

Scalable Symbolic Reasoning with Matrix-Based Brain-Inspired Representations and Vector-Space Acceleration

William Youngwoo Chung*, Hyunwoo Oh*, Hamza Errahmouni Barkam*, Calvin Yeung*, Mohsen Imani*

*Department of Computer Science, University of California, Irvine, USA

{chungwy1, hyunwoo, herrahmo, chyeung2, m.imani}@uci.edu

Abstract—Hyperdimensional Computing (HDC) enables robust, hardware-friendly symbolic computation, but mainstream complex-valued HDC uses commutative binding and relies on costly permutations to encode order and directionality. Generalized Holographic Reduced Representations (GHRR) replace commutative binding with non-commutative matrix multiplication, enabling native encoding of sequences, directed graphs, and hierarchies without permutation logic. However, naive GHRR incurs prohibitive matrix compute/storage overhead. We present a vector-space flattening method that preserves GHRR semantics while executing similarity, training updates, and inference directly using standard high-throughput dot-product engines. Additionally, we design a custom ASIC accelerator that fuses binding and similarity operations into a unified complex-valued data path, which supports high-throughput streaming via dual DMA engines and performs runtime normalization for accurate inference. Compared to a PyTorch baseline on an NVIDIA RTX 4090 GPU, the prototype delivers $1.36\times$ – $1.56\times$ higher throughput and achieves $16.2\times$ – $18.6\times$ better energy efficiency. These results demonstrate a scalable pathway to embedding brain-inspired symbolic reasoning in future AI accelerators.

I. INTRODUCTION

Symbolic reasoning at scale demands representations that are both (i) expressive enough to encode structure (order, directionality, relations) and (ii) efficient enough to satisfy the power and latency budgets of modern accelerators. Hyperdimensional Computing (HDC) represents entities as high-dimensional vectors [1] and composes them using bundling (superposition) and binding (association), making it naturally amenable to parallel hardware. However, the most widely used complex-valued HDC implementation, Fourier Holographic Reduced Representations (FHRR) [2], uses *commutative* binding and therefore cannot natively represent order or direction. As a result, sequences and graphs typically require additional permutation logic, which induces irregular data movement and can undermine the efficiency of prior FPGA and ASIC accelerators [3], [4].

Generalized Holographic Reduced Representations (GHRR) addresses this limitation by redefining binding as *non-commutative* matrix multiplication, enabling order-sensitive composition without explicit permutations [5]. The key barrier is practicality: a naive implementation operates in matrix space (e.g., $\mathbb{C}^{m\times m}$ per hypervector element), making inference and training substantially more expensive than standard element-wise binding.

This work makes GHRR scalable through (1) a *vector-space flattening* scheme that preserves GHRR similarity while reducing inference to standard dot products, and (2) a permutation-free ASIC accelerator that fuses binding and similarity into a unified complex-valued streaming datapath. Our 28 nm prototype integrates runtime normalization, interleaved real/imag buffers, and dual-DMA streaming, achieving 1.54 TFLOPS peak throughput in 4.77 mm^2 at 4.22 W. Against an optimized PyTorch baseline on an NVIDIA RTX 4090, it delivers 1.36 – $1.56\times$ higher throughput and 16.2 – $18.6\times$ better energy efficiency.

II. BACKGROUND AND RELATED WORK

HDC/VSA represents symbols as high-dimensional vectors and composes them via bundling and binding [1], [6], [7], [8], [9], [10]. Prior HDC graph and relational methods build structured representations but still depend on commutative binding and/or permutations for structural encoding [11], [12], [13]. Hardware acceleration has been explored across FPGA/ASIC/PIM platforms, often focusing on bundling/similarity while leaving permutation-heavy order encoding as a memory bottleneck [3], [14], [15], [4], [16]. Since permutations are bandwidth-bound and stress the memory wall [17], we instead leverage GHRR’s non-commutative binding to eliminate permutation logic and make it practical via vector-space flattening [5]. Permutations are bandwidth-bound data movement that disrupts GPU tensor-core utilization and requires shuffle/crossbar logic on ASIC/FPGA designs [4], [17], motivating permutation-free order-sensitive binding.

III. GHRR AND VECTOR-SPACE FLATTENING

A. GHRR Encoding

A GHRR hypervector is

$$H = [A^{(1)}, \dots, A^{(D)}]^\top \in \mathbb{C}^{D\times m\times m}, \quad A^{(j)} \in U(m), \quad (1)$$

with non-commutative binding $H_1 * H_2 = [A^{(j)}B^{(j)}]_{j=1}^D$ and similarity

$$\delta(H_1, H_2) = \frac{1}{mD} \Re \left[\sum_{j=1}^D \text{tr} \left(A^{(j)} (B^{(j)})^\dagger \right) \right], \quad (2)$$

which reduces to FHRR when $m = 1$ [5].

B. Flattening Equivalence

Using $\text{tr}(X^\dagger Y) = \text{vec}(X)^\dagger \text{vec}(Y)$, we define $\text{flat}(H) \in \mathbb{C}^{Dm^2}$ by concatenating $\text{vec}(A^{(j)})$. Then

$$\delta(H_1, H_2) = \frac{1}{mD} \Re[\text{flat}(H_1)^\dagger \text{flat}(H_2)], \quad (3)$$

so similarity search and class-memory dot products execute as standard dot products in vector space, avoiding matrix-form inference cost while preserving GHRR semantics.

IV. HARDWARE ACCELERATION

We co-design a permutation-free accelerator for GHRR that targets its two core operations: binding-based encoding and similarity-based inference. The system comprises an *Encoder* that produces GHRR hypervectors, an *Inference* block that compares encodings against a codebook using normalized similarity, and a lightweight *Host Interface* with a dual-channel *DMA engine* for streaming inputs and codebook vectors between off-chip memory and on-chip buffers. Fig. 1 visualizes the top-level architecture of the custom GHRR accelerator.

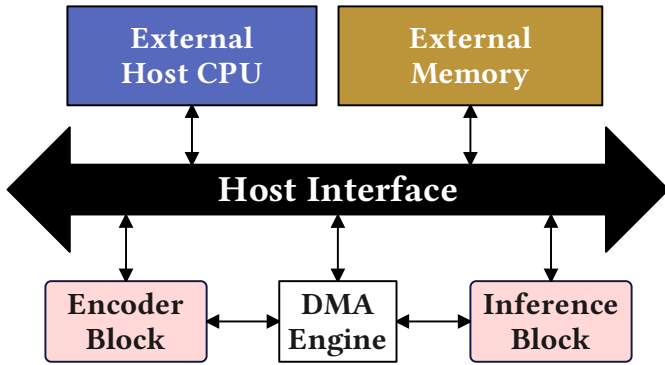


Fig. 1: Top-level architecture of the GHRR accelerator. Host interface coordinates encoder and inference blocks via DMA streaming from external memory.

Encoder. The encoder implements complex-valued binding with a pipelined multiply-accumulate datapath operating on interleaved real/imaginary buffers (shape $(D, 2)$). Inputs are mapped through per-dimension unitary transforms and a complex exponential stage, then streamed to downstream inference buffers.

Inference. The inference block evaluates similarity via normalized dot products across P candidates in parallel using an interleaved real-valued layout of the complex hypervectors. Vector norms are accumulated at runtime and used to scale dot products through reciprocal factors, ensuring magnitude-invariant similarity. A lightweight top-1 selector returns the best-matching codebook entry. The pipeline overlaps data movement, dot-product accumulation, normalization, and selection to sustain high utilization as D and batch size grow.

Eliminating Permutation. By eliminating positional permutations, the design avoids irregular memory access and shuffle/crossbar overhead. Binding and comparison execute as regular, streaming vector operations, improving hardware efficiency for order-sensitive symbolic reasoning.

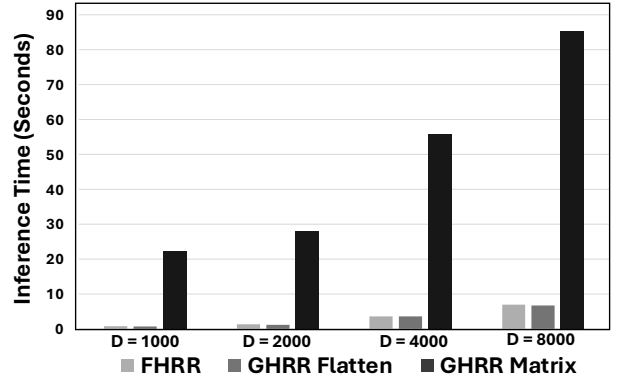


Fig. 2: Inference cost of FHRH, GHRR Flatten, and GHRR Matrix on a synthetic sequence based dataset.

V. RESULTS

Flattening validation. We benchmark FHRH, GHRR (Flatten), and GHRR (Matrix) on a synthetic sequence classification task (10 classes, sequence length 5) on an RTX 3060Ti over 50 trials. Fig. 2 shows that GHRR (Flatten) matches FHRH inference time within $0.95\times-1.05\times$ across $D \in \{1000, 2000, 4000, 8000\}$, while matrix-form GHRR is substantially slower ($38.2\times$, $16.8\times$, $15.5\times$, and $12.3\times$ at the same D), confirming that vector-space flattening is essential for scalable GHRR inference.

Hardware evaluation. For a representative configuration ($D=8000$, $m=2$, $P=8$), the prototype occupies 4.77 mm^2 , consumes 4.22 W , and achieves 1.54 TFLOPS peak throughput. In Figure 3, we compare our accelerator to an optimized PyTorch baseline on an NVIDIA RTX 4090. Our accelerator provides $1.36-1.56\times$ higher throughput and $16.2-18.6\times$ better energy efficiency, verifying the benefit of our custom GHRR accelerator for scaling symbolic reasoning architectures.

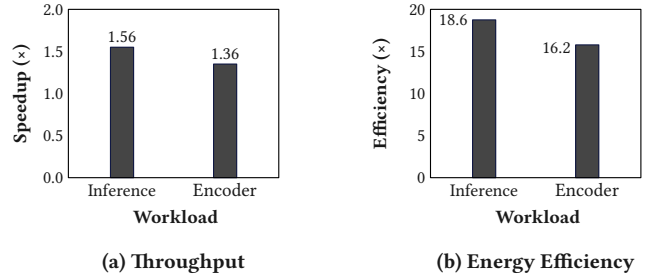


Fig. 3: Comparison of throughput and energy efficiency against an NVIDIA RTX 4090 baseline.

VI. CONCLUSION

We demonstrate a practical pathway for permutation-free, order-sensitive HDC by combining GHRR with vector-space flattening and a hardware design that unifies binding and normalized similarity. Flattening reduces GHRR inference to standard dot products while preserving similarity semantics, and the prototype delivers $1.36-1.56\times$ higher throughput and $16.2-18.6\times$ better energy efficiency than an RTX 4090 baseline.

REFERENCES

- [1] P. Kanerva, "Hyperdimensional computing: An introduction to computing in distributed representation with high-dimensional random vectors," *Cognitive Computation*, vol. 1, no. 2, pp. 139–159, 2009.
- [2] T. A. Plate, "Holographic reduced representations," *IEEE Transactions on Neural Networks*, vol. 6, no. 3, pp. 623–641, 1995.
- [3] M. Imani, S. Salamat, S. Gupta, J. Huang, and T. Rosing, "Fach: Fpga-based acceleration of hyperdimensional computing by reducing computational complexity," in *Proceedings of the 24th Asia and South Pacific Design Automation Conference*, pp. 493–498, 2019.
- [4] H. Chen, A. Zakeri, F. Wen, H. E. Barkam, and M. Imani, "Hypergraf: Hyperdimensional graph-based reasoning acceleration on fpga," in *2023 33rd International Conference on Field-Programmable Logic and Applications (FPL)*, pp. 34–41, IEEE, 2023.
- [5] C. Yeung, Z. Zou, and M. Imani, "Generalized holographic reduced representations," *arXiv preprint arXiv:2405.09689*, 2024.
- [6] T. Plate, *Holographic Reduced Representation: Distributed Representation for Cognitive Structures*. Stanford, CA: CSLI Publications, 2003.
- [7] M. Bennett, J. Clark, and M. Imani, "Hyperdimensional computing: A fast, robust and interpretable paradigm for artificial intelligence," *PLoS Computational Biology*, vol. 20, no. 1, p. e1012426, 2024.
- [8] P. Neubert, S. Schubert, and P. Protzel, "An introduction to hyperdimensional computing for robotics," *KI-Künstliche Intelligenz*, vol. 33, no. 4, pp. 319–330, 2019.
- [9] M. Issa, S. Shahhosseini, Y. Ni, T. Hu, D. Abraham, A. M. Rahmani, N. Dutt, and M. Imani, "Hyperdimensional hybrid learning on end-edge-cloud networks," in *2022 IEEE 40th International Conference on Computer Design (ICCD)*, pp. 652–655, IEEE, 2022.
- [10] Y. Ni, W. Y. Chung, S. Cho, Z. Zou, and M. Imani, "Efficient exploration in edge-friendly hyperdimensional reinforcement learning," in *Proceedings of the Great Lakes Symposium on VLSI 2024*, pp. 111–118, 2024.
- [11] J. Kang, M. Zhou, A. Bhansali, W. Xu, A. Thomas, and T. Rosing, "Relhd: A graph-based learning on fefet with hyperdimensional computing," in *2022 IEEE 40th International Conference on Computer Design (ICCD)*, pp. 553–560, IEEE, 2022.
- [12] H. Chen, Y. Ni, A. Zakeri, Z. Zou, S. Yun, F. Wen, B. Khaleghi, N. Srinivasa, H. Latapie, and M. Imani, "Hdreason: Algorithm-hardware codesign for hyperdimensional knowledge graph reasoning," *arXiv preprint arXiv:2403.05763*, 2024.
- [13] P. Poduval, H. Alimohamadi, A. Zakeri, F. Imani, M. H. Najafi, T. Givargis, and M. Imani, "Graphd: Graph-based hyperdimensional memorization for brain-like cognitive learning," *Frontiers in Neuroscience*, vol. 16, 2022.
- [14] M. Imani, S. Patil, and T. S. Rosing, "Masc: Ultra-low energy multiple-access single-charge tcam for approximate computing," in *2016 Design, Automation & Test in Europe Conference & Exhibition (DATE)*, pp. 373–378, IEEE, 2016.
- [15] Y. Ni, Y. Kim, T. Rosing, and M. Imani, "Algorithm-hardware co-design for efficient brain-inspired hyperdimensional learning on edge," in *2022 Design, Automation & Test in Europe Conference & Exhibition (DATE)*, pp. 292–297, IEEE, 2022.
- [16] H. E. Barkam, S. E. Jeon, S. Yun, C. Yeung, Z. Zou, X. Jiao, N. Srinivasa, and M. Imani, "Hyperdimensional computing for resilient edge learning," in *2023 IEEE/ACM International Conference on Computer Aided Design (ICCAD)*, pp. 1–8, IEEE, 2023.
- [17] W. Y. Chung, H. Errahmouni Barkam, T. Das, and M. Imani, "Robust reasoning and learning with brain-inspired representations under hardware-induced nonlinearities," in *Proceedings of the Great Lakes Symposium on VLSI 2025*, pp. 968–975, 2025.