THE CIRCLE WAS HIDING A HELIX: HELICAL REDUCED REPRESENTATIONS FOR EXACT VECTOR SYMBOLIC BINDING

Anonymous authors
Paper under double-blind review

ABSTRACT

Vector Symbolic Architectures (VSAs) store symbols as vectors and combine them by binding and bundling. A bundle of K bound pairs in D dimensions holds only about $D/\ln N$ pairs, where N is the codebook size. The cause of this limit lies in the binding operation, not in the vectors.

Binding is multiplication in a group G, and bundling is addition in the group algebra $\mathbb{C}[G]$. If G is compact, every representation is unitary. Every bound term then has magnitude 1, superposed terms are separable only by statistics, and the capacity bound follows. Bundling cannot escape to the Lie algebra of G either, because distributivity forces any such operation to be a representation.

Helical Reduced Representations (HeRR) bind on the non-compact group $\mathbb{Z}$. Their vectors are Laurent polynomials. Binding is polynomial multiplication, bundling is polynomial addition, and nesting depth is polynomial degree. Circular convolution is this ring modulo $t^D - 1$, so existing VSAs are quotients of HeRR. Storing a HeRR in fixed space is a choice of characters of $\mathbb{Z}$, and three independent capacity budgets follow: dimension, number of evaluation points, and numerical precision. Non-unitary characters, absent from compact groups, supply the third. Two exact decoders recover a bundle without observing crosstalk, so the codebook need not be random.

At 128 complex numbers of storage, HeRR retrieves 96 role–filler pairs at accuracy 1.00, against 0.13 for FHRR at equal storage. With a correlated codebook, FHRR reaches 0.00 at 8 pairs while HeRR remains exact.

The binding capacity limit is therefore a property of compact binding groups, not of vector representations.

1 INTRODUCTION

A Vector Symbolic Architecture (VSA) represents symbols as high-dimensional vectors [1; 2; 3]. A VSA has two operations. Binding combines two vectors into one vector. The result is not similar to either input. Bundling adds several vectors into one vector. The result is similar to every input. With these two operations, a VSA can store a data structure in one vector of fixed size.

VSAs have a capacity limit. Let a bundle contain K bound pairs. Let each vector have D elements. Let the codebook contain N symbols. Retrieval of one pair fails when K is larger than about $D/\ln N$ [1; 4]. The other $K - 1$ pairs cause crosstalk. Crosstalk is the noise that the other pairs add to the retrieved vector.

Prior work reduces the effect of crosstalk. Sparse block codes [5; 6], residue number systems [7], and shift-equivariant codes [8] add structure to the vectors. Resonator networks [9; 10] and linear-code factorizers [11] improve the decoder. These methods change the constant in the capacity limit. They do not remove the limit.

We take a different route. A phasor vector element is a complex number of modulus 1. In IEEE double precision it occupies 128 bits and carries one phase. The magnitude is fixed at 1 and carries no information. We ask what the unused bits can hold, and answer the question with representation theory. Section 2.4 proves that the magnitude must be constant whenever binding is multiplication

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

in a compact group, because a compact group has only unitary representations. The statement is an implication, not a survey. Table 1 lists the architectures whose binding satisfies its hypothesis, and Section 2.3 treats two architectures that do not.

The paper has five results.

1. A VSA whose binding is a group operation is a module over the group algebra $\mathbb{C}[G]$. Table 1 gives G and its representation for the standard architectures (Section 2).
2. If G is compact, then every representation is unitary and stores phase only, and the capacity bound follows (Section 2.4). Bundling cannot be moved to the Lie algebra of G (Section 2.5).
3. Helical Reduced Representations (HeRR), a VSA on the non-compact group $\mathbb{Z}$ (Section 3). Its vectors are Laurent polynomials. Existing VSAs are quotients of it. It has three independent capacity budgets (Section 3.3).
4. Two exact decoders (Section 3.4). Because the decoders are exact, the codebook can be non-random.
5. Experiments that measure the three budgets, exact retrieval, correlated codebooks, and nesting depth (Section 4).

The paper is self-contained. Section 2.1 gives all definitions from algebra. Section 3.4 gives the decoding algorithms in full.

2 Vector Symbolic Architectures Are Group Algebras

2.1 Definitions

Definition 2.1 (Group)**.** *A group G is a set with an operation $(a, b) \mapsto ab$. The operation is associative. There is an identity element e with $ea = ae = a$. Every element a has an inverse a^{-1} with $aa^{-1} = e$. The group is abelian if $ab = ba$ for all a, b.*

Definition 2.2 (Group algebra)**.** *The group algebra $\mathbb{C}[G]$ is the set of finite formal sums $\sum_g c_g\, g$, with $c_g \in \mathbb{C}$ and $g \in G$. Addition adds the coefficients of equal group elements. Multiplication is $(\sum_g a_g g)(\sum_h b_h h) = \sum_{g,h} a_g b_h\, (gh)$. The group algebra is a ring.*

Definition 2.3 (Representation)**.** *A representation of G on a vector space V is a map ρ from G to invertible linear maps on V with $\rho(ab) = \rho(a)\rho(b)$. A representation extends to $\mathbb{C}[G]$ by linearity. A character is a one-dimensional representation $\chi : G \to \mathbb{C} \setminus \{0\}$. A representation is unitary if every $\rho(g)$ satisfies $\rho(g)^* \rho(g) = I$. A character is unitary if $|\chi(g)| = 1$ for all g.*

Definition 2.4 (Compact group)**.** *A topological group is compact if it is compact as a topological space. Every finite group is compact. The circle group $\mathrm{U}(1) = \{z \in \mathbb{C} : |z| = 1\}$ is compact. The integers $\mathbb{Z}$ and the real numbers $\mathbb{R}$ are not compact.*

Fact 2.5 (Weyl's unitarian trick)**.** *Every finite-dimensional representation of a compact group is equivalent to a unitary representation [12; 13]. For finite groups, this is Maschke's theorem [14].*

Fact 2.6 (Characters of $\mathbb{Z}$)**.** *Every character of $\mathbb{Z}$ has the form $n \mapsto z^n$ for one non-zero complex number z. The character is unitary if and only if $|z| = 1$.*

Definition 2.7 (Phasor)**.** *A phasor is a complex number of modulus 1, $e^{i\theta}$. A phasor vector is a vector in $\mathbb{C}^D$ whose elements are all phasors.*

Definition 2.8 (VSA operations)**.** *Let $a, b \in \mathbb{C}^D$. Elementwise multiplication is $(a \odot b)_j = a_j b_j$. Circular convolution is $(a \circledast b)_k = \sum_j a_j b_{(k-j) \bmod D}$. The discrete Fourier transform (DFT) F satisfies $F(a \circledast b) = F(a) \odot F(b)$.*

2.2 Binding as a group operation

Let G be a group and ρ a representation of G on V. A VSA over (G, ρ) does the following.

- Each atomic symbol is one element $a \in G$. Its vector is $\rho(a)v_0$ for one fixed vector v_0.
- Bind: $\mathrm{bind}(a, b) = ab$, the group product.
- Bundle: $\mathrm{bundle}(a_1, \ldots, a_K) = \sum_k a_k$, the sum in $\mathbb{C}[G]$.
- Unbind: $\mathrm{unbind}(s, a) = a^{-1}s$.

Table 1 gives each published VSA in this form.

VSA	Group G	Representation	Compact?
Binary Spatter Codes (XOR) [2]	$\mathbb{Z}_2^D$	sign	finite
MAP (bipolar product) [3]	$\{\pm 1\}^D$	sign	finite
HRR (circular convolution) [1]	$\mathbb{Z}_D$	regular	finite
FHRR (phasor product) [1]	$\mathrm{U}(1)^D$	identity	compact
Fractional power encoding [1; 15]	one-parameter subgroup of $\mathrm{U}(1)^D$	identity	compact
MBAT (orthogonal matrices) [16; 18]	$\mathrm{O}(D)$	defining	compact
GHRR (unitary matrices) [19]	$\mathrm{U}(m)^k$	defining	compact
Residue HDC [7]	$\mathbb{Z}_{p_1} \times \cdots \times \mathbb{Z}_{p_k}$	characters	finite
HeRR (ours)	$\mathbb{Z}$	any character	**non-compact**

Table 1: Binding operations as group multiplication. Every group listed above HeRR is compact, so each satisfies the hypothesis of Proposition 2.9. Two architectures that do not are treated in Section 2.3.

The identification is exact. Circular convolution is multiplication in the group algebra $\mathbb{C}[\mathbb{Z}_D]$ written in the standard basis. The DFT is the change of basis that splits $\mathbb{C}[\mathbb{Z}_D]$ into its D characters. This is why HRR becomes FHRR under the DFT. Bundling is addition in $\mathbb{C}[G]$ in every row of Table 1.

2.3 Architectures outside the compact case

Two architectures fall outside the hypothesis of Proposition 2.9. Matrix Binding of Additive Terms [16] originally bound with random matrices, which gives the group $\mathrm{GL}(D, \mathbb{R})$. Hadamard-derived Linear Binding [17] binds by elementwise multiplication and unbinds by elementwise division on real vectors, which gives the group $(\mathbb{R}^*)^D$. Both groups are non-compact.

Neither architecture stores information in the magnitude. Both treat non-unit magnitude as a defect. The matrix method was later restricted to orthogonal matrices, because random matrices are numerically unstable [18]. The elementwise method constrains its initial distribution so that no element is near zero, because division by a small element is unstable [17]. The non-compact direction has therefore been reached before and discarded. We use it.

2.4 A compact group stores phase only

Proposition 2.9. *Let G be compact. Let ρ be a finite-dimensional representation of G. Then ρ is equivalent to a unitary representation. For every $g \in G$, $\rho(g)$ has operator norm 1. For every product $g_1 g_2 \cdots g_n$ of atoms, $\rho(g_1 g_2 \cdots g_n)$ has operator norm 1.*

Proof. The first statement is Fact 2.5. A unitary map has operator norm 1. A product of unitary maps is unitary. □

Proposition 2.9 has a direct consequence for capacity. Let $s = \sum_k \rho(r_k f_k)$ be a bundle of K bound pairs. Unbind with role r_q. The result is $\rho(f_q) + \sum_{k \neq q} \rho(r_q^{-1} r_k f_k)$. Every term has norm 1. The algebra does not distinguish the target from the other $K-1$ terms. The decoder must distinguish them by statistics. Random codebooks make the other terms incoherent. Their sum then has norm about $\sqrt{K-1}$. Retrieval succeeds while the target norm 1 exceeds the noise by a margin that depends on N. This gives the known bound $K \approx D/(c \ln N)$ [4].

2.5 The Lie algebra cannot bundle

Fractional power encoding computes with the exponents of phasors [1; 15]. The exponents form the Lie algebra $\mathfrak{g}$ of $G = \mathrm{U}(1)^D$. In $\mathfrak{g}$, binding is addition. It is natural to ask whether bundling can also be defined in $\mathfrak{g}$. The answer is no.

Proposition 2.10. *Let binding in $\mathfrak{g}$ be addition: $\mathrm{bind}(X, Y) = X + Y$. Let bundling have the form $B(X_1, \ldots, X_K) = \sum_k \psi(X_k)$ for one continuous map ψ. Suppose bundling distributes over*

binding:

$$\mathrm{bind}\big(A, B(X_1, \ldots, X_K)\big) = B(X_1 + A, \ldots, X_K + A) \quad \textit{for all } A. \tag{1}$$

Then $\psi(X) = \exp(\lambda \cdot X)$ for one linear map λ. Thus ψ is a group representation, and B is bundling in $\mathbb{C}[G]$.

Proof. Set $K = 1$ in (1). Then $\psi(X + A) = \varphi(A)\psi(X)$ for one map φ. Set $X = 0$. Then $\psi(A) = \varphi(A)\psi(0)$. Thus $\psi(X + A) = \psi(X)\psi(A)/\psi(0)$. Define $\eta(X) = \psi(X)/\psi(0)$. Then $\eta(X + A) = \eta(X)\eta(A)$. This is Cauchy's exponential functional equation. Its only continuous solutions are $\eta(X) = \exp(\lambda \cdot X)$ [20]. □

Proposition 2.10 shows that $\mathbb{C}[G]$ is the only home for bundling. The choice is therefore the group G, not the location of the operation.

3 Helical Reduced Representations: Binding on $\mathbb{Z}$

3.1 Construction

Definition 3.1 (Helical Reduced Representation)**.** *A Helical Reduced Representation (HeRR) is a finite sum*

$$R(t) = \sum_n c_n t^n, \qquad n \in \mathbb{Z}, \quad c_n \in \mathbb{C}^D. \tag{2}$$

Each c_n is a coefficient. Each n is a height. The variable t is formal. The name refers to the helix, which is the universal cover of the circle: a height records the winding count that the circle group $\mathbb{Z}_D$ discards. An atom is one term $c\,t^0$, with c a phasor vector. An atom at height h is $c\,t^h$.

Definition 3.2 (Operations)**.**

- *Bind: $R \cdot S$. Coefficients of equal total height multiply elementwise. Heights add. For single terms, $(c\,t^h)(d\,t^g) = (c \odot d)\,t^{h+g}$.*
- *Bundle: $R + S$. Terms of equal height add. Terms of different height stay separate.*
- *Unbind: multiply by the inverse term $\bar{c}\,t^{-h}$, where $\bar{c}$ is the elementwise conjugate.*
- *Fold: substitute $t = 1$. All heights map to zero. The result is a vector in $\mathbb{C}^D$.*

Proposition 3.3. *(a) Unbind is exact: $(\bar{c}\,t^{-h})(c\,t^h) = \mathbf{1}\,t^0$. (b) Fold is a ring homomorphism, and folding HeRR gives FHRR. (c) Substituting $t^D = 1$ and applying the inverse DFT gives HRR. (d) A bundle of K atoms at distinct heights is a K-sparse polynomial. (e) The degree of a term equals the sum of the heights of the atoms that were bound to form it.*

Proof. (a) $\bar{c} \odot c = \mathbf{1}$ for phasors. (b) Substitution is a ring homomorphism. With $t = 1$, bind is $\odot$ and bundle is $+$, which is FHRR. (c) With $t^D = 1$, the ring is $\mathbb{C}[\mathbb{Z}_D] \otimes \mathbb{C}^D$. The DFT identifies $\mathbb{C}[\mathbb{Z}_D]$ with the algebra of circular convolution. (d) and (e) follow from Definition 3.2. □

Statement (e) says that nesting depth is polynomial degree. Statement (b) says that every existing VSA is the fold of HeRR. The fold sends every height to zero. The fold discards the heights.

3.2 Compression is a choice of characters

A HeRR has no fixed size. To store it in fixed size, evaluate it at M points $t_1, \ldots, t_M \in \mathbb{C} \setminus \{0\}$:

$$S_m = R(t_m) = \sum_n c_n t_m^n \in \mathbb{C}^D, \qquad m = 1, \ldots, M. \tag{3}$$

The storage is the $M \times D$ array S. Evaluation is a ring homomorphism. Therefore bind and unbind can be applied to S directly. Each evaluation point t_m is a character of $\mathbb{Z}$ (Fact 2.6). Two families of characters are used.

Unitary characters. Let $\omega = e^{2\pi i/M}$. Let $t_m = \omega^{m-1}$ for $m = 1, \ldots, M$. Then a term at height h contributes $\omega^{(m-1)h}$ to S_m. The mean over m is

$$\frac{1}{M} \sum_{m=1}^{M} \omega^{(m-1)h} = \begin{cases} 1 & h \equiv 0 \pmod M, \\ 0 & \text{otherwise.} \end{cases} \tag{4}$$

This is the DFT on $\mathbb{Z}_M$. The heights modulo M are called coils. There are M coils.

Budget	Mechanism	Retrieval	Ceiling (pairs)	Prior VSAs
Dimension D	statistical incoherence	approximate	about $D/(c \ln N)$	yes
Coils M	unitary characters, eq. (4)	exact	M (known heights); $M/2$ (unknown)	as block codes
Precision	non-unitary character $t = B$	exact	about $\lfloor p/\log_2 B \rfloor$ levels per coil	no

Table 2: Capacity budgets of HeRR. p is the mantissa width in bits.

Non-unitary characters. Let $t = B$ with real $B > 1$. Then a term at height h contributes $c_h B^h$. Terms of different height have different magnitude. Compact groups have no non-unitary characters (Fact 2.5). The group $\mathbb{Z}$ does.

Both families. Let $t_m = B^{1/M}\omega^{m-1}$. Write a height as $h = \ell M + c$ with $c \in \{0, \dots, M-1\}$. Then

$$t_m^h = B^{\ell + c/M}\, \omega^{(m-1)c}. \tag{5}$$

The phase factor selects the coil c. The magnitude factor separates the levels ℓ inside one coil.

3.3 Three capacity budgets

Equation (5) gives three independent quantities that a representation can use (Table 2).

The precision budget does not depend on D. It depends on the floating-point mantissa. With $p = 53$ bits (IEEE double precision), $B = 64$ gives at most 8 to 9 levels. The practical limit is cancellation residue. The mean in (4) cancels terms of magnitude B^ℓ. The residue is of order $B^{\ell_{\max}} 2^{-p}$. The residue must be smaller than the smallest term, which has magnitude 1. Two constraints act in opposite directions. A small base gives many levels but breaks the peel step, which needs $B \gtrsim QK_c/\pi$. A large base makes the peel safe but reduces the level count, so the product ML falls below K. The optimum is therefore interior. Section 4.5 measures it.

3.4 Decoders

Decoder A: unknown heights (sparse interpolation). Let the bundle have K terms with unknown heights h_k and coefficients c_k. Evaluate at $t_m = r\omega^{m-1}$ with $r \geq 1$. Then

$$S_m = \sum_k c_k z_k^{m-1}, \qquad z_k = r^{h_k}\omega^{h_k}. \tag{6}$$

Each column of S is a sum of K exponentials. The heights are the same for every column. The matrix pencil method [21; 22] recovers the z_k as follows.

1. Set $L = \lfloor M/2 \rfloor$. For each column d, form the $L \times (M - L + 1)$ Hankel matrix H_d with entries $S_{i+j,d}$.
2. Concatenate $H_1, \dots, H_D$ horizontally into H.
3. Compute the singular value decomposition $H = U\Sigma V^*$. Set $\hat{K}$ to the number of singular values above a tolerance.
4. Let U_K be the first $\hat{K}$ columns of U. Let $U_\uparrow$ be U_K without its last row and $U_\downarrow$ be U_K without its first row.
5. The eigenvalues of $U_\uparrow^+ U_\downarrow$ are the z_k. Here $U_\uparrow^+$ is the pseudo-inverse.
6. Recover $h_k = \mathrm{round}\big(\log z_k / \log(r\omega)\big)$. Recover the coefficients by least squares on the Vandermonde system (6).

Exact recovery requires $M \geq 2K$ [23; 24; 21; 22].

Decoder B: known heights (mean and peel). Let the query supply role r_q and its height h_q. Let phases be quantised to Q levels. Do the following.

1. Unbind on the samples: $S_m \leftarrow S_m \odot \bar{r}_q \cdot t_m^{-h_q}$. The target is now at height 0.
2. Select the coil: $v \leftarrow \frac{1}{M}\sum_m S_m\, \omega^{-(m-1)c_q}$. By (4), only terms in coil c_q remain.
3. Normalise: $v \leftarrow v/B^{c_q/M}$. Terms in coil c_q now have magnitudes B^ℓ.

K pairs	FHRR (2048-d)	HeRR, Decoder B
31	1.00	1.00
64	1.00	1.00
128	1.00	0.90
256	0.87	0.67
512	0.30	0.30

Table 3: Retrieval accuracy with unitary characters only. Heights are distinct for $K \leq 64$ and collide for $K > 64$.

4. Peel. While $\max_j |v_j| > 1/2$: set $\ell = \text{round}(\log \max_j |v_j| / \log B)$. Set $\theta_j = \text{round}(Q \arg(v_j / B^\ell)/2\pi) \bmod Q$. Set $u_j = \exp(2\pi i \theta_j / Q)$. Record (ℓ, u). Set $v \leftarrow v - B^\ell u$.
5. Return the recorded vector at level ℓ_q.

Step 4 is exact when the sum of all lower terms is smaller than half a phase step. This holds when $B \gtrsim QK_c/\pi$, where K_c is the number of terms in the coil.

Consequence. Neither decoder observes crosstalk. Both recover one vector, then compare it with the codebook. The codebook does not need to be random.

4 Results

4.1 Experimental protocol

All comparisons use equal storage, counted in complex numbers. FHRR uses dimension $M \cdot D$. HeRR uses M evaluation points and coefficients of dimension D. The task is role–filler retrieval. A bundle stores K pairs $r_k \odot f_k$. The query gives one role r_q. Retrieval is correct when the decoded vector is nearest to the correct filler in the codebook. Accuracy is the fraction of correct retrievals over 30 to 40 trials. Codebook size N is 500, or 1000 when $K > 500$. Two codebooks are used: random (independent uniform phases) and correlated (all vectors within angular spread 0.3 of one prototype). The code is in the file `spring_vsa.py`.

4.2 Properties of the ring

The following properties were verified numerically. Bind followed by unbind returns the input to machine precision. Bundling two terms of different height gives two terms. Folding at $t = 1$ equals the FHRR bundle to machine precision. Decoder A recovers $K = 16$ pairs from M evaluation points with accuracy 1.00 for $M \geq 40$, 0.68 for $M = 32$, and 0.05 for $M = 24$. This confirms the bound $M \geq 2K$.

4.3 Decoder B with unitary characters

With known heights, Decoder B needs no singular value decomposition. Table 3 gives accuracy for $M = 64$, $D = 32$, random codebook.

Retrieval is exact up to $K = M$. For $K > M$, heights must collide. Each coil then holds K/M terms in D dimensions. Accuracy converges to the FHRR value. With unitary characters only, HeRR equals block-partitioned FHRR.

4.4 The non-unitary character

Table 4 uses one evaluation point $t = B = 64$, storage D complex numbers, phases quantised to $Q = 16$.

HeRR fails at $K = 10$ for every D. The cause is precision: $64^9 \approx 2^{54}$ exceeds the 53-bit mantissa. The ceiling is independent of D.

D	K pairs	FHRR (D)	HeRR, magnitude
16	8	0.15	1.00
32	9	0.35	1.00
64	9	0.80	1.00
16, 32, 64	10	0.10 – 0.65	0.75 – 0.85

Table 4: Retrieval accuracy with the magnitude channel. Accuracy is 1.00 through $K = 9$ at every D.

K pairs	FHRR (128-d)	HeRR (8×16)
16	0.87	1.00
32	0.53	1.00
48	0.20	1.00
64	0.17	1.00
96	0.13	1.00

Table 5: Retrieval accuracy with coils and levels, random codebook.

4.5 Both characters together

Table 5 uses $M = 8$ coils, $D = 16$, $B = 16$, and 12 levels. Storage is 128 complex numbers for both methods.

HeRR retrieves 96 pairs with accuracy 1.00 in 128 numbers. FHRR at the same storage has accuracy 0.13. The coil and precision budgets multiply.

Figure 1(c) sweeps the base B at fixed storage, with the level count set to $L = \lfloor 50/\log_2 B \rfloor$. Accuracy at $K = 96$ is 0.24 at $B = 2$, 0.88 at $B = 8$, 1.00 at $B = 16$, and undefined above $B = 32$ because $ML < K$. The optimum is interior, as Section 3.3 predicts.

4.6 Correlated codebook

Table 6 repeats Table 5 with the correlated codebook.

FHRR fails at 8 pairs. With a correlated codebook, the crosstalk is a bias toward the prototype, not zero-mean noise. More dimensions do not remove a bias. HeRR does not observe crosstalk. Its accuracy is unchanged until the coils are exhausted.

4.7 Nesting depth

A chain of depth d is $k_{d-1} \odot (\cdots \odot (k_1 \odot x))$. The bundle holds one target chain and several distractor chains of the same depth. The query supplies all keys of the target chain. Storage is 32 complex numbers: HRR uses 64 real numbers, FHRR uses 32 dimensions, HeRR uses 8 coils of dimension 4. In HeRR, each key has a height. Two settings are used. In the random setting, key heights are random. In the assigned setting, the first key height of each chain is set so that the chain degree modulo M is distinct from every other chain.

Real-valued HRR loses accuracy with depth. Its unbinding uses an approximate inverse, and the error grows at each level. FHRR does not lose accuracy with depth. Its unbinding is exact. FHRR loses accuracy only with the number of distractors. HeRR with random heights behaves as block-partitioned FHRR. HeRR with assigned heights is exact at every depth and every distractor count tested. Chain degree is an address. An encoder that controls degrees controls collisions.

5 Discussion

5.1 Summary of contributions

The paper makes four claims that are new to the VSA literature.

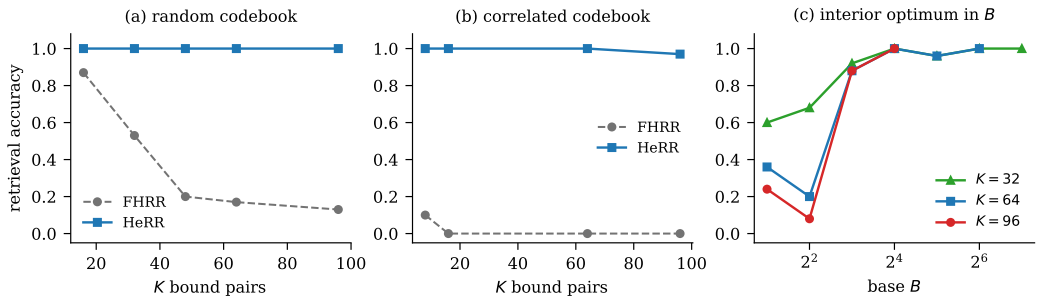


Figure 1: Retrieval accuracy at equal storage of 128 complex numbers. (a) Random codebook. (b) Correlated codebook; FHRR fails at 8 pairs because crosstalk is a bias, not zero-mean noise. (c) Accuracy against the base B at fixed storage, with level count $L = \lfloor 50/\log_2 B \rfloor$. Missing points are configurations where $ML < K$.

K pairs	FHRR (128-d)	HeRR (8×16)
8	0.10	1.00
16	0.00	1.00
64	0.00	1.00
96	0.00	0.97

Table 6: Retrieval accuracy with a correlated codebook.

1. Compact binding groups store phase only, and the capacity bound follows (Proposition 2.9). Bundling cannot be moved to the Lie algebra (Proposition 2.10).
2. HeRR, a VSA on the non-compact group $\mathbb{Z}$, exists. Every prior VSA is its fold (Proposition 3.3).
3. Non-unitary characters store information in the magnitude channel. Numerical precision is a capacity budget independent of dimension (Tables 2 and 4).
4. Retrieval is exact by sparse interpolation or by mean-and-peel. Therefore the codebook does not need to be random (Table 6).

5.2 Limitations

With a random codebook, in the approximate regime ($K > M$), HeRR has the same accuracy as FHRR at equal storage (Table 3). The gains are exactness, addressable structure, and the precision budget. They are not a larger item count in the approximate regime.

The precision budget requires floating-point arithmetic. Bipolar, binary, and in-memory hyperdimensional hardware have no mantissa. On this hardware, only the dimension and coil budgets are available.

FHRR is already exact under nesting (Table 7). Our depth result is exactness with assigned degrees. It is not a property that FHRR lacks in the approximate regime.

Decoder A costs one singular value decomposition of an $L \times (M - L + 1)D$ matrix. Decoder B costs one mean and at most $p/\log_2 B$ peel steps. Both are cheaper than a resonator network, which iterates to convergence.

5.3 Relation to prior work

Group-theoretic VSAs. Yu et al. [25] define finite-group VSAs and characterise their expressible similarity matrices. Ozen [26] identifies HRR with the group algebra $\mathbb{C}[\mathbb{Z}_n]$ and proves that unitary group VSAs are maximally expressive under stated assumptions. Both works stay within compact groups. Ozen's theorem holds within the unitary regime. It does not cover the magnitude channel, because unitary representations have none. GHRR [19] and orthogonal MBAT [18] choose unitary or

Depth	Distractors	HRR (real)	FHRR	HeRR, random	HeRR, assigned
2	7	0.50	0.60	0.45	1.00
4	7	0.43	0.73	0.53	1.00
8	3	0.23	0.98	0.65	1.00
16	3	0.20	0.95	0.83	1.00
16	7	0.10	0.73	0.65	1.00

Table 7: Retrieval of the innermost atom of a nested chain at 32 complex numbers of storage.

orthogonal matrices for numerical stability. The instability they avoid is the unmanaged non-unitary channel that Section 3.2 manages.

Capacity theory. Frady, Kleyko and Sommer [4] derive the capacity of VSA superposition as a linear recurrent network with orthogonal weights. They report that many VSA models have the same capacity. Proposition 2.9 accounts for this: the models they compare are unitary. The same work uses contracting recurrent weights (magnitude below 1) as a forgetting mechanism. The non-unitary character $t = B > 1$ is the same mechanism with the opposite sign and the opposite purpose. Plate [1], Thomas et al. [27], and Clarkson et al. [28] give further capacity bounds. All are per dimension. None treats precision as a resource.

Structure inside compact groups. Sparse block codes [5; 6], residue number systems [7], and recursive binding [8] add addressable structure inside finite cyclic groups. The coils of Section 3.2 are the unitary-character case of these methods. The levels have no counterpart in them.

Fractional power encoding. Fractional power encoding [1; 15] and Vector Function Architectures [15] compute in the Lie algebra of $\mathrm{U}(1)^D$. Proposition 2.10 shows why these methods have no bundling operation in the algebra.

Decoders. Resonator networks [9; 10] and their variants [29; 30] are iterative and approximate. Raviv [11] achieves exact recovery with random linear codes over $\mathrm{GF}(2)$. Decoder A is the first exact decoder for VSA bundles based on sparse interpolation. Unlike $\mathrm{GF}(2)$ codes, it gives depth as degree and admits correlated codebooks. The recovery guarantee comes from the sparse interpolation literature [23; 24; 21; 22].

Precision. qFHRR [31] shows that FHRR needs only 3 to 4 phase bits per dimension. Section 3.3 shows what the remaining bits can store. Spiking phasor implementations report that discarding the modulus loses information [32]. Proposition 2.9 makes this loss precise.

5.4 Extensions

The group $\mathbb{Z}$ has one non-compact direction. The group $\mathbb{Z}^r$ has r independent non-compact directions. One direction can address depth and another can address slots. A non-abelian group gives order-dependent binding without permutations. We do not study these extensions here.

AI use statement

Large language models were used in this work for four purposes, in line with the ICLR policy on AI use. They were used for literature retrieval and discovery, to locate related work in the VSA and sparse interpolation literatures. They were used during research ideation and execution, in a dialogue that produced the framing of binding as a group operation and the construction of Section 3, and to write the experiment code in the supplementary material. They were used to draft and to polish sections of the text. They were used to draft the statements and proofs of Propositions 2.9 and 2.10. The author checked every proof, ran every experiment reported here, and takes full responsibility for all claims, proofs, and results in the paper.

REPRODUCIBILITY STATEMENT

The proofs of Propositions 2.9, 2.10 and 3.3 are given in full in Sections 2.4, 2.5 and 3. Both decoders are given as complete numbered algorithms in Section 3.4. All parameters of every experiment are listed in Appendix A. The supplementary material contains a single self-contained NumPy script that reproduces every table and figure, with a fixed random seed.

A EXPERIMENTAL DETAILS

Codebooks. A random phasor vector has phases drawn independently and uniformly from $[0, 2\pi)$. A quantised phasor vector has phases drawn from the Q-th roots of unity, $Q = 16$. A correlated codebook draws one prototype phase vector ϕ. Each codebook vector has phases $\phi + 0.3\,\varepsilon$, with ε drawn from the standard normal distribution, then quantised.

Evaluation points. For Decoder A, $t_m = e^{2\pi i(m-1)/H}$ with $H = 1024$ and $m = 1, \ldots, M$. For Decoder B with unitary characters, $t_m = e^{2\pi i(m-1)/M}$. For Decoder B with both characters, $t_m = B^{1/M} e^{2\pi i(m-1)/M}$.

Height assignment. In Tables 3 and 5, heights are drawn without replacement from $\{0, \ldots, H-1\}$ (Table 3, $H = 1024$) or from $\{0, \ldots, LM-1\}$ with L levels (Table 5). In Table 3 for $K > 64$ and in Table 7 (random setting), heights are drawn with replacement from $\{0, \ldots, M-1\}$. In Table 7 (assigned setting), the first key height of chain k is $(a_k - \sum_{j>1} h_{k,j}) \bmod M$, where $a_1, \ldots, a_{K+1}$ are distinct values in $\{0, \ldots, M-1\}$.

Baselines. FHRR: vectors in $\mathrm{U}(1)^{MD}$. Bind is $\odot$. Bundle is $+$. Unbind is $\odot$ with the conjugate. HRR: real vectors in $\mathbb{R}^{2MD}$ with elements drawn from $\mathcal{N}(0, 1/2MD)$. Bind is circular convolution. Unbind is circular convolution with the involution $a_k^\dagger = a_{-k \bmod 2MD}$. Retrieval in both baselines is by maximum real inner product with the codebook.

Decoder parameters. Decoder A: $L = \lfloor M/2 \rfloor$, singular value tolerance 10^{-6} relative to the largest singular value, $\hat{K}$ capped at $L - 1$. Decoder B: $Q = 16$, peel stop threshold $1/2$, at most 12 peel steps.

REFERENCES

[1] T. A. Plate. *Distributed Representations and Nested Compositional Structure*. PhD thesis, University of Toronto, 1994; "Holographic reduced representations," *IEEE Trans. Neural Networks* 6(3):623–641, 1995; *Holographic Reduced Representation*. CSLI, 2003.

[2] P. Kanerva. Hyperdimensional computing. *Cognitive Computation* 1(2):139–159, 2009.

[3] R. W. Gayler. Vector symbolic architectures answer Jackendoff's challenges for cognitive neuroscience. *ICCS/ASCS*, pp. 133–138, 2003.

[4] E. P. Frady, D. Kleyko, F. T. Sommer. A theory of sequence indexing and working memory in recurrent neural networks. *Neural Computation* 30(6):1449–1513, 2018.

[5] M. Laiho, J. H. Poikonen, P. Kanerva, E. Lehtonen. High-dimensional computing with sparse vectors. *IEEE BioCAS*, 2015.

[6] E. P. Frady, D. Kleyko, F. T. Sommer. Variable binding for sparse distributed representations: theory and applications. *IEEE TNNLS* 34(5):2191–2204, 2023; arXiv:2009.06734. M. Hersche et al. Factorizers for distributed sparse block codes. arXiv:2303.13957, 2023.

[7] C. J. Kymn, D. Kleyko, E. P. Frady, C. Bybee, P. Kanerva, F. T. Sommer, B. A. Olshausen. Computing with residue numbers in high-dimensional representation. *Neural Computation* 37(1):1–37, 2025; arXiv:2311.04872.

[8] D. A. Rachkovskij, D. Kleyko. Recursive binding for similarity-preserving hypervector representations of sequences. *IJCNN*, 2022; arXiv:2201.11691.

[9] E. P. Frady, S. J. Kent, B. A. Olshausen, F. T. Sommer. Resonator networks, 1. *Neural Computation* 32(12):2311–2331, 2020.

[10] S. J. Kent, E. P. Frady, F. T. Sommer, B. A. Olshausen. Resonator networks, 2. *Neural Computation* 32(12):2332–2388, 2020.

[11] N. Raviv. Linear codes for hyperdimensional computing. *Neural Computation* 36(6):1084–1120, 2024; arXiv:2403.03278.

[12] G. B. Folland. *A Course in Abstract Harmonic Analysis*, 2nd ed. CRC Press, 2016.

[13] D. Bump. *Lie Groups*, 2nd ed. Springer, 2013.

[14] J.-P. Serre. *Linear Representations of Finite Groups*. Springer, 1977.

[15] E. P. Frady, D. Kleyko, C. J. Kymn, B. A. Olshausen, F. T. Sommer. Computing on functions using randomized vector representations. arXiv:2109.03429, 2021. B. Komer. *Biologically Inspired Spatial Representation*. PhD thesis, University of Waterloo, 2020.

[16] S. I. Gallant, T. W. Okaywe. Representing objects, relations, and sequences. *Neural Computation* 25(8):2038–2078, 2013.

[17] M. M. Alam, E. Raff, S. Biderman, T. Oates, J. Holt. A Walsh Hadamard derived linear vector symbolic architecture. *NeurIPS*, 2024; arXiv:2410.22669.

[18] S. I. Gallant. Orthogonal matrices for MBAT vector symbolic architectures, and a "soft" VSA representation for JSON. arXiv:2202.04771, 2022.

[19] C. Yeung, Z. Zou, M. Imani. Generalized holographic reduced representations. arXiv:2405.09689, 2024.

[20] J. Aczél. *Lectures on Functional Equations and Their Applications*. Academic Press, 1966.

[21] Y. Hua, T. K. Sarkar. Matrix pencil method for estimating parameters of exponentially damped/undamped sinusoids in noise. *IEEE Trans. ASSP* 38(5):814–824, 1990.

[22] R. Roy, T. Kailath. ESPRIT—estimation of signal parameters via rotational invariance techniques. *IEEE Trans. ASSP* 37(7):984–995, 1989.

[23] M. Ben-Or, P. Tiwari. A deterministic algorithm for sparse multivariate polynomial interpolation. *STOC*, pp. 301–309, 1988.

[24] M. Giesbrecht, G. Labahn, W. Lee. Symbolic–numeric sparse interpolation of multivariate polynomials. *J. Symbolic Computation* 44(8):943–959, 2009.

[25] T. Yu, Y. Zhang, Z. Zhang, C. De Sa. Understanding hyperdimensional computing for parallel single-pass learning. *NeurIPS* 35, 2022; arXiv:2202.04805.

[26] R. Ozen. *Exploring Symmetry and Representation in Vector Symbolic Architectures*. MCogSc thesis, Carleton University, Ottawa, 2024. `https://carleton.scholaris.ca/items/b0b13bda-1d65-469e-a9fd-e51e9169c8f1`

[27] A. Thomas, S. Dasgupta, T. Rosing. A theoretical perspective on hyperdimensional computing. *JAIR* 72:215–249, 2021.

[28] K. Clarkson, S. Ubaru, E. Yang. Capacity analysis of vector symbolic architectures. arXiv:2301.10352, 2023.

[29] J. Langenegger, G. Karunaratne, M. Hersche, L. Benini, A. Sebastian, A. Rahimi. In-memory factorization of holographic perceptual representations. *Nature Nanotechnology* 18(5):479–485, 2023.

[30] C. Yeung, P. Poduval, M. Imani. Self-attention based semantic decomposition in vector symbolic architectures. arXiv:2403.13218, 2024.

[31] S. Snyder, H. Poursiami, M. Parsa. qFHRR: rethinking Fourier holographic reduced representations through quantized phase and integer arithmetic. arXiv:2604.25939, 2026.

[32] E. P. Frady, F. T. Sommer. Robust computation with rhythmic spike patterns. *PNAS* 116(36):18050–18059, 2019.