Time as Irreversible Distinction Update:
Finite Records, Causal Ordering, and Register-Time Collapse

Yining Wu
Independent Researcher
yining.wu@alumni.upenn.edu
(Dated: May 2026)

Time is often treated either as a continuous coordinate in physical equations or as a global ordering whose arrow is explained by statistical mechanics, cosmology, or boundary conditions. FDS-O2 gives a narrower operational account: for a finite physical observer, usable time is the ordered structure of irreversible distinction updates in finite records. Building on FDS-O1, where an observer is defined as a finite distinction-register and measurement as stable record formation, this paper defines a temporal register as a physical system that records, orders, timestamps, updates, synchronizes, and reuses distinctions under finite memory, finite clock precision, finite buffering, finite communication bandwidth, finite latency, and finite thermodynamic budgets. The central claim is not that coordinate time is unreal or that microscopic dynamics must be nonunitary. It is that finite systems access temporal order only through register time: causal dependencies, clock labels, logs, residues, and update traces carried by finite records. We make causal dependency prior to temporal labeling: record z_j is operationally after z_i when the update that forms z_j depends on z_i through an accessible finite update chain. When finite memory forces overwrite, compression, projection, or garbage collection, the update map is many-to-one. Its non-injectivity can be quantified by $\mathcal{L}_U = H(X|Z)$, the residual set of compatible past states given the current record. This loss defines an operational arrow: future records may contain traces of past records, but erased distinctions cannot be reconstructed from the current finite record without external memory. The paper adds synchronization bottlenecks between finite observers, latency-induced apparent causal inversion, finite clock coarse-graining, dissipative projection, and long-context temporal collapse in AI-like finite-window systems. Deterministic simulations illustrate memory-fill-driven non-injectivity, fixed-memory overwrite versus append-only logging, latency-based order inversion, clock-tick collisions, synchronization ambiguity, projection-semigroup loss, and context-window temporal collapse. The paper does not solve the cosmological arrow of time, the Past Hypothesis, the full Second Law, general relativity, or quantum measurement. Its contribution is narrower: it converts finite record formation into a concrete operational model of time as causal, irreversible, finite-register update.

Scope and Boundary of the Theory. This paper does not claim that microscopic laws are fundamentally nonunitary, that coordinate time in equations is invalid, that relativity requires a preferred foliation, or that cosmology no longer requires boundary conditions. It does not derive general relativity, quantum measurement, or the thermodynamic Second Law in full. It treats *register time*: the finite record structure through which bounded observers access temporal order. Coordinate time, proper time, and clock readings remain valid in their standard domains. O2 asks what finite records must carry for a physical system to use temporal order at all.

Claim-status summary

Table I summarizes the central FDS-O2 claims, their epistemic status, and the conditions under which they should be weakened or rejected.

Keywords: finite observer; time; register time; causal dependency; irreversible update; finite records; non-injective update; dissipative projection; synchronization bottleneck; clock precision; latency; buffering; Landauer principle; finite memory; semigroup; information thermodynamics; long-context AI; active finite distinction systems.

INTRODUCTION

From finite measurement to finite time

FDS-O1 defined an observer as a finite distinction-register: a physical system that registers, preserves, updates, orders, and communicates distinctions through finite records, finite channels, finite buffers, and finite thermodynamic budgets [1]. O1 treated measurement as stable finite record formation. The present paper asks the next operational question: what is time for such a finite register?

The answer is deliberately modest. A finite observer does not directly access an ideal continuum. It accesses temporal order through records: timestamps, dependency edges, update traces, residues, logs, memory states, clock ticks, and causal communication. A system says that event a occurred before event b only if it has a record structure that supports that comparison.

TABLE I. Central FDS-O2 claims, epistemic status, and demotion or failure conditions.

Claim	**Status**	**What would weaken or falsify it**
Register time is ordered finite-record update	Operational definition	Finite observers use temporal order without any record ordering, update history, clock record, causal marker, dependency edge, or memory of change
Causal dependency precedes temporal labels	Structural proposition	Stable temporal order is obtained solely from labels even when update dependencies and causal traces are removed or scrambled
Non-injective update induces an operational arrow	Conditional theorem	A finite register repeatedly overwrites or compresses records yet reconstructs erased distinctions from the current finite record without external memory, hidden reservoirs, or additional records
Synchronization is a finite-channel record-exchange problem	Relativity-compatible bridge	Bounded observers establish global simultaneity without signal exchange, finite bandwidth, latency, synchronization records, or assumptions about propagation
Buffers and latency can distort recorded order	Testable prediction	Load-dependent latency and finite buffers never produce delayed records, order inversions, timestamp gaps, or apparent cause-effect reversal under controlled acquisition stress
Finite clock precision coarsens time	Operational bridge	Arbitrarily dense events are totally ordered by a bounded finite clock without tick collisions, synchronization error, or external ordering information
Dissipative projection carries housekeeping cost when physically implemented by irreversible reuse	Physical bridge	Repeated logically irreversible temporal record reuse violates generalized Landauer accounting after all reservoirs, correlations, feedback records, and work sources are included

Without a record-bearing ordering structure, temporal comparison is not operationally available to that finite system.

O2 therefore reformulates time at the register level. If records could be updated reversibly and indefinitely logged, past states could in principle remain reconstructible. But finite systems have finite memory, finite clocks, finite buffers, finite update throughput, finite communication bandwidth, and finite energy. Under sustained acquisition, a finite temporal register must overwrite, compress, project, externalize, or garbage-collect records. These operations are generally many-to-one. The operational arrow is the direction in which non-injective record updates accumulate.

Coordinate time, clock time, and register time

To prevent a common misunderstanding, O2 distinguishes coordinate time from the finite time available to a physical observer. Table II summarizes the terminology.

The central boundary statement is:

$$\text{O2 does not deny coordinate time; it models finite access to temporal order.} \tag{1}$$

Coordinate time may remain an excellent parameter in physical equations. Register time is what a finite observer can use after records, clocks, buffers, and updates have done their work.

Main thesis

The main thesis is:

$$\text{register time} = \text{causally ordered irreversible distinction update.} \tag{2}$$

A finite temporal register produces records

$$r_n = (z_n, \kappa_n, d_n, \eta_n), \tag{3}$$

where z_n is a record state, κ_n is a clock label or ordering marker, d_n is a dependency pointer or causal trace, and η_n is metadata controlling stability, confidence, latency, provenance, or synchronization. A time label is not the root of order. It is a finite annotation attached to an update process.

A generic temporal update has the form

$$z_{n+1} = U_n(z_n, x_{n+1}, b_n, \theta_n), \tag{4}$$

where x_{n+1} is incoming measurement content, b_n is buffer state, and θ_n represents clock, control, synchronization,

TABLE II. Coordinate time, clock time, and register time. O2 studies the finite record layer, not a replacement for coordinate time in equations.

Term	Meaning	Status in O2
Coordinate time t	Continuous parameter used in physical equations or coordinate charts	Idealized mathematical parameter; not denied
Proper time τ	Time measured along a worldline by an ideal or physical clock	Compatible with O2; implemented through physical clock records
Clock time	Readout of a clock device or oscillator register	Finite labels with tick width, drift, jitter, wraparound, and synchronization limits
Register time	Ordered finite-record structure available to a bounded observer	Main object of O2
Historical order	Causal/update dependency order among records	More primitive than labels inside the finite register

or update context. If U_n is non-injective over possible pre-update states, no inverse exists inside the accessible finite record. The missing distinctions are not merely unknown; they are no longer carried by the register.

Relation to earlier time drafts

Earlier time drafts argued that truncation forms a semigroup without inverses and that the arrow of time follows from algebraic irreversibility. The current O2 version narrows and operationalizes that claim. It does not claim to solve the entire arrow-of-time problem or to remove the need for statistical mechanics and cosmology. It uses the semigroup idea only at the level of finite records: update operations that overwrite, compress, or discard distinctions are non-invertible maps on the register state space. The empirical content is the predicted pattern of finite-register signatures: reconstruction-depth limits, non-injectivity growth, synchronization uncertainty, clock-tick collisions, buffer-shaped latency, apparent causal inversion, and temporal-collapse behavior in finite-window systems.

RELATED WORK

The arrow of time and irreversibility

The asymmetry of time has been central to physics since Eddington's discussion of the arrow of time [3]. Statistical-mechanical accounts relate macroscopic irreversibility to entropy increase, typicality, and boundary conditions, while acknowledging the role of low-entropy initial conditions or the Past Hypothesis [4–6]. Nonequilibrium thermodynamics and dissipative structures explain how local order can persist through entropy export [7, 8]. FDS-O2 does not replace these accounts. It addresses the lower operational layer: a finite observer accesses temporal order only through finite records, and finite record updates can be non-invertible even when a larger embedding dynamics is modeled reversibly.

Information, erasure, and reversible computation

Landauer's principle links logically irreversible erasure to thermodynamic cost [9]; Bennett showed that computation can be arranged reversibly when sufficient memory and garbage management are available [10]. Modern information thermodynamics refines this accounting by including feedback, correlations, nonequilibrium reservoirs, and stochastic dynamics [8, 11]. O2 uses this tradition conservatively. It does not claim that every update is a Landauer erasure. It claims that finite temporal registers under sustained load must eventually trade memory growth against irreversible record reuse or externalization.

Causal ordering and finite clocks

In distributed systems, Lamport showed that event ordering is not merely a clock label but a structure constrained by causal communication and message order [14]. In causal-set approaches, causal order is treated as a primitive partial order for spacetime structure [15]. O2 is compatible with these traditions but does not identify register time with spacetime discreteness. Its primitive is finite record update: a bounded physical system uses a partial order only when dependency information, clock records, or causal traces survive in the register.

FINITE TEMPORAL REGISTERS

Definition 1 (Finite temporal register)**.** *A finite temporal register is a finite distinction-register that can record, order, update, timestamp, retain, synchronize, and communicate distinctions over a finite window. It consists of a record state space $\mathcal{Z}$, update maps U_n, clock labels κ_n, buffer states b_n, synchronization messages m_n, and retention rules specifying which past distinctions remain accessible.*

Definition 2 (Temporal record)**.** *A temporal record is a tuple*

$$r_n = (z_n, \kappa_n, d_n, \eta_n), \tag{5}$$

where z_n is a record class, κ_n is a finite clock label or ordering marker, d_n is a dependency pointer or causal *trace, and η_n is metadata controlling stability, confidence, latency, provenance, or synchronization.*

Definition 3 (Operational temporal order)**.** *For records r_i and r_j, write $r_i \prec_{\mathcal{O}} r_j$ when the finite observer $\mathcal{O}$ has an accessible dependency edge, clock comparison, append order, causal trace, or synchronization record supporting the claim that r_i precedes r_j. The relation $\prec_{\mathcal{O}}$ is observer-relative but not arbitrary: it is constrained by retained records, causal communication, clock precision, and update dependence.*

The relation $\prec_{\mathcal{O}}$ need not be a total order. Two events may be incomparable to a finite observer if they fall within the same clock tick, arrive through unsynchronized channels, are buffered with ambiguous latency, or have had their ordering metadata erased.

Causal dependency before temporal labels

O2 treats causal/update dependency as more primitive than clock labels. Clock labels help compare records, but they do not by themselves create the update chain that makes one record depend on another.

Definition 4 (Update dependency)**.** *Let z_i and z_j be accessible record states. Write*

$$z_i \prec_U z_j \tag{6}$$

if z_j is generated by a finite composition of update maps whose domain includes a state carrying z_i or a dependency trace of z_i.

Proposition 1 (Causal dependency is prior to register time)**.** *If a finite observer judges z_i to be operationally before z_j on the basis of its own register, then at least one of the following must be accessible: an update dependency $z_i \prec_U z_j$, a clock comparison $\kappa_i < \kappa_j$, an append-order trace, or a communication/synchronization record. Among these, update dependency is the causal substrate: a label can be copied or corrupted, but the formation of z_j from z_i requires a physical update chain.*

Sketch. A temporal comparison is an operational claim made by a finite register. Such a claim must be carried by some retained distinction: a dependency edge, a label, a log order, or a synchronization record. A clock label is itself a record produced by an update process. If all dependency, append, and communication traces are erased or scrambled, labels alone can no longer determine which record causally contributed to which later record. Thus labels are useful finite annotations, but causal/update dependence supplies the physical ordering substrate within the register. ☐

This proposition connects O2 to causal graphs and distributed-system logical clocks without reducing physical time to software time. O2 claims only that finite systems need retained dependency information to use temporal order.

TEMPORAL CAPACITY AND BUDGET CROSSING

A finite register has finite temporal capacity. A schematic capacity ledger is

$$C_{\text{time}}(t) = \min\{C_{\text{rec}}, C_{\text{clk}}, C_{\text{buf}}, C_{\text{upd}}, C_{\text{log}}, C_{\text{ext}}^{\text{eff}}, C_{\text{sync}}\}. \tag{7}$$

The terms denote stable record capacity, clock-label capacity, buffering capacity, update-throughput capacity, log memory, effective externalized access, and synchronization capacity over a window.

For a task family Ψ_t, distortion tolerance ε, and retention/verification window τ, let

$$R_{\text{min}}^{(\tau)}(\varepsilon; \Psi_t) \tag{8}$$

be the minimal temporal distinction demand required to meet the task. The O2 temporal capacity deficit is

$$\Delta_{\text{O2}}(t) = R_{\text{min}}^{(\tau)}(\varepsilon; \Psi_t) - C_{\text{time}}(t). \tag{9}$$

Theorem 1 (Finite temporal update-exit theorem)**.** *Let $\mathcal{O}$ be a finite temporal register. If*

$$R_{\text{min}}^{(\tau)}(\varepsilon; \Psi_t) > C_{\text{time}}(t), \tag{10}$$

then full-fidelity temporal reconstruction at distortion ε is impossible unless at least one exit occurs: coarser clock ticks, increased latency, buffering, externalized logging, lossy compression, record overwrite, garbage collection, task relaxation, boundary enlargement, or failure.

Proof. $R_{\text{min}}^{(\tau)}$ is the minimum number of task-relevant temporal distinctions required to meet the ordering target. C_{time} is the maximum number of temporal distinctions the finite register can stabilize, label, synchronize, update, retain, and access in the same window. If demand exceeds capacity, a full-fidelity temporal record would require distinctions not represented by any admissible record state, log, timestamp, synchronization message, or update sequence. The register must therefore reduce demand, increase capacity, store backlog, externalize records, erase or compress prior records, relax the task, or fail. ☐

The empirical content of the theorem is the predicted co-occurrence of exits near independently estimated crossing: reconstruction depth saturates, overwrite or garbage-collection events increase, clock ticks collide, synchronization ambiguity grows, buffer latency rises, local order inversions appear, and temporal reports become compressed, delayed, merged, or missing.

NON-INJECTIVE UPDATE AND DISSIPATIVE PROJECTION

Let X denote the set of possible pre-update register states and Z the set of post-update record states. A physical update implements a map

$$U : X \to Z. \tag{11}$$

If U is injective and the register retains the required garbage information, the update can be reversed in principle. If U is many-to-one, then distinct pre-update states can produce the same post-update record.

Definition 5 (Non-injectivity loss)**.** *For a finite update $U : X \to Z$, define the preimage multiplicity of $z \in Z$ as*

$$m_U(z) = |U^{-1}(z)|. \tag{12}$$

The average non-injectivity loss is

$$\mathcal{L}_U = \mathbb{E}_{z\sim P_Z}\left[\log_2 m_U(z)\right]. \tag{13}$$

Equivalently, under the induced joint distribution on pre-update and post-update states,

$$\mathcal{L}_U = H(X|Z). \tag{14}$$

This quantity is the number of bits of compatible past that remain unresolved after reading the current record. It is a direct operational measure of lost reconstructibility.

Proposition 2 (Arrow strength from non-injectivity)**.** *For a finite register, the operational arrow associated with an update U strengthens as $\mathcal{L}_U = H(X|Z)$ increases. If $\mathcal{L}_U = 0$, the current record preserves enough information to reconstruct the pre-update state. If $\mathcal{L}_U > 0$, multiple possible past states are compatible with the same current record.*

If the many-to-one update is physically implemented through irreversible reset, overwrite, compression, or garbage collection, generalized Landauer accounting gives the lower-bound form

$$Q_{\min} \geq k_{\mathrm{B}} T \ln 2\, \mathcal{L}_U, \tag{15}$$

when $\mathcal{L}_U$ is measured in bits and the usual thermodynamic assumptions and caveats are satisfied. Recent reviews emphasize that practical erasure processes often dissipate more than the ideal Landauer bound and that finite-time, finite-bath, nonequilibrium, non-Markovian, and quantum regimes require generalized accounting [22]. The abstract mathematical map is not automatically dissipative. The heat belongs to the physical implementation when discarded distinctions are irreversibly erased or overwritten.

Definition 6 (Dissipative projection)**.** *A dissipative projection is a physical many-to-one record update*

$$\Pi_{\mathrm{diss}} : X \to Z \tag{16}$$

that discards distinctions and stabilizes a smaller record space through irreversible overwrite, reset, many-to-one compression, or garbage collection. Its abstract projection structure gives non-injectivity; its physical implementation supplies thermodynamic cost.

Compatible histories. A large value of $H(\mathrm{Past} \mid Z_t)$ means that the current finite record Z_t is compatible with many possible prior histories. The observer has not retained a unique reconstructible past; it has retained an equivalence class of histories that project to the same present record. This resembles the language of compatible or consistent histories [19], but the claim here is operational rather than interpretive: FDS-O2 does not posit branching worlds or a new quantum interpretation. It only states that finite non-injective record updates merge multiple past trajectories into the same accessible present state.

The family of projections P_K retaining only depth K obeys

$$P_K \circ P_L = P_{\min(K,L)}. \tag{17}$$

This is a semigroup under composition. It is not a group when discarded temporal distinctions cannot be reconstructed from the retained finite record.

SYNCHRONIZATION, LATENCY, AND APPARENT CAUSAL INVERSION

Synchronization bottleneck

Consider two finite observers O_A and O_B with local register times $\{a_i\}$ and $\{b_j\}$. To establish a shared temporal order, they must exchange records. Let C^{AB}_{chan} be the bit capacity of the record-exchange channel during a synchronization interval, and let L_{AB} be latency with uncertainty δL_{AB}. If the order relation to be established has entropy $H(\Delta_{AB})$, a schematic lower bound on synchronization uncertainty is

$$\delta t_{\mathrm{sync}} \gtrsim \delta L_{AB} + \frac{H(\Delta_{AB})}{C^{AB}_{\mathrm{chan}}} + \delta t_{\mathrm{clock}}. \tag{18}$$

This is not a derivation of relativity. It states the operational fact that cross-boundary simultaneity claims require finite signal exchange, finite channels, and finite clock records. In relativistic settings, finite signal propagation speed supplies an additional physical boundary on record exchange, consistent with the operational constraints on simultaneity introduced by special relativity [16].

Latency-induced apparent causal inversion

Let two source events obey the source order $e_i \prec e_j$ with times $t_i < t_j$. The finite register records them at

$$\hat{t}_i = t_i + \ell_i, \qquad \hat{t}_j = t_j + \ell_j, \tag{19}$$

where ℓ_i and ℓ_j are load-dependent latencies. A recorded inversion occurs when

$$\hat{t}_i > \hat{t}_j, \tag{20}$$

which is equivalent to

$$\ell_i - \ell_j > t_j - t_i. \tag{21}$$

This creates an apparent effect-before-cause record without violating physical causality. The violation is in the finite register's acquired order, not in the source process.

A sufficient engineering criterion for maintaining causally consistent records is

$$C_{\text{upd}} > h_{\text{env}} + h_{\text{sync}} + h_{\text{overhead}}, \tag{22}$$

where h_{env} is the entropy rate of the relevant event stream, h_{sync} is the record-exchange burden, and h_{overhead} is the cost of metadata, timestamps, buffers, and error correction. If the update capacity falls below this demand, the register must delay, drop, merge, reorder, or coarsen records.

Finite clocks and partial orders

A clock is itself a finite distinction-register. It maps physical change into labels

$$\kappa : I \to \{0, 1, \ldots, K-1\} \tag{23}$$

within an operational interval I. If multiple events map to the same label, their order is not available from the clock alone. Clock capacity over a window can be written schematically as

$$C_{\text{clk}}(\tau) = \log_2 N_{\text{ticks}}(\tau) - H_{\text{jitter}} - H_{\text{sync}} - H_{\text{wrap}}. \tag{24}$$

Finite clocks therefore naturally produce partial orders under high event density, jitter, drift, wraparound, or synchronization limits.

NUMERICAL MODELS AND SIMULATIONS

The simulations are deterministic synthetic demonstrations. They are not fits to physical detector data, clock data, cosmological observations, AI-system benchmarks, or human time perception. They provide a reproducible diagnostic template for the O2 model. All figures and CSV outputs are generated by `code/generate_results.py`.

Memory fill and non-injectivity

The first simulation contrasts cumulative distinction history with finite retained record capacity. Incoming distinctions accumulate over update index. Once cumulative history exceeds capacity, lost bits grow. The non-injectivity index is $H(\text{past}|\text{record})$, and the arrow-strength proxy increases as memory fill forces more aggressive projection.

Fixed overwrite versus append-only logging

The second simulation compares fixed-memory overwrite with expanded append-only logging. Fixed memory bounds storage at the cost of immediate overwrite. Append-only logging preserves reconstructibility longer but accumulates storage demand and later cleanup cost.

Latency-induced apparent causal inversion

The third simulation sends an ordered event stream through a finite buffer with load-dependent latency. When latency differences exceed source event gaps, the recorded order locally inverts. This produces apparent effect-before-cause observations without violating source causality.

Finite clock precision

The fourth simulation maps a bursty event stream into finite clock ticks. As tick width grows, multiple events collide in the same clock label and temporal order becomes partial or ambiguous.

Synchronization bottleneck

The fifth simulation models two finite observers attempting to establish shared order through a finite record-exchange channel. Synchronization uncertainty decreases with channel capacity but remains bounded below by latency uncertainty, jitter, and clock uncertainty.

Projection semigroup and non-injectivity

The sixth simulation uses a toy family of projections that retain only depth k from an initial depth K_0. The number of compatible microhistories per record grows as retained depth falls. The loss $H(\text{past}|\text{record})$ quantifies how much past remains unreconstructible.

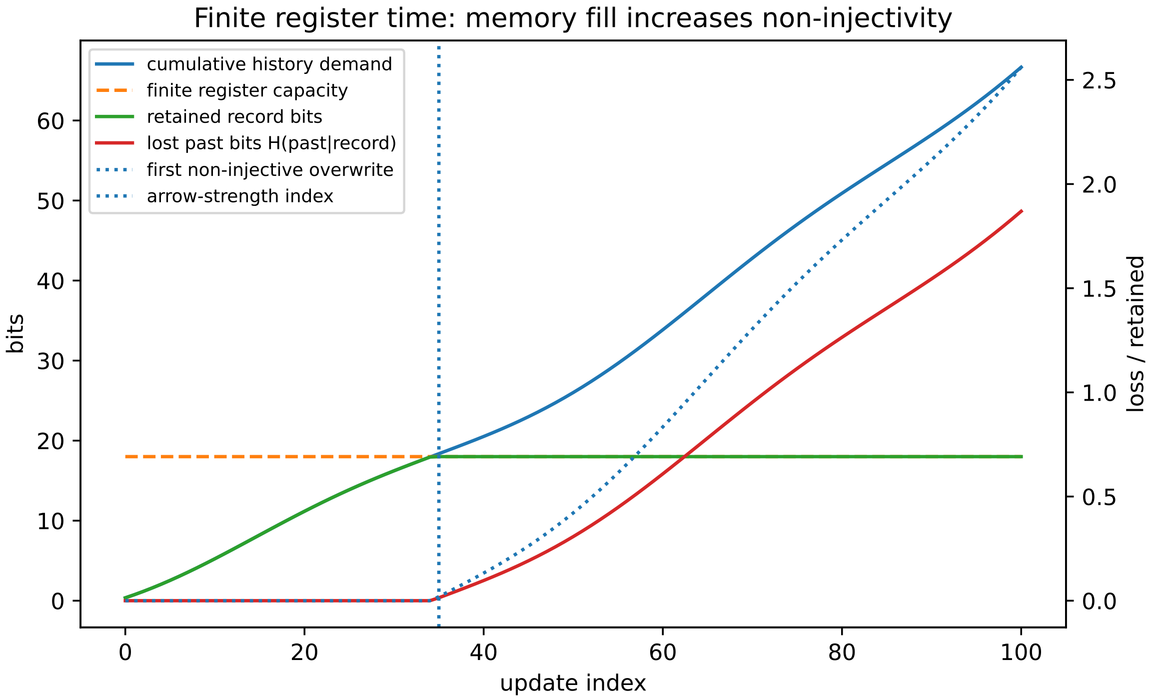

FIG. 1. Finite register time: memory fill increases non-injectivity. Cumulative history demand eventually exceeds finite register capacity. The retained record saturates while compatible past information $H(\text{past}|\text{record})$ grows. The arrow-strength index is a normalized loss of reconstructible past.

Long-context temporal collapse

The seventh simulation treats a long-context AI-like system as a finite temporal register. Once task history exceeds the context window, exact historical order is replaced by compression, summary, or discard. The model predicts temporal step confusion, false memory of order, and plan drift when task-history demand crosses finite context capacity.

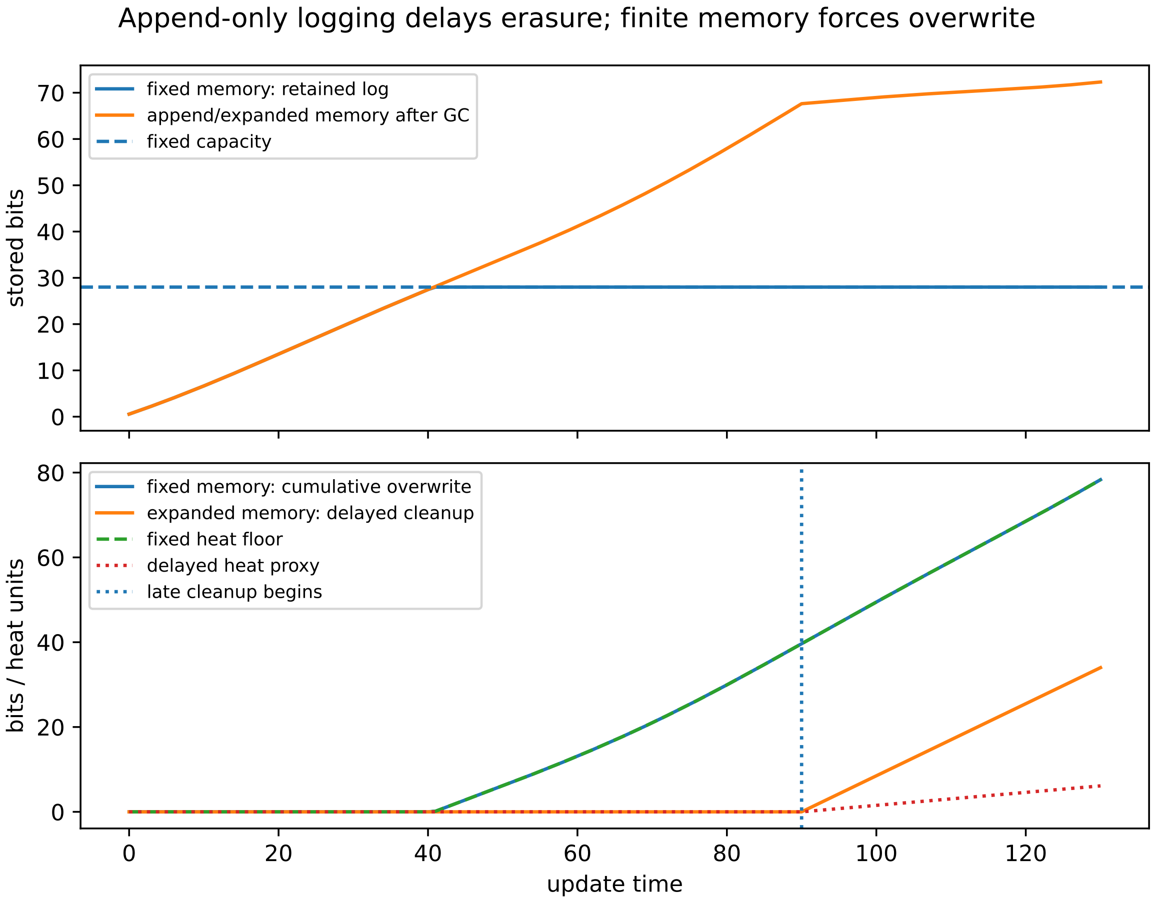

FIG. 2. Append-only logging delays erasure; finite memory forces overwrite. Fixed memory keeps storage bounded but increases overwritten distinctions and heat-floor proxy. Expanded memory lowers immediate erasure but accumulates physical storage and delayed garbage collection.

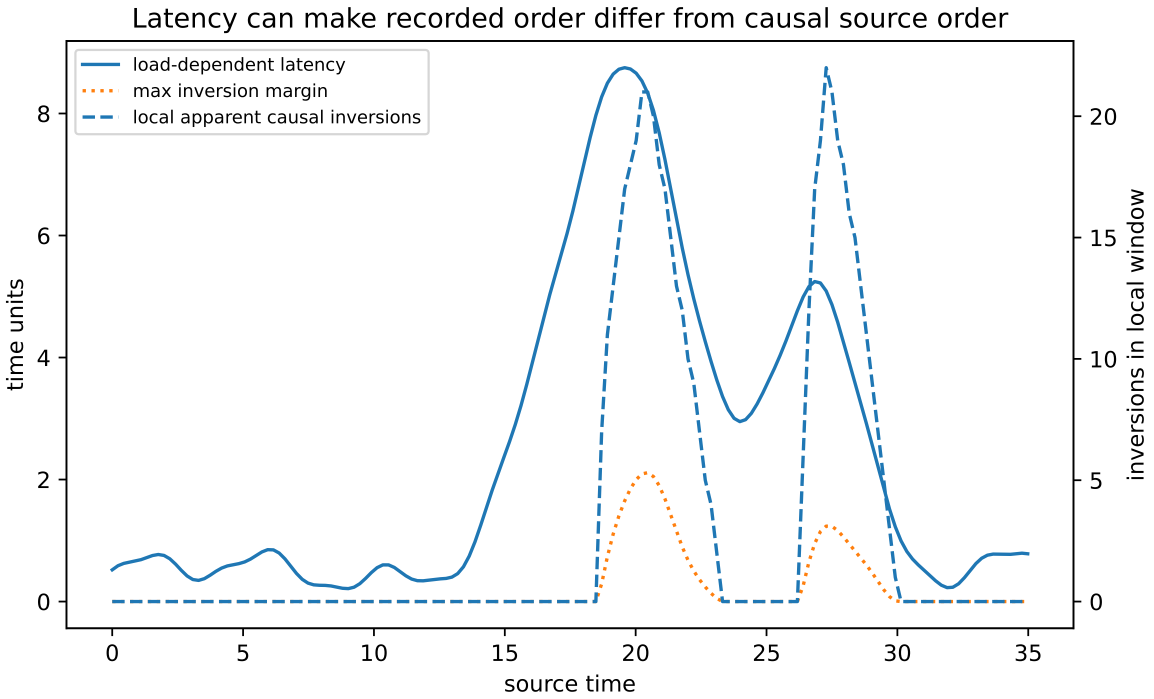

FIG. 3. Latency can make recorded order differ from causal source order. Under load, latency differences exceed event separation and generate local apparent causal inversions in the record stream.

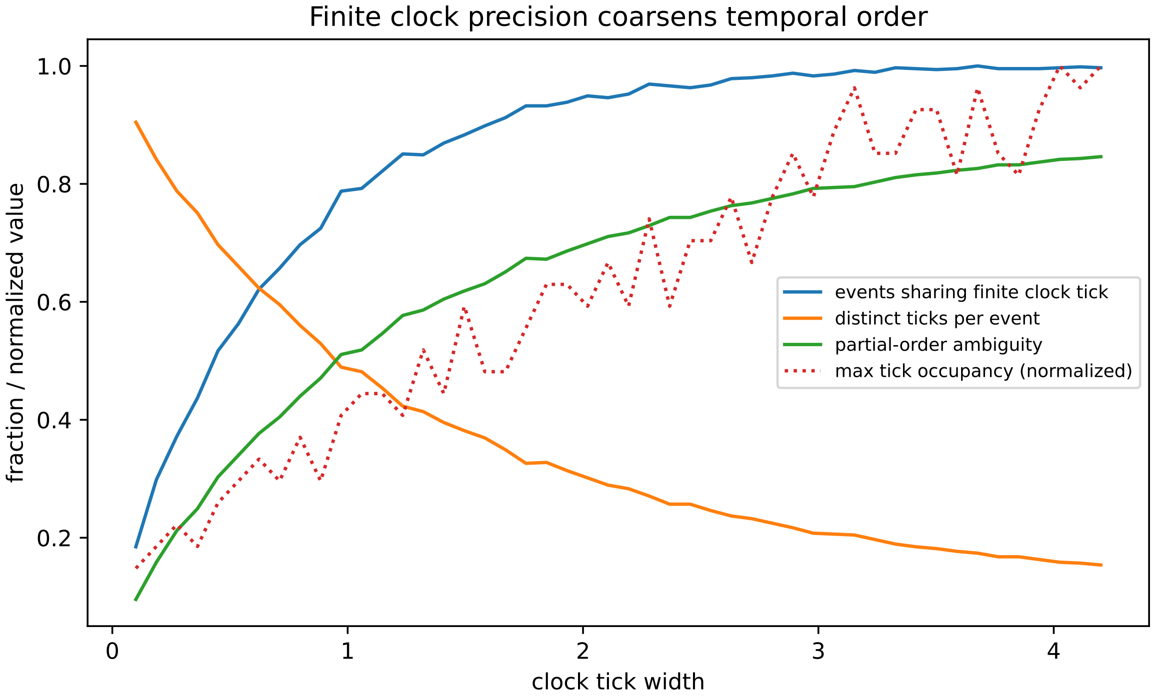

FIG. 4. Finite clock precision coarsens temporal order. Larger tick widths increase collisions and partial-order ambiguity while reducing the number of distinct labels available per event.

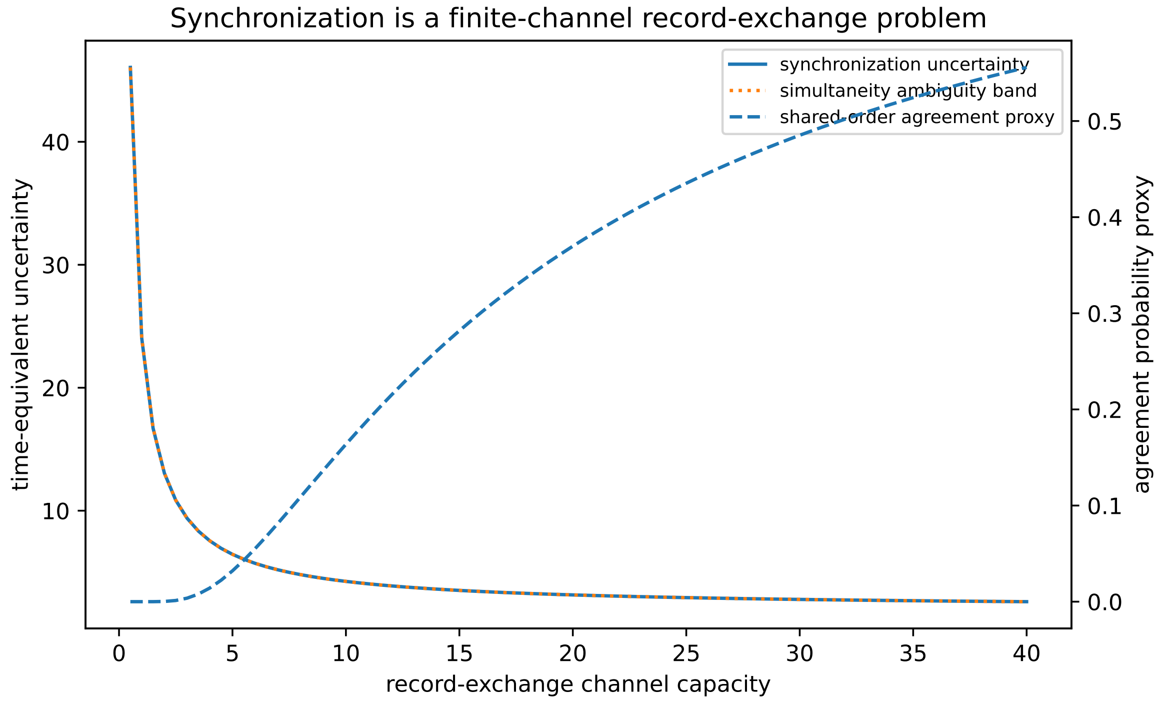

FIG. 5. Synchronization is a finite-channel record-exchange problem. As communication capacity increases, shared-order uncertainty decreases, but latency and jitter impose an ambiguity band. This is an operational account, not a derivation of relativistic simultaneity.

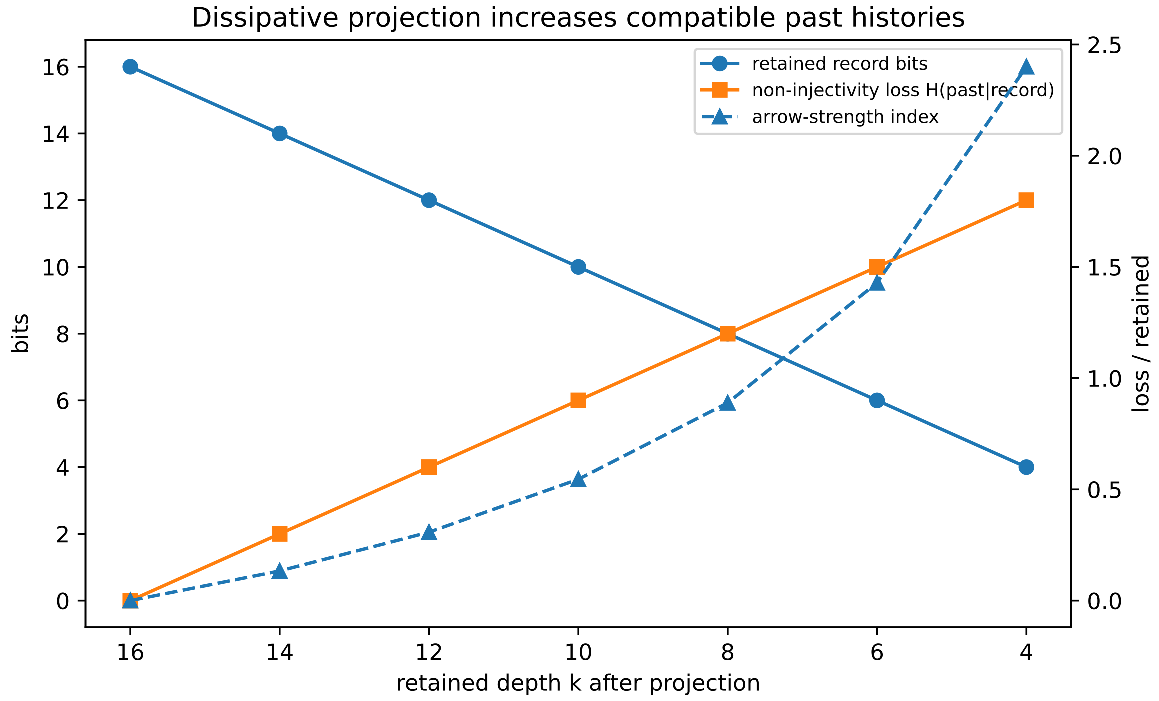

FIG. 6. Dissipative projection increases compatible past histories. Retaining fewer temporal distinctions increases non-injectivity loss $H(\text{past}|\text{record})$ and the arrow-strength index.

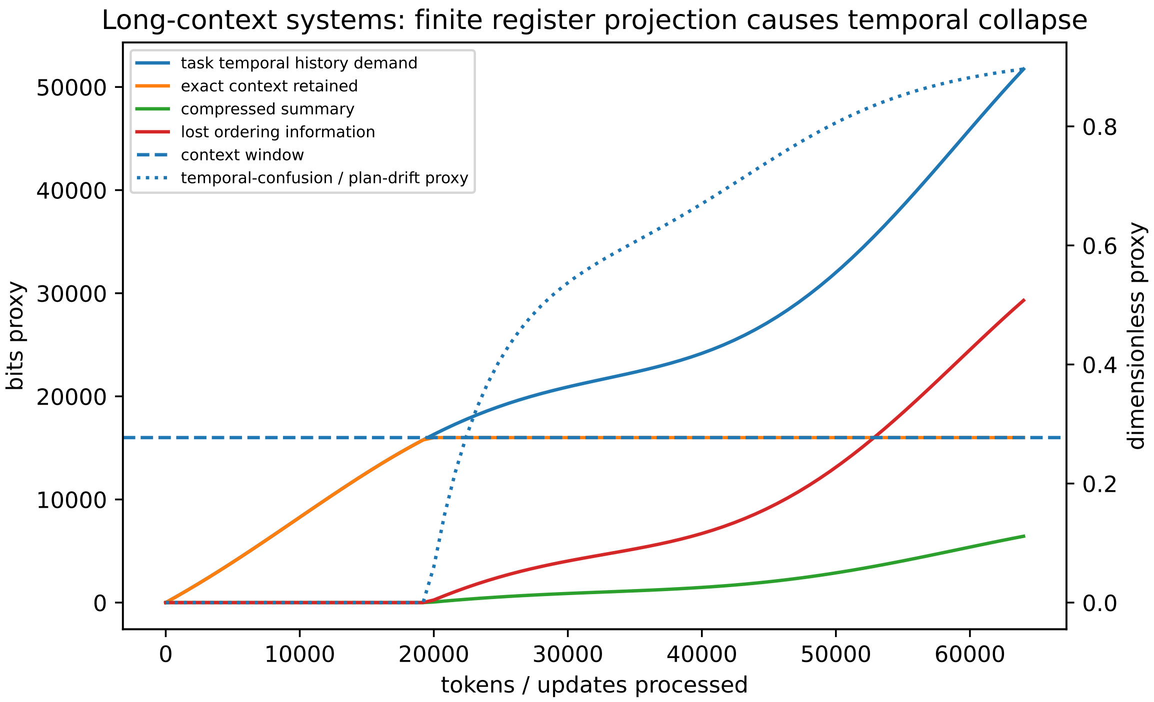

FIG. 7. Long-context systems as finite temporal registers. When task-history demand exceeds exact context capacity, the system must summarize or discard. The lost-order proxy increases together with temporal-confusion and plan-drift proxies. This is an engineering projection, not a claim about subjective experience.

DOMAIN-SPECIFIC PROJECTIONS

Distributed acquisition and control

In high-rate acquisition, robotics, and distributed control, apparent temporal failures often occur upstream of planning or inference. Buffers fill, events are timestamped late, logs are garbage-collected, channels desynchronize, and causally earlier events can be recorded after later events. O2 provides a finite-register diagnostic: if $C_{\rm upd}$ and $C_{\rm sync}$ fall below event entropy rate and synchronization burden, temporal order degrades before the controller sees the world.

Long-context AI and temporal-collapse failure

Transformer-based and large language model systems provide a useful engineering example of finite temporal registers with bounded context windows [17, 18]. A long-context AI system can be treated as a finite temporal register with context capacity $C_{\rm ctx}$. Empirical studies show that information access can degrade with position and context length, even when the nominal context window is large [20, 21]. Let $\mathcal{H}_t$ be the task history and $R_{\rm task}(t)$ the number of temporal distinctions required to preserve step order, commitments, dependencies, and causal state. If

$$R_{\rm task}(t) > C_{\rm ctx}, \tag{25}$$

then the system must apply a projection

$$\pi_{\rm ctx} : \mathcal{H}_t \to \widetilde{\mathcal{H}}_t. \tag{26}$$

The resulting history is not the original past; it is a compressed, truncated, or distributional representation. This can produce temporal hallucination, step-order confusion, lost commitments, plan drift, and false memory of earlier instructions. This is an engineering analogy and diagnostic projection, not a claim that language models possess physical time experience.

Human and biological time

Biological time perception also depends on finite record formation, attention, working memory, and update rate. O2 suggests that some timing distortions can be interpreted as finite-register effects: event density exceeds update capacity, attention gates update frequency, or memory compression merges adjacent temporal distinctions. This is a projection for future empirical work, not a biological theory completed here.

EXPERIMENTAL PROTOCOLS

Protocol 1 (Finite temporal budget-crossing test)**.** *Specify a finite register boundary, clock source, record carrier, buffer, and retention window. Estimate* $C_{\rm rec}$, $C_{\rm clk}$, $C_{\rm buf}$, $C_{\rm upd}$, $C_{\rm log}$, $C_{\rm ext}^{\rm eff}$, *and* $C_{\rm sync}$. *Increase event entropy rate, synchronization burden, or history length until* $R_{\rm min}^{(\tau)}$ *crosses* $C_{\rm time}$. *Measure reconstruction depth, tick collisions, buffer occupancy, order inversions, overwrite events, externalization, and heat proxy.*

Protocol 2 (Causal-dependency degradation test)**.** *Feed a controlled causal event graph into a finite logging system. Remove or compress dependency edges while keeping raw timestamps available. O2 predicts that temporal order judgments should degrade when dependency traces are removed, even if clock labels remain.*

Protocol 3 (Latency inversion test)**.** *Feed a timestamped ordered event stream through a finite buffer under increasing load. Compare source order with recorded order. O2 predicts local order inversions when latency variability exceeds event separation.*

Protocol 4 (Synchronization bottleneck test)**.** *Let two observers synchronize through a channel of adjustable capacity and latency. Measure shared-order agreement and simultaneity ambiguity as a function of channel capacity. O2 predicts a finite ambiguity band controlled by latency uncertainty and record-exchange capacity.*

Protocol 5 (Projection-depth reconstruction test)**.** *Retain only the last* K *layers, bits, or dependency edges of an event history. Measure the number of compatible past histories consistent with the current record. O2 predicts growth of* $H(\text{past}|\text{record})$ *as retained depth decreases.*

LIMITATIONS AND FALSIFICATION

First, O2 is an operational finite-record model, not a complete theory of time. It does not solve the cosmological low-entropy initial condition problem. Second, it does not deny reversible microdynamics. It states that finite accessible records can evolve by non-injective update maps even when a larger embedding dynamics is reversible. Third, it does not define a preferred foliation of spacetime. Each observer's temporal access is implemented by local clocks, causal communication, and finite record order. Fourth, Landauer accounting applies only to physically implemented logically irreversible operations under relevant thermodynamic assumptions. Fifth, simulations in this paper are synthetic demonstrations, not empirical tests.

The strong version of FDS-O2 would be weakened or rejected by any of the following:

1. a finite physical observer that uses temporal order without any record order, clock label, update trace, causal dependency, synchronization record, or memory of change;

2. stable temporal order obtained solely from labels after dependency edges, causal traces, and synchronization metadata are removed or scrambled;

3. repeated finite-memory overwrite or many-to-one compression with perfect reconstruction of erased distinctions from the current finite record alone;

4. arbitrary event density totally ordered by bounded clocks without tick collisions, latency uncertainty, or external ordering information;

5. bounded observers establishing global simultaneity without finite signal exchange, latency assumptions, clock synchronization, or record communication;

6. no change in reconstruction depth, latency, ordering ambiguity, buffer occupancy, synchronization uncertainty, or erasure events when temporal demand crosses independently estimated temporal capacity;

7. logically irreversible temporal record reuse below generalized Landauer accounting after all reservoirs, feedback records, correlations, and work sources are included.

CONCLUSION

A finite observer does not use time as a magical external primitive. It uses temporal order through finite records: timestamps, dependency edges, update traces, logs, residues, memory states, communication messages, and clocks. Once this is made explicit, time inherits the finite-register constraints developed in O1. A temporal register has limited memory, limited clock precision, limited update throughput, limited buffer capacity, limited synchronization bandwidth, and limited thermodynamic budget.

The main result is operational: when temporal demand exceeds temporal capacity, the observer cannot maintain full-fidelity order. It must coarsen clock ticks, buffer, delay, overwrite, compress, externalize, garbage-collect, relax the task, or fail. Causal dependency is prior to temporal labeling because labels themselves are finite records generated by physical update chains. The arrow of register time is the direction in which non-injective updates accumulate. Future records can preserve traces of the past, but erased distinctions cannot be reconstructed from the current finite record alone.

In this operational sense, the arrow available to a finite observer is the price of finite memory: to keep updating, the system must forget, compress, overwrite, or externalize parts of its own past. O2 therefore supplies the operational trident's second anchor. O1 states what it means to measure as a finite record. O2 states what it means for records to become temporally ordered under causal, irreversible update. O3 should next treat the macroscopic thermodynamic cost of maintaining such finite records under memory reuse.

Finally, FDS-O2 suggests an operational time-resolution/energy-throughput tradeoff. A finite register can use finer temporal ticks only by increasing its update rate, synchronization rate, memory turnover, or irreversible housekeeping operations. If each tick requires stabilizing b_{tick} bits and the tick interval is Δt, the update throughput must satisfy $C_{\mathrm{upd}} \gtrsim b_{\mathrm{tick}}/\Delta t$. When these updates require reset, overwrite, compression, or garbage collection, the housekeeping power has a Landauer-style lower bound $P_{\mathrm{hk}} \gtrsim k_{\mathrm{B}} T \ln 2 \cdot b_{\mathrm{erase}}/\Delta t$. This is not a derivation of the quantum energy-time uncertainty relation. It is a macroscopic finite-register constraint: sharper operational time requires more physical updating.

Notation Summary

Simulation Parameters

The simulations are deterministic and use fixed synthetic parameters in `code/generate_results.py`. Figure 1 uses an update index $n \in [0, 100]$, a rising input-bit stream, and an 18-bit register capacity. Figure 2 compares a 28-bit fixed-memory register with append-only logging and delayed garbage collection after update 90. Figure 3 uses 160 source events with load pulses that increase latency and cause local order inversion. Figure 4 maps a bursty event stream into finite clock ticks with tick widths from 0.10 to 4.2. Figure 5 uses a synchronization load of 22 bits and finite channel capacities from 0.5 to 40 bits per exchange interval. Figure 6 uses an initial depth $K_0 = 16$ and retained depths $k \in \{16, 14, 12, 10, 8, 6, 4\}$. Figure 7 uses a 16,000-token context window proxy and a growing temporal task-history demand. No proprietary data, detector data, clock data, cosmological data, AI benchmark data, or human-subject data are used.

Reproducibility Checklist

1. Code availability: all simulation code is included in the replication package.

2. Deterministic execution: the random seeds are

TABLE III. FDS-O2 notation summary.

Symbol	Meaning
$\mathcal{O}$	finite observer or distinction-register
$\mathcal{R}$	finite temporal register
$\mathcal{Z}$	finite record state space
z_n	record state after update n
r_n	temporal record tuple $(z_n, \kappa_n, d_n, \eta_n)$
κ_n	finite clock label or ordering marker
d_n	dependency pointer or causal trace
U_n	update map from current record and input to next record
$\mathcal{H}_n$	possible histories up to update n
π_n	projection from histories to finite records
$\prec_{\mathcal{O}}$	operational temporal order available to observer $\mathcal{O}$
$\prec_U$	causal/update dependency relation
C_{time}	accessible temporal ordering capacity
C_{sync}	synchronization capacity between finite observers
C_{clk}	clock-label capacity after jitter and synchronization uncertainty
C_{upd}	update-throughput capacity
$R_{\min}^{(\tau)}(\varepsilon; \Psi)$	minimal task-relevant temporal distinction demand
Δ_{O2}	temporal capacity deficit
$\mathcal{L}_U$	non-injectivity loss $H(X\|Z)$ of update U
P_{hk}	housekeeping power associated with irreversible record reuse

fixed.

3. Figure reproduction: running `python code/generate_results.py` regenerates all figures and CSV outputs.
4. Data status: all numerical outputs are synthetic demonstrations generated from the stated model.
5. Platform independence: the code uses standard Python scientific libraries.

Boundary of Applicability

FDS-O2 applies to systems that implement finite temporal records: they have record carriers, finite clocks, finite buffers, finite update throughput, finite retention windows, finite synchronization channels, and finite memory reuse constraints. It does not apply to ideal mathematical time parameters with unbounded precision, nor does it make claims about nonunitary fundamental dynamics or a preferred cosmic foliation. It is compatible with reversible microscopic dynamics and standard relativity when those theories are interpreted as descriptions of larger state spaces, coordinate conventions, or invariant relations rather than as finite observer records.

CODE AVAILABILITY

The simulation code used to generate Figs. 1–7 is included in the accompanying replication package under `code/generate_results.py`. Running the script regenerates all figures (PDF and PNG) and CSV tables in a single pass.

AI ASSISTANCE DISCLOSURE

AI-assisted tools were used for language polishing, structural feedback, LaTeX drafting support, and code-debugging assistance. The author reviewed and edited all content and remains responsible for all claims, references, simulations, and conclusions. No AI system is listed as an author.

---

[1] Y. Wu, "Observer as a Finite Distinction Register: Measurement Capacity, Dynamic Bottlenecks, and Budget-Crossing Signatures," Zenodo (2026), doi:10.5281/zenodo.20248792.

[2] Y. Wu, "Finite Distinguishability Budgets and Maintenance Bounds for Physical Observers," Zenodo (2026), doi:10.5281/zenodo.20234249.

[3] A. S. Eddington, *The Nature of the Physical World.* Cambridge University Press, Cambridge (1928).

[4] D. Z. Albert, *Time and Chance.* Harvard University Press, Cambridge, MA (2000).

[5] S. Carroll, *From Eternity to Here: The Quest for the Ultimate Theory of Time.* Dutton, New York (2010).

[6] R. Penrose, *The Road to Reality: A Complete Guide to the Laws of the Universe.* Jonathan Cape, London (2004).

[7] G. Nicolis and I. Prigogine, *Self-Organization in Nonequilibrium Systems.* Wiley, New York (1977).

[8] U. Seifert, "Stochastic thermodynamics, fluctuation theorems and molecular machines," Reports on Progress in Physics **75**, 126001 (2012).

[9] R. Landauer, "Irreversibility and heat generation in the computing process," IBM Journal of Research and Development **5**, 183–191 (1961).

[10] C. H. Bennett, "The thermodynamics of computation–a review," International Journal of Theoretical Physics **21**, 905–940 (1982).

[11] J. M. R. Parrondo, J. M. Horowitz, and T. Sagawa, "Thermodynamics of information," Reviews of Modern Physics **87**, 45–67 (2015).

[12] A. Berut et al., "Experimental verification of Landauer's principle linking information and thermodynamics," Nature **483**, 187–189 (2012).

[13] Y. Jun, M. Gavrilov, and J. Bechhoefer, "High-precision test of Landauer's principle in a feedback trap," Physical Review Letters **113**, 190601 (2014).

[14] L. Lamport, "Time, clocks, and the ordering of events in a distributed system," Communications of the ACM **21**, 558–565 (1978).

[15] R. D. Sorkin, "Causal sets: Discrete gravity," in *Lectures on Quantum Gravity*, edited by A. Gomberoff and D. Marolf, Springer (2005), arXiv:gr-qc/0309009.

[16] A. Einstein, "On the electrodynamics of moving bodies," *Annalen der Physik* **17**, 891–921 (1905).

[17] A. Vaswani et al., "Attention is all you need," Advances in Neural Information Processing Systems 30 (2017).

[18] T. B. Brown et al., "Language models are few-shot learners," Advances in Neural Information Processing Systems 33, 1877–1901 (2020).

[19] R. Omnès, "Consistent interpretations of quantum mechanics," *Reviews of Modern Physics* **64**, 339–382 (1992).

[20] N. F. Liu et al., "Lost in the middle: how language models use long contexts," *Transactions of the Association for Computational Linguistics* **12**, 157–173 (2024).

[21] A. Modarressi et al., "NoLiMa: long-context evaluation beyond literal matching," arXiv:2502.05167 (2025).

[22] P. Chattopadhyay et al., "Landauer principle and thermodynamics of computation," *Reports on Progress in Physics* **88**, 086001 (2025).