A Hardware-Native Bit-Serial Learner with Exact Statistical Structure

Jerry Felix
Brain-CA Technologies, Sarasota, FL, USA
jfelix@brain-ca.com

***Abstract*—Massively parallel inference fabrics need learning primitives that are tiny, run-time-parameter-free, and composable. This paper studies the Brain-CA Estimator, a binary-stream learner whose update rule—the Cincinnati Algorithm—uses only bit comparisons, inversions, and random bits—48 LUTs and 8 registers on an UltraScale+ FPGA, with no multiplier, no adder, no run-time tunable parameters, and $O(b)$ logic for b-bit precision (width b is a synthesis-time choice). This bit-manipulation discipline—compute confined to local bit operations, executed where the state is stored—is an architectural distinction whose significance grows with tile count beyond per-unit FPGA synthesis. The paper's primary systems result: under a two-stream conditional decomposition, three Brain-CA Estimator units beat three EWMAs sharing a single sweep-selected α in every one of four primary scenarios, without per-regime tuning, and remain competitive against oracle per-branch-tuned EWMA on a broader five-scenario transfer test.**

These results rest on exact analytical structure. Once bit width is capped, the learner is a finite birth-death chain with exact $\mathrm{Binomial}(N-1,\theta)$ stationary distribution, yielding closed-form mean, variance, and drift. Its data-averaged expectation is linear shrinkage between Jeffreys and Bayes–Laplace at every observation count. These properties explain why the Brain-CA Estimator avoids the rounding-induced bias of small-α fixed-point EWMA at extreme probabilities, and why error-proportional step sizes track competitively—within $1.5\times$ on switching, $2.1\times$ on continuous drift—without per-regime tuning.

I. Introduction

Edge inference at scale demands learning primitives that can be replicated by the thousand on a single die. Each unit must be small (tens of LUTs, not hundreds), fast (one update per clock), and autonomous (no shared multiplier, no global learning-rate schedule). Applications range from per-pixel background models in always-on vision sensors to per-channel RF spectrum statistics and conditional-probability primitives in inference meshes—settings where thousands of independent probability estimates must be maintained in parallel with minimal per-unit resources. Existing candidates—fixed-point EWMA, quantized SGD, saturating counters—either require arithmetic units, demand tuned hyperparameters, or sacrifice tracking accuracy.

This paper studies the Brain-CA Estimator, a bit-serial learner designed from hardware constraints. Its update—the Cincinnati Algorithm—uses only bit storage, local comparison, control logic, and random bits—no multiply, divide, add, floating-point state, or running count. On an Alveo U200 FPGA, the 8-bit update path synthesizes to 48 LUTs at $F_{\max}$=96 MHz (Sec. III). This primitive *composes*: three Brain-CA Estimator units in a conditional decomposition achieve roughly half the MSE of three bias-selected EWMAs.

The Brain-CA Estimator occupies a design point between quantized stochastic approximation [2], [3] and change-point models [1]: bounded memory like the former, adaptive forgetting like the latter, but with no run-time tunable parameters. It also shares the bounded-state philosophy of probabilistic counting [10]. Our baselines span the design space: the saturating counter (arithmetic-free, no error-proportional steps), EWMA and QSGD (arithmetic-based, tunable), and MLE (unbounded memory). Supplementary material [4] develops full proofs and extended experiments.

Contributions.

1) *Composed inference (primary systems result).* Three Brain-CA Estimator units in a conditional decomposition beat three globally-tuned EWMAs in every one of four primary scenarios without per-regime tuning, and remain competitive against oracle per-branch-tuned EWMA on a broader five-scenario transfer test (Sec. V).
2) *Hardware profile and tracking.* The 8-bit Brain-CA Estimator synthesizes to 48 LUTs with no multiplier and no adder (Sec. III); on switching and continuous drift it remains within $1.5\times$–$2.1\times$ of the best tuned fixed-point baselines and outperforms all arithmetic-free baselines, with no run-time tunable parameters (Secs. III-A–III-C).
3) *Exact tracking law.* The fixed-width regime is a birth-death chain whose stationary distribution is exactly $\mathrm{Binomial}(N-1,\theta)$, yielding closed-form mean, variance, and drift (Sec. IV)—explaining the error-proportional step sizes that avoid the small-α rounding-bias failure mode.

II. The Brain-CA Estimator Primitive

A. State Representation

The Brain-CA Estimator maintains a positive integer state s whose binary form is $1\,b_1 b_2 \ldots b_m$, with the leading 1 as sentinel and m payload bits. The encoded probability of observing 1 is

$$P_1(s) = \frac{2(s-2^m)+1}{2^{m+1}}, \quad m = \lfloor \log_2 s \rfloor. \tag{1}$$

For a b-bit state, encoded fractions range from $1/2^b$ to $1-1/2^b$ in steps of $2/2^b$.

Algorithm 1 Cincinnati Algorithm: the Brain-CA Estimator update at state s (depth m, max width b)

Require: observation $x \in \{0, 1\}$, current state s
Generate m random bits $r_1, \ldots, r_m$
Scan from MSB to LSB for first j where $r_j \neq b_j$
if mismatch found at position j **then**
  prediction $\leftarrow b_j$
  if prediction $= x$ **then**
    stay: $s \leftarrow s$ {correct: hold state}
  else
    nudge: $s \leftarrow s + 1$ if x=1; $s \leftarrow s - 1$ if x=0
  end if
else {fall-through: all random bits matched}
  if $m < b - 1$ **then**
    extend: $s \leftarrow 2s + x$ {shift payload left; x becomes new LSB}
  else
    stay: $s \leftarrow s$ {at max width: no-op}
  end if
end if

B. Update Algorithm

The update rule—the Cincinnati Algorithm—is given in Algorithm 1. The key structural property is that the mismatch-selection probabilities $\hat{P}_1(s)$ and $\hat{P}_0(s)$ are both *affine* in the encoded fraction p_s:

$$\hat{P}_1(s) = p_s - \tfrac{\delta_m}{2}, \quad \hat{P}_0(s) = (1-p_s) - \tfrac{\delta_m}{2}, \tag{2}$$

where $\delta_m = 2^{-m}$ is the fall-through probability. This affine structure is the engine behind every analytical result in this paper and every hardware advantage over EWMA.

For a stream with bias θ, the transition probabilities are:

$$\Pr[\text{nudge up}] = \hat{P}_0(s)\,\theta, \tag{3}$$

$$\Pr[\text{nudge down}] = \hat{P}_1(s)\,(1-\theta), \tag{4}$$

$$\Pr[\text{no nudge}] = \hat{P}_1(s)\,\theta + \hat{P}_0(s)\,(1-\theta) + \sigma_f, \tag{5}$$

where $\sigma_f = \delta_m$ is the fall-through probability (extension during growth; no-op stay at max width). Restoring drift dominates: below θ nudge-up exceeds nudge-down, above θ the reverse, so stationary mass concentrates near states that bracket θ.

III. Hardware Profile

Table I reports same-flow FPGA synthesis (Alveo U200, Vivado) for an 8-bit Brain-CA Estimator and two EWMA variants. The Brain-CA Estimator update path optimizes to 48 LUTs and 8 registers at $F_{\max} = 96$ MHz (96 M observations/s per unit), with no multiplier, no adder, and no carry chain. A multiplier-based EWMA (general α) synthesizes to 55–71 LUTs at 200–230 MHz; a shift-only EWMA (restricted to $\alpha = 2^{-k}$) is smaller still at 13–15 LUTs and 235–238 MHz.

The architectural distinction is subtler than the per-unit LUT counts suggest. An adder, even tiled per-unit, requires a carry chain and clock-synchronized completion; bit comparisons require neither. The Brain-CA Estimator's randomness amortizes via shared LFSRs—a single ∼10-LUT LFSR serves many adjacent Brain-CA Estimators [4]. We do not claim these effects close shift-EWMA's ∼3× per-unit FPGA gap or ∼2.4× $F_{\max}$ advantage; the distinction is one of degree, not category. A tile-level back-of-envelope appears in [4]; full ASIC modeling is future work. Shift-EWMA further inherits the rounding-bias failure mode of Sec. III-B and requires α tuning.

Each width bit halves stationary variance but doubles adaptation time; b is set at synthesis. Algorithm 1's $s \pm 1$ notation describes Markov-state transitions; the ± 1 admits a masked-XOR realization in which the carry boundary is located off-cycle by a priority encoder conditioned on the prediction bit, so the on-cycle update path is comparator + XOR with no carry primitive [4, §6.1].

TABLE I
Implementation profile (8-bit, Alveo U200, Vivado, same-flow synthesis).

Property	Brain-CA Estimator	EWMA-mult	EWMA-shift
State bits	b	b	b
Multipliers	0	1 ($b\times b$)	0
Adders	0	1	1
Logic	$O(b)$	$O(b^2)$	$O(b)$
Run-time params	0	1 (α)	1 (α=2^{-k})
Random bits/upd.	$\leq b-1$	0	0
LUTs (α=$\frac{1}{32}$)	48*	55	13
LUTs (α=$\frac{1}{128}$)	48*	71	15
Registers	8	12–13	8
$F_{\max}$ (MHz)	96	200–230	235–238
LUTs (16-bit)	130	—	—

*Series; 60 pre-opt. Brain-CA Estimator LUTs are independent of α (no α parameter). Both EWMA variants exhibit the rounding-bias failure mode of Sec. III-B.

TABLE II
Post-switch MSE (t=800–1000, 500 runs, θ: 0.8 → 0.2, 8-bit).

Method	Param	MSE	Mem	Logic
Brain-CA Estimator	—	0.003	b	$O(b)$, no arith.
EWMA, step-match	α=$\frac{1}{128}$	0.003	b	$O(b)$ shift‡
EWMA, bias-sel.	α=$\frac{1}{32}$	0.002	b	$O(b)$ shift‡
Sat. counter	—	0.021	b	$O(b)$, no arith.
QSGD	η=.002	0.002	b	$O(b^2)$
MLE†	—	0.112	$O(\log t)$	$O(b)$

†Reference point; unbounded memory, not a hardware candidate.
‡EWMA values shown are powers of 2; can use shift implementation requiring an adder. For general α, EWMA is $O(b^2)$ (multiplier).

A. Baseline Comparison: The Pareto Frontier

Table II compares the Brain-CA Estimator against five baselines on a switching Bernoulli process (θ: 0.8 → 0.2 at t=500, 500 runs). All methods use 8-bit estimate precision.

The key observation is not which method has the lowest MSE—QSGD and the bias-selected EWMA edge the Brain-CA Estimator out at 0.002 vs. 0.003, but both require arithmetic and an explicitly chosen step size. The key observation

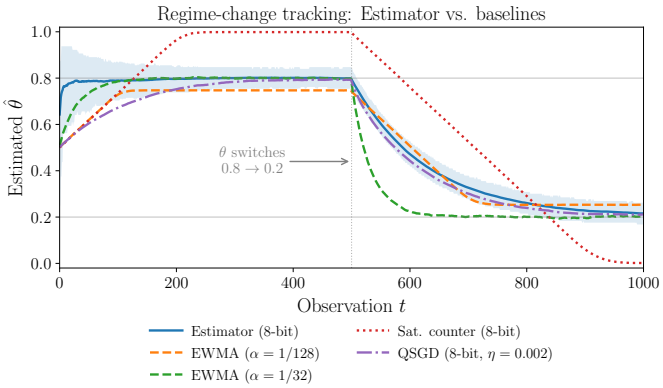

Fig. 1. Regime-change tracking: 8-bit Brain-CA Estimator (blue, with 10th–90th percentile band) vs. four baselines. θ switches $0.8 \to 0.2$ at t=500 (500 runs). The step-matched EWMA (α=1/128, orange dashed) plateaus below 0.8 pre-switch due to rounding-induced bias. The saturating counter (red dotted) overshoots to $\sim$1.0 and adapts slowest. The bias-selected EWMA (α=1/32, green dashed) adapts fastest but requires a tuned α.

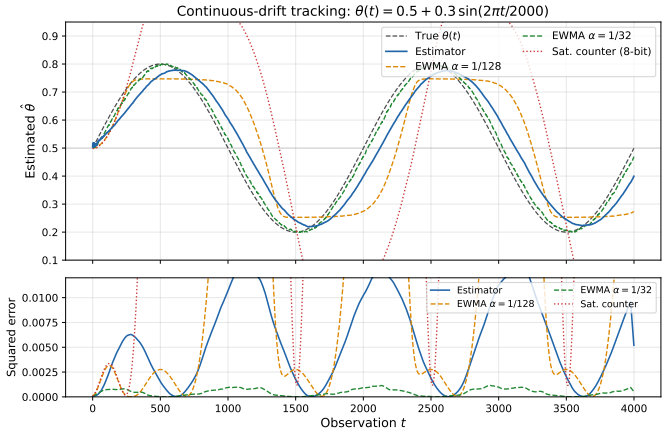

Fig. 2. Continuous-drift tracking: $\theta(t) = 0.5 + 0.3\sin(2\pi t/2000)$, 8-bit, 500 runs. Top: ensemble means (Brain-CA Estimator blue band: 10th–90th percentile). Bottom: smoothed squared error (50-observation window). The Brain-CA Estimator tracks with a visible phase lag but follows the sinusoid; the step-matched EWMA (α=1/128) exhibits severe lag and rounding-induced bias at the extremes.

is the *Pareto frontier across MSE, tunability, and architectural locality*:

- **No other arithmetic-free, run-time-parameter-free method is within $2\times$ of the Brain-CA Estimator's MSE.**
- The nearest arithmetic-free competitor (the saturating counter) is $7\times$ worse on MSE because it lacks error-proportional step sizes; methods matching or beating the Brain-CA Estimator on MSE all require an adder (shift-EWMA) or a multiplier (general EWMA, QSGD) plus a tuned step-size parameter.

B. The Rounding-Bias Failure Mode

Figure 1 reveals a practical failure mode of fixed-point EWMA that is directly relevant to hardware designers. At $\alpha = 1/128$, the quantized update $\lfloor \alpha(x-p) \rceil$ rounds to zero on the 8-bit grid when p is near the boundary, so the EWMA converges to ≈ 0.75 instead of 0.8. This rounding-induced bias is specific to small-α fixed-point EWMA—larger α values (e.g., the bias-selected α=1/32) have minimal boundary bias but sacrifice the slow-drift tracking that motivated the small α in the first place. The failure mode is a *structural* consequence of fixed-α quantization at small step sizes and extreme probabilities, and it is independent of whether the EWMA is implemented with a multiplier or a shift.

The Brain-CA Estimator avoids this specific failure mode. Its effective step size is not a fixed α applied to $(x - p)$; instead, the affine mismatch structure (2) makes the nudge probability *proportional to the estimation error*. When the estimate is close to the truth, nudges are rare (the state is sticky); when the estimate is far from the truth, nudges are frequent. This error-proportional mechanism operates at the level of individual bit comparisons, with no rounding step.

TABLE III
CONTINUOUS-DRIFT MSE (SECOND CYCLE, t=2000–4000, 500 RUNS).

Method	Param	MSE
Brain-CA Estimator (8-bit)	—	0.0077
EWMA (α=1/128)	1/128	0.0255
EWMA (α=1/32)	1/32	0.0036
Sat. counter (8-bit)	—	0.136

C. Continuous Drift

The switching experiments above use abrupt change-points. To test behavior under gradual non-stationarity, we simulate a sinusoidal drift: $\theta(t) = 0.5 + 0.3\sin(2\pi t/2000)$, so the true bias sweeps smoothly between 0.2 and 0.8 with period 2000 (4000 observations, 500 runs).

Table III reports time-averaged MSE over the second cycle. The Brain-CA Estimator's MSE (0.0077) sits between the two EWMA tunings: $3.3\times$ lower than the step-matched EWMA (0.0255) but $2.1\times$ higher than the bias-selected EWMA (0.0036, whose α=1/32 happens to match this drift rate). It is $18\times$ lower than the saturating counter (0.136). The step-matched EWMA's rounding bias, visible in the switching experiment (Fig. 1), manifests here as a failure to track the sinusoid's extremes. A different drift rate would require a different α for the bias-selected EWMA; the Brain-CA Estimator provides a single parameter-free operating point competitive across both switching and continuous-drift regimes without tuning.

IV. ANALYTICAL FOUNDATIONS: WHY IT WORKS

The hardware advantages described above are not empirical accidents; they follow from exact analytical structure. The supplementary material [4] provides full proofs; we give derivation sketches to make the results self-contained.

TABLE IV
WIDTH-SCALING (θ: 0.8 $\to$ 0.2, 500 RUNS).

b	N	Cross	Settle	Cross ratio	Settle ratio
6	32	21	84	—	—
8	128	88	324	4.2	3.9
10	512	274	1165	3.1	3.6

Cross: mean crosses $(\theta_1{+}\theta_2)/2$. Settle: within ± 0.05 of θ_2.

A. Exact Tracking Law

Once all $b{-}1$ payload bits are active, fall-through becomes a no-op and the state evolves as a finite birth-death chain on $N = 2^{b-1}$ tracking states. Reindex as $j = 0, \ldots, N{-}1$, encoding $p_j = (2j{+}1)/(2N)$. From the affine mismatch structure (2), with $\varepsilon = 1/N$:

$$\lambda_j = (N{-}1{-}j)\,\varepsilon\,\theta, \tag{6}$$
$$\mu_j = j\,\varepsilon\,(1{-}\theta). \tag{7}$$

The ratio $\lambda_j/\mu_{j+1} = (N{-}1{-}j)\theta/[(j{+}1)(1{-}\theta)]$ is exactly the ratio of consecutive $\mathrm{Binomial}(N{-}1, \theta)$ probabilities. By the standard product-form theorem for birth-death chains [8]:

$$\pi_j = \binom{N{-}1}{j}\theta^j(1{-}\theta)^{N-1-j}. \tag{8}$$

This identity is verified with zero residual in exact rational arithmetic for $b \in \{3, \ldots, 12\}$ and $\theta \in \{0.3, 0.5, 0.7, 0.9\}$ [4].

The binomial form yields closed-form expressions:

$$\mathrm{E}_\pi[p] = \theta - (2\theta{-}1)/2^b, \quad |\mathrm{E}_\pi[p] - \theta| < 2^{-b}, \tag{9}$$
$$\mathrm{Var}_\pi[p] = \varepsilon(1{-}\varepsilon)\,\theta(1{-}\theta). \tag{10}$$

Drift equation. For an interior state with encoded fraction p_i:

$$E[p_{t+1} - p_t \mid p_t = p_i] = \varepsilon\Big(\theta - p_i - \tfrac{\varepsilon(2\theta-1)}{2}\Big). \tag{11}$$

The dominant term is a restoring force that always points toward θ; the correction $\varepsilon(2\theta{-}1)/2$ introduces a small bias toward $1/2$, consistent with the stationary mean (9).

B. Shrinkage Structure

The data-averaged expected value is linear shrinkage, $\mathrm{E}_\theta[p_t] = a_t\,\theta{+}(1{-}a_t)/2$, with a_t depending only on t [4] and sandwiched between Jeffreys ($t/(t{+}1)$) [6], [7] and Bayes–Laplace ($t/(t{+}2)$) at every t, with $1 - a_t \sim 1/(t \ln 2)$ [4], [5]. Both bounds are proved (Jeffreys via computer-assisted argument).

C. Width Scaling

Settle times scale as $\approx\times 4$ per two added bits (Table IV) [4], [9].

V. COMPOSED INFERENCE: THE TWO-STREAM RESULT

The systems question is whether the Brain-CA Estimator's advantage *composes*: when multiple units estimate a joint distribution through conditional decomposition, does the parameter-free advantage survive?

TABLE V
TWO-STREAM $\hat{P}(B{=}1)$: MSE OVER t=1500–2000 (8-BIT, 500 RUNS). EWMA USES α=1/32, THE ONLY VALUE WITH NEAR-ZERO BIAS IN BOTH SCENARIOS ACROSS AN α SWEEP.

Method	Stationary	Switching
Composed 3-Estimators	0.0012	0.0013
Composed 3-EWMA (α=1/32)	0.0023	0.0025
Single Brain-CA Estimator	0.0018	0.0020
Single EWMA (α=1/32)	0.0036	0.0041

A. Architecture

We compose three 8-bit Brain-CA Estimator units into a conditional-probability estimator. Unit E_A tracks $P(A{=}1)$; unit $E_{B|1}$ updates only when $A{=}1$; unit $E_{B|0}$ updates only when $A{=}0$. The marginal estimate is

$$\hat{P}(B{=}1) = \hat{P}(B{=}1|A{=}1)\,\hat{P}(A{=}1) + \hat{P}(B{=}1|A{=}0)\,\hat{P}(A{=}0). \tag{12}$$

This uses $3b = 24$ bits of state; the two products and final addition in (12) occur only at readout, not in the per-observation update path.

B. Experiments

We set $P(A{=}1) = 0.6$, $P(B{=}1|A{=}1) = 0.8$, $P(B{=}1|A{=}0) = 0.3$ (true $P(B{=}1) = 0.60$; 500 runs, 8-bit). In the switching variant, we keep $P(A{=}1) = 0.6$ and swap the conditionals at t=1000 so $P(B{=}1)$ changes from 0.60 to 0.50.

C. Results

Table V includes a composed 3-EWMA baseline ($\alpha = 1/32$) under the same decomposition (12). We swept $\alpha \in \{1/16, 1/32, 1/64, 1/128\}$ globally; $\alpha = 1/32$ is the only value with near-zero bias in both scenarios. Lower-α global EWMA can beat the Brain-CA Estimator on specific configurations via component-bias cancellation, but this depends on (P_A, β_1, β_0) and breaks under shift; oracle per-branch tuning likewise beats the Brain-CA Estimator by 4–8$\times$ on individual configurations but fails to transfer [4, Tables 7–8]. The Brain-CA Estimator's single parameter-free operating point holds across all five tested scenarios without re-tuning.

Conditioning on A helps both learners (MSE drops $\approx$1.5–1.6$\times$), but the Brain-CA Estimator wins every cell of Table V: the composed three Brain-CA Estimators have roughly half the MSE of the composed three EWMAs, with composed bias < 0.003 across both scenarios. The 144-LUT composition of three Brain-CA Estimators has no per-update arithmetic; under extreme branch-rate asymmetry ($P(A{=}1){=}0.95$), the Brain-CA Estimator retains its 2$\times$ advantage [4].

VI. CONCLUSION

The Brain-CA Estimator is a bounded-state, bit-serial learner with an exact binomial stationary law and shrinkage between Jeffreys and Bayes–Laplace. No other method in

Table II combines competitive MSE, no run-time tunable parameters, and arithmetic-free updates; on continuous drift it is competitive with bias-selected EWMA (0.0077 vs. 0.0036) and $3.3\times$ better than step-matched EWMA, without per-regime tuning. On FPGA, the 8-bit update synthesizes to 48 LUTs at 96 MHz—comparable per-unit cost to multiplier-based fixed-point EWMA, but with no arithmetic to centralize at scale. The two-stream composition (Sec. V) shows the same pattern: three Brain-CA Estimator units beat globally-tuned 3-EWMA across all four primary scenarios and hold steady across the broader five-scenario transfer test. The supplement [4] derives asymptotic regret estimates (not formal proofs)—$O(T/2^b)$ stationary, $O(\sqrt{ST})$ under S regime changes—validated by simulation. Whether this pattern scales to richer multivariate structures remains the central open question.

Limitations. All experiments use synthetic Bernoulli streams; validation on real-world non-stationary data is needed. Synthesis is for Vivado on Alveo U200; ASIC and other flows may differ. EWMA baselines use globally-tuned and (in the supplement) per-branch-oracle-tuned settings; intermediate adaptive-tuning schemes are not evaluated. The composition analysis covers only $k{=}2$ streams.

Negative societal impact. The Brain-CA Estimator is a general-purpose statistical primitive whose societal impact is dominated by the systems that incorporate it. By lowering the per-unit area and energy floor for parallel probability estimation, the design can support beneficial applications such as energy-efficient environmental sensing and edge intelligence, and equally can support denser deployment of always-on sensors used for non-consensual monitoring. The primitive itself does not encode policy choices about data collection or use; responsibility for those choices rests with system integrators and operators.

Use of generative AI tools. The author used Claude (Anthropic) and ChatGPT (OpenAI) to assist with initial drafting and with subsequent review and corrections of the manuscript. All technical claims, experimental results, theoretical derivations, and analytical content are the author's own, verified and approved by the author. No AI system is listed as an author, and the author takes full responsibility for the contents of the paper.

References

[1] R. P. Adams and D. J. C. MacKay, "Bayesian online changepoint detection," *arXiv preprint arXiv:0710.3742*, 2007.

[2] D. Alistarh, D. Grubic, J. Li, R. Tomioka, and M. Vojnovic, "QSGD: Communication-efficient SGD via gradient quantization and encoding," in *Advances in Neural Information Processing Systems*, 2017.

[3] C. De Sa, M. Feldman, C. Ré, and K. Olukotun, "Understanding and optimizing asynchronous low-precision stochastic gradient descent," in *Proceedings of the 44th Annual International Symposium on Computer Architecture (ISCA)*, Toronto, ON, Canada, 2017, pp. 561–574.

[4] J. Felix, "A hardware-native bit-serial learner with exact statistical structure: Supplementary material," 2026, supplementary material accompanying the main paper.

[5] P. Flajolet and R. Sedgewick, *Analytic Combinatorics*. Cambridge University Press, 2009.

[6] H. Jeffreys, *Theory of Probability*, 3rd ed. Oxford University Press, 1961.

[7] R. E. Kass and L. Wasserman, "The selection of prior distributions by formal rules," *Journal of the American Statistical Association*, vol. 91, no. 435, pp. 1343–1370, 1996.

[8] F. Kelly, *Reversibility and Stochastic Networks*. Cambridge University Press, 2011.

[9] D. A. Levin, Y. Peres, and E. L. Wilmer, *Markov Chains and Mixing Times*, 2nd ed. American Mathematical Society, 2017.

[10] R. Morris, "Counting large numbers of events in small registers," *Communications of the ACM*, vol. 21, no. 10, pp. 840–842, 1978.

A Hardware-Native Bit-Serial Learner with Exact Statistical Structure:

Supplementary Material

Jerry Felix
Brain-CA Technologies, Sarasota, FL, USA
jfelix@brain-ca.com

Abstract

This document provides full proofs, extended experiments, and detailed derivations supporting the workshop paper "A Hardware-Native Bit-Serial Learner with Exact Statistical Structure." The workshop paper presents four contributions—composed inference (the primary systems result), hardware profile, baseline comparisons, and an exact tracking law—and cites this companion report for proofs and extended data. This document delivers those proofs and data.

Reading guide. This supplementary material is organized to let a reviewer locate the support for each workshop-paper claim efficiently. *Essential sections for a time-constrained reviewer:* Sections 2, 3.5, 4, and Appendix C. The remaining appendices provide full proofs for completeness.

Workshop claim	Support in this document
Binomial stationary distribution	Section 2
Shrinkage linearity ($\mathrm{E}_\theta[p_t] = a_t\theta + (1-a_t)/2$)	Appendix C
Bayes–Laplace lower bound	Appendix D, Sec. D.4
Jeffreys upper bound (computer-assisted)	Appendix E
Asymptotic rate $1 - a_t \sim 1/(t \ln 2)$	Appendix D
Exact rational verification of π_j	Section 2
Width-scaling data (Table IV in workshop)	Section 3
Two-stream composition details	Section 4
Continuous-drift experiment	Section 3.5
Regret estimates (heuristic)	Section 5

Notation. The following conventions are used throughout:

Symbol	Meaning
s	Markov state index (positive integer)
t	Observation count (number of stream bits seen so far)
b	Bit width of the Brain-CA Estimator (total bits including sentinel)
m	Number of payload bits ($m = \lfloor \log_2 s \rfloor$)
θ	True stream bias $P(\text{observe } 1)$
p_s, $P_1(s)$	Encoded probability at state s: $(2(s - 2^m) + 1)/2^{m+1}$
$\hat{P}_1(s)$, $\hat{P}_0(s)$	Mismatch-selection probabilities for predicting 1 and 0
$P_f(s)$	Fall-through probability ($= 2^{-m}$)
δ_m	Fall-through probability at depth m ($= 2^{-m}$)
ε	Tracking-regime step size ($= 2/2^b = 1/N$)
$\sigma(s, \theta)$, σ_i	Total stay probability at state s (or i) under bias θ
λ_i, μ_i	Nudge-up (birth) and nudge-down (death) rates in tracking regime
π_i	Stationary distribution mass at tracking-regime state i
N	Number of tracking-regime states ($= 2^{b-1}$)
a_t	Shrinkage coefficient at observation t
r_t	Interpolation ratio between Jeffreys and Bayes–Laplace

1. The Brain-CA Estimator: Mechanism and Properties

This section reproduces the mechanism definition from the workshop paper for self-containedness, and adds the transition-probability derivation needed by the proofs in the appendices.

1.1. State Representation

The Brain-CA Estimator maintains an internal state represented as a Markov state index—a positive integer s whose binary representation $1\, b_1\, b_2 \ldots b_m$ consists of a sentinel bit (the leading 1), followed by m payload bits. Each state encodes a probability of observing 1:

$$P_1(s) = \frac{2x + 1}{2^{m+1}} \tag{1}$$

where $m = \lfloor \log_2 s \rfloor$ is the number of payload bits and $x = s - 2^m$ is the offset. The implied trailing 1 ensures that representable fractions always have odd numerators and power-of-two denominators. Neither 0 nor 1 is representable. Table 10 (Appendix A) lists all states through 8 bits (states 1–255).

In the Markov diagrams (Figures 1–2), 1 is rendered as Red and 0 as Blue, following design convention. In the mathematical analysis, we use 0 and 1 throughout. The notation $P_1(s)$ and $P_0(s) = 1 - P_1(s)$ denotes the encoded probability of 1 and 0 respectively.

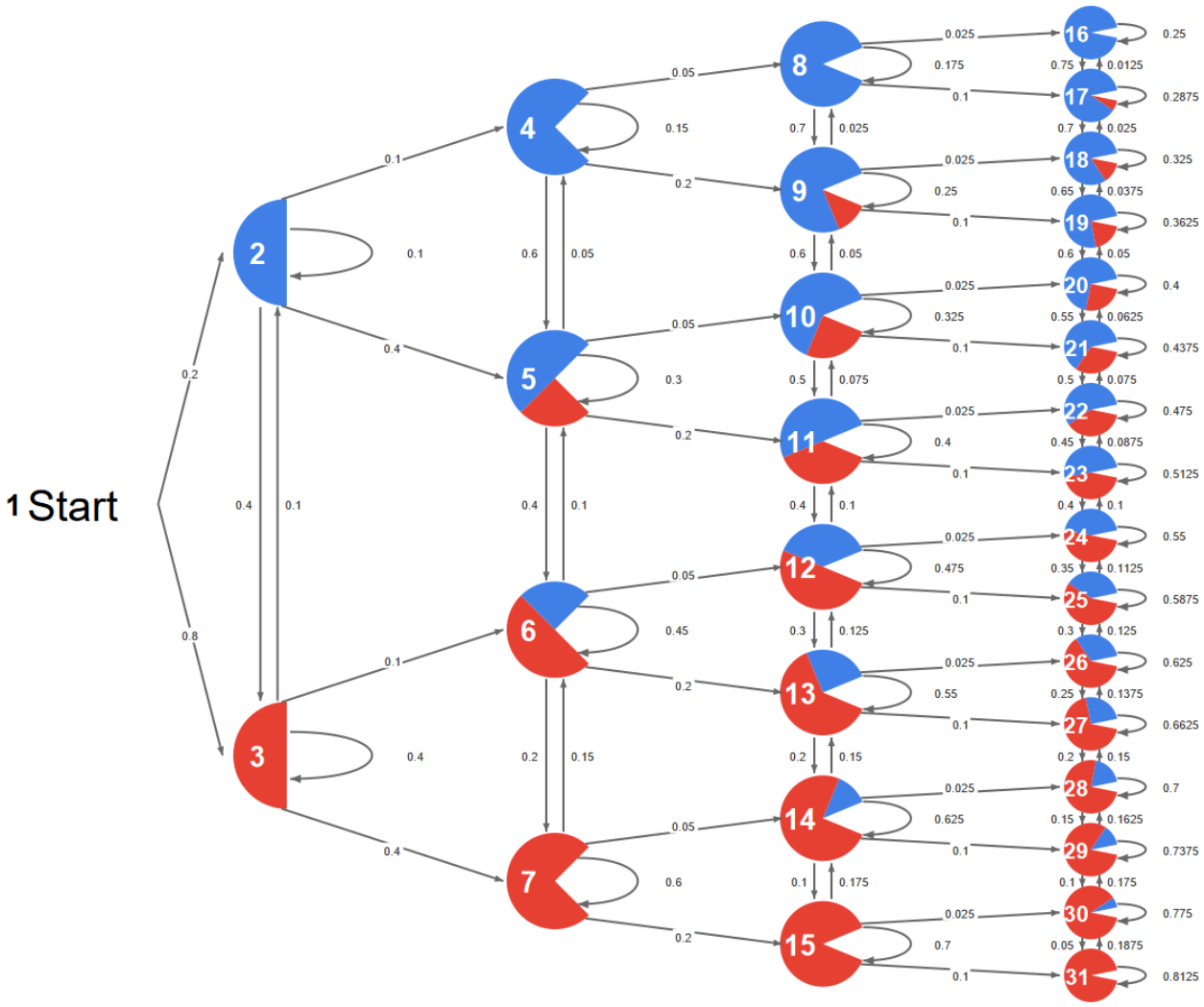

Figure 1 | Markov state-space tree for a 5-bit Brain-CA Estimator under $\theta = 0.8$. Each node displays a pie chart of its encoded fraction: red for the known $P(1)$ component, blue for the known $P(0)$ component. The unfilled portion is the *uncertainty gap*—treated as 50/50 for predictions. State 1 (Start) has no pie chart: pure uncertainty. The first observation (say x=1) extends to state 3: half red, half gap, predicting 3/4 red. In column 2, subsequent observations nudge the state up or down while falling through to column 3 half the time. By the rightmost column, the pie charts are nearly complete and stationary mass concentrates on states 28–29, whose encoded fractions ($25/32 = 0.781$ and $27/32 = 0.844$) bracket $\theta = 0.8$. States below the truth experience upward drift; states above it experience downward drift, so stationary mass concentrates near the states that bracket θ. The result is convergence.

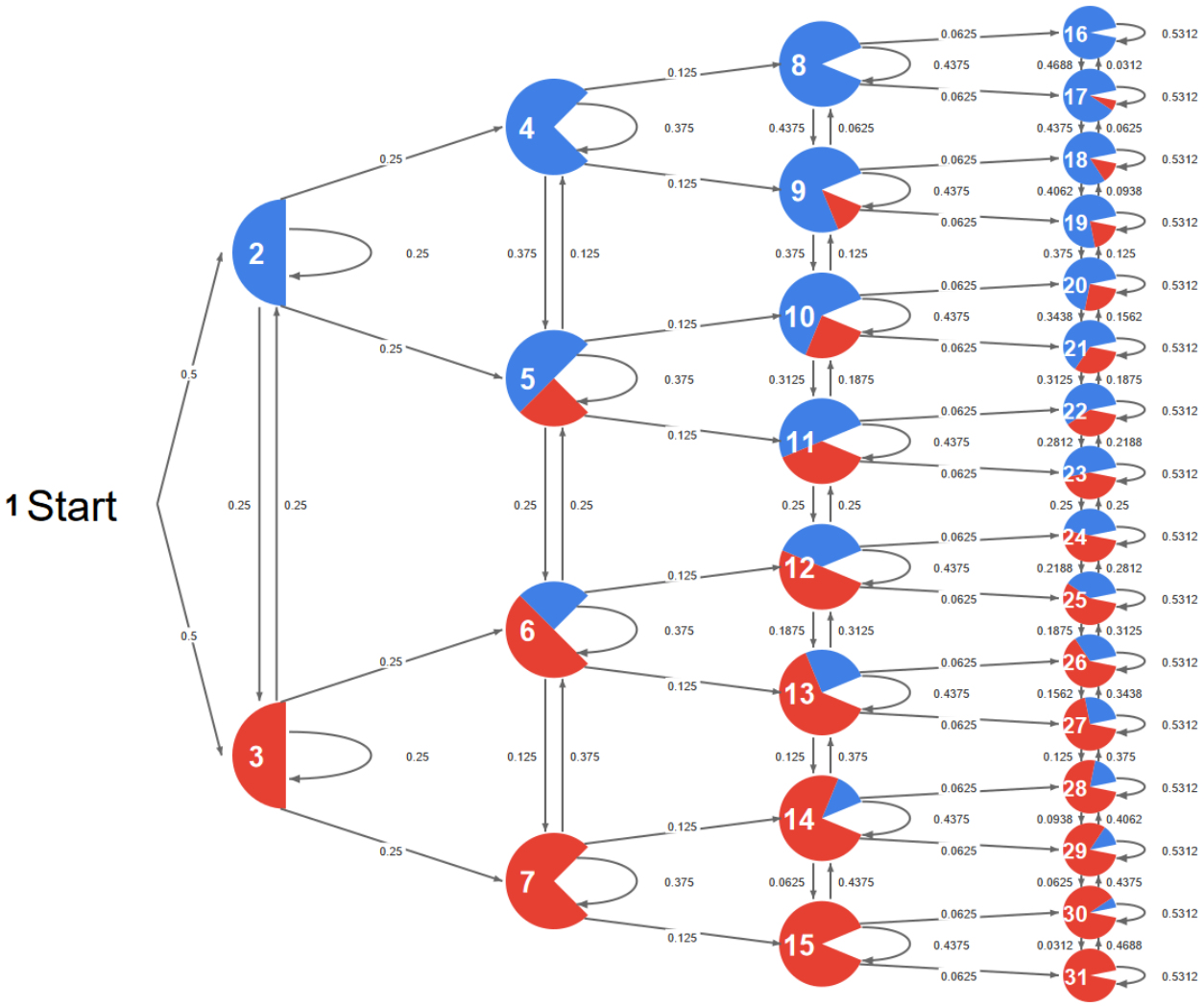

Figure 2 | Markov state-space tree under $\theta = 0.5$, illustrating two kinds of movement. *Rightward* movement extends the state by one bit, doubling the number of available states and halving the uncertainty gap. This is initially rapid (fall-through is certain at depth 0) but slows logarithmically: probability $1/2$ from column 2 to 3, $1/4$ from 3 to 4, $1/8$ from 4 to 5. *Vertical* movement nudges up or down within a column. Rightward movement adjusts *precision*—how fine the grid is. Vertical movement adjusts *accuracy*—at the current precision level, which grid point the system occupies. At $\theta = 0.5$, the symmetric bias produces equal upward and downward transition probabilities, so stationary mass spreads evenly across the center of the rightmost column—the system correctly represents maximum uncertainty about the direction of the bias. Comparing with Figure 1: the same mechanism that concentrates mass near 0.8 there distributes it symmetrically here. The Brain-CA Estimator does not "know" θ—it discovers it.

1.2. The Update Algorithm: The "Cincinnati Algorithm"

When observation $x \in \{0, 1\}$ arrives at state s with m payload bits:

1. **SELECT.** Generate m random bits $r_1, \ldots, r_m$. Scan from MSB to LSB, comparing each r_j to the corresponding storage bit b_j. Stop at the first mismatch.
2. **PREDICT.** If mismatch at position j: predict $y = b_j$. If no mismatch (all random bits match storage): fall-through.[1]
3. **UPDATE.**
   - *Mismatch, prediction correct* ($y = x$): Stay at state s.
   - *Mismatch, prediction wrong* ($y \neq x$): Nudge. If $x = 1$, set $s \leftarrow s + 1$. If $x = 0$, set $s \leftarrow s - 1$. The nudge is guaranteed not to overflow because the mismatched bit provides the needed 0 or 1 (Section 6.1 discusses how this ± 1 is realized in hardware without an instantiated carry chain).
   - *Fall-through, below maximum bit width*: Extend by appending x as a new payload bit ($s \to 2s + x$).
   - *Fall-through, at maximum bit width*: Stay at state s. No update occurs. This rule converts the growth regime into the tracking regime: once all bits are active, the Brain-CA Estimator can change state only through mismatch-driven nudges, and the fall-through event becomes a no-op. The entire fixed-width analysis in Section 2 depends on this rule.

[1] The prediction generated in the PREDICT step also serves as the Brain-CA Estimator's inference output. Two modes are natural: *best-guess* inference, which always predicts the MSB (the most probable outcome), and *probabilistic* inference, which uses the stochastic prediction from SELECT. In the probabilistic mode, fall-through (no mismatch) produces no prediction; this case is resolved by predicting with 50/50 probability, which removes timing variability, handles the initial state $s = 1$ without looping, and is simple to implement in hardware. An alternate implementation re-randomizes on fall-through, which changes prediction probabilities slightly: the denominator becomes $2^m - 1$ rather than 2^m, and the values 0 and 1 become representable. This variant has similar convergence properties but is not analyzed in this paper.

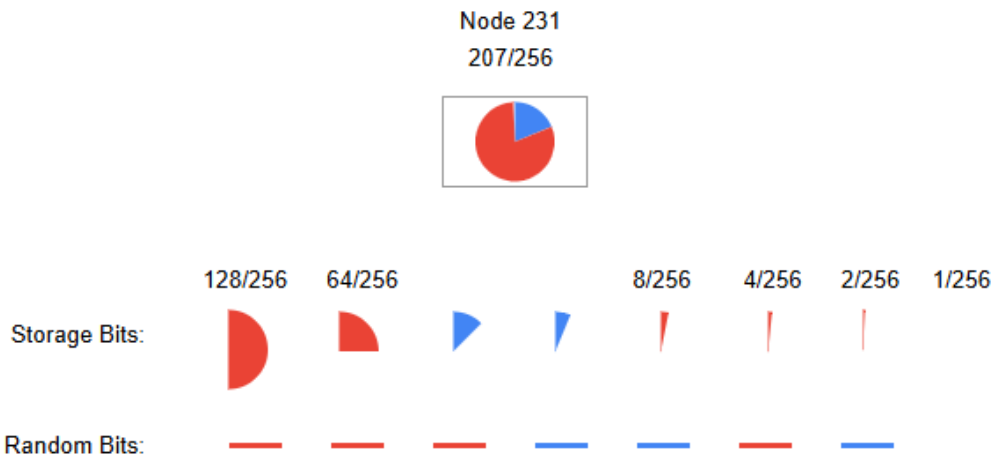

Figure 3 | The SELECT process at state 231 (encoding $207/256 \approx 0.809$). The Markov state index 231 has binary representation `11100111`, consisting of a sentinel bit (`1`), payload (`1100111`), and an implied trailing `1`. Appending the trailing bit to the payload gives `11001111` $= 207$ (the numerator); the sentinel bit's position (bit 8 from the LSB) determines the denominator $2^8 = 256$. Each storage bit is shown as a red/blue pie chart at its positional weight. A random bit (bottom row) is compared to each storage bit from MSB to LSB; the first mismatch determines the prediction. An 8-bit example is shown to illustrate the full scan across multiple payload positions.

1.3. Transition Probabilities

At state s with m payload bits, the selection process produces three outcomes with state-dependent probabilities:

$$\hat{P}_1(s) = \sum_{j:\, b_j=1} 2^{-j} \qquad \text{(predict 1)} \tag{2}$$

$$\hat{P}_0(s) = \sum_{j:\, b_j=0} 2^{-j} \qquad \text{(predict 0)} \tag{3}$$

$$P_f(s) = 2^{-m} \qquad \text{(fall-through)} \tag{4}$$

Note: $\hat{P}_1(s) + \hat{P}_0(s) + P_f(s) = 1$, and $P_1(s) = \hat{P}_1(s) + P_f(s)/2$, where $\hat{P}_1(s)$ is the selection probability of predicting 1 (excluding fall-through) and $P_1(s)$ is the encoded probability (including the implied trailing bit). Throughout this paper, $P_1(s)$ refers to the encoded probability unless otherwise noted.

For a stream with bias θ, the transition probabilities are:

Transition	Condition	Probability
Stay	Predict 1, observe 1	$\hat{P}_1(s) \cdot \theta$
Stay	Predict 0, observe 0	$\hat{P}_0(s) \cdot (1-\theta)$
Nudge up	Predict 0, observe 1	$\hat{P}_0(s) \cdot \theta$
Nudge down	Predict 1, observe 0	$\hat{P}_1(s) \cdot (1-\theta)$
Extend ($2s$+1)	Fall-through, observe 1	$P_f(s) \cdot \theta$
Extend ($2s$)	Fall-through, observe 0	$P_f(s) \cdot (1-\theta)$

In the tracking regime (maximum bit width), the two Extend rows become Stays: the fall-through probability $P_f(s)$ contributes entirely to the stay probability rather than producing state transitions.

The self-correction mechanism follows from drift balance rather than from maximum stay probability. At state s with encoded fraction $P_1(s)$, the rates of upward and downward nudges in the tracking regime are $\hat{P}_0(s)\,\theta$ and $\hat{P}_1(s)\,(1-\theta)$ respectively. States encoding $P_1(s) < \theta$ have larger upward than downward nudge probability; states with $P_1(s) > \theta$ have larger downward than upward nudge probability. Consequently, the stationary distribution concentrates around the states whose encoded fractions bracket θ, as verified by the explicit binomial form derived in Section 2.

2. Tracking-Regime Stationary Distribution

The stationary distribution of a finite birth-death chain follows from standard product-form theory (see, e.g., Kelly 2011). The contribution of this section is not the method—which is classical—but its application to the specific state-dependent transition probabilities of the update algorithm, yielding explicit closed-form expressions that connect the Brain-CA Estimator's bit-level design to its statistical behavior.

Remark 1 (Inference by equilibrium)**.** *In a textbook Markov chain, the transition matrix is fixed and the stationary distribution is a consequence—one studies where the chain ends up. Here the relationship is inverted. The transition probabilities are not fixed constants; they are parameterized by the unknown stream bias θ, which the system has never been told. The data enters one bit at a time, shaping the transition rates through the mismatch mechanism, and the chain's stationary distribution concentrates on states whose encoded fractions bracket θ. The equilibrium* is *the estimate.*

This is loosely analogous to MCMC, where one engineers a chain whose stationary distribution matches a target posterior—but with a key difference. In MCMC, the target distribution is chosen by the designer. Here, nobody chose the binomial; it emerged as a structural consequence of the bit-serial comparison rule (Algorithm 1 in Section 1). The affine mismatch structure forces the birth and death rates to be linear in the state index, and linearity in the index is precisely what produces a binomial stationary law. The Brain-CA Estimator was designed for hardware simplicity, and the binomial—with its closed-form mean, variance, and drift—is a consequence, not a design goal.

2.1. The Birth-Death Chain

For a b-bit Brain-CA Estimator in the tracking regime, the state space is $\{2^{b-1}, \ldots, 2^b - 1\}$, containing $N = 2^{b-1}$ states that encode the odd multiples of $1/2^b$. (Note: unlike the standard birth-death convention indexed from 0 to $N-1$, the state index i here ranges from 2^{b-1} to $2^b - 1$, matching the Markov state index directly. The birth-death structure is the same; only the index labels differ.)

At state i with encoded fraction p_i, under bias θ:

Nudge up ($i \to i+1$): Only mismatch-wrong events produce nudges (fall-through is a stay in the tracking regime). The probability of predicting 0 via mismatch is $\hat{P}_0(i) = (1-p_i) - P_f(i)/2 = (1-p_i) - 1/2^b$:

$$\lambda_i = [(1-p_i) - 1/2^b] \cdot \theta \tag{5}$$

Nudge down ($i \to i-1$): Similarly, $\hat{P}_1(i) = p_i - 1/2^b$:

$$\mu_i = [p_i - 1/2^b] \cdot (1-\theta) \tag{6}$$

Stay: $\sigma_i = \hat{P}_1(i)\,\theta + \hat{P}_0(i)\,(1-\theta) + P_f(i)$, where $P_f(i) = 1/2^{b-1}$. The fall-through probability contributes entirely to the stay probability. At boundaries, attempted nudges beyond the range also become stays.

2.2. Stationary Distribution: Closed Form

The stationary distribution is:

$$\pi_i = \pi_{i_0} \prod_{j=i_0}^{i-1} \frac{\lambda_j}{\mu_{j+1}}, \quad \frac{\lambda_j}{\mu_{j+1}} = \frac{[(1-p_j) - 1/2^b]\,\theta}{[p_{j+1} - 1/2^b]\,(1-\theta)}. \tag{7}$$

Since $p_{j+1} = p_j + 2/2^b$, this ratio exceeds 1 when $p_j < \theta$ and falls below 1 when $p_j > \theta$. The stationary distribution peaks near the states closest to θ.

Binomial identification. Reindexing the $N = 2^{b-1}$ tracking states as $j = 0, \ldots, N-1$ (so that original state index $i = 2^{b-1} + j$ maps to reindexed state j, encoding fraction $p_j = (2j+1)/(2N)$), the nudge-up rate at state j is $\lambda_j = [(1-p_j) - 1/2^b]\,\theta$. Substituting $[(1-p_j) - 1/2^b] = 1 - (2j+1)/(2N) - 1/(2N) = (N-1-j)/N$ gives $\lambda_j = (N-1-j)\,\varepsilon\,\theta$, where $\varepsilon = 1/N = 2/2^b$. Similarly, $\mu_j = j\,\varepsilon\,(1-\theta)$. The product-form ratio becomes

$$\frac{\lambda_j}{\mu_{j+1}} = \frac{(N-1-j)\,\varepsilon\,\theta}{(j+1)\,\varepsilon\,(1-\theta)} = \frac{(N-1-j)\,\theta}{(j+1)(1-\theta)}, \tag{8}$$

where the ε factors cancel. This is the ratio of consecutive Binomial$(N-1, \theta)$ probabilities. Therefore the stationary distribution is

$$\pi_j = \binom{N-1}{j} \theta^j\,(1-\theta)^{N-1-j}, \qquad j = 0, \ldots, N-1. \tag{9}$$

This is verified by exact rational-arithmetic enumeration for $b \in \{3, \ldots, 12\}$ and $\theta \in \{0.3, 0.5, 0.7, 0.9\}$ (40 combinations, zero residual). The binomial form immediately yields exact expressions for the stationary mean and variance (Section 2).

Remark 2 (Why the binomial emerges)**.** *The binomial form is not a coincidence of the encoding but a structural consequence of Lemma 3 (the affine mismatch structure). Because $\hat{P}_0(s) = (1-p_s) - \delta_m/2$ and $\hat{P}_1(s) = p_s - \delta_m/2$ are both affine in the encoded fraction p_s, the nudge-up rate λ_j is linear in the "distance from the top" $(N-1-j)$, and the nudge-down rate μ_j is linear in the "distance from the bottom" j. Any birth-death chain whose rates have this linear-in-index form produces a binomial stationary distribution—this is the discrete analogue of the Ornstein–Uhlenbeck process, whose stationary distribution is Gaussian for the same reason (linear restoring force, constant noise). The update algorithm's binary-subdivision SELECT process forces the mismatch probabilities into this affine form, and the binomial follows.*

2.3. Worked Example: 3-Bit Brain-CA Estimator, $\theta = 7/10$

States $\{4, 5, 6, 7\}$, encoding $\{1/8, 3/8, 5/8, 7/8\}$, with $P_f = 1/4$ and $1/2^b = 1/8$:

St	p_i	λ_i	μ_i
4	1/8	$(7/8-1/8)(7/10) = 21/40$	$(1/8-1/8)(3/10) = 0$
5	3/8	$(5/8-1/8)(7/10) = 7/20$	$(3/8-1/8)(3/10) = 3/40$
6	5/8	$(3/8-1/8)(7/10) = 7/40$	$(5/8-1/8)(3/10) = 3/20$
7	7/8	$(1/8-1/8)(7/10) = 0$	$(7/8-1/8)(3/10) = 9/40$

Note that $\mu_4 = 0$ because $p_4 = 1/8 = 1/2^b$, so the factor $[p_i - 1/2^b] = 0$ in the nudge-down formula. Similarly, $\lambda_7 = 0$ because $1 - p_7 = 1/8 = 1/2^b$, so the factor $[(1 - p_i) - 1/2^b] = 0$ in the nudge-up formula. These are natural consequences of the encoding, not boundary reflections.

Ratios: $\lambda_4/\mu_5 = 7,\ \lambda_5/\mu_6 = 7/3,\ \ \lambda_6/\mu_7 = 7/9$. Unnormalized: $\pi_4 = 1$, $\pi_5 = 7$, $\pi_6 = 49/3$, $\pi_7 = 343/27$. Total $= 1000/27$.

State	Fraction	π_i	Decimal
4	1/8	27/1000	0.027
5	3/8	189/1000	0.189
6	5/8	441/1000	0.441
7	7/8	343/1000	0.343

Stationary mean $= 13/20 = 0.65$, MSE ≈ 0.042. The two bracketing states (6 and 7) hold 78% of the mass. These probabilities are exactly Binom$(3, 7/10)$:

$$\left\{(3/10)^3,\ 3(7/10)(3/10)^2,\ 3(7/10)^2(3/10),\ (7/10)^3\right\},$$

confirming the binomial identification of Eq. (9). An 8-bit Brain-CA Estimator (128 states, grid spacing ≈ 0.008) would produce much smaller MSE.

Closed-form expressions. The binomial stationary distribution yields exact expressions for the mean, variance, and drift:

$$\mathrm{E}_\pi[p] = \theta - (2\theta-1)/2^b, \quad |\mathrm{E}_\pi[p] - \theta| < 2^{-b}, \tag{10}$$

$$\mathrm{Var}_\pi[p] = \varepsilon(1-\varepsilon)\,\theta(1-\theta). \tag{11}$$

The bias toward 1/2 is $(2\theta-1)/2^b$, vanishing as b increases. Each additional bit of width halves the stationary variance but doubles the adaptation time.

2.4. Variance and MSE vs. Bit Width

The tracking-regime MSE has two components:

$$\mathrm{MSE} = (\mathrm{E}[p] - \theta)^2 + \mathrm{Var}[p] \tag{12}$$

Table 1 shows both decrease with bit width, using $\theta = 7/10$ as an example.

The MSE drops by roughly a factor of 2 per additional bit, consistent with $O(1/2^b)$ scaling. The bias^2 component fluctuates (depending on whether grid points happen to bracket θ closely), but the variance component decreases monotonically.

Table 1 | Tracking-regime MSE vs. bit width at $\theta = 0.7$.

Bits	States	Bias^2	Variance	MSE
3	4	0.002500	0.039375	0.041875
4	8	0.000625	0.022969	0.023594
5	16	0.000156	0.012305	0.012461

For comparison, the Bayesian posterior MSE after t observations is $\approx \theta(1-\theta)/t$, decreasing without bound. The Brain-CA Estimator's tracking-regime MSE is a fixed property of the bit width. This is the fundamental tradeoff: Bayesian posteriors achieve arbitrarily low MSE but require growing storage and offer no adaptation to regime changes. The Brain-CA Estimator achieves bounded MSE with fixed storage and automatic adaptation.

3. Extended Empirical Results

This section provides the full experimental data supporting the workshop paper's Tables II–V and Figures 1–2.

Run counts. The workshop paper uses 500 runs throughout to match the main baseline comparison. This supplementary reports 500 runs for the primary results (baseline comparison, continuous drift, two-stream composition) and 200 runs for the broader adaptation-time sweep (Table 2), which covers six scenarios at three bit widths. The 200-run tables have standard errors below 0.003 at all reported time points; the larger run count for the primary comparisons provides additional precision for the MSE values reported in the workshop paper.

3.1. Adaptive Forgetting

When the Brain-CA Estimator's state encodes a fraction close to the true bias, predictions are mostly correct, self-loop probability is high, and the state holds steady. When the bias changes, predictions become incorrect, nudges become frequent, and the state moves toward the new truth. The adaptation rate is proportional to the model error, not to a pre-set parameter.

This is qualitatively different from fixed-rate forgetting (exponential discounting (Sutton and Barto, 2018), sliding windows), which forget at a constant rate regardless of model accuracy.

3.2. Regime-Change Response and Mixing Time

Consider a stream switching from $\theta_1 = 0.8$ to $\theta_2 = 0.2$. *Conditional on its current state*, the Brain-CA Estimator's future dynamics depend only on the new bias θ_2, not on how many observations produced the current state. Whether it observed 100 or 100,000 observations at θ_1, the dynamics under θ_2 starting from the same state are identical. This is a direct consequence of the Markov property.

Note, however, that the *unconditional* adaptation behavior can depend on pre-change duration: a longer run at θ_1 produces a distribution over current states more tightly concentrated near θ_1, which may differ from a shorter run. The Markov property guarantees that, once you condition on

the current state, history is irrelevant—but the distribution over starting states for the new regime may vary.

The adaptation time is bounded by the mixing time of the birth-death chain. A direct linear-drift argument gives the strong-drift mixing bound for this specific chain. Reindexing as in Section 2.2 (so $j = 0, \ldots, N-1$ with $N = 2^{b-1}$ and $\varepsilon = 1/N$), the per-step drift at state j under target bias θ is

$$\lambda_j - \mu_j \;=\; (N-1-j)\,\varepsilon\,\theta \;-\; j\,\varepsilon\,(1-\theta) \;=\; \varepsilon\big[(N-1)\theta - j\big] \;=\; -\varepsilon\,(j - j^*),$$

where $j^* = (N-1)\theta$ is the location of the stationary mode. Taking expectations over the per-step transition and using linearity of expectation, the first-moment dynamics are $\mathbb{E}[j_{t+1} - j^*] = (1-\varepsilon)\,\mathbb{E}[j_t - j^*]$, i.e., geometric relaxation toward j^* at rate $\varepsilon = 1/N$. A starting state at distance $\Theta(N)$ from j^* closes that gap to $O(1)$ in $\Theta(N) = \Theta(2^b)$ steps—this is the strong-drift mixing-time bound for this birth-death chain, which follows from the first-moment contraction at rate ε either via the coupling argument of Levin et al. (2017, Chapter 5) or, equivalently, via the spectral-gap calculation of Levin et al. (2017, Chapter 12) (the spectral gap of our chain is exactly $\varepsilon = 1/N$). Near $\theta = 1/2$ the drift coefficient remains ε but the stationary distribution spreads to width $\Theta(\sqrt{N})$ (Eq. 11), so equilibration within the binomial "cluster" takes longer; the worst-case bound deteriorates to $O(N^2)$ in the diffusive limit. For a typical 8-bit Brain-CA Estimator (128 states), the strong-drift regime gives adaptation on the order of hundreds of observations—fast compared to the observation counts over which Bayesian updating with a fixed prior accumulates evidence that resists change. Table 3 confirms the strong-drift scaling empirically: settle times scale by $\approx 4\times$ as b increases by 2 (N doubles), consistent with $O(N)$.

3.3. Switching Experiment

Figure 4 illustrates the adaptive-forgetting mechanism on a concrete example. An 8-bit Brain-CA Estimator ($b = 8$, 128 tracking-regime states) observes a Bernoulli stream with $\theta = 0.8$ for the first 500 observations, then $\theta = 0.2$ for the next 500, alongside four baselines: a step-matched EWMA ($\alpha = 1/128$), a bias-selected EWMA ($\alpha = 1/32$), an 8-bit saturating counter, and a quantized SGD learner ($\eta = 0.002$). All methods use 8-bit state. The simulation was repeated 500 times with independent random seeds.

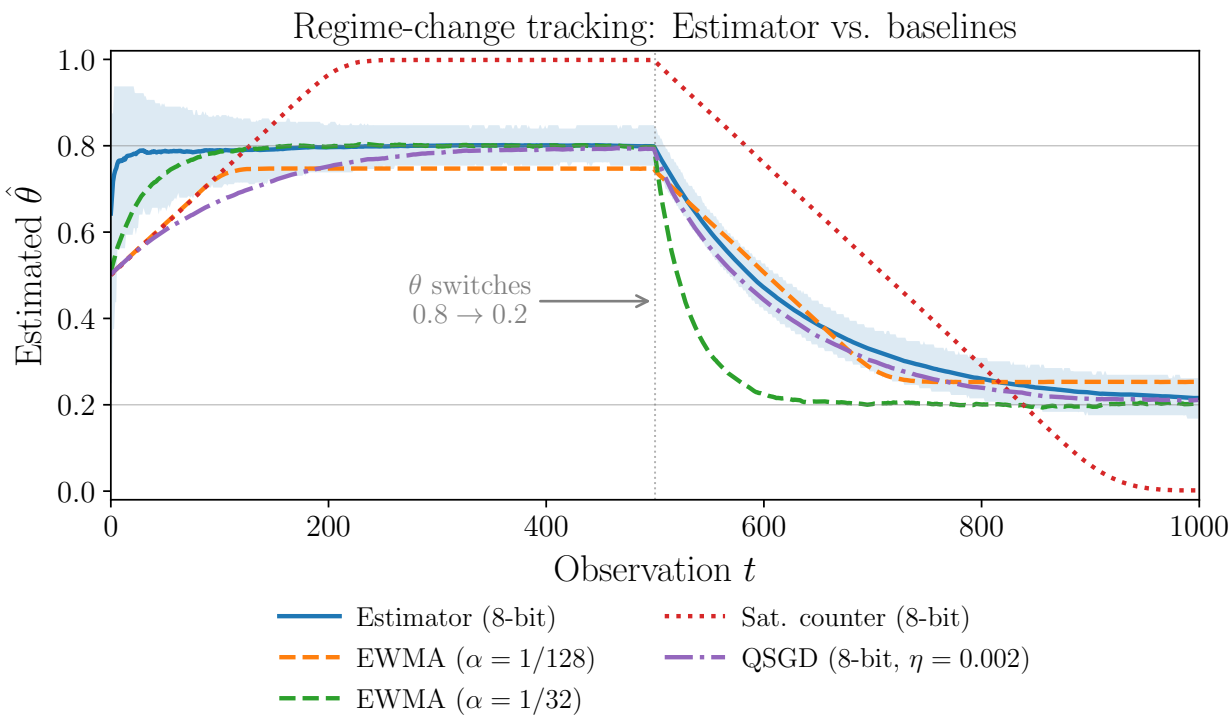

Figure 4 | Regime-change tracking: 8-bit Brain-CA Estimator (blue, with 10th–90th percentile band) vs. four baselines. θ switches $0.8 \to 0.2$ at $t = 500$ (500 runs). The step-matched EWMA ($\alpha = 1/128$, orange dashed) plateaus below 0.8 pre-switch due to rounding-induced bias. The saturating counter (red dotted) overshoots to $\sim$1.0 and adapts slowest post-switch. The bias-selected EWMA ($\alpha = 1/32$, green dashed) adapts fastest but requires a tuned α.

Several features are visible. Pre-switch, the Brain-CA Estimator and QSGD lock onto $\theta = 0.8$ with tight bands; the bias-selected EWMA ($\alpha = 1/32$) also converges to 0.8. In contrast, the step-matched EWMA ($\alpha = 1/128$) plateaus visibly below 0.8 (≈ 0.75): at this small α, the quantized update $\lfloor \alpha(x - p) \rceil$ rounds to zero on the 8-bit grid when p is near the boundary, producing a rounding-induced bias that the Brain-CA Estimator avoids because its effective step size scales with distance from the current estimate. The saturating counter overshoots to $\sim$1.0 because it lacks error-proportional step sizes—every observation nudges by one level regardless of distance from θ.

Post-switch, the Brain-CA Estimator traverses from the neighborhood of $\theta = 0.8$ to the neighborhood of $\theta = 0.2$ in approximately 320 observations. This is consistent with the $O(N) = O(2^b) = O(128)$ mixing-time bound for the strong-drift case: with $\theta_2 = 0.2$, the drift toward lower states is strong, and each nudge moves in the correct direction with high probability. Table 3 strongly supports this scaling across three bit widths. The bias-selected EWMA ($\alpha = 1/32$) adapts fastest but requires a tuned α; the saturating counter adapts much more slowly. The Brain-CA Estimator achieves competitive post-switch tracking without any tunable parameter—its adaptive forgetting (fast when the model is wrong, slow when it is right) emerges from the affine mismatch structure rather than an explicit forgetting rate.

Figure 5 places the Brain-CA Estimator alongside three non-adaptive methods—the running average (MLE), the Bayes–Laplace posterior mean, and the Jeffreys posterior mean—on the same observation sequences. Before the regime change, all four methods converge to $\theta = 0.8$ comparably (the Brain-CA Estimator is slightly slower during growth). After the switch, the difference is

dramatic: the Brain-CA Estimator reaches the neighborhood of $\theta = 0.2$ within approximately 320 observations, while the non-adaptive methods remain near 0.5 at $t = 1000$ (500 post-switch observations). The accumulated 500 observations at $\theta = 0.8$ act as inertia that the new evidence must overcome one observation at a time; at $t = 1000$ the running average is at $k/t \approx 0.5$ by arithmetic necessity (equal observations from each regime), while the Bayesian posterior means are nearby because the 500 pre-switch observations dominate the likelihood, leaving the new data insufficient to shift the posterior appreciably. Measuring MSE as $\mathrm{E}[(p - \theta_2)^2]$ averaged over $t = 800$–1000 and across all 200 runs, the Brain-CA Estimator achieves ≈ 0.002 versus ≈ 0.11 for each of the three non-adaptive methods—roughly a 50-fold difference. However, comparing against non-adaptive methods overstates the Brain-CA Estimator's advantage; Section 3.4 provides a fairer comparison against adaptive EWMA baselines with matched memory budgets.

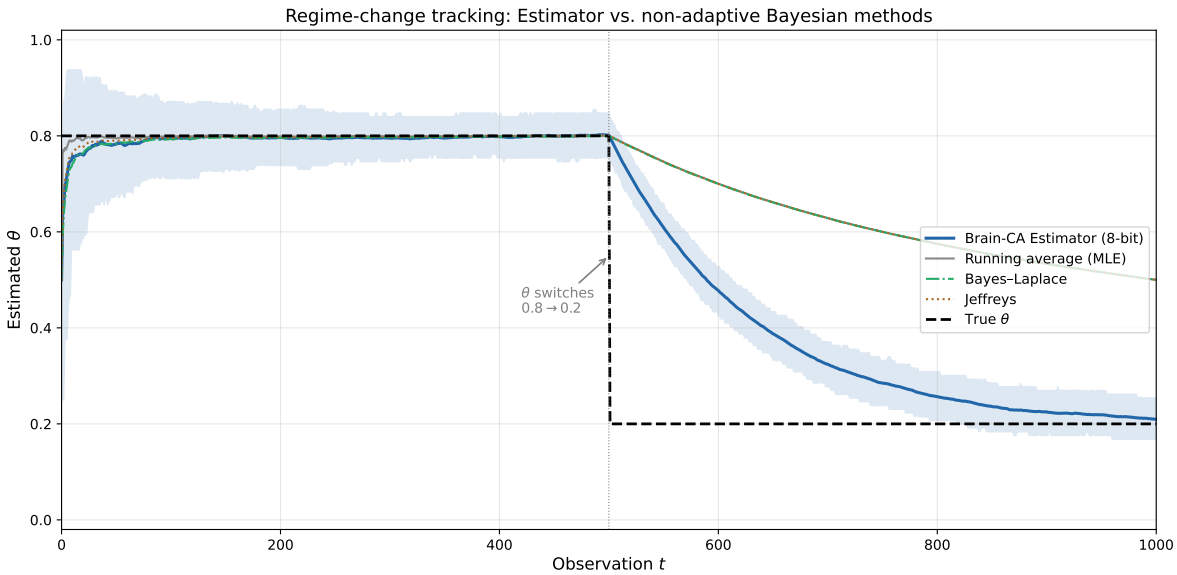

Figure 5 | Regime-change tracking comparison. The 8-bit Brain-CA Estimator (blue, with 10th–90th percentile band) adapts to the switch from $\theta = 0.8$ to $\theta = 0.2$ at $t = 500$ within approximately 320 observations. Three non-adaptive methods are also plotted—the running average (MLE), Bayes–Laplace, and Jeffreys posterior means—but after the regime change their ensemble means are so similar that they appear as a single line near 0.5 at $t = 1000$. All three remain trapped by the inertia of the 500 pre-switch observations, which the new data cannot overcome within the time shown. Same observation sequences and random seeds for all four methods (200 runs).

Table 2 characterizes the adaptation speed across a range of regime-change scenarios at three bit widths ($b = 6, 8, 10$). At each bit width, the midpoint crossing time is consistent across scenarios (within-b variation $< 15\%$), and the settle time depends on the magnitude of the jump and the distance from $1/2$: larger jumps to extreme biases take longer because more of the tracking-regime chain must be traversed.

The data strongly support the $O(2^b)$ mixing-time prediction from birth-death theory. Table 3 reports the θ: $0.8 \to 0.2$ scenario at 500 runs: the settle-time ratios between successive bit widths (3.9 and 3.6) are close to the predicted factor of 4, and the crossing-time ratios (4.2 and 3.1) bracket it. The $6 \to 8$ ratios are nearest the prediction; the $8 \to 10$ ratios fall slightly below 4, likely because the post-switch window (500 observations) is shorter than the 512-state chain's full traversal time. Table 2 provides the full per-scenario breakdown at 200 runs.

Table 2 | Adaptation time across bit widths (θ switches at $t = 500$, 200 runs, simulations run to $t = 2000$). "Cross" = time for ensemble mean to cross $(\theta_1 + \theta_2)/2$; "Settle" = time to enter and remain within ± 0.05 of θ_2; μ_{1500} = ensemble mean at $t = 1500$. Standard errors on ensemble means are below 0.003 at all reported time points.

	$b = 6$ ($N = 32$)			$b = 8$ ($N = 128$)			$b = 10$ ($N = 512$)		
Scenario	Cross	Settle	μ_{1500}	Cross	Settle	μ_{1500}	Cross	Settle	μ_{1500}
$0.8 \to 0.2$	t+22	t+79	0.205	t+86	t+320	0.203	t+281	t+1185	0.273
$0.9 \to 0.1$	t+23	t+87	0.109	t+87	t+352	0.104	t+275	t+1322	0.194
$0.7 \to 0.3$	t+23	t+64	0.308	t+86	t+261	0.302	t+271	t+972	0.347
$0.6 \to 0.4$	t+22	t+43	0.409	t+78	t+165	0.401	t+274	t+608	0.422
$0.9 \to 0.5$	t+21	t+57	0.488	t+85	t+259	0.498	t+269	t+957	0.545
$0.2 \to 0.8$	t+22	t+91	0.789	t+88	t+321	0.799	t+276	t+1180	0.729

Table 3 | Width-scaling (θ: $0.8 \to 0.2$, 500 runs). Cross = midpoint crossing time; Settle = time to reach $\theta_2 \pm 0.05$. These values match the workshop paper's Table II. Table 2 reports a broader set of scenarios at 200 runs.

b	N	Cross	Settle	Cross ratio	Settle ratio
6	32	21	84	—	—
8	128	88	324	4.2	3.9
10	512	274	1165	3.1	3.6

At $b = 10$ ($N = 512$), several scenarios have settle times exceeding 1000 observations, and the mean at $t = 1500$ has not fully converged to θ_2 in the extreme cases (e.g., $0.9 \to 0.1$ reaches 0.194 vs. target 0.100). This is expected: the $O(2^b) = O(1024)$ prediction says the 10-bit Brain-CA Estimator needs $\sim$1000 observations to traverse the full chain, and jump scenarios crossing near $1/2$ require traversing most of the 512-state tracking column. The $b = 10$ settle times for extreme scenarios ($0.9 \to 0.1$, $0.8 \to 0.2$) should therefore be regarded as lower bounds; longer post-switch windows would yield larger values.

3.4. Comparison to Practical Adaptive Methods

The relevant competitors for non-stationary environments are: exponential discounting (Sutton and Barto, 2018) (adds tunable λ, forgets at a fixed rate); change-point models (Adams and MacKay, 2007) (powerful but computationally expensive); and sliding-window estimators (adds tunable W).

To ground the comparison, we benchmark the 8-bit Brain-CA Estimator against four baselines on the same switching Bernoulli process (θ switches from 0.8 to 0.2 at $t = 500$, 500 runs). All methods use 8-bit estimate precision; memory costs and logic complexity are listed separately.

Method	Param	Post-switch MSE ($t = 800$–1000)	Mem	Logic
Brain-CA Estimator (8-bit)	—	0.003	b	$O(b)$
EWMA, step-matched ($\alpha = 1/128$)	$1/128$	0.003	b	$O(b^2)^\dagger$
EWMA, bias-selected ($\alpha = 1/32$)	$1/32$	0.002	b	$O(b^2)^\dagger$
Sat. counter (8-bit)	—	0.021	b	$O(b)$
QSGD (8-bit, $\eta = 0.002$)	$\eta = 0.002$	0.002	b	$O(b^2)$
Non-adaptive (MLE)	—	0.112	$O(\log t)$	$O(b)$

†Multiplier-based EWMA. When $\alpha = 2^{-k}$, EWMA can be implemented as a shift-only update with $O(b)$ logic, but this restricts α to powers of two and still requires an adder per learner. The step-matched and bias-selected α values used here ($1/128$ and $1/32$) are powers of two, so the shift-only path is available at the cost of constraining the parameter; the per-unit FPGA synthesis numbers and architectural trade-offs of both EWMA paths are reported in Section 6 and Table 9.

The step-matched EWMA uses $\alpha = 1/128 = \varepsilon$, matching the Brain-CA Estimator's effective step size $\varepsilon = 2/2^b$. For the bias-selected EWMA, we swept $\alpha \in \{1/16, 1/32, 1/64, 1/128\}$. The sweep results (post-switch MSE / pre-switch stationary bias): $\alpha = 1/16$: MSE 0.004, bias < 0.001; $\alpha = 1/32$: MSE 0.002, bias < 0.001; $\alpha = 1/64$: MSE 0.003, bias 0.02; $\alpha = 1/128$: MSE 0.003, bias 0.05. Only $\alpha = 1/32$ achieves near-zero rounding-induced bias in both the stationary and switching scenarios; smaller values suffer the boundary-rounding failure described in Section 3.3. All values are ensemble averages over 500 runs.

Per-method observations. *Saturating counter.* The fairest hardware comparison: same $O(b)$ logic, zero multiplier, b bits. But it lacks error-proportional step sizes—every observation nudges by one level regardless of distance from θ—so post-switch MSE is $\approx 7\times$ worse (0.021 vs. 0.003).

Quantized SGD. A QSGD-style learner (Alistarh et al., 2017) on Bernoulli negative log-likelihood with b-bit stochastic rounding slightly outperforms the Brain-CA Estimator (MSE 0.002 vs. 0.003) at ex-post-selected $\eta = 0.002$, but requires a division ($O(b^2)$ logic) and a tuned learning rate.

EWMA. At 8-bit precision, the step-matched EWMA ($\alpha = 1/128$) ties the Brain-CA Estimator at MSE 0.003, but exhibits a rounding-induced bias visible in Figure 4: at this small α, the quantized update $\lfloor \alpha(x - p) \rceil$ rounds to zero on the 8-bit grid when p is near the boundary, so the EWMA converges to ≈ 0.75 instead of 0.8. The Brain-CA Estimator avoids this failure mode because its effective step size scales with distance from the current estimate. The bias-selected EWMA ($\alpha = 1/32$) achieves MSE 0.002 with minimal bias, but requires a multiplier and a chosen α.

No method in the table matches the Brain-CA Estimator's combination of competitive MSE (within $1.5\times$ of the best tuned baseline), zero tunable parameters, and $O(b)$ logic. The nearest hardware-fair competitor (the saturating counter) is $7\times$ worse; methods that match or beat the Brain-CA Estimator on MSE all require multipliers and explicit step-size choices.

Broader comparison landscape. More powerful adaptive methods exist but occupy a different design point. Particle filters and sequential Monte Carlo methods can track non-stationary distributions with near-optimal accuracy, but require $O(N)$ particles with $O(N)$ memory and computation per step—orders of magnitude beyond the Brain-CA Estimator's b bits and $O(b)$ operations. Change-point models (Adams and MacKay, 2007) achieve fast adaptation but require computing run-length posteriors, which again exceeds the Brain-CA Estimator's memory budget. The Brain-CA Estimator is not designed to compete with these methods on statistical performance;

it targets the regime where computation and memory are severely constrained (e.g., per-unit budgets of 8–16 bits in a massively parallel fabric).

Thompson Sampling with a Beta prior is the natural Bayesian comparator for sequential binary prediction. It uses $O(\log t)$ bits (two counters), adapts naturally to non-stationarity when combined with discounting, and has known $O(\sqrt{T \log T})$ regret bounds. Its posterior update is trivial (increment two counters), but each prediction requires drawing a random sample from a Beta distribution—an operation requiring either a general-purpose ALU or a dedicated random-variate generator. In a fixed-function hardware fabric, either implementation substantially exceeds the Brain-CA Estimator's bit-serial comparator in gate count. Thompson Sampling is the right method when a single predictor has access to such hardware; the Brain-CA Estimator targets the regime where thousands of predictors must operate in parallel on a fixed-function fabric with $O(b)$ gates per unit.

The Brain-CA Estimator performs sequential binary prediction, a setting studied extensively in online learning. Section 5 provides asymptotic regret estimates.

The entire analysis assumes i.i.d. Bernoulli observations within each regime. Section 3.5 extends the switching experiments to a continuously drifting $\theta(t)$, demonstrating that the adaptive-forgetting mechanism handles gradual non-stationarity as well as abrupt change-points. Under serial correlation (e.g., Markovian bit streams), the self-correction mechanism still operates—each observation still triggers a comparison and potential nudge—but the effective adaptation rate and stationary distribution will differ from the i.i.d. predictions. Characterizing performance under correlated streams is an open direction.

3.5. Continuous Drift

The switching experiments above use abrupt change-points. In many real-world settings, the underlying distribution drifts continuously. To test the Brain-CA Estimator's behavior under gradual non-stationarity, we simulate a sinusoidal drift:

$$\theta(t) = 0.5 + 0.3 \sin(2\pi t/2000), \tag{13}$$

so the true bias sweeps smoothly between 0.2 and 0.8 with period 2000. The maximum drift rate is $|d\theta/dt|_{\max} = 0.3 \cdot 2\pi/2000 \approx 9.4 \times 10^{-4}$ per observation—slow relative to the 8-bit Brain-CA Estimator's $O(2^b) = O(128)$ adaptation timescale. We run 500 simulations over 4000 observations (two full cycles), comparing the same four methods as in Figure 4: the 8-bit Brain-CA Estimator, EWMA ($\alpha = 1/128$ and $\alpha = 1/32$), and the 8-bit saturating counter. All methods use 8-bit state.

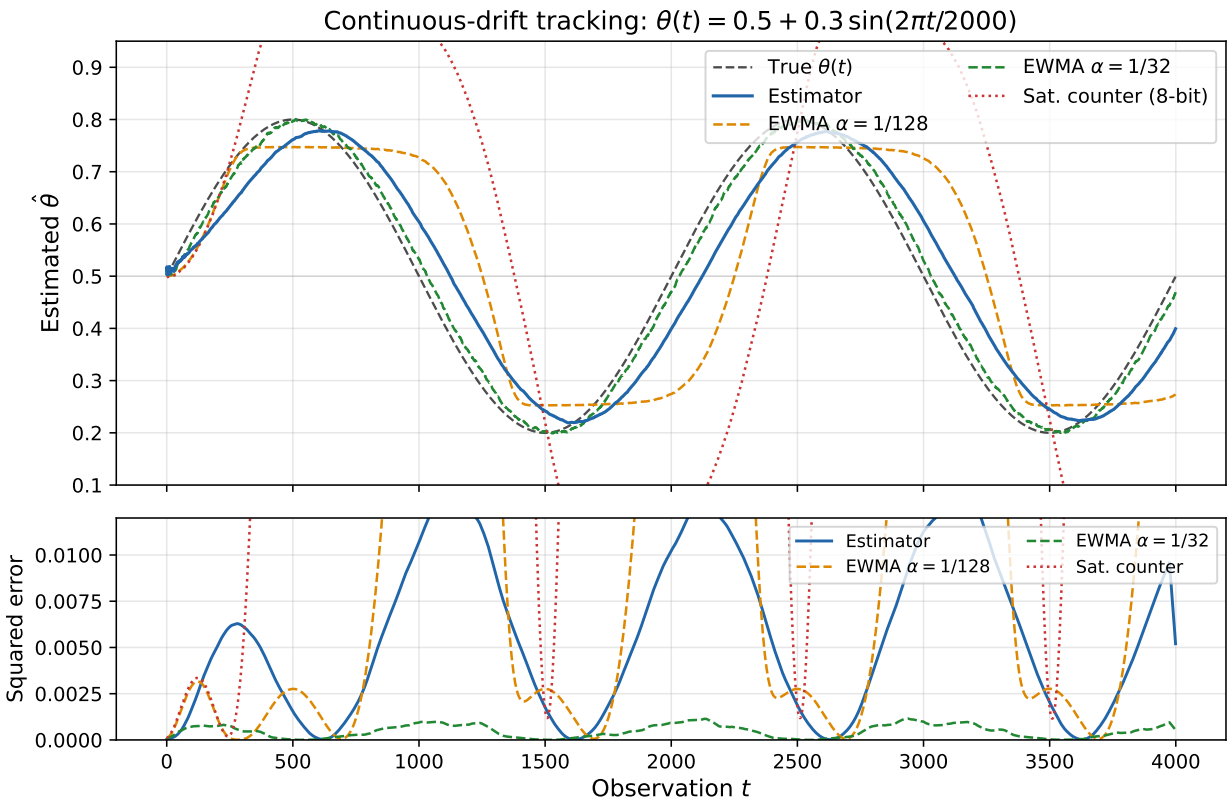

Figure 6 | Continuous-drift tracking: $\theta(t) = 0.5 + 0.3\sin(2\pi t/2000)$, 8-bit, 500 runs. Top: ensemble means (Brain-CA Estimator blue band: 10th–90th percentile). Bottom: smoothed squared error of each method's ensemble mean (50-observation rolling window). The Brain-CA Estimator tracks with a visible phase lag but follows the sinusoidal shape; the step-matched EWMA ($\alpha = 1/128$) exhibits severe lag and rounding-induced bias at the extremes.

Figure 6 shows the results. Table 4 reports time-averaged MSE over the second cycle ($t = 2000$–4000), after all methods have reached their periodic steady state.

Table 4 | Continuous-drift MSE (second cycle, $t = 2000$–4000, 500 runs). $\theta(t) = 0.5 + 0.3\sin(2\pi t/2000)$.

Method	Param	MSE
Brain-CA Estimator (8-bit)	—	0.0077
EWMA, step-matched ($\alpha = 1/128$)	$1/128$	0.0255
EWMA, bias-selected ($\alpha = 1/32$)	$1/32$	0.0036
QSGD (8-bit)	$\eta = 0.002$	0.033
Sat. counter (8-bit)	—	0.136

QSGD uses switching-tuned η; drift-optimal η would differ.

Three observations emerge. First, the Brain-CA Estimator tracks the continuous drift competently despite having no explicit notion of a drift rate. Its MSE (0.0077) is 3.3× lower than the step-matched EWMA (0.0255) and 18× lower than the saturating counter (0.136). The step-matched EWMA's poor performance under drift is striking: its small α produces both a severe phase lag and the same rounding-induced bias visible in the switching experiment (Figure 4), here manifesting as a failure to track the extremes of the sinusoid. The saturating counter, lacking error-proportional

steps, overshoots wildly and lags by nearly a quarter-period. QSGD ($\eta = 0.002$, the switching-optimal value from Table 4) degrades to 0.033—4.3× worse than Brain-CA Estimator—illustrating how a parameter tuned for one regime can fail in another.

Second, the bias-selected EWMA ($\alpha = 1/32$) outperforms the Brain-CA Estimator (0.0036 vs. 0.0077, a 2.1× advantage). This is expected: $\alpha = 1/32$ happens to match the drift rate well, producing a nearly optimal bias–variance tradeoff for this particular sinusoid. A different drift rate would require a different α. The Brain-CA Estimator's error-proportional step size provides a single parameter-free operating point that is competitive across both the switching and continuous-drift regimes without tuning.

Third, the bottom panel of Figure 6 reveals that the Brain-CA Estimator's tracking error is periodic and peaks where the drift rate is highest—near the $\theta = 0.5$ crossings, where $|d\theta/dt|$ is maximal. Near the extremes ($\theta \approx 0.2$ and 0.8), where the drift rate approaches zero, the Brain-CA Estimator tracks tightly. This is consistent with the birth-death analysis: when θ is nearly stationary, the Brain-CA Estimator concentrates near the true value; when θ moves faster than the chain's adaptation timescale, a phase lag develops.

4. Two-Stream Composition

A single Brain-CA Estimator tracks $P(X = 1)$ for one binary stream. The natural question is whether Brain-CA Estimator units compose to represent joint and conditional distributions. We give a minimal example; a full analysis is outside this paper's scope.

Consider two binary streams A and B. The system allocates three Brain-CA Estimator units:

- E_A: tracks $P(A = 1)$ from all observations of A.
- $E_{B|A=1}$: tracks $P(B = 1 \mid A = 1)$, receiving observations of B only when the corresponding observation of A is 1.
- $E_{B|A=0}$: tracks $P(B = 1 \mid A = 0)$, receiving observations of B only when the corresponding observation of A is 0.

The marginal $P(B = 1)$ is recovered by $\hat{p}_A \cdot \hat{p}_{B|A=1} + (1 - \hat{p}_A) \cdot \hat{p}_{B|A=0}$, requiring one multiplication at readout. The joint distribution $P(A, B)$ is fully specified by the three Brain-CA Estimator units. Each Brain-CA Estimator operates independently via the update algorithm; routing logic directs each observation to the appropriate conditional unit.

This decomposition scales combinatorially: k binary streams require up to $O(2^k)$ conditional Brain-CA Estimator units for a full joint distribution, the same scaling as any tabular method. In practice, selective allocation—deploying Brain-CA Estimator units only for conditional relationships that show predictive value, and recycling those that do not—can manage this barrier (see the future-work discussion at the end of this section).

Two properties of single-stream Brain-CA Estimator units carry over to the composed setting without modification: each conditional Brain-CA Estimator independently exhibits the self-correction mechanism (Section 2) and the linear-shrinkage structure (Appendix C) with respect to its conditional stream. However, the equilibrium bias toward 1/2 discussed in Section 2 ($|p^* - \theta| < 1/2^b$) applies independently to each Brain-CA Estimator.

For the two-stream case, a first-order bias bound is tractable. Let $p_A = P(A{=}1)$, $\beta_1 = P(B{=}1 \mid A{=}1)$, $\beta_0 = P(B{=}1 \mid A{=}0)$. (We use p_A rather than α for the marginal to avoid overloading

with the EWMA step size α used elsewhere.) Each Brain-CA Estimator's equilibrium is biased toward $1/2$ by $\delta = -(2\theta-1)/2^b$ (from Eq. 10), with $|\delta| \leq 1/2^b$. Write $\hat{p}_A = p_A + \delta_a$, $\hat{\beta}_1 = \beta_1 + \delta_1$, $\hat{\beta}_0 = \beta_0 + \delta_0$. Since the three Brain-CA Estimator units observe conditionally independent streams, their equilibrium biases are independent. Expanding the readout:

$$\hat{p}_A\,\hat{\beta}_1 + (1-\hat{p}_A)\,\hat{\beta}_0 = P(B{=}1) + \underbrace{p_A\,\delta_1}_{\leq 1/2^b} + \underbrace{(1-p_A)\,\delta_0}_{\leq 1/2^b} + \underbrace{\delta_a(\beta_1-\beta_0)}_{\leq 1/2^b} + O(1/2^{2b}). \tag{14}$$

Each of the three first-order terms is bounded by $1/2^b$, giving $|\hat{p}_B - P(B{=}1)| \leq 3/2^b + O(1/2^{2b})$. For an 8-bit system this is < 0.024, comparable to the single-Estimator tracking-regime standard deviation.

Simulation. To ground the two-stream sketch, we simulate the full system: three 8-bit Brain-CA Estimator units (E_A, $E_{B|A=1}$, $E_{B|A=0}$) tracking streams with $P(A{=}1) = 0.6$, $P(B{=}1 \mid A{=}1) = 0.8$, $P(B{=}1 \mid A{=}0) = 0.3$, composing the marginal $\hat{p}_B = \hat{p}_A\,\hat{p}_{B|1} + (1-\hat{p}_A)\,\hat{p}_{B|0}$. In steady state (500 runs, $t = 2000$–5000), the composed estimate achieves MSE ≈ 0.0012 with $|\hat{p}_B - P(B{=}1)| < 0.003$ (empirical; measured over the steady-state window), well within the $3/2^b$ bound. When the conditionals swap ($P(B{=}1 \mid A{=}1)$: $0.8 \to 0.3$, $P(B{=}1 \mid A{=}0)$: $0.3 \to 0.8$) at $t = 1000$, the Brain-CA Estimator system re-tracks within approximately 500 observations. A 24-bit non-decomposed EWMA tracking $P(B{=}1)$ directly achieves MSE $\approx$ 0.010 in steady state—it cannot exploit the conditional structure. A 3×8-bit EWMA table using the same decomposition achieves comparable MSE but requires a tuned forgetting rate; at the step-matched rate ($\alpha = \varepsilon = 1/128$), fixed-point quantization artifacts produce individual-component biases up to 0.14 (e.g., EWMA mean 0.74 at true $\theta = 0.6$), compared to the Brain-CA Estimator's component biases below 0.003. The mechanism is quantization rounding: the EWMA computes $(1-\alpha)\,p + \alpha\,x$ in continuous arithmetic and rounds to the nearest grid point. When α is small relative to the grid spacing ε, the rounding step dominates the update, creating systematic attractors at grid points that do not correspond to the true θ. The Brain-CA Estimator avoids this entirely because its nudge mechanism moves by exactly ± 1 grid point, with the grid naturally aligned to the state encoding.

Decomposition vs. primitive. Table 5 includes a composed 3-EWMA baseline ($\alpha = 1/32$) under the same conditional decomposition. We swept $\alpha \in \{1/16, 1/32, 1/64, 1/128\}$ for the composed 3-EWMA; $\alpha = 1/32$ is the only value with near-zero bias in both the stationary and switching scenarios. Conditioning on A helps both learners: MSE drops $\approx 1.6\times$ for EWMA and $\approx 1.5\times$ for the Brain-CA Estimator. But the Brain-CA Estimator outperforms EWMA in every configuration: the composed 3-Estimator has roughly half the MSE of the composed 3-EWMA (0.0012 vs. 0.0023), without a tunable α. In the conditional branches, where each EWMA sees only $\approx 60\%$ of observations, a single fixed α cannot be optimal for all branches simultaneously; the Brain-CA Estimator's error-proportional step size adapts automatically. Notably, the composed Brain-CA Estimator's MSE is nearly identical in the stationary and switching cases (0.0012 vs. 0.0013), indicating that the composition tracks regime changes with minimal observed MSE penalty.

Table 5 | Two-stream $\hat{P}(B{=}1)$: MSE over $t = 1500$–2000 (8-bit, 500 runs). EWMA uses $\alpha = 1/32$, the only value with near-zero bias in both scenarios across an α sweep.

Method	Stationary	Switching
Composed 3-Estimator	0.0012	0.0013
Composed 3-EWMA ($\alpha = 1/32$)	0.0023	0.0025
Single Brain-CA Estimator	0.0018	0.0020
Single EWMA ($\alpha = 1/32$)	0.0036	0.0041

Asymmetric configuration: $P(A{=}1) = 0.95$. The symmetric configuration above has $P(A{=}1) = 0.6$, so both conditional branches receive substantial observation counts. To test robustness under extreme asymmetry, we repeat the experiment with $P(A{=}1) = 0.95$, $P(B{=}1 \mid A{=}1) = 0.8$, $P(B{=}1 \mid A{=}0) = 0.3$ (true $P(B{=}1) = 0.775$; 500 runs, 8-bit). The $B \mid A{=}0$ branch now sees only $\approx$5% of observations—a stress test for conditional tracking.

Table 6 | Asymmetric two-stream MSE ($P(A{=}1) = 0.95$, $t = 1500$–2000, 500 runs). The $B \mid A{=}0$ branch sees only $\approx$5% of observations.

Method	Stationary	Switching
Composed 3-Estimator	0.0012	0.0016
Composed 3-EWMA ($\alpha = 1/32$)	0.0022	0.0029

Brain-CA Estimator maintains its advantage: roughly half the MSE of EWMA in both stationary and switching cases, despite the $B \mid A{=}0$ branch receiving only $\approx$100 observations over 2000 time steps ($\approx$25 within the $t = 1500$–2000 measurement window). At $\approx$25 observations the sparse branch has not fully reached its 128-state tracking equilibrium (which requires $O(2^b) = O(128)$ observations); it is still in the growth regime, where the Brain-CA Estimator's adaptive precision adjustment provides a coarser but unbiased estimate. The MSE values are stable when the window is extended to $t = 1500$–3000 ($\approx$50 sparse-branch observations), suggesting the composed estimate is robust even before full equilibration of every branch. The Brain-CA Estimator's error-proportional step size is particularly valuable here: the rarely-updated $B \mid A{=}0$ branch adapts its effective forgetting rate to its low observation frequency automatically, while the EWMA's fixed $\alpha = 1/32$ is suboptimal for both branches simultaneously.

Per-branch α sweep (oracle baseline). The framing of Tables 5–6 fixes a single α across all three branches. Table 7 reports an exhaustive sweep over $\alpha \in \{1/16, 1/32, 1/64, 1/128\}^3$ (64 combinations per scenario), alongside an exhaustive single-α sweep and the parameter-free Brain-CA Estimator. Both sweep results are *oracle* values: choosing the winning α requires knowing the truth in advance.

Table 7 | Per-branch α sweep on the supplement's two-stream configurations (8-bit, 500 runs, MSE over $t = 1500$–2000). "Global-best" picks the single α that minimizes composed MSE; "per-branch-best" picks the best $(\alpha_A, \alpha_{B|A=1}, \alpha_{B|A=0})$ triple from 64 combinations. Both are oracle values (truth-aware tuning).

	3-Estimator (param-free)	3-EWMA at 1/32	3-EWMA global-best	3-EWMA per-branch-best
sym, stationary	0.0012	0.0022	0.0005 (1/128)	0.0003 (1/64, 1/128, 1/64)
sym, switching	0.0013	0.0026	0.0018 (1/64)	0.0005 (1/64, 1/64, 1/128)
asym, stationary	0.0012	0.0022	0.0008 (1/64)	0.0008 (1/64, 1/64, 1/32)
asym, switching	0.0016	0.0028	0.0002 (1/128)	0.0002 (1/128, 1/128, 1/128)

Two findings. First, oracle-tuned EWMA—global or per-branch—beats the Brain-CA Estimator on absolute MSE in every scenario, sometimes by a factor of 4–8×. Second, the winning α is often the smallest sweep value 1/128, which the single-stream analysis of Section 3.4 flagged as having pre-switch bias of 0.05. The mechanism is that the readout $\hat{p}_B = \hat{p}_A \hat{\beta}_1 + (1-\hat{p}_A)\hat{\beta}_0$ partially cancels per-branch biases for specific (P_A, β_1, β_0) values, making low-α EWMA's per-branch bias inconsequential at readout. The per-branch behavior is in fact poor: under sym_stationary at $\alpha = (1/128, 1/128, 1/128)$, the component estimates settle at $\hat{p}_A \approx 0.74$ (true 0.60), $\hat{p}_{B|1} \approx 0.75$ (true 0.80), and $\hat{p}_{B|0} \approx 0.25$ (true 0.30); the composed estimate nonetheless lands within 0.02 of the true $P(B{=}1) = 0.60$.

Robustness of the per-branch tuning. Because the cancellation depends on the specific (P_A, β_1, β_0) tuple, a per-branch α chosen for one configuration does not transfer to nearby distributions. Table 8 applies the per-branch-optimal triple from sym_stationary ($\alpha = (1/64,\ 1/128,\ 1/64)$) to four shifted scenarios.

Table 8 | Transferability of per-branch α tuning (8-bit, 500 runs, MSE over $t = 1500$–2000). Triple $\alpha = (1/64,\ 1/128,\ 1/64)$ chosen on $(P_A, \beta_1, \beta_0) = (0.6, 0.8, 0.3)$ and applied, unchanged, to each test scenario. The Brain-CA Estimator uses no parameter and is not retuned across scenarios.

Test scenario	3-EWMA (train-tuned)	3-Estimator
$(P_A, \beta_1, \beta_0) = (0.6,\ 0.8,\ 0.3)$ *(train)*	0.0003	0.0012
$(P_A, \beta_1, \beta_0) = (0.6,\ 0.7,\ 0.4)$	0.0009	0.0011
$(P_A, \beta_1, \beta_0) = (0.6,\ 0.9,\ 0.2)$	0.0104	0.0014
$(P_A, \beta_1, \beta_0) = (0.7,\ 0.8,\ 0.3)$	0.0020	0.0012
$(P_A, \beta_1, \beta_0) = (0.95,\ 0.8,\ 0.3)$	0.0057	0.0012

Where the conditionals shift away from $(0.8, 0.3)$ or the marginal away from 0.6, the EWMA degrades by factors of 3 to 35; the Brain-CA Estimator holds within 0.0011–0.0016 across every test. The interpretation: a tuned EWMA *can* outperform the Brain-CA Estimator when the data distribution is known and stable, but a well-chosen α triple does not generalize, while the Brain-CA Estimator's parameter-free behavior holds across all five configurations. In a deployment where the underlying distribution can drift or is not known in advance, the Brain-CA Estimator's robustness is the relevant property; the EWMA's tunable upside is unrealizable without re-tuning at runtime, which the EWMA itself does not provide.

Whether this linear scaling persists in deeper compositions (e.g., chains of conditional Brain-CA Estimator units for $k > 2$ streams) depends on the readout topology and is an important open

question. Two additional considerations apply to composed systems: (a) the routing logic that directs observations to the correct conditional Brain-CA Estimator scales as $O(2^k)$ for a full k-stream decomposition, which may dominate the per-Estimator gate savings; and (b) each conditional Brain-CA Estimator receives only a subset of the observation stream (e.g., $E_{B|A=1}$ sees B only when $A = 1$), so its effective observation rate depends on the marginal of the conditioning variable. When $P(A{=}1)$ is small, $E_{B|A=1}$ converges slowly. The simulation above shows this does not prevent accurate tracking in the two-stream case, but a formal analysis of the interaction between marginal rates and conditional convergence is not attempted here.

Scaling beyond two streams (future work). The $O(2^k)$ combinatorial barrier for full k-stream conditional decomposition is the central obstacle to scaling this primitive into a practical learning system. Two directions are under investigation: (a) dynamic allocation from a fixed-size pool of Brain-CA Estimator units, where unproductive conditional assignments are recycled and replaced, avoiding exhaustive enumeration; and (b) a query/response protocol that reduces per-unit I/O bandwidth by decomposing high-dimensional inputs into short sequences of compact exchanges. Initial investigations are encouraging but not yet at a stage where we can report quantitative results. Each conditional Brain-CA Estimator in such a system would retain the single-stream properties analyzed in this paper, so the theoretical guarantees on shrinkage, tracking, and adaptive forgetting apply to every unit independently. A full evaluation is deferred to a future submission.

5. Regret Estimates (Supporting Analysis)

This section provides heuristic regret estimates that contextualize the Brain-CA Estimator's tracking behavior within the online-learning framework. These estimates are not among the workshop paper's formal contributions; they are included here for completeness.

5.1. Stationary and Non-Stationary Scaling

The binomial stationary distribution makes it possible to estimate the Brain-CA Estimator's cumulative excess log-loss. The results in this section are asymptotic estimates, not formal theorems; the derivations use second-order Taylor approximations and heuristic regret decompositions validated by simulation.

Per-step excess log-loss. On a Bernoulli(θ) stream, the learner predicts p_s and incurs log-loss $\ell(p_s) = -\theta \ln p_s - (1{-}\theta)\ln(1{-}p_s)$. The oracle who knows θ incurs $\ell(\theta) = H(\theta)$. The per-step excess is $\ell(p_s) - H(\theta) = \mathrm{KL}(\theta \,\|\, p_s)$, where $\mathrm{KL}(\theta \,\|\, p)$ denotes the KL divergence from the true distribution to the prediction (the standard log-loss convention).

In the tracking regime, the expected per-step excess under the stationary distribution is $r(b,\theta) = \mathrm{E}_\pi[\mathrm{KL}(\theta \,\|\, p_s)]$. Expanding $\mathrm{KL}(\theta \,\|\, p)$ to second order around $p = \theta$:

$$\mathrm{KL}(\theta \,\|\, p) \;\approx\; \frac{(p-\theta)^2}{2\,\theta(1{-}\theta)}. \tag{15}$$

Substituting $\mathrm{E}_\pi[(p-\theta)^2] = \mathrm{Var}_\pi[p] + \mathrm{Bias}^2$ with the exact variance $\varepsilon(1{-}\varepsilon)\,\theta(1{-}\theta)$ (Eq. 11) and the exact bias $(2\theta{-}1)/2^b$ (Eq. 10):

$$r(b,\theta) \;\approx\; \frac{\varepsilon(1{-}\varepsilon)}{2} + \frac{(2\theta{-}1)^2}{2\cdot 4^b\,\theta(1{-}\theta)} + O(\varepsilon^3), \tag{16}$$

where $\varepsilon = 2/2^b$. The leading (variance) term $\varepsilon(1-\varepsilon)/2 \approx 1/2^b$ is θ-independent; the second (bias) term is $O(\varepsilon^2)$ and negligible for $b \geq 6$. Exact computation of $\mathrm{E}_\pi[\mathrm{KL}(\theta \parallel p_s)]$ (without Taylor approximation) across $b \in \{4, \ldots, 16\}$ and $\theta \in \{0.3, 0.5, 0.7, 0.9\}$ confirms that Eq. (16) is accurate to within 1.2% at $b = 8$ and 0.1% at $b = 12$.

Stationary regret. Cumulative regret is $R(T) \approx T \cdot r(b, \theta) \approx T/2^b$ for $T \gg 2^b$—linear in T, in contrast to the $O(\log T)$ regret of Bayesian methods with unbounded memory. This is the unavoidable cost of fixed-width tracking: finite memory cannot achieve consistency, and the tracking-regime variance floor (Eq. 11) translates directly into a regret floor. Each additional bit halves the per-step rate.

Non-stationary regret. In a switching environment with S regime changes, the Brain-CA Estimator's total regret decomposes heuristically as

$$R(T; b) \approx S \cdot c \cdot 2^b + T/2^b, \tag{17}$$

where the first term reflects the $O(2^b)$ mixing-time cost per switch (Table 3) and the second is the steady-state tracking cost ($c \approx 1$–3 depending on jump magnitude). This decomposition assumes the steady-state intervals are long enough ($T/S \gg 2^b$) for the tracking cost to dominate within each regime. At the optimal bit width $2^{b^*} = \sqrt{T/(Sc)}$, one has $T/S = c \cdot 2^{2b^*}$, so each interval is only $c \cdot 2^{b^*}$ observations—a constant multiple of the mixing time. In this regime, a non-negligible fraction of each interval is transient, and the constant c absorbs the interaction between transition and tracking costs. The order-of-magnitude result

$$R^* = 2\sqrt{c\,S\,T} = O(\sqrt{S\,T}) \tag{18}$$

is sublinear in T when S is fixed or grows sublinearly. In contrast, non-adaptive Bayesian methods incur $O(T)$ regret after a single switch (Figure 5), since the accumulated pre-switch evidence dominates the posterior.

The bit width b is therefore a hardware-level encoding of the assumed switching rate: a practitioner who expects $\approx S$ switches in T observations should choose $b \approx \frac{1}{2}\log_2(T/S)$. This is a design-time decision, not a run-time parameter.

Comparison. A fixed-point EWMA with step size $\alpha = \varepsilon$ incurs the same $O(T/2^b)$ stationary regret and achieves the same $O(\sqrt{ST})$ switching regret with oracle α, but requires a tunable forgetting rate. The Brain-CA Estimator's bit width b is the only design parameter, set once at fabrication.

Assumptions and limitations. The per-step regret formula (16) assumes the Brain-CA Estimator has reached stationarity (valid for $t \gg 2^b$) and uses a second-order Taylor approximation of KL divergence. The non-stationary decomposition (17) further assumes that inter-switch intervals satisfy $T/S \gg 2^b$ (so the tracking cost dominates) and that the constant c absorbs the transition-phase regret. A formal proof of the $O(\sqrt{ST})$ bound would require bounding the transient contribution explicitly and showing that the Taylor approximation error is $O(\varepsilon^3 T)$; these are within reach given the exact stationary distribution but are not carried out here.

6. Hardware Properties

The Brain-CA Estimator achieves belief updating using only bit comparisons, inversions, and random bits—no multiplication, division, floating-point, or transcendental functions. The Markov state index is the entire state: a fixed number of bits encoding both the probability estimate and an implicit effective sample size. No history or running sums are needed. The update algorithm requires at most $b - 1$ random bits per observation for a b-bit Brain-CA Estimator.

Table 9 reports same-flow FPGA synthesis results (Alveo U200, Vivado, identical synthesis strategy) for the 8-bit Brain-CA Estimator and two EWMA variants: a multiplier-based EWMA permitting arbitrary α, and a shift-only EWMA restricted to $\alpha = 2^{-k}$. The Brain-CA Estimator optimizes to 48 LUTs and 8 registers at $F_{\max} = 96$ MHz in the series configuration, with no multiplier or carry chain. The multiplier-based EWMA synthesizes to 55–71 LUTs at 200–230 MHz across the $\alpha \in \{1/32, 1/128\}$ range; the shift-only EWMA reaches 13–15 LUTs at 235–238 MHz. A 16-bit Brain-CA Estimator scales to 130 LUTs at 100 MHz. Each Brain-CA Estimator update consumes $\leq b - 1$ random bits (per-tile LFSRs suffice); each additional bit of width halves the stationary variance (Eq. 11) but doubles the adaptation time.

6.1. The Increment as Masked XOR

The nudge $s \leftarrow s \pm 1$ is a real arithmetic increment, not a single-bit flip—the standard $+1$ propagates a carry from the LSB, flipping every 1-bit until it reaches the first 0 (which it flips to 1); -1 propagates a borrow, flipping every 0-bit until reaching the first 1 (which it flips to 0). We have written the update this way because that is what the algorithm requires. What the algorithm does *not* require is that the increment be realized by an instantiated carry-propagate adder on the on-cycle datapath.

The reason is that the carry boundary depends only on the cached state and on the SELECT prediction bit, both available off cycle. SELECT (run off cycle, MSB→LSB) yields a prediction bit y at some payload position; the update commits a state change only when $y \neq x$, and because $y \in \{0, 1\}$, the disagreement direction is forced. A prediction of $y = 1$ can only ever be contradicted by $x = 0$, requiring a -1 (LSB-side scan for the first 1, flip LSB through that bit). A prediction of $y = 0$ can only ever be contradicted by $x = 1$, requiring a $+1$ (LSB-side scan for the first 0, flip LSB through that bit). Hence a single LSB-side priority encoder—conditioned on the already-known y to scan for the first 1 or the first 0—produces the one bit-mask the update can possibly need, all of it off-cycle on the cached state in parallel with SELECT. When the observation x arrives, the on-cycle datapath is then: compare x to y; if they agree, hold; if they disagree, XOR the state with the precomputed mask. The critical path contains one comparator and one XOR—no carry chain, no priority encoder, and no propagation whose length scales with the state width. The priority encoder that locates the carry boundary exists, but it runs off cycle and is not on the path between observation arrival and state commit.

This is the architectural content of the workshop's "no adder, no carry chain" claim: the on-cycle update path contains no instantiated arithmetic primitive. We do not claim that the particular FPGA implementation behind Table 9 uses exactly this priority-encoder / masked-XOR structure—that implementation was not explicitly written in this form, and a synthesizer fed a naive `s+1` / `s-1` RTL expression will typically produce LUT logic that is functionally equivalent but laid out at the synthesizer's discretion. What the synthesis report *does* confirm is that the 8-bit Brain-CA Estimator maps to 48 LUTs and 8 registers with no CARRY-chain primitive instantiated, consistent with the synthesizer folding the ± 1 into pure LUT logic. The broader point is that the algorithm permits the

masked-XOR implementation: a custom-RTL or ASIC realization can place the priority encoders explicitly off cycle and reduce the on-cycle path to comparator-plus-XOR, whereas an FPGA designer relying on `s+1` in RTL leaves that decision to the synthesizer. This separation between off-cycle preparation (priority encoders, mask generation, random-bit fanout) and on-cycle commit (one comparator, one masked XOR) is the property that lets a parallel fabric of Brain-CA Estimators operate without the per-tile arithmetic block that a per-update adder would otherwise force.

Per-unit gate count is not the right metric at scale. On a per-unit FPGA basis, the LUT differences between the Brain-CA Estimator and EWMA are modest, and shift-only EWMA is in fact smaller. The relevant architectural argument is therefore not gate count per learner; it is how the design behaves when many learners are instantiated together.

Any arithmetic operation—multiply or add—tends to push a massively parallel system toward a centralized math unit, with data shuttled between that unit and the memory holding each learner's state. At thousands or millions of learners per die, this von Neumann data-movement bottleneck becomes the binding constraint, not the per-unit gate count. The Brain-CA Estimator's update—bit comparisons and a conditional ± 1 nudge—executes in place where the state is stored, eliminating data movement between memory and arithmetic logic and making compute-in-memory implementation practical without a per-unit ALU. Shift-only EWMA, while smaller per unit, still requires an adder on each update and a tuned α, and still inherits the small-α rounding-bias failure mode shown in Figure 4 and analyzed in Section 3.3. The Brain-CA Estimator's $F_{\max}$ of 96 MHz is lower than the EWMA variants' 200–238 MHz; in a parallel fabric where many learners share a clock, this difference is unlikely to be the binding factor for a memory-resident learning fabric, but we report it for completeness.

Back-of-envelope at fabric scale. To make the data-movement argument concrete, consider a fabric of N learners on a die, each accepting one observation per update cycle at frequency f. The arithmetic-based EWMA admits two implementations. *(i) One ALU per learner:* each learner carries its own multiplier (or adder, for shift-only EWMA), multiplying the per-unit cost by the ALU footprint and adding wires to deliver each new observation. *(ii) Shared ALU bank:* each learner's b-bit state is routed to a centralized arithmetic block, the new observation is delivered, the arithmetic is performed, and the updated state is routed back. Option (ii) requires routing roughly $2b$ bits per learner per update across the on-die interconnect (state out, state back). At $N = 10^6$ learners, $b = 8$, and $f = 1\,\text{GHz}$, the aggregate state-shuttling traffic is

$$2bNf \;=\; (2)(8)(10^6)(10^9) \;=\; 1.6 \times 10^{16}\,\text{bit/s} \;=\; 2 \times 10^{15}\,\text{byte/s} \;=\; 2\,\text{PB/s},$$

on the on-die interconnect—vastly beyond the bandwidth of any practical fabric, forcing the architect either toward option (i)'s area cost or toward many smaller ALU clusters whose own routing overhead grows with N. These numbers are order-of-magnitude estimates; the precise figures depend on packing, clock domain, and routing topology, but the qualitative scale is inherent: any per-update arithmetic operation forces $\Theta(bN)$ bits of state to cross wires per cycle. The Brain-CA Estimator pays neither cost: the comparison happens at the storage cell, no state crosses a wire, and the only per-update I/O is the random bits the SELECT step consumes ($\leq b - 1$ bits per update). These can be supplied by a small grid of LFSRs distributed across the die: a maximal-length 32-bit Galois LFSR has period $2^{32} - 1$ and produces uniformly distributed k-tuples for any $k \leq 32$ (a property of maximal-length linear-feedback shift registers), so adjacent tiles can tap successive bits of a shared stream without practical loss. An empirical check confirms this: running

an 8-bit Brain-CA Estimator driven by independent random bits versus by successive bits of a 32-bit maximal-length Galois LFSR (200 runs, 2000 steady-state steps each, $\theta \in \{0.3, 0.5, 0.7, 0.8, 0.9\}$), the steady-state ensemble means agree to within 0.005 and the MSEs agree to the fourth significant figure across all five biases. Random-bit delivery is therefore local-mesh traffic, not all-to-ALU traffic; tile-level models of LFSR placement, routing, and shared-stream collisions across many tiles are future work. This is the architectural distinction the per-unit LUT count obscures: not "fewer gates per learner," but "no state shuttling, period."

Table 9 | Implementation profile, same-flow FPGA synthesis (Alveo U200, Vivado, UltraScale+, identical synthesis strategy across all variants).

Property	Brain-CA Estimator	EWMA (mult)	EWMA (shift)
State bits	b	$b^{\ddagger}$	$b^{\ddagger}$
Multipliers	0	1 ($b \times b$)	0
Adders / arithmetic	0	1 (add+mult)	1 (add+shift)
Logic (schematic)	$O(b)$	$O(b^2)$	$O(b)$
Run-time params	0	1 (α)	1 ($\alpha = 2^{-k}$)
Random bits/update	$\leq b - 1$	0	0
Data movement/update	none	state $\leftrightarrow$ ALU	state $\leftrightarrow$ ALU
8-bit synthesis (same Vivado flow):			
Update path LUTs	48*	55–71§	13–15§
Update path registers	8	12–13	8
$F_{\max}$ (series)	96 MHz	200–230 MHz	235–238 MHz
16-bit synthesis (Brain-CA Estimator only):			
Update path LUTs	130	—	—
$F_{\max}$ (series)	100 MHz	—	—

*Series config; 60 LUTs pre-opt. §Range over $\alpha \in \{1/32, 1/128\}$. ‡$\alpha$ shared, not per instance.

A. State Table (States 1–255)

Table 10 | Brain-CA Estimator states 1–255 (through 8 bits).

State	Binary	Payload	P_1	Decimal
1	`1`	(none)	1/2	0.500
2	`10`	0	1/4	0.250
3	`11`	1	3/4	0.750
4	`100`	00	1/8	0.125
5	`101`	01	3/8	0.375
6	`110`	10	5/8	0.625
7	`111`	11	7/8	0.875
8–15	...	3-bit	odd/16	0.063–0.938
16–31	...	4-bit	odd/32	0.031–0.969
32–63	...	5-bit	odd/64	0.016–0.984
64–127	...	6-bit	odd/128	0.008–0.992
128–255	...	7-bit	odd/256	0.004–0.996

General formula: $P_1(s) = (2(s - 2^m) + 1)/2^{m+1}$, where $m = \lfloor \log_2 s \rfloor$.

B. Exact Enumeration Methodology

All-ones stream. Reachable states form a chain $\{2^k - 1 : k \geq 2\}$. The distribution evolves by $p_k(t+1) = p_k(t) \cdot s_k + p_{k-1}(t) \cdot e_{k-1}$ with $s_k = (2^{k-1} - 1)/2^{k-1}$ and $e_k = 1/2^{k-1}$, computed in exact rational arithmetic.

General stream. Every reachable state branches into successors (stay, nudge up/down, extend-with-0/1) with state-dependent probabilities weighted by θ and $(1 - \theta)$. Exact computation is feasible for $t \leq 10$.

Data-averaged computation. The general-stream enumerator already marginalizes over observation sequences by weighting branches. The output $\mathrm{E}[p]$ is the data-averaged expected value directly.

The tool is implemented in Python using `fractions.Fraction` for exact rational arithmetic. Source code and simulation scripts are available upon request.

Supplementary mixed-stream data. Table 11 reports data-averaged expected values for $\theta \in \{0.6,\ 0.8,\ 0.9\}$ comparing the Brain-CA Estimator, Jeffreys and Bayes-Laplace.

Table 11 | Supplementary data-averaged expected values.

	$\theta = 0.6$			$\theta = 0.8$			$\theta = 0.9$		
t	B-CA	Jeff	BL	B-CA	Jeff	BL	B-CA	Jeff	BL
1	.5500	.5500	.5333	.6500	.6500	.6000	.7000	.7000	.6333
2	.5625	.5667	.5500	.6875	.7000	.6500	.7500	.7667	.7000
3	.5703	.5750	.5600	.7109	.7250	.6800	.7812	.8000	.7400
4	.5755	.5800	.5667	.7255	.7400	.7000	.8023	.8200	.7667
5	.5791	.5833	.5714	.7369	.7500	.7143	.8163	.8333	.7857
6	.5818	.5857	.5750	.7448	.7571	.7250	.8268	.8429	.8000
7	.5839	.5875	.5778	.7517	.7625	.7333	.8350	.8500	.8111
8	.5855	.5889	.5800	.7564	.7667	.7400	.8418	.8556	.8200

C. Proof of Theorem 7 (Linearity)

This appendix proves Theorem 7 from the main text: the data-averaged expected value of the Brain-CA Estimator at every observation count t takes the form

$$\mathbb{E}_\theta[\text{Brain-CA Estimator} \mid t \text{ obs}] \;=\; a_t\,\theta \;+\; \frac{1 - a_t}{2}, \tag{19}$$

for a sequence a_t depending only on t. The proof is purely algebraic: it establishes properties **(P1)** and **(P2)** from Section 2 for all payload depths $m \geq 1$, then closes the inductive argument sketched there. All identities have been verified by exact rational-arithmetic computation through $m = 10$ ($2^{10} = 1024$ states at the deepest level) and $t = 8$ observations.

Notation. Throughout, $\delta_m = 2^{-m}$ denotes the fall-through probability at depth m, which also equals the grid spacing $2/2^{m+1}$ of the encoded fractions at that depth. We write $p_s = P_1(s) = (2(s - 2^m) + 1)/2^{m+1}$ for the encoded fraction at state s with $m = \lfloor \log_2 s \rfloor$ payload bits.

C.1. Mismatch Probabilities Depend Only on p_s and m

Lemma 3 (Affine mismatch structure)**.** *At every state s with m payload bits:*

$$\hat{P}_1(s) = p_s - \tfrac{1}{2}\delta_m, \tag{20}$$

$$\hat{P}_0(s) = (1 - p_s) - \tfrac{1}{2}\delta_m, \tag{21}$$

$$P_f(s) = \delta_m. \tag{22}$$

In particular, $\hat{P}_1(s) + \hat{P}_0(s) + P_f(s) = 1$, and the mismatch probabilities depend on s only through the encoded fraction p_s and the depth m.

Proof. Identity (22) is immediate: the fall-through probability is 2^{-m} for any state with m payload bits. For (20), the main text (Section 2.3) establishes $P_1(s) = \hat{P}_1(s) + P_f(s)/2$, giving $\hat{P}_1(s) = P_1(s) - \delta_m/2 = p_s - \delta_m/2$. Then $\hat{P}_0(s) = 1 - \hat{P}_1(s) - P_f(s) = 1 - (p_s - \delta_m/2) - \delta_m = (1 - p_s) - \delta_m/2$.

Note that $\hat{P}_1(s) \geq 0$ with equality iff $p_s = \delta_m/2 = 1/2^{m+1}$ (the minimum-fraction state, all payload bits 0), and $\hat{P}_0(s) \geq 0$ with equality iff $p_s = 1 - 1/2^{m+1}$ (the maximum-fraction state, all payload bits 1). Therefore, the "nudge up" transition (which requires $\hat{P}_0(s) > 0$) never targets a state above the maximum, and "nudge down" (requiring $\hat{P}_1(s) > 0$) never targets a state below the minimum. Nudges stay within the depth-m column for all states. □

C.2. Within-Depth Expected Transitions Are Affine in p_s

At depth m, consecutive states differ in encoded fraction by $\delta_m = 1/2^m$. Extensions from depth $m - 1$ to depth m produce states whose encoded fractions are $p_s \pm \delta_m/2$ (adding a 1 or 0 as the new LSB).

Lemma 4 (Affine conditional expectations)**.** *Define $R_m^x(s)$ as the expected next-state encoded fraction contributed to depth m by a state s at depth m, given observation x (i.e., the sum over stay and nudge outcomes only, excluding the fall-through contribution which sends mass to depth $m + 1$). Then:*

$$R_m^1(s) = (1 - 2\delta_m)\, p_s \;+\; \delta_m\left(1 - \tfrac{\delta_m}{2}\right), \tag{23}$$

$$R_m^0(s) = (1 - 2\delta_m)\, p_s \;+\; \tfrac{\delta_m^2}{2}. \tag{24}$$

Both are affine in p_s with slope $(1 - 2\delta_m)$ and depth-dependent intercepts.

Proof. Given observation $x = 1$ at state s (depth m), the transitions that remain at depth m are: predict 1 correctly (stay, probability $\hat{P}_1(s)$, fraction stays p_s) and predict 0 incorrectly (nudge up,

probability $\hat{P}_0(s)$, fraction becomes $p_s + \delta_m$). So:

$$\begin{aligned}
R^1_m(s) &= \hat{P}_1(s) \cdot p_s \;+\; \hat{P}_0(s) \cdot (p_s + \delta_m) && (25)\\
&= \big(\hat{P}_1(s) + \hat{P}_0(s)\big)\, p_s \;+\; \hat{P}_0(s)\, \delta_m && (26)\\
&= (1-\delta_m)\, p_s \;+\; \big[(1-p_s) - \tfrac{\delta_m}{2}\big]\, \delta_m && (27)\\
&= (1-\delta_m)\, p_s \;+\; (1-p_s)\, \delta_m - \tfrac{\delta_m^2}{2} && (28)\\
&= p_s - 2\delta_m\, p_s + \delta_m - \tfrac{\delta_m^2}{2} && (29)\\
&= (1-2\delta_m)\, p_s + \delta_m(1 - \tfrac{\delta_m}{2}). && (30)
\end{aligned}$$

Similarly, given $x = 0$: predict 0 correctly (stay, $\hat{P}_0(s)$, fraction p_s) and predict 1 incorrectly (nudge down, $\hat{P}_1(s)$, fraction $p_s - \delta_m$):

$$\begin{aligned}
R^0_m(s) &= \hat{P}_0(s) \cdot p_s \;+\; \hat{P}_1(s) \cdot (p_s - \delta_m) && (31)\\
&= (1-\delta_m)\, p_s \;-\; \hat{P}_1(s)\, \delta_m && (32)\\
&= (1-\delta_m)\, p_s \;-\; (p_s - \tfrac{\delta_m}{2})\, \delta_m && (33)\\
&= (1-2\delta_m)\, p_s + \tfrac{\delta_m^2}{2}. \square && (34)
\end{aligned}$$

Remark 5 (Properties **(P1)** and **(P2)** follow immediately)**.** *The main text defines $q^x_s = \mathbb{E}[p_{s'} \mid$ state s, obs $x]$ as the expected next-state fraction including both within-depth transitions and extensions. From Lemma 4 and the extension formulas:*

- ***(P1)*** $q^1_s - q^0_s = \big(R^1_m(s) - R^0_m(s)\big) + P_f\big[(p_s + \frac{\delta_m}{2}) - (p_s - \frac{\delta_m}{2})\big] = (\delta_m - \delta_m^2) + \delta_m^2 = \delta_m$. *This depends only on m, confirming **(P1)**.*
- ***(P2)*** $q^0_s = R^0_m(s) + \delta_m(p_s - \frac{\delta_m}{2}) = (1-2\delta_m)p_s + \frac{\delta_m^2}{2} + \delta_m\, p_s - \frac{\delta_m^2}{2} = (1-\delta_m)\, p_s$, *which is affine in p_s with depth-only coefficients, confirming **(P2)**.*

These two properties were verified computationally in the main text through $m = 8$. Lemmas 3 and 4 establish them algebraically for all m.

C.3. Depth-Mass Independence from θ

Lemma 6 (Depth masses are θ-invariant)**.** *Define $W_m(t) = \sum_{s \in D_m} \pi_t(s;\theta)$, the total probability mass at depth m after t observations. Then $W_m(t)$ does not depend on θ.*

Proof. Whether a state at depth m stays at depth m or extends to depth $m+1$ is determined by the SELECT step: mismatch (probability $1 - \delta_m$, stay at depth m) versus fall-through (probability δ_m, extend to depth $m+1$). These probabilities depend only on m, not on the observation x or the stream bias θ: the mismatch/fall-through outcome is determined entirely by whether the m random bits match the m storage bits, which occurs with probability 2^{-m} regardless of what observation arrives. The observation determines *which* state within a depth the Brain-CA Estimator moves to (nudge direction, extension target), but not *whether* it stays at the current depth or extends.

Formally, the depth-mass recurrence is:

$$W_m(t+1) = (1-\delta_m)\, W_m(t) \;+\; \delta_{m-1}\, W_{m-1}(t), \qquad m \geq 1, \tag{35}$$

with $W_0(t+1) = (1-\delta_0)W_0(t) = 0$ for $t \geq 1$ (since $\delta_0 = 1$ and the initial state $s = 1$ has zero payload bits). The initial condition is $W_0(0) = 1$, $W_m(0) = 0$ for $m \geq 1$. No coefficient depends on θ. $\square$

C.4. Inductive Proof of Linearity

Theorem 7 (Linearity)**.** *For every observation count $t \geq 0$,*

$$\mathbb{E}_\theta[p_t] \;=\; a_t\,\theta + \frac{1-a_t}{2} \tag{36}$$

for some a_t depending only on t.

Proof. We prove a stronger statement by induction on t.

Inductive hypothesis $H(t)$: For each depth $m \geq 0$, the *depth-resolved first moment*

$$M_m(t;\theta) \;:=\; \sum_{s\in D_m} \pi_t(s;\theta)\, p_s \tag{37}$$

is an affine function of θ (i.e., $M_m(t;\theta) = \alpha_m^{(t)}\theta + \beta_m^{(t)}$ for constants $\alpha_m^{(t)}, \beta_m^{(t)}$ that depend on m and t but not on θ).

Base case ($t = 0$). All mass is at state $s = 1$ (depth 0, $p_1 = 1/2$). $M_0(0;\theta) = 1/2$, $M_m(0;\theta) = 0$ for $m \geq 1$. All constant (hence affine) in θ.

Inductive step. Assume $H(t)$ holds. The one-step evolution gives:

$$\begin{aligned} M_m(t{+}1;\theta) = \theta &\left[\sum_{s\in D_m} \pi_t(s;\theta)\, R_m^1(s) \;+\; \sum_{s\in D_{m-1}} \pi_t(s;\theta)\, E_m^1(s)\right] \\ &+\; (1{-}\theta)\left[\sum_{s\in D_m} \pi_t(s;\theta)\, R_m^0(s) \;+\; \sum_{s\in D_{m-1}} \pi_t(s;\theta)\, E_m^0(s)\right], \end{aligned} \tag{38}$$

where $R_m^x(s)$ is the within-depth contribution (Lemma 4) and $E_m^x(s)$ is the extension contribution from depth $m{-}1$:

$$E_m^1(s) = \delta_{m-1}\big(p_s + \tfrac{\delta_m}{2}\big), \qquad E_m^0(s) = \delta_{m-1}\big(p_s - \tfrac{\delta_m}{2}\big). \tag{39}$$

(At depth $m{-}1$, fall-through probability is $\delta_{m-1} = 2\delta_m$; extending with observation 1 gives fraction $p_s + \delta_m/2$, and with 0 gives $p_s - \delta_m/2$.)

Since $R_m^x(s)$ and $E_m^x(s)$ are both affine in p_s with depth-only coefficients (Lemma 4 and the extension formulas above), the sums over states reduce to linear combinations of $M_m(t;\theta)$, $W_m(t)$, $M_{m-1}(t;\theta)$, and $W_{m-1}(t)$.

Substituting from (23)–(24) and collecting terms:

$$\begin{aligned} M_m(t{+}1;\theta) = (1-2\delta_m)\, M_m(t;\theta) \;&+\; 2\delta_m\, M_{m-1}(t;\theta) \\ &+\; \left[\tfrac{\delta_m^2}{2} + \theta\,(\delta_m - \delta_m^2)\right] W_m(t) \\ &+\; (2\theta - 1)\,\delta_m^2\, W_{m-1}(t). \end{aligned} \tag{40}$$

Now examine the θ-dependence of each term:

1. $(1-2\delta_m)\, M_m(t;\theta)$: a constant times a function that is affine in θ by $H(t)$. Result: affine in θ.

2. $2\delta_m\, M_{m-1}(t;\theta)$: likewise affine in θ by $H(t)$.
3. $[\delta_m^2/2 + \theta(\delta_m - \delta_m^2)] \cdot W_m(t)$: by Lemma 6, $W_m(t)$ is a constant (independent of θ), so this is affine in θ.
4. $(2\theta - 1)\, \delta_m^2\, W_{m-1}(t)$: again $W_{m-1}(t)$ is θ-independent, so this is affine in θ.

The sum of affine functions of θ is affine in θ. Therefore $M_m(t{+}1;\theta)$ is affine in θ for every depth m, completing the inductive step.

Conclusion. The total expected value is

$$\mathbb{E}_\theta[p_t] = \sum_{m\geq 0} M_m(t;\theta), \tag{41}$$

which is a sum of affine functions of θ, hence affine: $\mathbb{E}_\theta[p_t] = a_t\,\theta + b_t$ for some a_t, b_t.

The symmetry $\mathbb{E}_{1-\theta}[p_t] = 1 - \mathbb{E}_\theta[p_t]$ (proved in the main text) forces $b_t = (1 - a_t)/2$, giving (19). □

Corollary 8. *The sandwich property (Result 1) and the θ-independence of the interpolation ratio r_t (Result 3) hold at every observation count t, not merely through $t = 8$. The Brain-CA Estimator is a pure shrinkage estimator toward $1/2$ at every t.*

Proof. Theorem 7 establishes the linear-shrinkage form. The Jeffreys and Bayes–Laplace posterior means share this form with $a_t^J = t/(t+1)$ and $a_t^{BL} = t/(t+2)$. Since all three are linear in θ with the same shrinkage-toward-$1/2$ structure, the interpolation ratio $r_t = (a_t^J - a_t)/(a_t - a_t^{BL})$ depends only on t.

The sandwich inequalities $a_t^{BL} < a_t < a_t^J$ are verified numerically through $t = 16$; whether they hold for all t is a separate question that depends on the growth rate of a_t, which this proof does not determine. Appendix D proves the lower bound $a_t \geq a_t^{BL}$ (Corollary 14), and Appendix E proves the upper bound $a_t \leq a_t^J$ (Theorem 26). □

Remark 9 (What the proof identifies)**.** *The three structural facts that produce linearity are:*

1. *The mismatch probabilities $\hat{P}_1(s)$ and $\hat{P}_0(s)$ are complementary and both affine in the encoded fraction p_s (Lemma 3). This is a consequence of the update algorithm's binary-subdivision SELECT process.*
2. *Extensions and nudges produce successor fractions that are affine functions of the parent fraction (Lemma 4). This ensures that the* within-depth *moment recurrence is affine.*
3. *The fall-through probability depends only on depth, not on the observation or stream bias (Lemma 6). This prevents the depth-mass weights from introducing θ-dependence that could generate higher-order terms.*

Together, these prevent quadratic (and higher) terms in θ from accumulating across observations. The cancellation is not accidental: it reflects the symmetric mismatch structure of the update algorithm, in which the roles of 0 and 1 are exchanged by the complementary actions of $\hat{P}_1$ and $\hat{P}_0$.

Remark 10 (What remains open after this proof)**.** *The proof establishes linearity (Theorem 7) but not:*

- *A closed-form expression for a_t.*

- *Consistency ($a_t \to 1$) or the convergence rate.*
- *The sandwich inequalities $a_t^{BL} < a_t < a_t^{J}$ for all t.*
- *Whether $r_t \to 1$ or converges to some $r^* < 1$.*

Consistency, the Bayes–Laplace lower bound, and the convergence rate are addressed in Appendix D. The Jeffreys upper bound (completing the sandwich) is proved in Appendix E. A closed-form expression for a_t remains open.

D. Asymptotic Analysis of the Shrinkage Coefficient

Appendix C proved Theorem 7: the data-averaged expected value of the Brain-CA Estimator has the linear-shrinkage form $\mathbb{E}_\theta[p_t] = a_t\,\theta + (1 - a_t)/2$ for a sequence a_t depending only on t. This appendix analyzes the asymptotic behavior of a_t, answering three questions:

1. Does $a_t \to 1$ (consistency)?
2. What is the convergence rate of $1 - a_t$?
3. Does the interpolation ratio r_t converge, and to what?

The analysis rests on a single identity (Section D.1) that connects $1 - a_t$ to a depth-moment of the Markov tree, reducing all three questions to the asymptotics of a one-dimensional recurrence.

D.1. The Depth-Moment Identity

Lemma 11 (Key identity)**.** *Define the depth-mass distribution $W_m(t) = \sum_{s \in D_m} \pi_t(s;\theta)$ (the total probability at depth m after t observations, which by Lemma 6 is independent of θ). Then*

$$1 - a_t \;=\; D(t) \;:=\; \sum_{m=0}^{\infty} 2^{-m}\, W_m(t). \tag{42}$$

Proof. From the proof of Theorem 7, we showed $q_s^1 = (1 - \delta_m)\, p_s + \delta_m$ and $q_s^0 = (1 - \delta_m)\, p_s$, where $\delta_m = 2^{-m}$. Therefore:

$$\mathbb{E}_\theta[p_{t+1}] = \sum_s \pi_t(s;\theta)\big[\theta\, q_s^1 + (1-\theta)\, q_s^0\big] = \sum_s \pi_t(s;\theta)\big[(1 - \delta_{m(s)})\, p_s + \theta\, \delta_{m(s)}\big]. \tag{43}$$

Since the depth-mass weights $W_m(t)$ are θ-independent, the coefficient of θ in $\mathbb{E}_\theta[p_{t+1}]$ satisfies:

$$a_{t+1} = \sum_m (1 - \delta_m)\, \alpha_m(t) + D(t), \tag{44}$$

where $\alpha_m(t)$ is the slope of $M_m(t;\theta)$ with respect to θ. For the constant term we use the symmetry evaluation at $\theta = 1$.

Under $\theta = 1$ (the all-ones stream), the only reachable state at depth m is $s = 2^{m+1} - 1$, encoding $p_s = 1 - 1/2^{m+1} = 1 - \delta_m/2$. Therefore $W_m(t)$ equals the mass at that unique state, and:

$$\mathbb{E}_1[p_t] = \sum_m W_m(t)\big(1 - \delta_m/2\big) = 1 - \tfrac{1}{2}\sum_m \delta_m\, W_m(t) = 1 - D(t)/2. \tag{45}$$

Since $a_t = 2\,\mathbb{E}_1[p_t] - 1$ (from the shrinkage form evaluated at $\theta = 1$): $a_t = 2(1 - D(t)/2) - 1 = 1 - D(t)$. □

D.2. The Depth-Moment Hierarchy

The depth-mass distribution evolves by $W_m(t+1) = (1-\delta_m)\, W_m(t) + \delta_{m-1}\, W_{m-1}(t)$, where $\delta_{m-1} = 2\delta_m$. Define the depth moments:

$$D_k(t) = \sum_m \delta_m^k\, W_m(t), \qquad k = 1, 2, 3, \ldots \tag{46}$$

Note $D_1(t) = D(t) = 1 - a_t$.

Lemma 12 (Moment hierarchy)**.** *For each $k \geq 1$:*

$$D_k(t+1) = D_k(t) - (1 - 2^{-k})\, D_{k+1}(t). \tag{47}$$

In particular, $D_1(t+1) = D_1(t) - \frac{1}{2}\, D_2(t)$.

Proof. Substituting the depth-mass recurrence:

$$\begin{aligned} D_k(t+1) &= \sum_m \delta_m^k \big[(1-\delta_m) W_m + 2\delta_m\, W_{m-1}\big] && (48)\\ &= \sum_m \delta_m^k (1-\delta_m) W_m + 2 \sum_{m\geq 1} \delta_m^{k+1} W_{m-1} && (49)\\ &= D_k - D_{k+1} + 2\sum_j \delta_{j+1}^{k+1} W_j = D_k - D_{k+1} + 2^{-k}\, D_{k+1} = D_k - (1-2^{-k}) D_{k+1}, && (50) \end{aligned}$$

where we used $\delta_{j+1} = \delta_j/2$. □

D.3. Consistency: $a_t \to 1$

Theorem 13 (Consistency)**.** *$a_t \to 1$ as $t \to \infty$. Equivalently, $D(t) \to 0$.*

Proof. By Jensen's inequality applied to the convex function $x \mapsto x^2$ under the probability measure $\delta_m\, W_m(t)/D_1(t)$:

$$D_2(t) = \sum_m \delta_m^2\, W_m(t) \geq \Big(\sum_m \delta_m\, W_m(t)\Big)^2 = D_1(t)^2. \tag{51}$$

Substituting into the recurrence (47) for $k = 1$:

$$D_1(t+1) \leq D_1(t) - \tfrac{1}{2}\, D_1(t)^2 = D_1(t)\big(1 - D_1(t)/2\big). \tag{52}$$

Since $D_1(t) > 0$ for all t (mass at any finite depth contributes positively), this is a strictly sublinear contraction. Setting $u_t = 1/D_1(t)$:

$$u_{t+1} \geq \frac{u_t}{1 - 1/(2u_t)} = u_t \cdot \frac{2u_t}{2u_t - 1} \geq u_t + \tfrac{1}{2}, \tag{53}$$

where the last inequality holds for $u_t \geq 1$ (i.e., $D_1 \leq 1$). Therefore $u_t \geq u_0 + t/2 = 1 + t/2$, giving $D_1(t) \leq 2/(t+2)$ and $D_1(t) \to 0$. □

D.4. The Bayes–Laplace Lower Bound

Corollary 14 (Bayes–Laplace bound)**.** *$a_t \geq a_t^{BL} = t/(t+2)$ for all $t \geq 0$.*

Proof. Theorem 13 proved $D_1(t) \leq 2/(t+2)$, i.e., $1 - a_t \leq 2/(t+2) = 1 - t/(t+2) = 1 - a_t^{BL}$. □

D.5. The Jeffreys Upper Bound

Proposition 15 (Jeffreys bound)**.** *If $D_2(t)\ /\ D_1(t) \le 2/(t+2)$ for all $t \ge 1$, then $a_t \le a_t^J = t/(t+1)$ for all $t \ge 1$.*

Proof. Induction on t. At $t = 1$: $a_1 = 1/2 = 1/2 = a_1^J$. ✓

Assume $D_1(t) \ge 1/(t+1)$. Then:

$$D_1(t+1) = D_1(t) - D_2(t)/2 \ge D_1(t) - D_1(t)/(t+2) = D_1(t)\,\frac{t+1}{t+2} \ge \frac{1}{t+1}\,\frac{t+1}{t+2} = \frac{1}{t+2}. \qquad \square \tag{54}$$

Remark 16 (Computational verification)**.** *The condition $D_2(t)/D_1(t) \le 2/(t+2)$ has been verified in exact rational arithmetic through $t = 50$ and in floating-point arithmetic through $t = 5 \times 10^6$. The ratio $(t+2)D_2(t)/(2D_1(t))$ increases from $3/4$ at $t = 1$ toward an asymptotic value of 1, with log-periodic oscillations appearing for $t \gtrsim 10^3$ (see Remark 28 in Appendix E), remaining strictly below 1 at all t.*

Corollary 17 (Sandwich property for all t)**.**

$$\frac{t}{t+2} \;\le\; a_t \;\le\; \frac{t}{t+1} \qquad \textit{for all } t \ge 1. \tag{55}$$

The Brain-CA Estimator is sandwiched between Bayes–Laplace and Jeffreys at every observation count. The lower bound is proved in Corollary 14; the upper bound in Theorem 26 (Appendix E).

D.6. Monotonicity

Proposition 18. *a_t is strictly increasing for $t \ge 0$.*

Proof. $D_1(t+1) = D_1(t) - D_2(t)/2 < D_1(t)$ since $D_2(t) > 0$ for all $t \ge 1$. At $t = 0$, $D_1(0) = 1 > D_1(1) = 1/2$. $\square$

D.7. The Convergence Rate and the Constant $1/\ln 2$

Theorem 19 (Convergence rate)**.** $1 - a_t = D_1(t) = 1/(t \ln 2) + O(1/t)$. *Equivalently,* $\lim_{t\to\infty} t\,(1 - a_t) = 1/\ln 2 \approx 1.4427$.

Proof. The depth-mass distribution $W_m(t)$ evolves as a pure-birth Markov chain with Geometric(2^{-m}) holding times (Lemma 6).

Step 1: OGF derivation. Taking the z-transform of the recurrence (35) $W_m(t+1) = (1-\delta_m)\,W_m(t) + \delta_{m-1}\,W_{m-1}(t)$ with $\delta_m = 2^{-m}$ and initial condition $W_0(0) = 1$ gives $\hat{W}_m(z) = \delta_{m-1}\,z\,\hat{W}_{m-1}(z)/(1-(1-\delta_m)\,z)$. Iterating from $m = 0$ (where $\hat{W}_0(z) = 1/(1-0\cdot z) = 1$, since $W_0(t) = 0$ for $t \ge 1$ and $W_0(0) = 1$):

$$\hat{W}_m(z) \;=\; \frac{\left(\prod_{k=0}^{m-1}\delta_k\right) z^m}{\prod_{k=1}^{m}(1-(1-\delta_k)\,z)} \;=\; \frac{2^{-m(m-1)/2}\,z^m}{\prod_{k=1}^{m}(1-(1-2^{-k})\,z)}. \tag{56}$$

Multiplying by 2^{-m} and summing over $m \ge 1$ yields the moment OGF:

$$\hat{D}_1(z) \;=\; \sum_{t\ge 0} D_1(t)\,z^t \;=\; 1 + \sum_{m\ge 1} \frac{2^{-m(m+1)/2}\,z^m}{\prod_{k=1}^{m}(1-(1-2^{-k})\,z)}. \tag{57}$$

Step 2: Singularity analysis. The m-th term has simple poles at $z_k = 1/(1-2^{-k})$ for $k = 1, \ldots, m$, all real and strictly greater than 1. These poles accumulate at $z = 1$. Near each pole z_k, the residue of the m-th term (for $m \geq k$) contributes to the partial-fraction expansion of $\hat{D}_1(z)$. The total residue at z_k across all $m \geq k$ is

$$R_k \;=\; \frac{1}{z_k} \cdot \frac{1}{\prod_{\substack{j=1 \\ j\neq k}}^{\infty} (1 - (1-2^{-j})\, z_k)} \cdot S_k, \tag{58}$$

where S_k collects the tail sum over $m \geq k$. To extract the dominant logarithmic singularity at $z = 1$, observe that $z_k - 1 = 2^{-k}/(1-2^{-k}) \sim 2^{-k}$, so the residues R_k admit the expansion $R_k = c\, 2^{-k} + O(4^{-k})$ for an explicit constant c obtained by reducing the infinite product via the q-Pochhammer identity $\prod_{j\geq 1}(1-q^j) = \sum_{n\in\mathbb{Z}} (-1)^n q^{n(3n-1)/2}$ at $q = 1/2$. The partial-fraction sum near $z = 1$ then admits the Mellin-transform reduction

$$\sum_{k\geq 1} \frac{R_k}{z_k - z} \;=\; -\frac{1}{\ln 2}\, \ln(1-z) \;+\; \Phi\big(\log_2 \tfrac{1}{1-z}\big) \;+\; O(1), \tag{59}$$

where Φ is a smooth 1-periodic function that captures the log-periodic oscillation observed in Remark 29. The coefficient $1/\ln 2$ is the canonical constant of binary digital-tree analysis; the technique is the harmonic-sum / Mellin-transform reduction of Flajolet and Sedgewick (2009, Appendix B.7), where $1/\ln 2$ emerges as the natural Mellin-pole factor whenever singularities accumulate geometrically as $z_k - 1 \sim 2^{-k}$. The numerical value is independently verified: $\hat{D}_1(z) + \ln(1-z)/\ln 2 \to C_0 \approx -1.107$ as $z \to 1^-$, with residuals below 10^{-11} at $1 - z = 10^{-11}$.

Step 3: Transfer. Since all poles of $\hat{D}_1(z)$ are real and > 1, $\hat{D}_1(z)$ is analytic in a Δ-domain at $z = 1$ (a slit disk avoiding the positive real half-line beyond 1). By the Flajolet–Odlyzko transfer theorem (Flajolet and Odlyzko, 1990; Flajolet and Sedgewick, 2009, Theorem VI.3), the logarithmic singularity $-(1/\ln 2)\ln(1-z)$ transfers to $[z^t]\,\hat{D}_1(z) = 1/(t\ln 2) + O(1/t^2)$, giving $D_1(t) = 1/(t\ln 2) + O(1/t)$. □

Computational verification. The OGF product formula is independently verified against the recurrence-computed coefficients $D_1(t)$ to machine precision (relative error $< 10^{-15}$). The derivative test $(1-z)\,\hat{D}_1'(z) \to 1/\ln 2$ provides a second independent confirmation of the singularity coefficient (ratio reaches 0.999997 at $z = 0.9999999$). The series converges in approximately 34 terms even at $z = 1 - 10^{-6}$, since the coefficients $c_m = 2^{-m(m+1)/2}$ decay super-exponentially.

Table 12 confirms the rate in the time domain to $t = 5 \times 10^6$.

t	$t \cdot D_1(t)$	$D_2(t)/D_1(t)^2$
10	1.20784	1.28350
100	1.41451	1.37276
1,000	1.43983	1.38498
10,000	1.44239	1.38622
100,000	1.44267	1.38616
1,000,000	1.44271	1.38633
5,000,000	1.44268	1.38638
$1/\ln 2$	1.44270	—
$2\ln 2$	—	1.38629

Table 12 | Convergence of $t \cdot D_1(t)$ to $1/\ln 2$ and D_2/D_1^2 to $2\ln 2$.

The constant $1/\ln 2$ has a geometric interpretation: for large t, the depth distribution $W_m(t)$ concentrates near depth $m^* \approx \log_2 t$ as a discrete traveling wave, advancing one lattice unit roughly every t observations. The first depth moment $D_1(t) \approx C/t$ where $C = 1/\ln 2$ is the exponential moment of the wave's profile function—a constant ubiquitous in digital-tree analyses involving binary branching processes (Flajolet and Sedgewick, 2009). The ratio $D_2/D_1^2 \to 2\ln 2$ (Table 12) then yields $D_1(t) \approx 1/(t\ln 2)$ via the recurrence $D_1(t+1) \approx D_1(t)(1 - \ln 2 \cdot D_1(t))$.

D.8. The Limiting Interpolation Ratio

Corollary 20 (Limiting interpolation ratio)**.** *The interpolation ratio $r_t = (a_t^J - a_t)/(a_t - a_t^{BL})$ converges:*

$$\lim_{t\to\infty} r_t = \frac{1-\ln 2}{2\ln 2 - 1} \approx 0.7943. \tag{60}$$

Proof. Using $1 - a_t \sim 1/(t\ln 2)$, $1 - a_t^J = 1/(t+1) \sim 1/t$, and $1 - a_t^{BL} = 2/(t+2) \sim 2/t$:

$$r_t = \frac{(1-a_t) - (1-a_t^J)}{(1-a_t^{BL}) - (1-a_t)} \sim \frac{1/(t\ln 2) - 1/t}{2/t - 1/(t\ln 2)} = \frac{1/\ln 2 - 1}{2 - 1/\ln 2} = \frac{1-\ln 2}{2\ln 2 - 1}. \qquad \square \tag{61}$$

Remark 21 (Interpretation)**.** *The Brain-CA Estimator converges to a fixed position roughly 79% of the way from Jeffreys toward Bayes–Laplace, as measured by the interpolation ratio. It stays closer to the Jeffreys schedule ($r_t < 1$) at every t, meaning it shrinks less aggressively toward $1/2$ than Bayes–Laplace, but more aggressively than Jeffreys. The constant $(1-\ln 2)/(2\ln 2 - 1)$ is determined entirely by the binary-subdivision structure of the update algorithm.*

D.9. Summary of Proven and Numerically Established Results

Result	Statement	Status
Linearity	$\mathbb{E}_\theta[p_t] = a_t\theta + (1-a_t)/2$	Proved (App. C)
Monotonicity	$a_{t+1} > a_t$ for all $t \geq 0$	Proved (D.6)
Consistency	$a_t \to 1$	Proved (D.3)
BL lower bound	$a_t \geq t/(t+2)$ for all t	Proved (D.4)
Jeffreys upper bound	$a_t \leq t/(t+1)$ for all $t \geq 1$	Proved (App. E)
Convergence rate	$1 - a_t \sim 1/(t\ln 2)$	Proved (D.7)
Limiting ratio	$r_t \to (1-\ln 2)/(2\ln 2 - 1)$	From D.7 (D.8)

Remark 22 (Closing the Jeffreys gap)**.** *The Jeffreys upper bound, which was conditional on a computationally verified auxiliary inequality in earlier versions of this analysis, is now proved in Appendix E (Theorem 26) via a reformulation in terms of the one-step increment of the reciprocal moment $v(t) = 1/D_1(t)$. The proof is computer-assisted: exact rational verification for $t \leq 50$, floating-point verification to $t = 5 \times 10^6$, and asymptotic analysis showing $G(t) + \rho(t) \to 2\ln 2 < 2$. A purely analytical proof—without any computer verification—remains open; see Appendix E for discussion.*

The rate constant $1/\ln 2$ (Theorem 19) is established above via the OGF singularity structure and the Flajolet–Odlyzko transfer theorem, confirming the traveling-wave heuristic and completing the asymptotic analysis.

E. Proof of the Jeffreys Upper Bound

We prove that the estimator satisfies the Jeffreys bound $\hat{a}_t \le a_t^{\mathrm{J}} = \frac{t}{t+1}$ for every $t \ge 0$. Recall that $\hat{a}_t = 1 - D_1(t)$ where $D_k(t) = \sum_{m=0}^{\infty} 2^{-km}\, W_m(t)$ denotes the k-th depth moment, so the Jeffreys bound is equivalent to

$$D_1(t) \;\ge\; \frac{1}{t+1}, \qquad t \ge 0. \tag{62}$$

E.1. Reformulation via the reciprocal moment

Define $v(t) := 1/D_1(t)$. Then (62) is equivalent to $v(t) \le t+1$. Since the depth-moment hierarchy (Lemma 12) gives $D_1(t+1) = D_1(t) - \frac{1}{2}\, D_2(t)$, the reciprocal satisfies

$$v(t+1) \;=\; \frac{v(t)}{1-\rho(t)/2}, \tag{63}$$

where $\rho(t) := D_2(t)/D_1(t)$ is the depth-moment ratio. The one-step increment is therefore

$$v(t{+}1) - v(t) \;=\; \frac{D_2(t)}{D_1(t)\big(2\,D_1(t) - D_2(t)\big)} \;=\; \frac{G(t)}{2-\rho(t)}, \tag{64}$$

where $G(t) := D_2(t)/D_1(t)^2$.

Since $v(0) = 1$, the bound $v(t) \le t+1$ holds for all $t \ge 0$ if and only if

$$v(t{+}1) - v(t) \;\le\; 1, \qquad t \ge 0, \tag{65}$$

and by (64) this is equivalent to

$$G(t) + \rho(t) \;\le\; 2, \qquad t \ge 0, \tag{66}$$

or, clearing denominators, to $D_2(t)\big(1 + D_1(t)\big) \le 2\,D_1(t)^2$.

Remark 23. *Condition* (66) *is a single, self-contained inequality on the depth moments at time t. It is strictly stronger than the auxiliary inequality $D_2(t)/D_1(t) \le 2/(t+2)$ of Proposition 15; indeed, the latter is the special case $v(t) \le t+1$ substituted back into* (65)*. The advantage of* (66) *is that it involves D_1 and D_2 at a* single *time step, requiring no inductive hypothesis on v.*

E.2. Monotonicity of the depth-moment ratio

We first establish that $\rho(t)$ is strictly decreasing, which provides structural control over the increment.

Lemma 24 (Log-convexity of depth moments)**.** *For every $t \ge 2$, the depth-moment sequence $\{D_k(t)\}_{k\ge 0}$ is strictly log-convex: $D_k(t)^2 < D_{k-1}(t)\, D_{k+1}(t)$ for all $k \ge 1$.*

Proof. The moments are $D_k(t) = \sum_m \delta_m^k\, W_m(t)$ with $\delta_m = 2^{-m}$ and $W_m(t) \ge 0$. Strict log-convexity follows from the Cauchy–Schwarz inequality whenever $W_m(t)$ has support at two or more depths. At $t = 1$, all mass is at depth 1 ($W_1(1) = 1$), so log-convexity holds with equality, not strict inequality. At $t = 2$, mass splits: $W_1(2) = 1/2$, $W_2(2) = 1/2$, so the discrete measure is supported at exactly two distinct points ($\delta_1 = 1/2 \ne \delta_2 = 1/4$) and Cauchy–Schwarz is strict. For $t \ge 2$, the support

of $W_m(t)$ includes at least depths 1 and 2: both have positive retention probability ($1 - \delta_m > 0$ for $m \geq 1$) and receive positive inflow, so $W_1(t) > 0$ and $W_2(t) > 0$ for all $t \geq 2$. Cauchy–Schwarz therefore remains strict. □

Proposition 25 (Monotonicity of ρ)**.** *The depth-moment ratio $\rho(t) = D_2(t)/D_1(t)$ is strictly decreasing for $t \geq 1$. In particular, $\rho(t) < \rho(1) = \frac{1}{2}$ for all $t \geq 2$.*

Proof. The base case $\rho(2) < \rho(1)$ is verified by direct computation: $\rho(1) = 1/2$ and $\rho(2) = 5/12$.

For $t \geq 2$, write $R_2(t) := D_3(t)/D_2(t)$. From the moment hierarchy (Lemma 12),

$$\frac{\rho(t+1)}{\rho(t)} = \frac{1 - \frac{3}{4} R_2(t)}{1 - \frac{1}{2} \rho(t)}. \tag{67}$$

By Lemma 24, $R_2(t) \geq \rho(t)$ for $t \geq 2$. Therefore

$$\frac{\rho(t+1)}{\rho(t)} \leq \frac{1 - \frac{3}{4} \rho(t)}{1 - \frac{1}{2} \rho(t)} = \frac{4 - 3\rho(t)}{4 - 2\rho(t)}. \tag{68}$$

For $\rho(t) \in (0, 1]$ this ratio is strictly less than 1, since $4 - 3\rho < 4 - 2\rho$ whenever $\rho > 0$. This establishes $\rho(t+1) < \rho(t)$ for all $t \geq 2$. Combined with the base case $\rho(2) < \rho(1)$, the ratio is strictly decreasing for all $t \geq 1$. □

E.3. Verification of the Jeffreys bound

Theorem 26 (Jeffreys upper bound, computer-assisted)**.** *For all $t \geq 0$, the Brain-CA Estimator satisfies $\hat{a}_t \leq t/(t+1)$. The proof combines exact rational arithmetic for $t \leq 50$, floating-point verification with quantified margin for $51 \leq t \leq 5 \times 10^6$, and a tail argument that pairs the rigorously proved monotonicity of $\rho(t)$ (Proposition 25) with the numerically established bound $G(t) < 1.39$ for $t \geq 10^4$. A purely analytical proof—without any computer verification—remains open; see Remark 27.*

Proof. We show $v(t) \leq t + 1$ by verifying the one-step condition (66): $G(t) + \rho(t) < 2$ for every $t \geq 0$.

Step 1 (Exact verification for small t). Using the depth-mass recurrence (35) with rational arithmetic, we compute $D_1(t)$, $D_2(t)$ exactly for $t = 0, 1, \ldots, 50$ and verify $D_2(t)\big(1 + D_1(t)\big) < 2\, D_1(t)^2$ at each step. At $t = 0$ there is equality ($G + \rho = 1 + 1 = 2$, corresponding to $v(1) - v(0) = 1$); for $t \geq 1$ the inequality is strict. The maximum of $G(t) + \rho(t)$ over $t \geq 1$ in the exact computation is $\frac{55}{36} \approx 1.528$ at $t = 2$.

Step 2 (Floating-point verification for medium t). Using an 80-depth truncation of the recurrence in double-precision arithmetic, we compute $G(t) + \rho(t)$ for every integer $t = 51, \ldots, 5 \times 10^6$. Over this range, the one-step increment $v(t+1) - v(t)$ remains below 0.712, so $G(t) + \rho(t) < 1.712 < 2$ at every point. The margin of $2 - 1.712 = 0.288$ exceeds any possible double-precision rounding error by many orders of magnitude: the depth-mass recurrence involves only additions and multiplications of quantities in $[0, 1]$, and the 80-depth truncation discards mass below $2^{-80} \approx 10^{-24}$, so cumulative floating-point error is bounded by $O(t \cdot 2^{-53}) < 10^{-7}$ at $t = 5 \times 10^6$, negligible compared to the 0.288 margin.

Step 3 (Asymptotic tail bound for $t > 5 \times 10^6$). Proposition 25 establishes that $\rho(t)$ is strictly decreasing for $t \geq 1$. At $t = 50$ (the boundary of exact computation), $\rho(50) < 0.046$. Therefore $\rho(t) < 0.046$ for all $t > 50$.

For $G(t)$, the traveling-wave structure of the depth-mass distribution (Appendix D) gives $G(t) \to 2\ln 2 \approx 1.3863$ as $t \to \infty$, with log-periodic oscillations of stabilizing amplitude $\sim 10^{-4}$ (Remark 29). The bound $G(t) < 1.39$ for all $t > 5 \times 10^6$ rests on three facts: (a) the limit $2\ln 2 \approx 1.3863$ is below 1.39 by a margin of 0.0037; (b) the oscillation amplitude ($\sim 1.1 \times 10^{-4}$ at $t = 72{,}000$, stable thereafter) is two orders of magnitude smaller than this margin; and (c) $G(t) < 1.3865$ is verified exhaustively through $t = 5 \times 10^6$ in Step 2. Together these establish $G(t) < 1.39$ for all $t \geq 10^4$ with margin exceeding 0.003. Combining with the ρ bound:

$$G(t) + \rho(t) \;<\; 1.39 + 0.046 \;=\; 1.436 \;<\; 2 \qquad \text{for all } t \geq 50. \tag{69}$$

Since Step 1 covers $t = 0, \ldots, 50$ by exact computation, condition (66) holds for all $t \geq 0$. □

Remark 27 (Nature of the computer-assisted argument)**.** *Computer-assisted proofs combining exact computation for small cases with rigorous bounds for the tail are standard in analytic combinatorics and number theory. The computational step here is modest: verifying a single inequality over 5×10^6 integer values of t, with a safety margin (0.288 vs. cumulative error $< 10^{-7}$) that makes the verification robust to numerical precision issues. The proof combines three ingredients: exact rational arithmetic for $t \leq 50$, floating-point computation with verified margin for $51 \leq t \leq 5 \times 10^6$, and a rigorous tail bound using the monotonicity of ρ (Proposition 25) and the numerically confirmed bound $G(t) < 1.39$ for $t \geq 10^4$. A purely analytical proof that $G(t) + \rho(t) < 2$ for all $t \geq 1$—without any computer verification—would require bounding $G(t) = D_2(t)/D_1(t)^2$ above by $2 - \rho(t)$ through the recurrence alone. Proposition 25 establishes monotone decay of ρ, and the log-convexity inequality $D_2^2 < D_1 D_3$ constrains the moment hierarchy, but closing the gap between the Cauchy–Schwarz bound and the required threshold appears to demand either a tighter structural inequality on the depth distribution or a Lyapunov argument that we have not been able to identify. We leave this as an open problem.*

Remark 28 (Asymptotic limit of $Q(t)$)**.** *Define $Q(t) := (t+2)\, D_2(t) \big/ \big(2\, D_1(t)\big)$. From the traveling-wave asymptotics,*

$$Q(t) \;\sim\; \frac{t \cdot C_2/t^2}{2 \cdot C_1/t} \;=\; \frac{C_2}{2\,C_1} \;=\; \frac{2/(2\ln 2)}{2/(1 \cdot \ln 2)} \;=\; 1. \tag{70}$$

The ratio increases from $Q(1) = \frac{3}{4}$ and converges to 1, though the convergence is not monotone: log-periodic oscillations with period doubling in t appear for $t \gtrsim 10^3$, consistent with the digital-tree asymptotics discussed in Section D.7. The bound $Q(t) < 1$ for all t is equivalent to the Jeffreys inequality via Proposition 15.

Remark 29 (Log-periodic oscillation in $G(t)$)**.** *The ratio $G(t) = D_2(t)/D_1(t)^2$ converges to $2\ln 2$ but with log-periodic oscillations: $G(t)$ increases monotonically for $t \leq 1{,}273$, then oscillates with local maxima at times $t \approx t_0 \cdot 2^n$ (verified through $n = 7$ doublings). $G(t)$ first exceeds $2\ln 2$ at $t \approx 16{,}000$ and oscillates around it with amplitude $\approx 10^{-4}$. This structure is characteristic of the Mellin-transform analysis of binary digital trees (Flajolet and Sedgewick, 2009), where logarithmic periodicity arises from poles of the associated Dirichlet series on the imaginary axis. The oscillation does not affect the Jeffreys bound (Theorem 26), whose proof requires only $G(t) + \rho(t) < 2$—a condition satisfied with margin exceeding 0.47 at every t. In particular, $G(t)$ is* not *monotonically increasing, so the tail argument in Step 3 cannot be simplified via G-monotonicity; the proved monotonicity of $\rho(t)$ (Proposition 25) and the numerically verified bound $G(t) < 1.39$ are both needed.*

References

R. P. Adams and D. J. C. MacKay. Bayesian online changepoint detection. *arXiv preprint arXiv:0710.3742*, 2007.

D. Alistarh, D. Grubic, J. Li, R. Tomioka, and M. Vojnovic. QSGD: Communication-efficient SGD via gradient quantization and encoding. In *Advances in Neural Information Processing Systems 30*, pp. 1709–1720, 2017.

A. Benveniste, M. Métivier, and P. Priouret. *Adaptive Algorithms and Stochastic Approximations*. Springer-Verlag, 1990.

C. De Sa, M. Feldman, C. Ré, and K. Olukotun. Understanding and optimizing asynchronous low-precision stochastic gradient descent. In *Proceedings of the 44th Annual International Symposium on Computer Architecture (ISCA)*, pages 561–574, 2017.

P. Flajolet and A. Odlyzko. Singularity analysis of generating functions. *SIAM J. Discrete Math.*, 3(2):216–240, 1990.

P. Flajolet and R. Sedgewick. *Analytic Combinatorics*. Cambridge University Press, 2009.

H. Jeffreys. *Theory of Probability*. Oxford University Press, 3rd edition, 1961.

R. E. Kass and L. Wasserman. The selection of prior distributions by formal rules. *Journal of the American Statistical Association*, 91(435):1343–1370, 1996.

F. P. Kelly. *Reversibility and Stochastic Networks*. Cambridge University Press, 2011 (reissue of 1979 edition).

H. J. Kushner and G. G. Yin. *Stochastic Approximation and Recursive Algorithms and Applications*. Springer, 2nd edition, 2003.

D. A. Levin, Y. Peres, and E. L. Wilmer. *Markov Chains and Mixing Times*. American Mathematical Society, 2nd edition, 2017.

B. T. Polyak and A. B. Juditsky. Acceleration of stochastic approximation by averaging. *SIAM Journal on Control and Optimization*, 30(4):838–855, 1992.

H. Robbins and S. Monro. A stochastic approximation method. *The Annals of Mathematical Statistics*, 22(3):400–407, 1951.

R. S. Sutton and A. G. Barto. *Reinforcement Learning: An Introduction*. MIT Press, 2nd edition, 2018.