Wave-Based Semantic Memory with Resonance-Based Retrieval: A Phase-Aware Alternative to Vector Embedding Stores

Aleksandr Listopad
Independent Researcher
al@evacortex.ai

Abstract

Conventional vector-based memory systems rely on cosine or inner product similarity within real-valued embedding spaces. While computationally efficient, such approaches are inherently phase-insensitive and limited in their ability to represent contextual modulation, polarity, or structured semantic transformations.

We introduce a wave-based memory representation, in which embedding vectors are transformed into fixed-length complex-valued waveforms of the form $\psi(x) = A(x)\, e^{i\phi(x)}$. Here, x indexes vector dimensions, $A(x)$ encodes semantic amplitude, and $\phi(x)$ encodes contextual phase. This formulation supports a phase-aware similarity function — the *resonance score* — which reflects alignment in both amplitude and phase and intuitively quantifies constructive interference between semantic patterns.

Formally, the resonance score is computed as the squared magnitude of the elementwise complex sum of two patterns, aggregated across dimensions and normalized by their combined energy. A scale-alignment factor ensures boundedness in $[0, 1]$ and prevents dominance by high-energy patterns. This metric generalizes cosine similarity by incorporating phase coherence, enabling comparison between meaning-modulated representations that would otherwise appear similar in traditional spaces.

We implement this model in *ResonanceDB*, a source-available system that stores amplitude–phase patterns in memory-mapped binary segments and evaluates similarity using a deterministic comparison kernel. Compatibility with standard vector embeddings is preserved via a *sign–phase mapping* ($A(x) = |v(x)|$, $\phi(x) = 0$ if $v(x) \geq 0$, $\phi(x) = \pi$ otherwise), which maintains a valid polar form while retaining sign information. For vectors with non-negative entries, a simple zero-phase initialization ($\phi(x) = 0$) is also valid.

Empirical evaluation on synthetic and embedding-derived datasets shows that phase-enriched queries improve top-k retrieval in tasks involving negation, inversion, and contextual shift — distinctions often blurred under cosine-based retrieval. These results suggest that wave-based memory provides a cognitively inspired, phase-sensitive alternative to conventional vector stores, expanding the expressive capacity of semantic retrieval in reasoning-oriented applications.

***K*eywords** semantic memory · complex embeddings · resonance-based retrieval · embedding storage · cognitive memory · reasoning systems

1 Introduction

Vector embeddings are the dominant paradigm for encoding semantic information in modern AI systems. From retrieval-augmented generation (RAG) pipelines to knowledge graph embedding and neural search engines, high-dimensional vectors derived from pre-trained language models are widely used to approximate meaning through geometric proximity, typically relying on cosine distance or inner product [1, 2]. These representations are computationally efficient, differentiable, and compatible with approximate nearest neighbor (ANN) methods.

Despite their widespread adoption, vector-based embeddings exhibit fundamental limitations in tasks involving semantic polarity, contextual shifts, or compositional reasoning. Standard embedding spaces represent concepts as static points, but lack mechanisms to encode transformations such as negation, modality, or epistemic stance. For example, embeddings for "happy" and "not happy" are often closely aligned, despite having opposite meaning [3]. The geometry captures surface similarity, but fails to reflect operator-level modulation.

To address this, we introduce a wave-based memory representation in which semantic patterns are modeled as complex-valued waveforms of the form $\psi(x) = A(x)\, e^{i\phi(x)}$, where $A(x)$ encodes semantic amplitude and $\phi(x)$ encodes contextual phase. Here, x denotes the index of a vector dimension. This formulation draws on constructive interference, treating meaning not as a static vector but as a modulated wave pattern whose alignment with other patterns depends on both magnitude and phase. Operations such as negation, modality, or discourse shift correspond to phase transformations, while similarity is preserved through phase-sensitive interference energy.

We implement this model in a practical system, ResonanceDB, which stores and retrieves wave patterns as amplitude–phase pairs using memory-mapped binary segments. Similarity in ResonanceDB is evaluated via a deterministic comparison kernel that computes constructive interference energy between two complex patterns. This energy is maximal when patterns are in-phase and proportionally aligned, and minimal when anti-phased.

Importantly, ResonanceDB is compatible with standard vector embeddings: any real-valued vector can be mapped to a wave pattern via amplitude assignment and sign–phase initialization, enabling seamless integration into existing pipelines without retraining or re-indexing.

The goal of this work is not to replace vector embeddings, but to extend their expressiveness by introducing a complementary, phase-aware semantic substrate. Our contribution lies in formalizing a resonance-based similarity metric, implementing an efficient storage and retrieval system, and demonstrating empirically that this framework captures distinctions inaccessible to conventional vector methods.

2 WavePattern Architecture

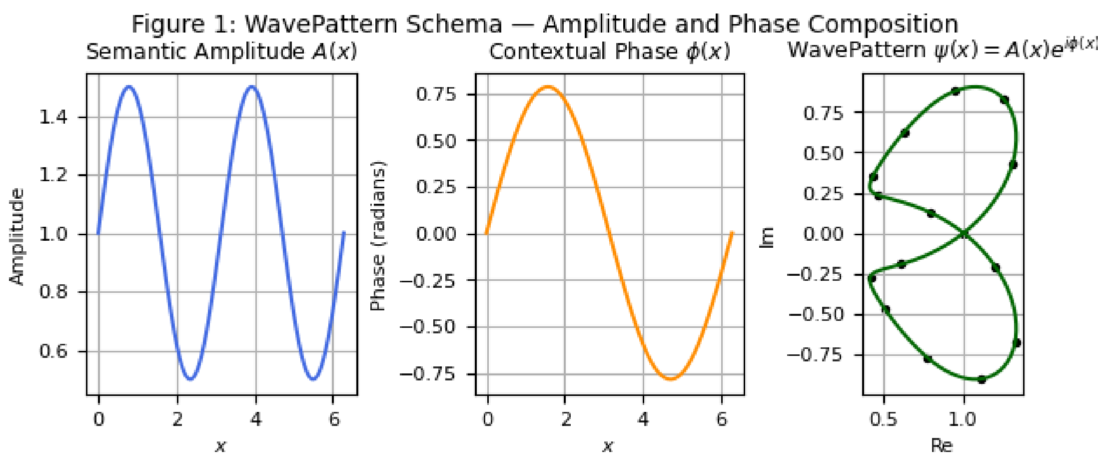

Figure 1: WavePattern schema. Top: amplitude $A(x)$. Middle: contextual phase $\phi(x)$. Bottom: complex trajectory $\psi(x) = A(x)\, e^{i\phi(x)}$ in the complex plane.

Each semantic pattern is modeled as a discrete complex-valued waveform

$$\psi(x) = A(x)\, e^{i\phi(x)}, \tag{1}$$

where x indexes vector dimensions, $A(x)$ is the real-valued amplitude encoding semantic intensity ($A(x) \geq 0$), and $\phi(x)$ is the phase component representing contextual modulation ($\phi(x) \in [-\pi, \pi)$). The resulting $\psi(x)$ is a pointwise complex representation of meaning, where amplitude governs salience and phase expresses structural or operator-level semantics.

Similarity between two waveforms $\psi_1(x)$ and $\psi_2(x)$ is computed using a resonance-based metric derived from constructive interference:

$$S(\psi_1, \psi_2) \;=\; \tfrac{1}{2} \cdot \frac{\sum_x |\psi_1(x) + \psi_2(x)|^2}{\sum_x \left(|\psi_1(x)|^2 + |\psi_2(x)|^2\right)} \cdot R, \tag{2}$$

where the scale-alignment factor R is

$$R = \frac{2\sqrt{E_1 E_2}}{E_1 + E_2}, \qquad E_1 = \sum_x |\psi_1(x)|^2, \;\; E_2 = \sum_x |\psi_2(x)|^2. \tag{3}$$

If $E_1 + E_2 = 0$, we define $S = 0$.

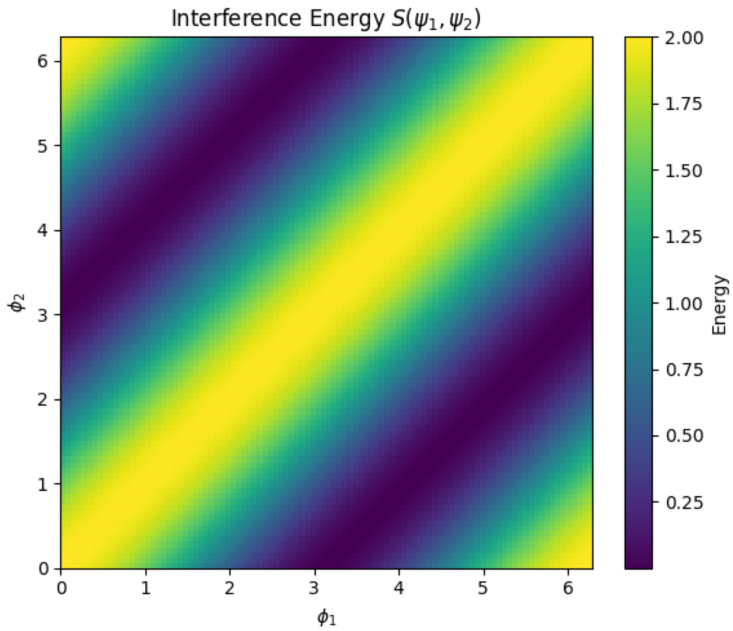

Figure 2: Resonance score $S(\psi_1, \psi_2)$ as a function of relative phase $\delta = \phi_2 - \phi_1$. Maximal resonance occurs at in-phase alignment ($\delta = 0$), minimal at anti-phase ($\delta = \pi$).

This scaling emphasizes interference between patterns of comparable energy and prevents dominance by high-intensity inputs. With purely real patterns ($\Im\psi = 0$) and equal norms ($E_1 = E_2$), Equation (2) reduces to $\frac{1+\cos\theta}{2}$, where θ is the angle between real vectors. For general vectors with arbitrary phases, S remains bounded in $[0, 1]$ and symmetric.

Patterns are stored as real-valued pairs (A, ϕ) and converted to complex form on demand. The system provides two backends: a scalar Java backend (`JavaKernel`, CPU baseline) and an *experimental* native SIMD backend (`NativeKernel`) that invokes native (C, AVX2) routines via the Panama Foreign Function & Memory API (Java 22).

3 Comparison with Vector Stores

Conventional vector similarity uses cosine similarity:

$$\text{cosine}(u, v) = \frac{u \cdot v}{\|u\| \, \|v\|}. \tag{4}$$

While efficient, cosine is phase-insensitive and may conflate contrasts (e.g., negation or epistemic shifts) that are structurally distinct.

Resonance operates on complex-valued waveforms and reflects both amplitude alignment and phase coherence. For ψ_1, ψ_2 derived from (A_1, ϕ_1) and (A_2, ϕ_2), $S(\psi_1, \psi_2)$ satisfies:

$$\begin{aligned}
&S \in [0, 1] && \text{(bounded)}\\
&S(\psi_1, \psi_2) = S(\psi_2, \psi_1) && \text{(symmetric)}\\
&S(\psi, \psi) = 1 && \text{(self-match)}\\
&S(\psi_1, \psi_2) < 1 \text{ if } \phi_1 \neq \phi_2 && \text{(phase-sensitive)}\\
&S(\psi_1, \psi_2) \to 0 \text{ when } \psi_2(x) = -\,\psi_1(x) && \text{(anti-phase)}\\
&S(e^{i\delta}\psi_1, e^{i\delta}\psi_2) = S(\psi_1, \psi_2) && \text{(global phase invariance).}
\end{aligned}$$

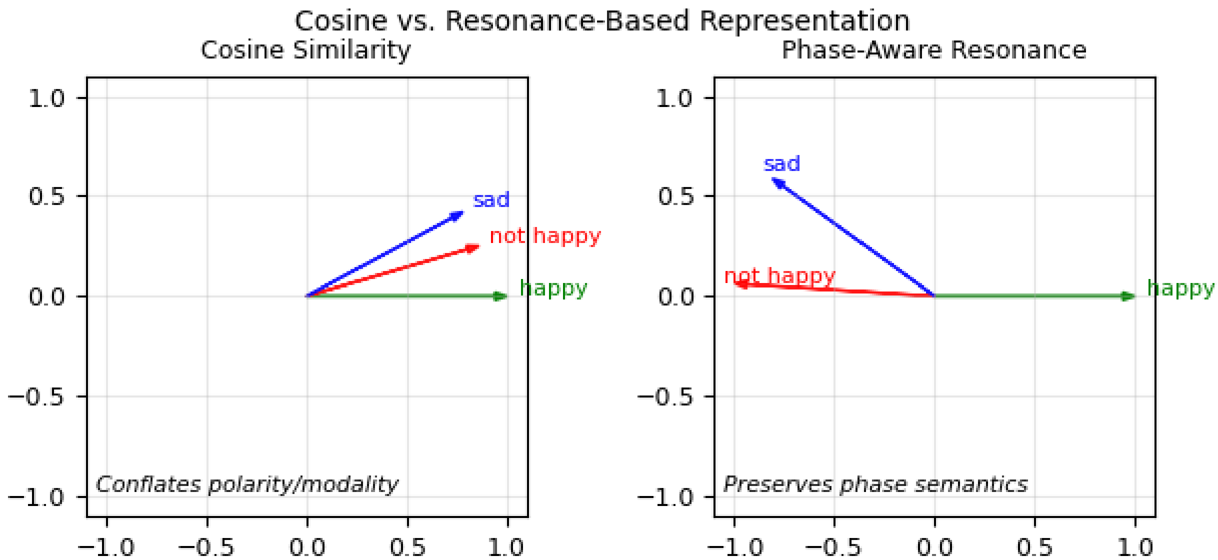

Figure 3: Illustration: Cosine similarity (left) can conflate polarity (e.g., "happy", "not happy", "sad" may appear close). Resonance-based space (right) distinguishes them via phase.

Mapping any real vector $v \in \mathbb{R}^L$ to $A(x) = |v(x)|,\ \phi(x) = 0$ if $v(x) \geq 0$ else π yields cosine-like behavior: for equal norms $S = \frac{1+\cos\theta}{2}$; in general, for fixed norms S is monotonic in $\cos\theta$ only in the purely real case.

4 Experimental Evaluation

We evaluate (i) latency and scalability, (ii) sensitivity to amplitude/phase variation, and (iii) retrieval precision under structured semantic perturbations.

4.1 Setup

Experiments ran on a single machine with **Windows 11 (10.0)**, **Java HotSpot 22.0.1** (64-bit), a **12-core Intel CPU** (3.60 GHz), and **32 GB RAM**, no GPU. Patterns had fixed length $L \in \{512, 1024\}$; dataset sizes ranged from 10^4 to 5×10^5. Embedding-derived inputs were mapped to amplitudes with sign–phase initialization; synthetic patterns varied phase to isolate phase effects. Top-k queries ($k \in \{1, 10, 100\}$) were executed under *cold-cache* conditions using a fixed **12-thread** Java executor with concurrent scans over memory-mapped segments.

All latency results below use the scalar backend `JavaKernel` (no SIMD). These figures represent a conservative baseline; the native AVX2 backend (`NativeKernel`) is experimental and excluded.

4.2 Latency and Scalability

On $L = 1024$ and 500K patterns (heap $\approx$ 512 MB), we observe interactive CPU-only latencies (cold cache, 12 threads) with `JavaKernel`:

- **top-1:** avg $\approx$ 135 ms, p95 $\approx$ 152 ms
- **top-10:** avg $\approx$ 138 ms, p95 $\approx$ 156 ms
- **top-100:** avg $\approx$ 145 ms, p95 $\approx$ 163 ms

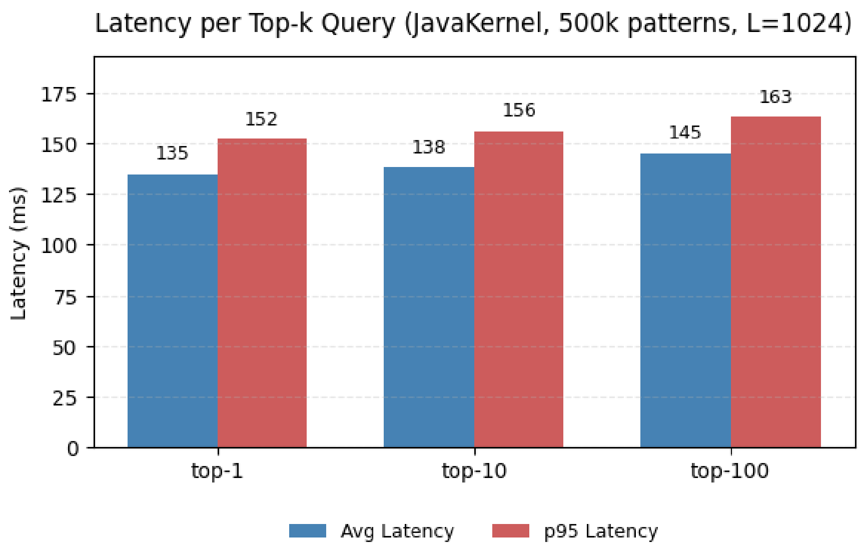

Figure 4: Latency per top-k query using JavaKernel ($L = 1024$, 500K patterns, 12 threads, cold cache). Average and 95th percentile (p95) latencies scale linearly with dataset size and grow modestly with k.

These measurements demonstrate that full waveform comparison — including per-element phase conversion and interference accumulation — is compatible with *interactive response times* on commodity CPUs, *without indexing or approximation*. Increased I/O throughput and SIMD acceleration are expected to reduce latency further.

5 Qualitative Analysis of Semantic Operators

While quantitative benchmarks establish improved retrieval metrics, a more intuitive picture emerges from visualizing distance distributions and interference patterns for structured semantic operators. We examined four representative transformations applied to the base concept *happy*: logical negation (NEG), controlled phase shift (SHIFT+), and intensity modulation (INT_UP, INT_DOWN).

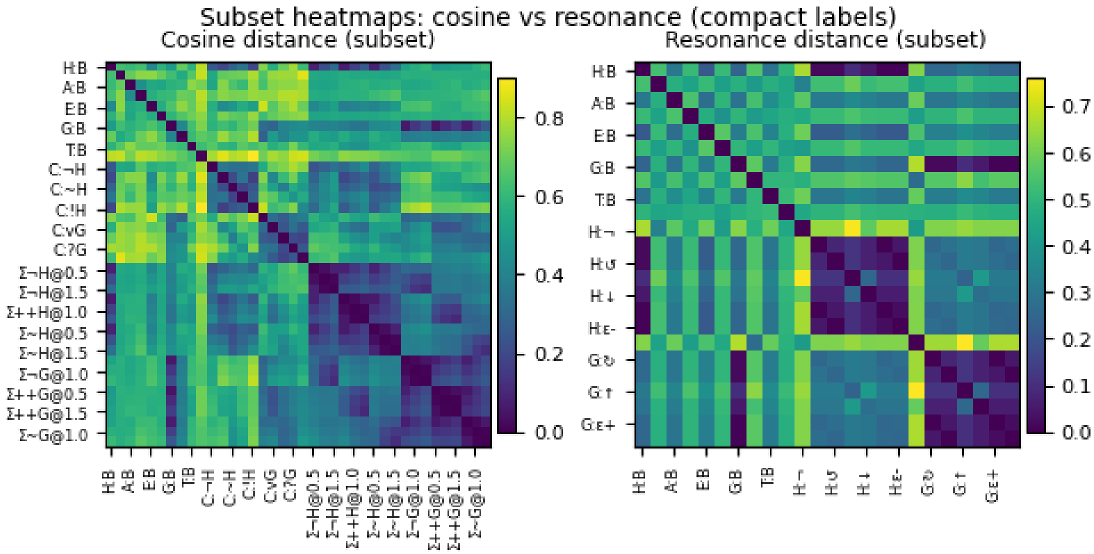

Figure 5: Operator Histograms. Cosine distributions collapse into broad, noisy clusters, while resonance produces compact and separable profiles that preserve operator semantics.

5.1 Cosine vs. Resonance Distributions

Figure 5 contrasts cosine and resonance distances for operator-enriched queries. Under cosine, distributions collapse into broad, noisy histograms centered near 0.6, with frequent spurious near-zero matches. Resonance, in contrast, yields compact, well-separated distributions:

- **NEG:** resonance pushes "not happy" away from "happy", while cosine conflates them.
- **SHIFT+:** resonance produces a narrow, stable band reflecting controlled phase drift; cosine reduces to random scatter.
- **INT_UP vs. INT_DOWN:** cosine fails to distinguish them, whereas resonance produces mirror-symmetric distributions, with INT_UP shifting outward from the base and INT_DOWN contracting inward.

5.2 Heatmap Structure

Heatmaps (Figure 6) reveal further differences. Cosine distance lacks structured block patterns: negated or intensified forms cluster indistinctly near their bases. Resonance heatmaps, however, exhibit clear block diagonals: operators form families with predictable interference profiles. NEG is anti-phased, SHIFT variants align in controlled offsets, and INT operators display amplitude-driven symmetry. These structured manifolds demonstrate that resonance retrieval respects operator semantics rather than treating them as noisy vector perturbations.

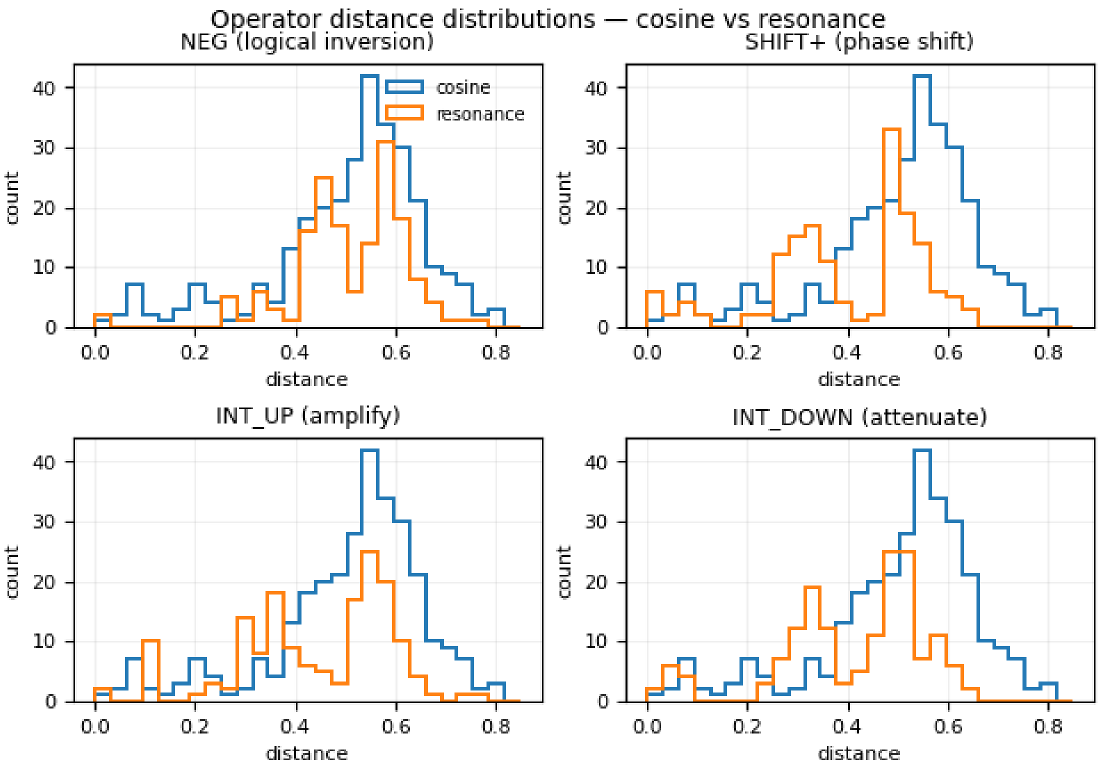

Figure 6: Side-by-Side Heatmaps. Cosine distances show noisy, unstructured clusters, whereas resonance produces clear block-diagonal manifolds aligned with operator families.

6 Related Work

6.1 Vector Similarity Search

The prevailing approach relies on vector embeddings and geometric similarity metrics. Libraries such as FAISS [1] implement efficient ANN search using quantization and indexing, typically grounded in cosine or inner product metrics [2]. While scalable, they treat embeddings as static points, lacking explicit operators for polarity, modality, or compositional recontextualization.

6.2 Holographic and Complex Embeddings

Holographic models [4] represent entities/relations via circular correlation inspired by associative memory. Complex-valued embeddings [5] introduce phase as a latent component; related phase-parameterized approaches include *ComplEx* [6] and *RotatE* [7], which model relations as rotations in the complex plane. Our approach elevates phase to a primary operator of semantic alignment, enabling interference-based retrieval sensitive to phase coherence.

6.3 Memory-Augmented Neural Architectures

Neural Turing Machines and Differentiable Neural Computers [8] add external, differentiable memory accessed by soft attention. ResonanceDB is non-trainable: fixed patterns, deterministic comparison, no runtime parameter updates — closer to content-addressable memory than to a learnable controller.

6.4 Cognitive and Neuromorphic Models of Meaning

Interference-based encodings have roots in cognitive and neuromorphic models. Frameworks such as Plate's HRR [9] and Kanerva's Hyperdimensional Computing [10] embed symbolic structure in high-dimensional vectors via algebraic or convolutional operations. Many remain theoretical or tied to specialized hardware; our system delivers a pragmatic implementation on conventional infrastructure.

7 Limitations

ResonanceDB deliberately forgoes training: phases are supplied by the encoder or specified procedurally, and there is no learnable phase modulator. This design improves interpretability but may limit adaptivity. Because similarity depends on phase coherence, the method can be sensitive to phase noise or high-frequency perturbations; smoothing or regularization may be required. The present retrieval is exact and scan-based, with complexity $O(NL)$ over N patterns of length L; at larger scales, memory-mapped I/O and per-query phase conversion can dominate latency. Although the scale-alignment factor R mitigates imbalance, amplitude calibration across heterogeneous corpora remains an engineering consideration. Approximate indexing is a natural direction for future work. The native SIMD backend is experimental; reported latencies use the scalar Java backend.

8 Discussion and Conclusion

We proposed a wave-based memory architecture with a resonance-based similarity that is sensitive to both magnitude and phase. Phase shifts naturally encode contrastive phenomena (negation, modality, role inversion) that are difficult to express in traditional vector geometries. Empirically, phase-enriched queries improve top-k retrieval on tasks where cosine fails to discriminate, while maintaining *interactive response times* on CPUs.

From an engineering perspective, ResonanceDB attains practical latency through fixed-length patterns, memory-mapped segments, and efficient accumulation; existing embeddings are compatible via sign–phase mapping, enabling plug-in use within RAG and search pipelines. All reported measurements are CPU-only with the scalar backend; SIMD acceleration and improved I/O throughput are expected to further reduce latency. Thus, wave-based memory is promising not only conceptually but also as an engineering substrate for reasoning systems, where phase can encode roles, hypotheses, or epistemic status.

A source-available prototype of ResonanceDB is available for academic use under the **Prosperity Public License 3.0**.[1] This license permits free use until revenue is generated, after which a commercial license is required.

Acknowledgments

References

[1] Jeff Johnson, Matthijs Douze, and Hervé Jégou. Billion-scale similarity search with GPUs. *IEEE Transactions on Big Data*, 7(3):535–547, 2021.

[2] Hervé Jégou, Matthijs Douze, and Cordelia Schmid. Product quantization for nearest neighbor search. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 33(1):117–128, 2011.

[3] Tomas Mikolov, Wen-tau Yih, and Geoffrey Zweig. Linguistic regularities in continuous space word representations. *NAACL-HLT*, pages 746–751, 2013.

[4] Maximilian Nickel, Lorenzo Rosasco, and Tomaso Poggio. Holographic embeddings of knowledge graphs. In *Proceedings of the 30th AAAI Conference on Artificial Intelligence*, pages 1955–1961, 2016.

[5] Chiheb Trabelsi, Olexa Bilaniuk, Ying Zhang, Dmitriy Serdyuk, Sandeep Subramanian, João Felipe Santos, Soroush Mehri, Negar Rostamzadeh, Yoshua Bengio, and Chris J. Pal. Deep complex networks. In *International Conference on Learning Representations (ICLR)*, 2018.

[6] Théo Trouillon, Johannes Welbl, Sebastian Riedel, Éric Gaussier, and Guillaume Bouchard. Complex embeddings for simple link prediction. In *Proceedings of the 33rd International Conference on Machine Learning (ICML)*, pages 2071–2080, 2016.

[7] Zhiqing Sun, Zhi-Hong Deng, Jian-Yun Nie, and Jian Tang. Rotate: Knowledge graph embedding by relational rotation in complex space. In *International Conference on Learning Representations (ICLR)*, 2019.

[8] Alex Graves, Greg Wayne, Ivo Danihelka, et al. Hybrid computing using a neural network with dynamic external memory. *Nature*, 538(7626):471–476, 2016.

[1] `https://prosperitylicense.com/`

[9] Tony A. Plate. *Holographic Reduced Representation: Distributed Representation for Cognitive Structures*. CSLI Publications, 2003.

[10] Pentti Kanerva. Hyperdimensional computing: An introduction to computing in distributed representation with high-dimensional random vectors. *Cognitive Computation*, 1(2):139–159, 2009.