Formal Representation of Hierarchical Foveated Context Memory with Resonant Consensus

Alan J Garcia

December 13, 2025

Abstract

While significant resources are invested in expanding the context windows of Large Language Models (LLMs), the prevailing approach to long-term memory remains fundamentally unstructured. Standard methods typically rely on flat retrieval of raw tokens, often resulting in hallucinations or loss of coherence over extended timelines. This document formalizes a Recurrent Hierarchical Memory system that prioritizes structural integrity over simple storage capacity. We define a rigorous framework for base-layer recurrence, hierarchical compression, and Distributed Passive Retrieval (DPR). Instead of relying on a single retrieval pass, this architecture utilizes a **Resonant Consensus Protocol** to map conflicting historical perspectives into a **Semantic Quadrant Topology**. This yields verifiable, high-fidelity memory superpositions where consensus is mathematically derived. The model prioritizes scalability and provenance, accepting higher latency to ensure the retrieval of verified consensus rather than statistical approximation.

1 System Primitives and Definitions

The system is defined over a discrete, totally ordered time domain.

Definition 1 (Time Indexing). *Let $t \in \mathbb{Z}^+$ be the discrete time index. The direction of time is defined as **right-to-left**, such that:*

- $t = 1$*: The most recent state (Current).*
- $t > 1$*: States further in the past.*
- $t = n$*: The oldest accessible state.*

Definition 2 (Message Feed). *Let $\mathcal{M} = \{m_1, m_2, \ldots, m_n\}$ be the ordered sequence of atomic message units.*

- *Each m_t represents a discrete request/response pair.*
- *m_t is indivisible (atomic) with respect to partition boundaries.*
- *Let $\mathcal{H}(x)$ denote the Shannon Entropy of data packet x.*

2 Base Layer Dynamics (L_0)

The base layer, denoted as L_0, functions as a dense recurrent chain. It represents the ”high-resolution” immediate history, analogous to a raw data feed before abstraction.

Definition 3 (Base Node State). *Let $L_0^{(t)}$ be the node at layer 0 and time index t.*

The state transition is defined by a generation function g_0 and a compressor function Λ.

$$g_0 : \mathcal{S}_{comp} \times \mathcal{M} \to \mathcal{S}_0 \tag{1}$$

$$\Lambda : \mathcal{S}_0 \to \mathcal{S}_{comp} \tag{2}$$

The recurrence relation for the state at time t is:

$$L_0^{(t)} = \Lambda\left(L_0^{(t+1)}\right) \parallel m_t \tag{3}$$

Where:

- $\parallel$ denotes the string concatenation operator.
- $L_0^{(t+1)}$ is the immediate predecessor (the node to the right in the diagram).
- The most recent message m_t is appended to the compressed history of the older node.

3 Hierarchical Abstraction (L_k)

Higher layers L_k (where $k > 0$) provide summarized abstractions of the layer below (L_{k-1}). These layers group multiple lower-level nodes into discrete intervals.

Definition 4 (Hierarchical Interval)**.** *Let $\mathcal{I}_{k,j} = [t_{start}, t_{end}]$ be the j-th time interval represented at hierarchy level k.*

The node L_k covering this interval is constructed by aggregating the sequence of constituent nodes from layer $k-1$ and applying a summarization function g_k.

$$L_k^{[t_{start},t_{end}]} = g_k\left(\Big\Vert_{i=t_{start}}^{t_{end}} L_{k-1}^{(i)}\right) \tag{4}$$

Operational Semantics:

- The operator $\Big\Vert_{i=a}^{b}$ denotes the ordered concatenation:

$$L_{k-1}^{(a)} \parallel L_{k-1}^{(a+1)} \parallel \ldots \parallel L_{k-1}^{(b)}$$

- This strictly preserves temporal order (e.g., if $t = 4$ is ”Cats” and $t = 5$ is ”Pizza”, the aggregate retains ”Cats $\parallel$Pizza”).

4 Agent State Sequence ($\mathcal{A}$)

The system maintains a sequence of agent states, representing the context configurations at different points in history.

Definition 5 (Agent Sequence)**.** *Let $\mathcal{A} = \{A_1, A_2, \ldots, A_N\}$ be the sequence of agent states, where N is the total number of historical agents tracked.*

4.1 Active Agent (A^*)

The active agent, denoted A^*, corresponds to the current state of the system ($A^* \equiv A_1$). It represents the ”foveated” slice through the hierarchy available for immediate inference.

Let Π_1 be the set of selected nodes across all hierarchy levels that form the current context for A^*.

$$A^* = \left(\mathop{\Big\Vert}\limits_{i=1}^{t_{cut}} L_0^{(i)} \right) \parallel \left(\mathop{\Big\Vert}\limits_{j \in \text{Indices}(L_1)} L_1^{(j)} \right) \parallel \ldots \parallel \left(\mathop{\Big\Vert}\limits_{z \in \text{Indices}(L_K)} L_K^{(z)} \right) \tag{5}$$

4.2 Historical Agents ($A_k, k > 1$)

Each historical agent A_k represents a frozen snapshot of the context state from a previous time step. Just as A^* is a cut through the hierarchy at $t = 1$, A_k is a cut through the hierarchy relative to a historical reference point τ_k.

$$A_k = \mathop{\Big\Vert}\limits_{node \in \Pi_k} node \tag{6}$$

Where Π_k is the set of intervals defining the context window for the agent at history index k.

4.3 Causal Dependency Graph

To differentiate Episodic Recall from simple document retrieval, we define the Causal Dependency Graph $\mathcal{G}_\pi = (V, E)$.

- V: The set of all nodes across all layers $\bigcup_k L_k$.
- E: The set of directed edges representing generative dependency. An edge $(u, v) \in E$ exists if node v was used as input to the generation function g that produced u.

Retrieving a ”Memory” in this system is formally defined as retrieving a subgraph $S \subseteq \mathcal{G}_\pi$ that preserves the transitive closure of dependencies, thus ensuring provenance.

5 Distributed Passive Retrieval (DPR)

To augment the active context A^*, the system utilizes a set of passive historical agents to perform targeted, high-fidelity verification of information lost in compression.

Definition 6 (Passive Agent Subset). *Let $\mathcal{P} \subset \mathcal{A} \setminus \{A^*\}$ be a sparse, equidistributed subset of historical agents passively running on distributed hardware. All agents in $\mathcal{P}$ share a common message queue MQ.*

5.1 L1 Time-Sharded Routing Function (Ψ)

To optimize retrieval efficiency and scalability, the L1 Index is partitioned into M disjoint temporal shards $S_1, S_2, \ldots, S_M$, where each shard contains indices for agents from a specific time interval.

The Routing Function Ψ maps a query q to a specific subset of relevant historical agents by aggregating searches across relevant time shards.

$$\Psi(q) = \bigcup_{j \in \text{TargetShards}(q)} \text{ANN}(q, S_j) \to \mathcal{P}_{target} \tag{7}$$

Where ANN(q, S_j) denotes an Approximate Nearest Neighbor search within shard S_j. Let $\mathcal{P}_{target} = \Psi(q)$ be the specific subset of passive agents selected for verification, such that $|\mathcal{P}_{target}| \ll |\mathcal{P}|$.

Mechanism 1 (Targeted RFI Broadcast)**.** *When A^* requires higher fidelity information regarding a compressed node L_k, it broadcasts a Request for Information (RFI) targeting only the subset identified by the routing function:*

$$MQ \leftarrow \mathit{RFI}(q, \Psi(q)) \tag{8}$$

5.2 L2 Parallel Verification and Confidence

For every targeted agent $p \in \Psi(q)$, the agent scans its local context structure in parallel using a high-precision Small Language Model (SLM) verification mechanism. If a match is found within a node at level L_i, the agent generates a response r_p.

Definition 7 (Confidence Function)**.** *The confidence $\mathcal{C}$ of a response r_p is defined as the product of the semantic verification score $\mathcal{V}$ and the inverse depth fidelity.*

$$\mathcal{C}(r_p) = \mathcal{V}(q, \mathit{context}_p) \cdot \frac{1}{1+i} \tag{9}$$

Where $\mathcal{V} \in [0, 1]$ represents the verification probability returned by the SLM.

6 Resonant Consensus Protocol

To mitigate hallucination and resolve conflicting historical accounts, the system employs a voting mechanism among the targeted agents before final injection.

6.1 Voting Population

The voting population $\mathcal{V}_{vote}$ is strictly constrained to the agents identified by the L1 Router.

$$\mathcal{V}_{vote} = \Psi(q) \subset \mathcal{P} \tag{10}$$

This constraint ensures the consensus process remains foveated and does not degrade into a global broadcast.

6.2 Semantic Quadrant Topology

Let r_k be a candidate response generated by agent $k \in \mathcal{V}_{vote}$. Every other agent $j \in \mathcal{V}_{vote}$ evaluates r_k against its own local context, generating an alignment score $E(r_k, j) \in [-1, 1]$.

We map each response r_k to a topological coordinate $\langle v^+, v^- \rangle$ based on the net alignment of positive and negative clusters within the voting population.

- **Symmetric Resonance (Consensus):** $v^+ > 0 \wedge v^- > 0$. High-entropy bridge concepts agreed upon by diverse contexts.
- **Asymmetry (Perspective):** $v^+ > 0 \wedge v^- < 0$ (or vice versa). Represents a partial truth valid only from specific historical perspectives.

6.3 Superposition Injection

The active agent A^* does not simply select the maximum confidence response. Instead, it aggregates the responses into a structured superposition object M_{super}.

$$M_{super} = \{r_{consensus}\} \cup \{r_{persp\_1}, r_{persp\_2}, \dots\} \tag{11}$$

Where $r_{consensus}$ are responses mapping to the Symmetric Resonance quadrant, and r_{persp} are high-confidence responses from Asymmetric quadrants. This allows A^* to generate a nuanced reply acknowledging both the agreed facts and the conflicting perspectives.

7 Scalability and Complexity Analysis

This section analyzes the asymptotic complexity of the Foveated Context architecture. The primary objective is to demonstrate that the computational cost of the system decouples from the total history length T, proving the theoretical feasibility of infinite context retention, independent of wall-clock latency constraints.

7.1 Inference Complexity

Let T be the total length of the historical message sequence.

- **Standard Transformer:** The self-attention mechanism scales quadratically with the total sequence length.
$$\mathcal{O}_{std} = O(T^2) \tag{12}$$
- **Foveated Active Agent (A^*):** The active agent maintains a compressed context of size C, where $C \ll T$ is bounded by the constrained optimization process.
$$\mathcal{O}_{fov} = O(C^2) \tag{13}$$
While C may be large (e.g., 32k tokens), it remains constant relative to the growth of history T. Thus, inference complexity is $O(1)$ with respect to T.

7.2 Retrieval Complexity (Hybrid DPR)

Let $N = |\mathcal{P}|$ be the total number of passive agents (proportional to T), distributed across M time shards. Let $k = |\Psi(q)|$ be the constant number of agents targeted by the L1 Router.

- **L1 Routing:** With time-sharding, the routing complexity depends on the shard size N/M. Assuming parallel shard queries:
$$\mathcal{O}_{L1} = O\left(\log \frac{N}{M}\right) \tag{14}$$
- **L2 Verification:** The verification cost is proportional only to the number of targeted agents and their local context size L_{ctx}.
$$\mathcal{O}_{L2} = O(k \cdot L_{ctx}) \tag{15}$$

Total System Scaling: The combined cost of inference and retrieval scales as $O(C^2 + \log \frac{N}{M})$. This logarithmic scaling confirms that the architecture remains computationally feasible even as $N \to \infty$, enabling infinite retention for deep research tasks where asynchronous retrieval latency is acceptable.

8 Constrained Optimization (The Interval Problem)

The structure of the hierarchy (specifically the size of intervals $\mathcal{I}$) is determined by an optimization process. The goal is to maximize the temporal range (context length) subject to information density constraints.

Decision Variables: Let $x_{t,k} \in \{0,1\}$ be a binary variable indicating whether time step t is represented by a node at hierarchy level k.

Optimization Objective 1 (Maximize Time Range)**.**

$$\textit{Maximize } J = \sum_{t=1}^{\infty} \sum_{k=0}^{K} x_{t,k} \tag{16}$$

Subject to Constraints:

1. **Unique Representation:** Each active time step must be represented by exactly one node in the hierarchy (no overlapping duplicates).

$$\sum_{k=0}^{K} x_{t,k} \leq 1, \quad \forall t \tag{17}$$

2. **Contiguity:** The context history must be continuous from $t = 1$ (no gaps).

$$x_{t,\cdot} = 0 \implies x_{t+1,\cdot} = 0 \tag{18}$$

3. **Entropy Bound (Foveation Constraint):** For any generated node N formed by an interval of messages, the accumulated Shannon entropy must not exceed a threshold $\mathcal{H}_{max}$.

$$\forall \text{ active nodes } N : \quad \mathcal{H}(N) \leq \mathcal{H}_{max} \tag{19}$$

A Appendix: Definitions

A.1 Time Steps (t)

In this model, a ”Time Step” t is a discrete index corresponding to a single event or message interaction cycle. The indexing is **reverse chronological**.

A.2 Ordered Concatenation ($\bigg\|$)

The symbol $\bigg\|$ (Big Parallel) represents the generalization of the string concatenation operator. Unlike summation ($\sum$), concatenation is **non-commutative**.

A.3 Shannon Information Entropy ($\mathcal{H}$)

Shannon entropy is used here as a measure of the ”information density” or ”complexity” of a context node. The constraint $\mathcal{H}(N) \leq \mathcal{H}_{max}$ ensures no single summarized node exceeds the model’s effective capacity.

A.4 Context Fidelity (Φ)

Fidelity $\Phi(L_k)$ represents the ratio of recoverable information in a node at hierarchy level k relative to the raw input. It serves as the theoretical justification for the depth penalty in the confidence function.

$$\lim_{k \to \infty} \Phi(L_k) = 0$$

A.5 Semantic Verification Space ($\mathcal{V}$)

The verification function $\mathcal{V}$ maps a query and a context node to a scalar probability score derived from SLM reasoning.

$$\mathcal{V} : \mathcal{Q} \times \mathcal{C} \to [0, 1]$$

This defines the metric space for the L2 verification step, ensuring the confidence score $\mathcal{C}$ remains a valid measure between 0 and 1.