delimit this example and do not certify novelty. The November 2026 real-period pinwheel preprint [8] is also recorded as current context, without relying on its computer-assisted theorem. A precise source audit and primary-literature search record accompany this version.

The first D03 continuation was also inspected directly, not inferred from its original prospectus. D03 now studies a *second hidden polarity* H belonging to the reference itself, with free observations $Z = C \oplus H$. Equal sensor/reference flip rates make a changed relative reading erase the orientation posterior contrast. Its paid trusted restoration is bundled into a probe, has no extra tick, and yields an exact report; the four-tick model has reliability threshold $3k + \delta$, where δ is its monetary/resource surcharge (called d in D03). Its necessity proof uses finite adverse observation runs of positive probability. We preserve a hash-identified snapshot of that continuation for this comparison.

In our earlier exact-reference model, the reference is exact whenever purchased, Q takes its own tick and allows a drift before a later probe, and the five-tick theorem follows from an age guard forcing adjacent Q, P. Consequently $4k$ is not a special case of D03's $3k + \delta$ merely because $\delta = k$ makes their numeric values coincide. The number and duration of actions, hidden variables, information access, feasible schedules, report risks and proof quantifiers differ. D03 resolves an informative hidden-reference filter and adversarial-history resource surcharge; D05 resolves the interaction of calibration *duration* with service deadlines and a quantitative posterior-risk guard. Neither establishes general hidden physical health or imperfect trusted restoration.

The high-level strict gap between recurrent physical readouts and reliable diagnosis is shared and should be presented once. The preferred editorial direction is one coordinated D03–D05 study with two explicitly separate model variants, shared physical preliminaries and aging attribution, and two model-specific theorems. This is a recommendation, not a completed merge or a claim of two independently novel discoveries. No unified theorem follows by juxtaposition. A merger should first reconcile paid versus free reference sampling, restoration timing, hidden-state transitions and the reliability criterion. Original and current D03 files were not edited.

B07 already contains orientation ambiguity, common-action reasoning and controlled invariance; B11 already studies evidence destroyed by repair. A physical service action in our coupled model does not erase a retained origin label or change the orientation process. Consequently no B11 theorem is rederived here. D06 architectural replacement and D10 microscopic work/heat remain distinct contracts; abstract k units are not joules.

The general hidden-health problem remains open. Beyond the recognized-erasure model, a useful next model would specify stochastic component failure, silently corrupted paid inspections, repair success and what failures reveal. It must distinguish almost-sure physical safety, finite-horizon survival probability and posterior report risk. Almost-sure indefinite survival may be impossible under unavoidable recurrent failure. Finite belief approximations require a certified conservative error bound; support information alone does not certify a numerical report-risk limit. No general abstraction theorem, general adaptive-superiority claim or device validation is supplied.

The original and revised versions, earlier internal review, new specialist and hostile AI reviews, author responses, independently written checking programs, exact data, counterexamples and source ledgers are preserved. The old closure applies only to the original hashes; new closure must identify this revision. External human review and a broader result-level scheduling/maintenance priority

audit remain required before a submission claim. The present result is a defensible extension of a technical note, not a claim of a new general theory.

10 Interpretation and next scientific step

Operationally, "Being" concerns a presently serviceable apparatus; "Becoming" includes service, resource inflow and reference renewal. The "MIND" connection is an actionable record that helps preserve future informative and reliable reporting. The distinction between continuing to obtain a reading and maintaining justified confidence now has an explicit finite-model witness. These interpretations add no theorem about subjective experience, selfhood or a natural system boundary.

The next step is to distinguish silently corrupted reference readings from recognized erasures, or to model genuinely hidden service condition, while retaining an exact comparison. For imperfect references, retain the full two-state posterior and compare a finite-horizon optimal belief controller with periodic calibration under identical costs. For hidden health, first decide whether safety is robust or probabilistic. A submission should follow only after that contract, a reproducible baseline comparison and the priority assessment are complete.

References

[1] Michal Ajdarów, Šimon Brlej, and Petr Novotný. Shielding in resource-constrained goal pomdps. In *Proceedings of the AAAI Conference on Artificial Intelligence*, 2023. URL `https://ojs.aaai.org/index.php/AAAI/article/view/26715`. Technical comparison checked against arXiv:2211.15349v1, 28 November 2022; introduction and model framing.

[2] B-Theory research programme. Causal self-diagnosis with noisy tests: Observation laws, irreversible access, and task recovery, 2026. URL `https://www.b-theory.com/read-online/causal-self-diagnosis/`. Version 0.3, 5 November 2026; research development draft prepared for Brian Theory.

[3] B-Theory research programme. Diagnosing while repairing: Information loss, adaptive order, and noisy readouts, 2026. URL `https://www.b-theory.com/manuscripts-html/diagnosing-while-repairing/`. Companion version 0.2.5, 8 November 2026. Byline: research development draft prepared for Brian Theory; AI-assisted derivation, exact computation, and simulated review.

[4] Tomáš Brázdil, Krishnendu Chatterjee, Martin Chmelík, Anchit Gupta, and Petr Novotný. Stochastic shortest path with energy constraints in pomdps. In *Proceedings of AAMAS*, 2016. URL `https://arxiv.org/abs/1602.07565`. Primary preprint; model-level comparison only.

[5] Jerry Ding, Alessandro Abate, and Claire Tomlin. Optimal control of partially observable discrete time stochastic hybrid systems for safety specifications. In *American Control Conference*, 2013. URL `https://www.cs.ox.ac.uk/people/alessandro.abate/publications/cDAT13.pdf`. Author-hosted primary manuscript; probabilistic safety formulation and dynamic-programming framework inspected.

[6] Luc Doyen and Michel De Lara. Stochastic viability and dynamic programming. *Systems & Control Letters*, 59(10):629–634, 2010. doi: 10.1016/j.sysconle.2010.07.008. URL `https://arxiv.org/abs/1002.1140`. Technical claims checked against arXiv:1002.1140v1; the retrieved PDF body displays 11 November 2018 despite its 2010 arXiv header.

[7] Songtao Feng and Jing Yang. Age of information minimization for an energy harvesting source with updating erasures: Without and with feedback, 2019. URL https://arxiv.org/html/1808.05141v2. Version 2, 10 May 2019; Sections II and IV inspected; average-age theorem not imported.

[8] Hiroshi Fujiwara, Grzegorz Gutowski, Katsuhisa Ouchi, and Alexander Wolff. A simple computer-assisted proof of the 5/6-bound for real-period pinwheel scheduling, 2026. URL https://arxiv.org/html/2611.10624v1. Preprint, 7 November 2026; introductory definitions and obstruction inspected; main proof not independently verified.

[9] Akitoshi Kawamura. Proof of the density threshold conjecture for pinwheel scheduling. In *Proceedings of the 56th Annual ACM Symposium on Theory of Computing*, pages 1816–1819, 2024. doi: 10.1145/3618260.3649757. URL https://www.kurims.kyoto-u.ac.jp/~kawamura/pinwheel/paper_e.pdf.

[10] Akitoshi Kawamura. Proof of the density threshold conjecture for pinwheel scheduling. *Proceedings of the National Academy of Sciences*, 123(32):e2530214123, 2026. doi: 10.1073/pnas.2530214123. URL https://arxiv.org/html/2606.27104v2. Author accepted version, 10 August 2026; introduction and scheduling definition inspected; computer-assisted proof not reproduced.

[11] Maël Montévil and Matteo Mossio. Biological organisation as closure of constraints. *Journal of Theoretical Biology*, 372:179–191, 2015. doi: 10.1016/j.jtbi.2015.02.029. URL https://montevil.org/publications/articles/2015-mm-organisation-closure-constraints/.

[12] PNAS correction. Correction for Kawamura, proof of the density threshold conjecture for pinwheel scheduling. Proceedings of the National Academy of Sciences, 123(44):e2634138123, 2026. URL https://pmc.ncbi.nlm.nih.gov/articles/PMC13732071/. Published online 26 October 2026; issue dated 3 November 2026.

[13] Bruno Sinopoli, Luca Schenato, Massimo Franceschetti, Kameshwar Poolla, Michael I. Jordan, and Shankar S. Sastry. Kalman filtering with intermittent observations. *IEEE Transactions on Automatic Control*, 49(9):1453–1464, 2004. URL https://users.ece.cmu.edu/~brunos/Publications/tacs04.pdf. Author-hosted manuscript; introduction and model inspected; covariance results not reproduced.

[14] Yin Sun and Benjamin Cyr. Sampling for data freshness optimization: Non-linear age functions. *Journal of Communications and Networks*, 21(3):204–219, 2019. URL https://arxiv.org/html/1812.07241v4. arXiv version 4, 3 May 2019; Section III-B, equations (11)–(12).

[15] Manav Vora, Pranay Thangeda, Michael N. Grussing, and Melkior Ornik. Welfare maximization algorithm for solving budget-constrained multi-component POMDPs. arXiv:2303.10302, 2023. URL https://arxiv.org/abs/2303.10302. Version 2, 14 May 2023. Model comparison checked in Sections 3.1 and 4.1; claims of general concavity or algorithm optimality are not used here.