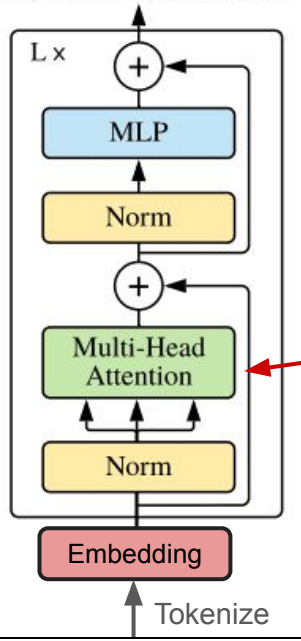
Complex KDA: Understanding and Enhancing the Expressivity of Kimi Delta Attention

Julien Siems*, Riccardo Grazzi*, Korbinian Pöppel*, Jaisidh Singh, Arber Zela, Timur Carstensen, Jenia Jitsev, Frank Hutter, Volkan Cevher, Antonio Orvieto, Aaron Klein
(*equal contribution)



Modern Language Modeling Architectures

Transformer

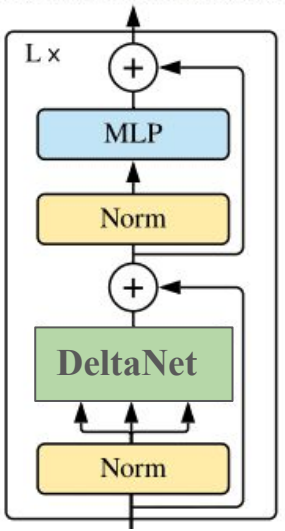


Alice, Bob and Carl each have a coin. Carl is the only one having a penny. Alice and Carl trade coins.

Channel Mixing
(Non-Linear)

Token Mixing
(Parallelizable)

DeltaNet



Linear RNNs:
More efficient for long sequences

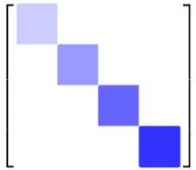
Linear RNNs

State matrix

Output

$$\boldsymbol{H}_i = \boldsymbol{A}(\boldsymbol{x}_i)\boldsymbol{H}_{i-1} + \boldsymbol{B}(\boldsymbol{x}_i), \quad \hat{\boldsymbol{y}}_i = \mathrm{dec}(\boldsymbol{H}_i, \boldsymbol{x}_i)$$

State-transition matrix



Diagonal:

Token Mix:	✓
Channel Mix:	×
Expressivity:	Parity
Examples:	Mamba, GLA

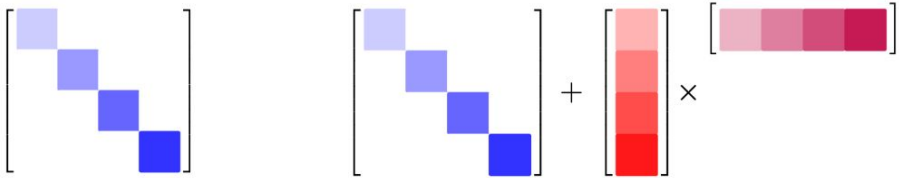
Linear RNNs

State matrix

Output

$$\boldsymbol{H}_i = \boldsymbol{A}(\boldsymbol{x}_i)\boldsymbol{H}_{i-1} + \boldsymbol{B}(\boldsymbol{x}_i), \qquad \hat{\boldsymbol{y}}_i = \mathrm{dec}(\boldsymbol{H}_i, \boldsymbol{x}_i)$$

State-transition matrix



	Diagonal:	**Rank 1 Update:**
Token Mix:	✓	✓
Channel Mix:	×	✓
Expressivity:	Parity	Reflections
Examples:	Mamba, GLA	DeltaNet, TTT, RWKV-7, KDA

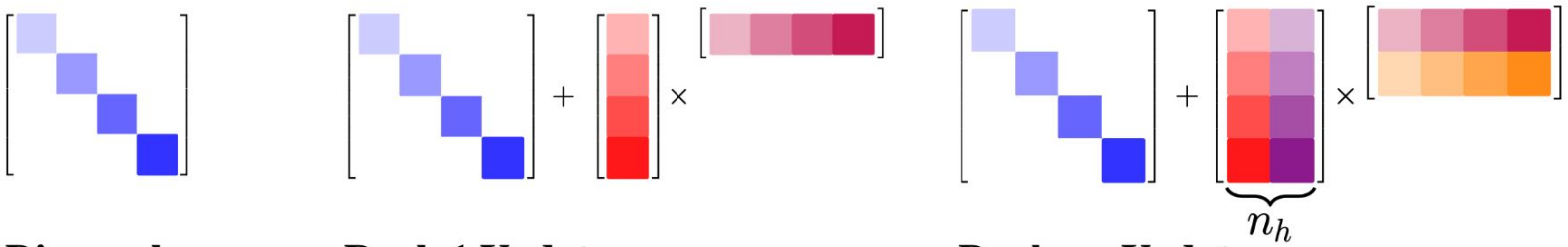
Linear RNNs

State matrix

Output

$$\boldsymbol{H}_i = \boldsymbol{A}(\boldsymbol{x}_i)\boldsymbol{H}_{i-1} + \boldsymbol{B}(\boldsymbol{x}_i), \qquad \hat{\boldsymbol{y}}_i = \mathrm{dec}(\boldsymbol{H}_i, \boldsymbol{x}_i)$$

State-transition matrix



	Diagonal:	**Rank 1 Update:**	**Rank n_h Update:**
Token Mix:	✓	✓	✓
Channel Mix:	×	✓	✓
Expressivity:	Parity	Reflections	Reflections, Rotations
Examples:	Mamba, GLA	DeltaNet, TTT, RWKV-7, KDA	**DeltaProduct**, SLiCE

Linear RNNs

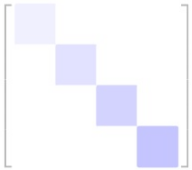
State matrix

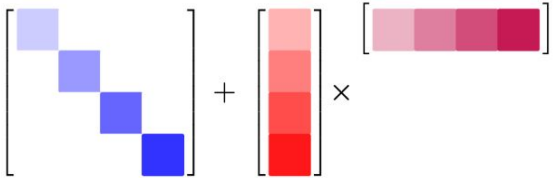
$$\boldsymbol{H}_i = \boldsymbol{A}(\boldsymbol{x}_i)\boldsymbol{H}_{i-1} + \boldsymbol{B}(\boldsymbol{x}_i),$$

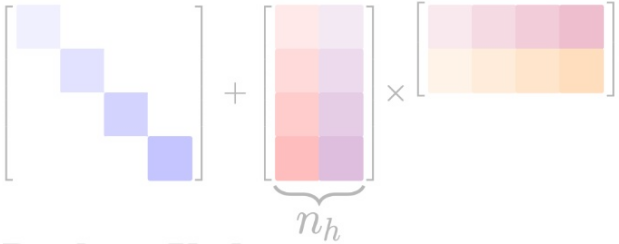
State-transition matrix

Output

$$\hat{\boldsymbol{y}}_i = \mathrm{dec}(\boldsymbol{H}_i, \boldsymbol{x}_i)$$







	Diagonal:	Rank 1 Update:	Rank n_h Update:
Token Mix:	✓	✓	✓
Channel Mix:	×	✓	✓
Expressivity:	Parity	Reflections ?	Reflections, Rotations
Examples:	Mamba, GLA	DeltaNet, TTT, RWKV-7, **KDA**	**DeltaProduct**, SLiCE

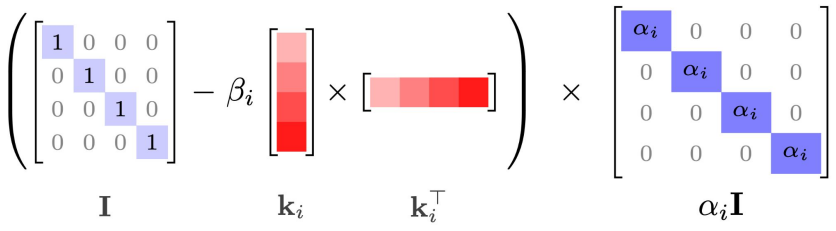
Linear RNNs

State matrix

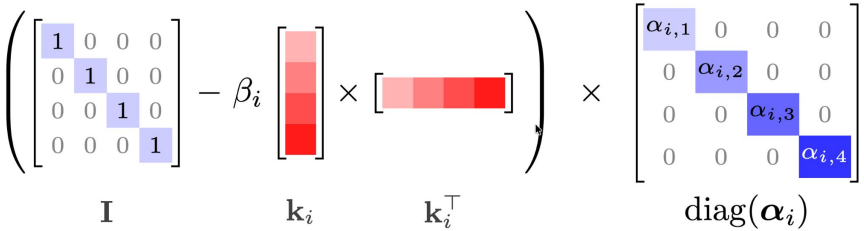
Output

$$\boldsymbol{H}_i = \boldsymbol{A}(\boldsymbol{x}_i)\boldsymbol{H}_{i-1} + \boldsymbol{B}(\boldsymbol{x}_i), \qquad \hat{\boldsymbol{y}}_i = \mathrm{dec}(\boldsymbol{H}_i, \boldsymbol{x}_i)$$

State-transition matrix



Gated DeltaNet



Kimi Delta Attention

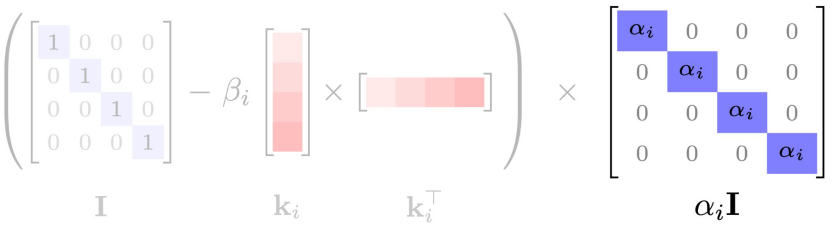
Linear RNNs

State matrix

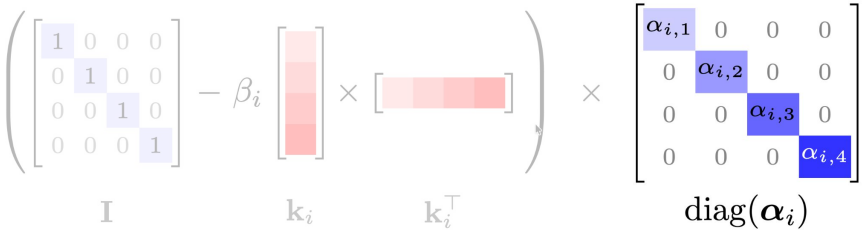
Output

$$\boldsymbol{H}_i = \boldsymbol{A}(\boldsymbol{x}_i)\boldsymbol{H}_{i-1} + \boldsymbol{B}(\boldsymbol{x}_i), \qquad \hat{\boldsymbol{y}}_i = \mathrm{dec}(\boldsymbol{H}_i, \boldsymbol{x}_i)$$

State-transition matrix



Gated DeltaNet



Kimi Delta Attention

Question of the paper: What expressivity does the full diagonal add?

Symmetry

When is $(\mathbf{I} - \beta\mathbf{k}\mathbf{k}^\top)\mathrm{Diag}(\alpha) = \mathrm{Diag}(\alpha)(\mathbf{I} - \beta\mathbf{k}\mathbf{k}^\top)$ **?**

$$\beta k_i k_j(\alpha_i - \alpha_j) = 0 \qquad \text{for all } i, j$$

Gated DeltaNet's A(xi) is **symmetric, but KDA need not be.**

Symmetry

So does KDA's A(xi) **have a complex spectrum?**



No, for positive gates $(\mathbf{I} - \beta\mathbf{k}\mathbf{k}^\top)\mathrm{Diag}(\alpha)$

is similar to $\mathrm{Diag}(\alpha)^{1/2}(\mathbf{I} - \beta\mathbf{k}\mathbf{k}^\top)\mathrm{Diag}(\alpha)^{1/2}$

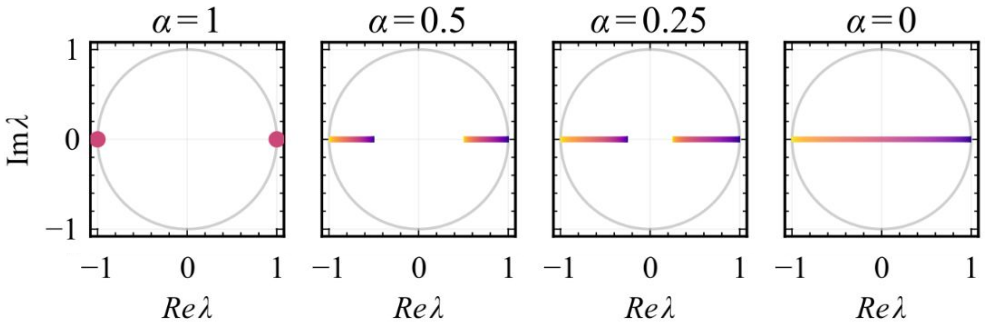


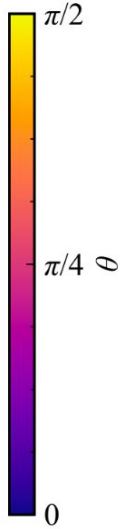
But negative gates break this symmetry!

Expanding the gate allows complex eigenvalues

$$\mathbf{D} = (\mathbf{I} - 2\mathbf{k}\mathbf{k}^\top)\mathrm{Diag}(\alpha, 1)$$

$$\boldsymbol{k} = (\cos\theta, \sin\theta)^\top \text{ and } \theta \in [0, \pi/2]$$

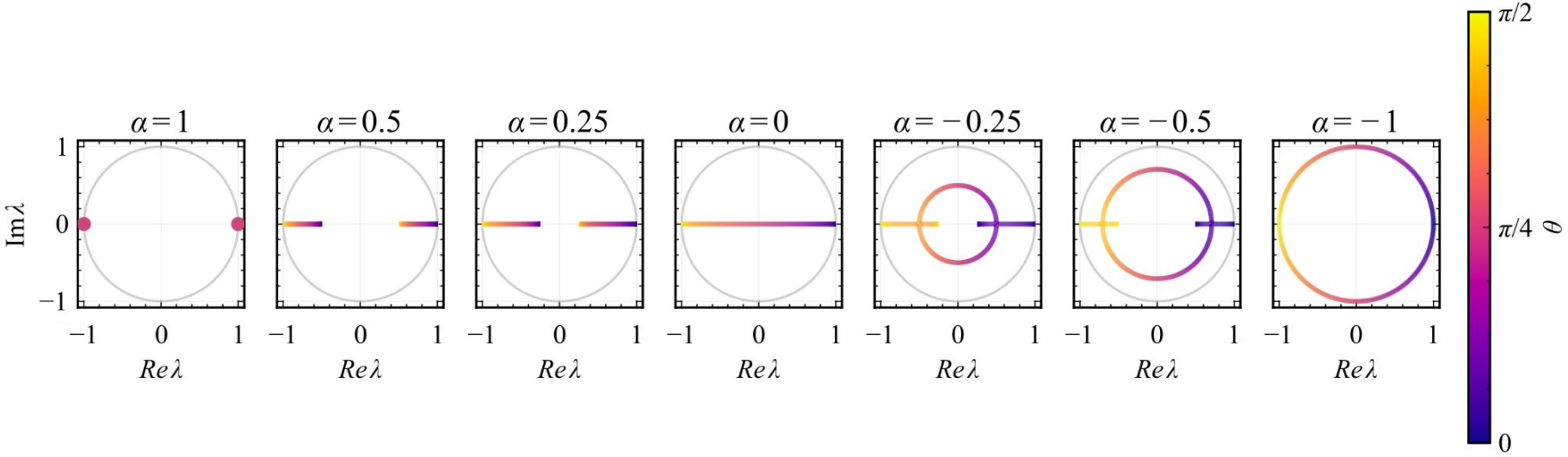




Expanding the gate allows complex eigenvalues

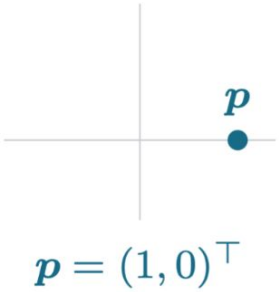
$$\mathbf{D} = (\mathbf{I} - 2\mathbf{k}\mathbf{k}^\top)\mathrm{Diag}(\alpha, 1)$$

$$\boldsymbol{k} = (\cos\theta, \sin\theta)^\top \text{ and } \theta \in [0, \pi/2]$$

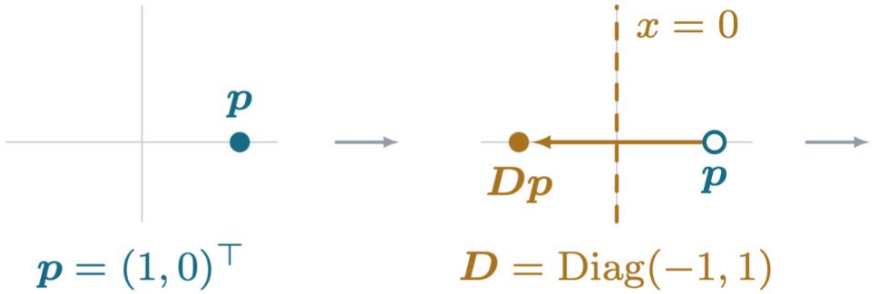


→ *Where does the rotation come from?*

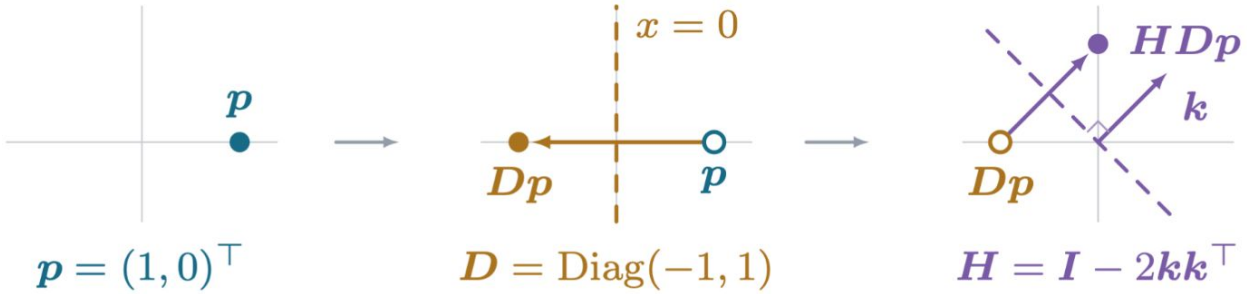
Kimi Delta Attention hides a rotation



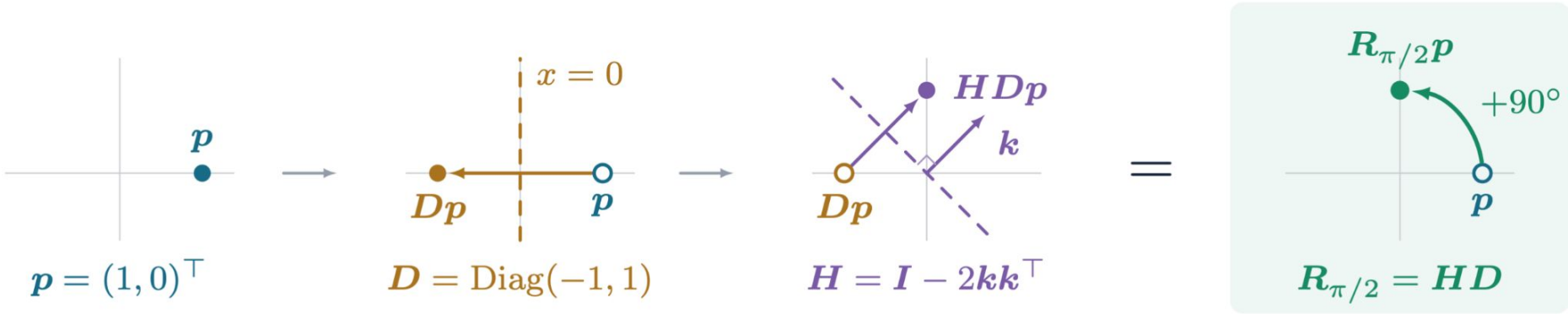
Kimi Delta Attention hides a rotation

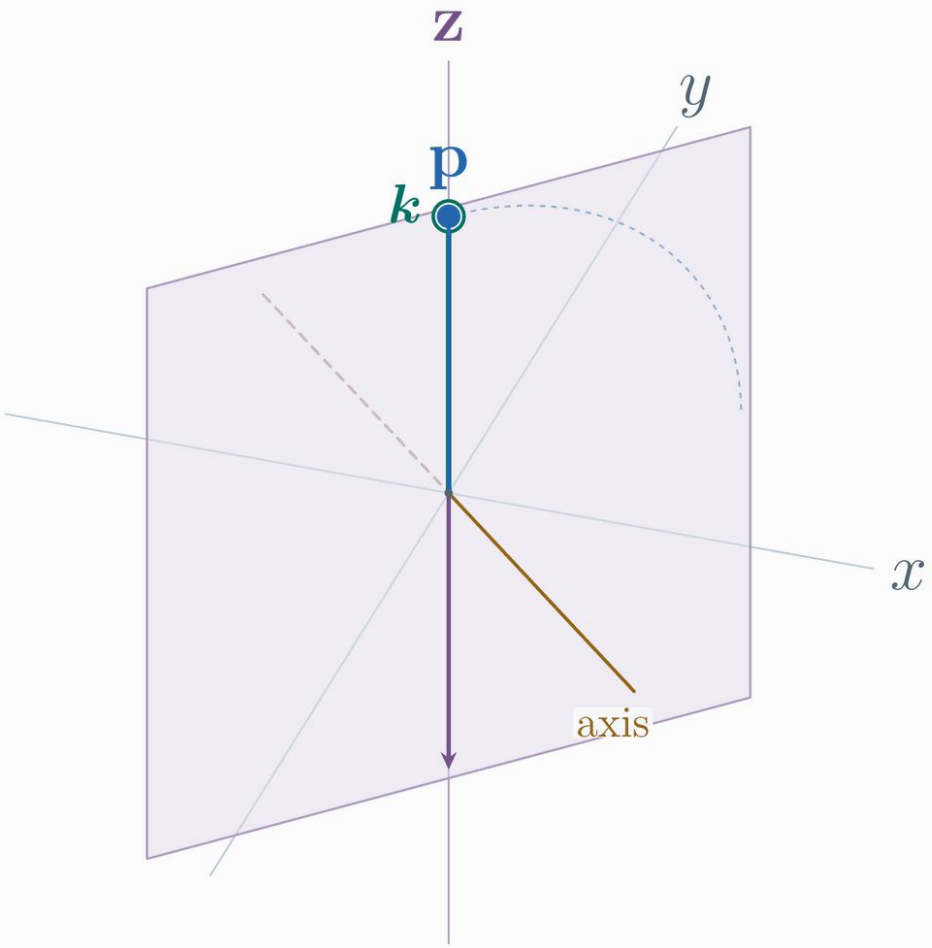


Kimi Delta Attention hides a rotation



Kimi Delta Attention hides a rotation





Diagonal gate

$D = \mathrm{diag}(1, 1, -1)$

$De_z = -e_z$ (xy-plane reflection)

Unit key

k_x 0.000

k_y 0.000

k_z 1.000

-1 0 1

State turn

$0.0°$

Key tilt

$0.0°$

KDA → Complex KDA

$$\boldsymbol{A}_t = (\boldsymbol{I} - \beta_t \boldsymbol{k}_t \boldsymbol{k}_t^\top)\,\mathrm{Diag}(\boldsymbol{\alpha}_t)$$

Kimi Delta Attention

$$\boldsymbol{\alpha}_t \in [0,1]^n$$

$$\beta_t \in [0,1]$$

Complex KDA (CKDA)

$$\boldsymbol{\alpha}_t \in [-1,1]^n$$

$$\beta_t \in [0,2]$$

From Diagonal-plus-rank-1 to Complex KDA (CKDA)

$$\boldsymbol{A} = \boldsymbol{D} + \boldsymbol{u}\boldsymbol{v}^\top \in \mathbb{R}^{n\times n}$$
$$A^\top A = I$$

$\longleftrightarrow$

$$\boldsymbol{A} = (\boldsymbol{I} - 2\boldsymbol{k}\boldsymbol{k}^\top)\boldsymbol{S}$$
$$\|\boldsymbol{k}\|_2 = 1, \boldsymbol{S} = \mathrm{Diag}(s_i), s_i \in \{-1, +1\}.$$

$\longrightarrow$ Complex KDA covers all orthogonal diagonal plus rank 1 matrices!

Spectrum of KDA

Necessary conditions for complex spectrum:

- At least two gate coordinates need to differ in sign AND
- The key vector needs to span the gates differing in sign AND
- $\beta > 1$.

Number of complex eigenvalues of CKDA:

- At most 1.
- Rotation is only planar.

Mamba-3 and CKDA state transitions

Mamba-3

$$\boldsymbol{A}_t = \rho_t \,\mathrm{blockdiag}\big(R(\theta_1), \ldots, R(\theta_m)\big)$$

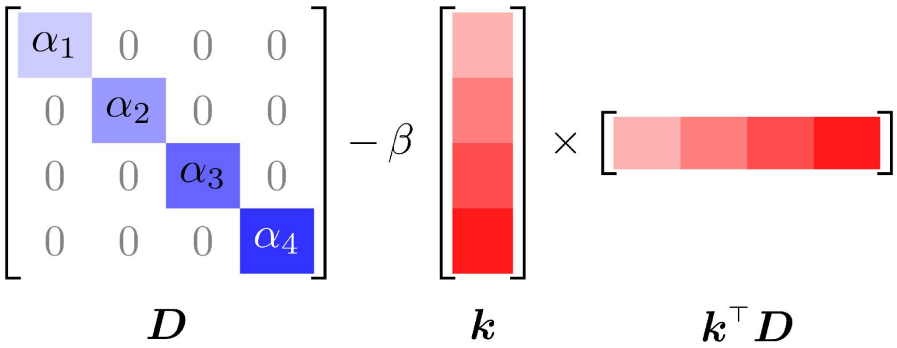
$$\rho \begin{bmatrix} c_1 & -s_1 & 0 & 0 \\ s_1 & c_1 & 0 & 0 \\ 0 & 0 & c_2 & -s_2 \\ 0 & 0 & s_2 & c_2 \end{bmatrix}$$

Fixed 2D rotation blocks

Transitions commute

Complex KDA

$$\boldsymbol{A}_t = (\boldsymbol{I} - \beta_t \boldsymbol{k}_t \boldsymbol{k}_t^\top)\,\mathrm{Diag}(\boldsymbol{\alpha}_t)$$

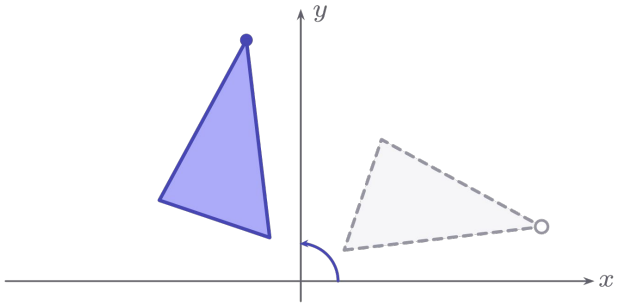


Diagonal plus rank one

Transitions need not commute

CKDA can realize any 2D rotation or reflection

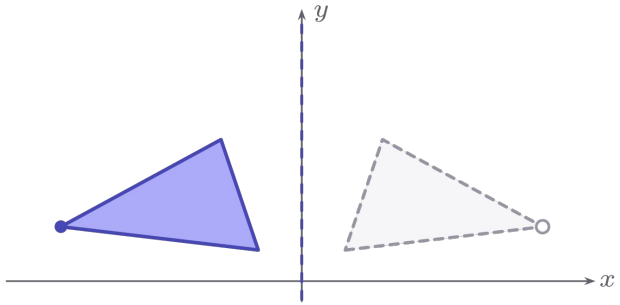
Rotation in both models



$$\rho \begin{pmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{pmatrix}$$

$$\det(\boldsymbol{A}) = +\rho^2$$

Reflection in CKDA only



$$\rho \begin{pmatrix} \cos\theta & \sin\theta \\ \sin\theta & -\cos\theta \end{pmatrix}$$

$$\det(\boldsymbol{A}) = -\rho^2$$

Linear RNNs

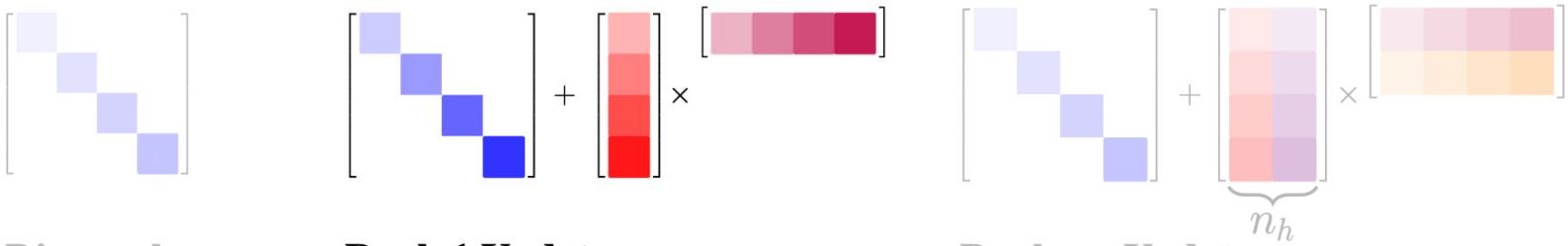
State matrix

$$\boldsymbol{H}_i = \boldsymbol{A}(\boldsymbol{x}_i)\boldsymbol{H}_{i-1} + \boldsymbol{B}(\boldsymbol{x}_i),$$

State-transition matrix

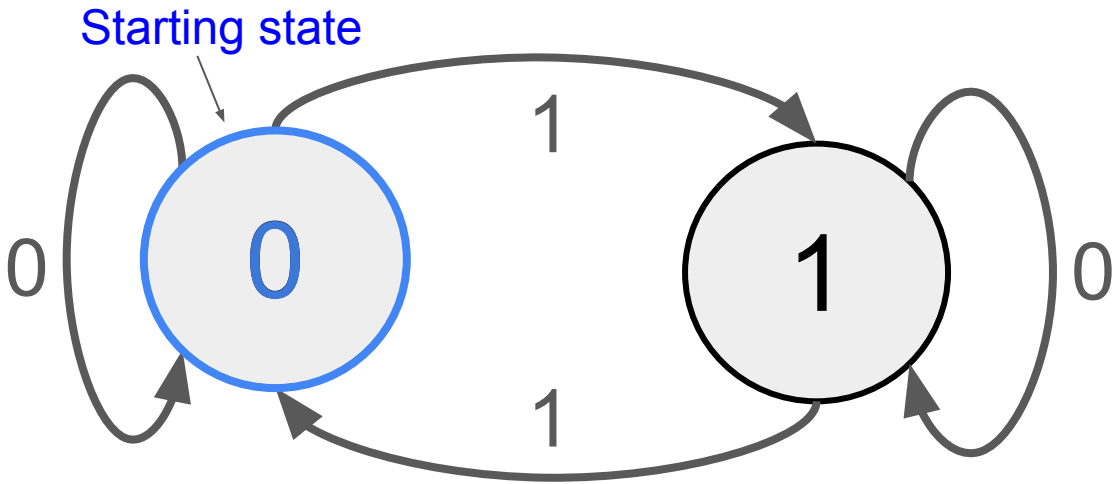
Output

$$\hat{\boldsymbol{y}}_i = \mathrm{dec}(\boldsymbol{H}_i, \boldsymbol{x}_i)$$



	Diagonal:	Rank 1 Update:	Rank n_h Update:
Token Mix:	✓	✓	✓
Channel Mix:	×	✓	✓
Expressivity:	Parity	Reflections **Rotations**	Reflections, Rotations
Examples:	Mamba, GLA	DeltaNet, TTT, RWKV-7, **KDA**	**DeltaProduct**, SLiCE

State-tracking (e.g. Parity)



Input bits	g_t	1	1	0	0	1	0
Parity State	s_t	1	0	0	0	1	1

Group state tracking (One Layer)

Group product:

$$g_t \in G, \qquad s_0 = e, \qquad s_t = g_t s_{t-1}$$

A linear RNN tracks the product in **one layer** if

$$\boldsymbol{S}_t = \boldsymbol{A}(g_t)\boldsymbol{S}_{t-1} + \boldsymbol{B}(g_t) \qquad d(\boldsymbol{S}_t, g_t) = s_t$$

Decoder (arbitrary)

The group is **tracked** if this is true for every possible sequence

The group is **realised** by the RNN if the decoder is one-to-one
=> number of reachable states = group elements (easier to learn)

CKDA tracking:

Constraint: $\boldsymbol{A}(g)$ should have the Complex KDA form

Our constructions have a **finite number of RNN states** and set

$$\boldsymbol{A}(g)^\top \boldsymbol{A}(g) = \boldsymbol{I}, \qquad \boldsymbol{B}(g) = 0 \quad \text{for every } g, \qquad \boldsymbol{S}_0 = \boldsymbol{I}$$

Orthogonal + CKDA matrix = **signed-Householder** matrix:

$$\boldsymbol{A}(g) = (\boldsymbol{I} - 2\boldsymbol{k}_g \boldsymbol{k}_g^\top)\boldsymbol{D}_g, \quad \boldsymbol{D}_g = \mathrm{diag}(\pm 1), \quad \|\boldsymbol{k}_g\|_2 = 1$$

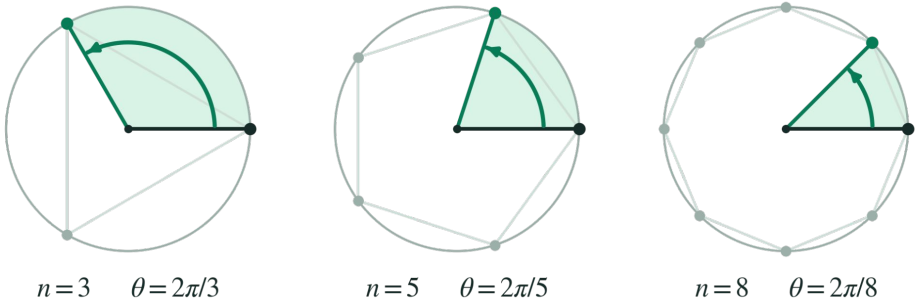
The unit vector is the reflection normal; the diagonal factor flips coordinate signs.

Cyclic and dihedral groups

Cyclic group Zn: 2D rotations with n equally spaced rotation angles

$$\boldsymbol{R}(\theta) = \left(\boldsymbol{I} - 2\boldsymbol{k}(\theta)\boldsymbol{k}(\theta)^\top\right)\boldsymbol{D}_0 \qquad \boldsymbol{k}(\theta) = \begin{pmatrix}\cos(\theta/2)\\ \sin(\theta/2)\end{pmatrix}, \qquad \boldsymbol{D}_0 = \mathrm{diag}(-1, 1)$$

$$\theta = 2\pi j/n, \qquad j = 0, \ldots, n-1$$



Dihedral group Dn:
also allow the **Householder reflections** by setting $\boldsymbol{D}_0$ to the identity

Every cyclic and dihedral group is realisable in d = 2

Which 3D rotations does CKDA allow?

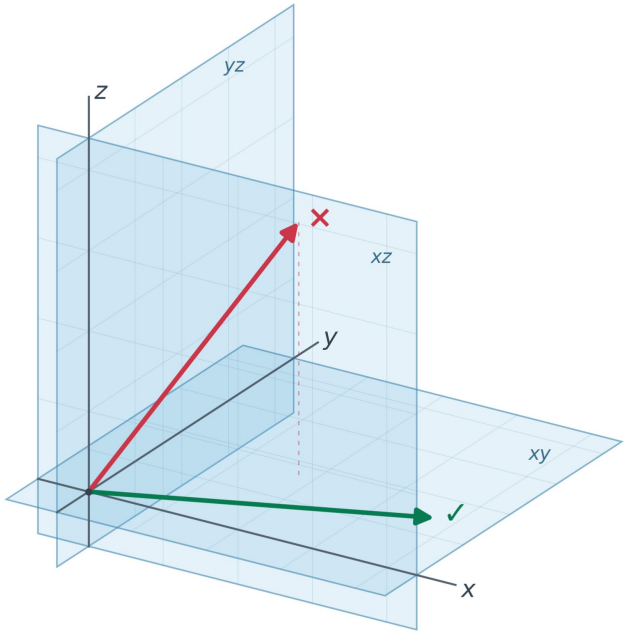
Let $\boldsymbol{R}_{\theta,\boldsymbol{a}}$ be a rotation with angle θ and unit axis $\boldsymbol{a}$.

For angles other than 0 or π:

$\boldsymbol{R}_{\theta,\boldsymbol{a}}$ is signed-Householder

$\Updownarrow$

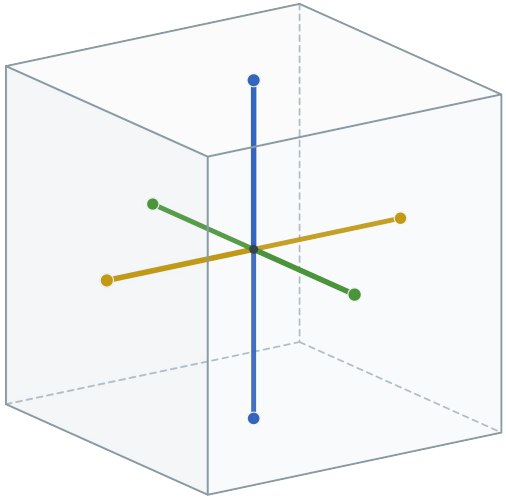
$a_i = 0$ for at least one coordinate i



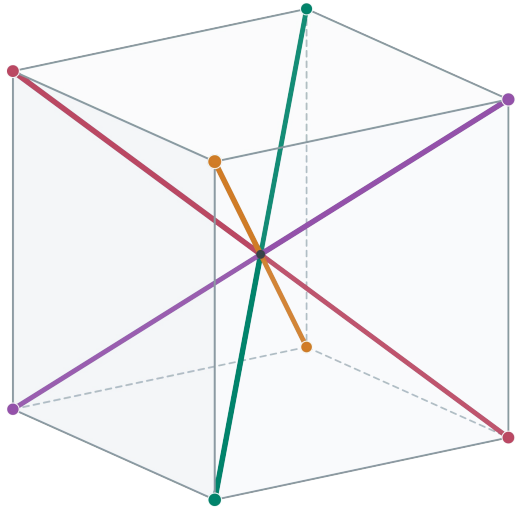
The rotation axis lies in a coordinate plane

Reorienting the cube (S4 and A4)

S4 = symmetry preserving rotations of a cube. 7 relevant rotation axis:



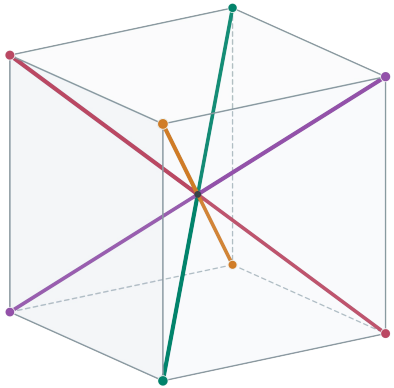
3 Face axis

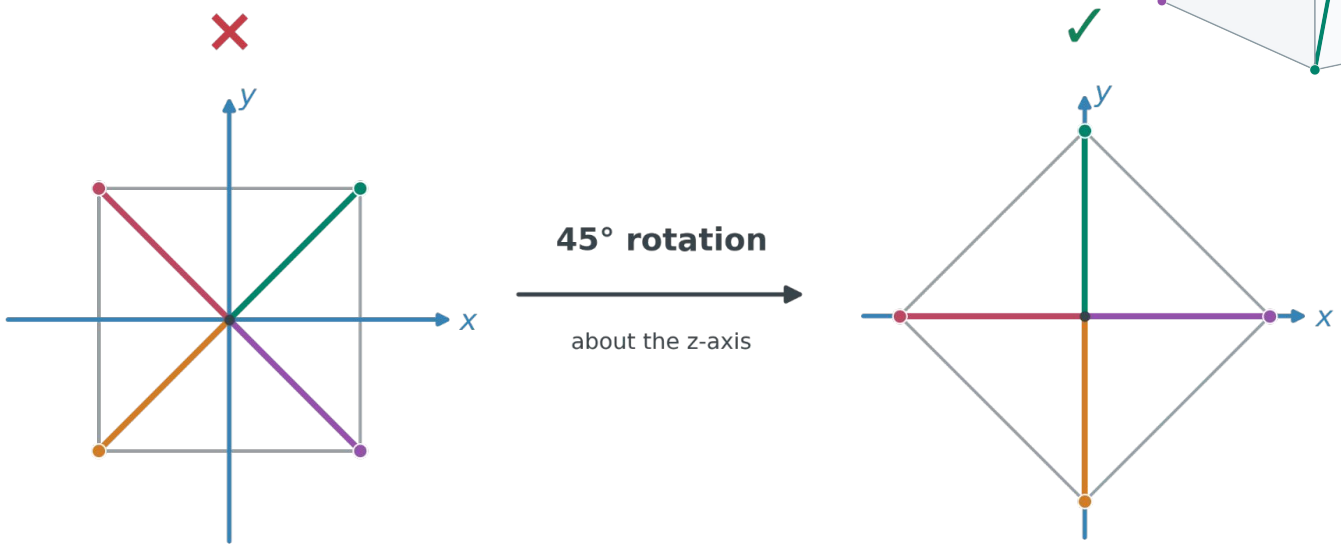


4 vertex axis

Reorienting the cube (S4 and A4)

S4 = symmetry preserving rotations of a cube. 7 relevant axis





=> Every relevant axis has now a zero coordinate.
S4 can be realized by one CKDA layer.

The three-dimensional obstruction for A5

A5 is the group of symmetry preserving rotations of a icosahedron with 16 relevant axis:



Lemma: Any plane contains at most 4 of the 16 relevant axes.
Hence, the 3 coordinate planes can contain at most 12 axes in total.

No 3D realization in any orientation.

Stronger **theorem**: no finite-state signed-Householder tracker in d = 3.

A 4D encoding for 3D rotations (Quaternions)

Lift any rotation $\boldsymbol{R}_{\theta,\boldsymbol{a}} \in SO(3)$ to a 4D signed-Householder rotation

$$E(\boldsymbol{R}_{\theta,\boldsymbol{a}}) = (\boldsymbol{I} - 2\boldsymbol{k}(\boldsymbol{a},\theta)\boldsymbol{k}(\boldsymbol{a},\theta)^\top)\boldsymbol{D}_0 \in SO(4)$$

$$\boldsymbol{k}(\boldsymbol{a},\theta) = \begin{pmatrix} \cos(\theta/2) \\ \sin(\theta/2)\boldsymbol{a} \end{pmatrix} \qquad \boldsymbol{D}_0 = \mathrm{diag}(-1,1,1,1)$$

$\boldsymbol{a}$: unit rotation axis $\quad \theta$: rotation angle

One Rotation plane: $\mathrm{span}\left\{\boldsymbol{e}_0, \begin{pmatrix} 0 \\ \boldsymbol{a} \end{pmatrix}\right\} \subset \mathbb{R}^4$

Tracking A5 with the appropriate decoder

$$f_+ : SO(4) \longrightarrow SO(3)$$

(1) preserves multiplication (homomorphism)

$$f_+(\boldsymbol{S}_1\boldsymbol{S}_2) = f_+(\boldsymbol{S}_1)f_+(\boldsymbol{S}_2)$$

(2) recovers each rotation

$$f_+\big(E(\boldsymbol{R})\big) = \boldsymbol{R}$$

Therefore $$f_+\big(E(\boldsymbol{R})\boldsymbol{S}\big) = f_+\big(E(\boldsymbol{R})\big)f_+(\boldsymbol{S}) = \boldsymbol{R}f_+(\boldsymbol{S})$$

Starting from the identity, induction gives **every 3D rotations product.**

For A5: 60 input rotations and 7,200 hidden states: decoder is **many-to-one**.

One CKDA layer tracks A5 in d = 4

Concurrent work (Linear RNNs with Quaternions).

Non-Abelian State Tracking with Quaternionic State Space Models

ICLR 2027 Conference Submission46254 Authors

18 Sep 2026 (modified: 13 Nov 2026) ICLR 2027 Conference Submission Everyone Revisions BibTeX CC BY 4.0

Keywords: state space models, sequence modeling, state tracking, expressivity, linear recurrent neural networks, quaternions, non-commutative dynamics, rotation groups, efficient architectures, language modeling

TL;DR: Icosian, a quaternionic state-space model with expressive non-commutative dynamics.

Symmetry in Time: A Representation Theory of Linear Recurrent Memory

Anif N. Shikder,[1,*] Ramit Dey,[1] Ján Mináč,[2,1] and Lyle E. Muller[3,1]

[1] *Fields Lab for Network Computation, Fields Institute, Toronto ON, Canada*
[2] *Department of Mathematics, Western University, London ON, Canada*
[3] *Department of Neuroscience, University of Texas at Dallas, Richardson TX, United States*

[Submitted on 12 Nov 2026]

Use **quaternions** directly => Can solve A5 but lack the expressiveness of Householder matrices

The obstruction to tracking S5

Theorem (spectral obstruction)

One layer, finitely many independent heads, arbitrary joint decoder:

$$\boldsymbol{S}_t^{(h)} = \boldsymbol{A}^{(h)}(g_t)\boldsymbol{S}_{t-1}^{(h)} + \boldsymbol{B}^{(h)}(g_t), \qquad \widehat{g}_t = d(\boldsymbol{S}_t)$$

Non-expansive transitions

$$\|\boldsymbol{A}^{(h)}(g)\|_2 \le 1 \qquad (\forall h, g)$$

Bounded reachable states

$$\sup_{\boldsymbol{S}\in\mathcal{R}} \|\boldsymbol{S}\|_F < \infty$$

State separability

$$\exists \varepsilon > 0 \;\; \forall \boldsymbol{S}, \boldsymbol{S}' \in \mathcal{R} : \\ d(\boldsymbol{S}) \neq d(\boldsymbol{S}') \Rightarrow \|\boldsymbol{S} - \boldsymbol{S}'\|_F \ge \varepsilon$$

Spectral bound

At most one non-real conjugate eigenvalue pair on the unit circle for each transition in each head:
(Covers **CKDA**, **Gated DeltaProduct3** but also **Mamba 3**)

Then tracking S5 (permutation of 5 objects) is impossible.

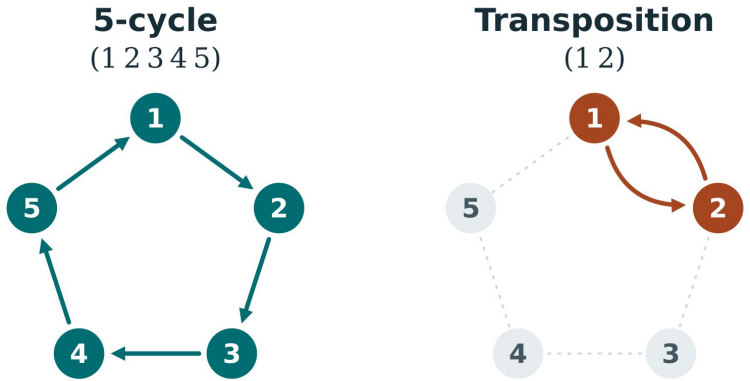
Proof sketch

1. Convert the layer into an orthogonal system

$$\boldsymbol{X}_t^{(h)} = \widetilde{\boldsymbol{A}}^{(h)}(g_t)\boldsymbol{X}_{t-1}^{(h)}, \qquad \widetilde{\boldsymbol{A}}^{(h)}(g)^\top \widetilde{\boldsymbol{A}}^{(h)}(g) = \boldsymbol{I}$$

That has a **finite number of reachable states** and where the transitions **preserve the the spectral bound**.

2. Craft a sequence involving powers of a **5-cycle** and a **transposition** and show that it can't be correctly decoded.



Complex KDA Tracking in One Layer (Theory)

✓ Symmetries of a polygon: Zn, Dn

✓ Rotations of a cube: S4

✓ Rotations of an icosahedron: A5

✗ All permutations of 5 items: S5

Matching DeltaProduct2. KDA can't solve parity.
GDN with negative eigenvalues can't solve the cyclic groups.

37

Multilayer Expressivity

- CKDA can simulate a DFA and WFA in **three** layers (DeltaNet needs four)
  DFA: deterministic finite automaton WFA: weighted finite automaton
- needs a further extension of β range to support expansive state (copying)
- resembles mostly the setup for DeltaNet, though modular counting (cyclic group) can be done in **one** layer instead of **two**

Reference Proofs: Peng, Bo, et al. 2025. “RWKV-7 ‘Goose’ with Expressive Dynamic State Evolution.” arXiv:2503.14456.
Merrill, William, et al. 2026. “Why Are Linear RNNs More Parallelizable?” arXiv:2603.03612.

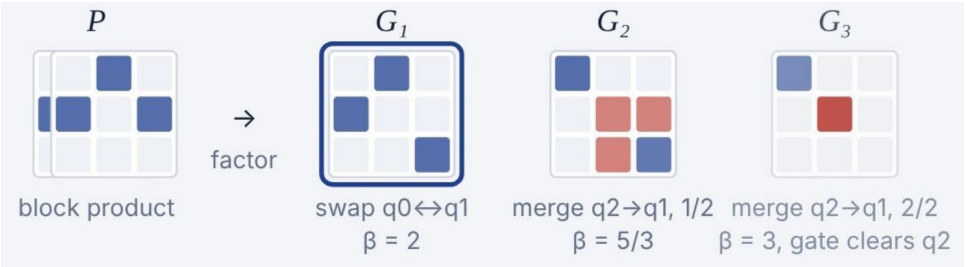
Multilayer Expressivity

Why do we need multiple layers at all?

- general WFA transition is an arbitrary matrix

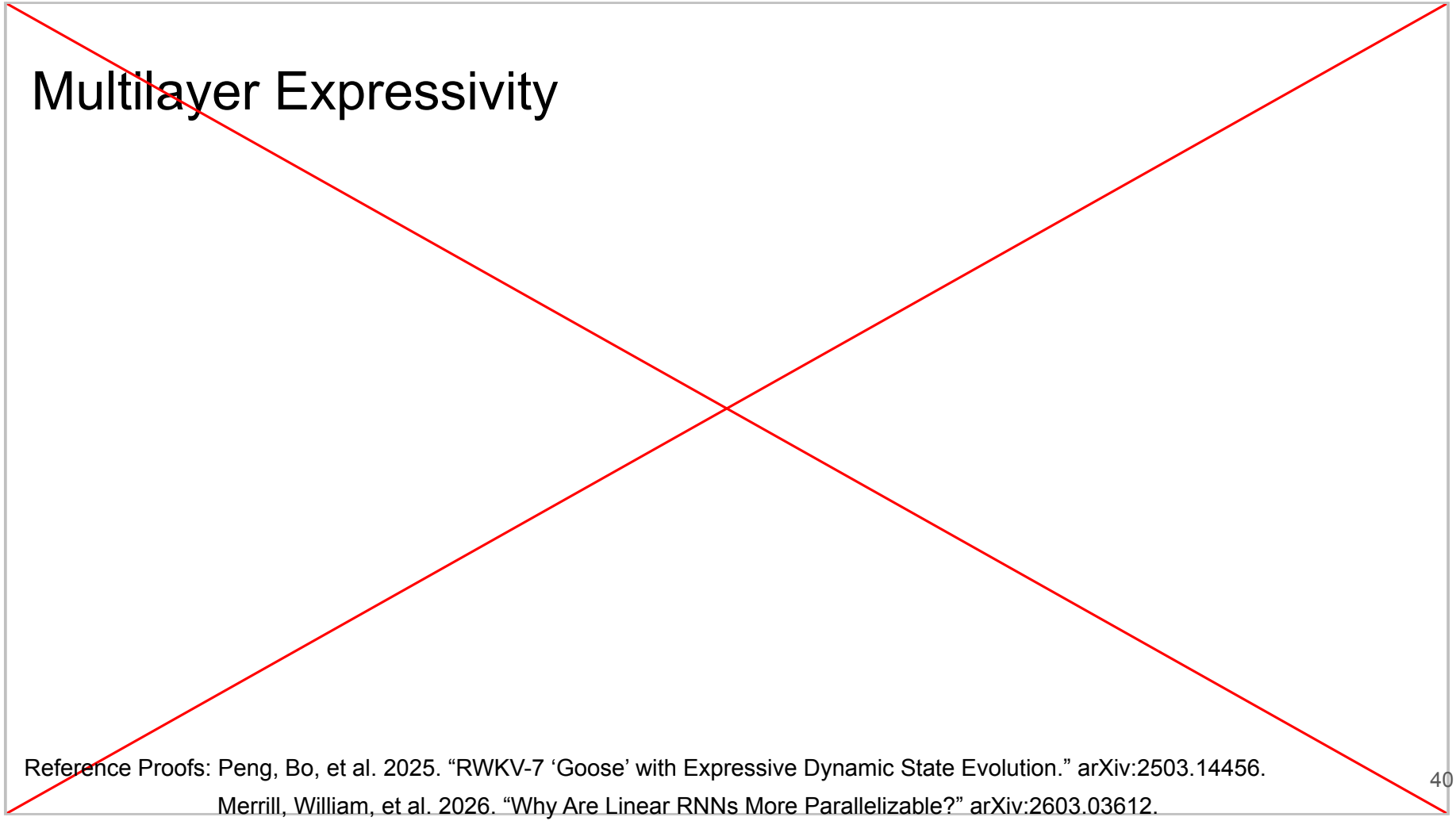


- trick: m transitions chained are also just such a matrix:
  decompose this product into m “CKDA factors” that can be applied per token



Reference Proofs: Peng, Bo, et al. 2025. “RWKV-7 ‘Goose’ with Expressive Dynamic State Evolution.” arXiv:2503.14456.
Merrill, William, et al. 2026. “Why Are Linear RNNs More Parallelizable?” arXiv:2603.03612.

Multilayer Expressivity

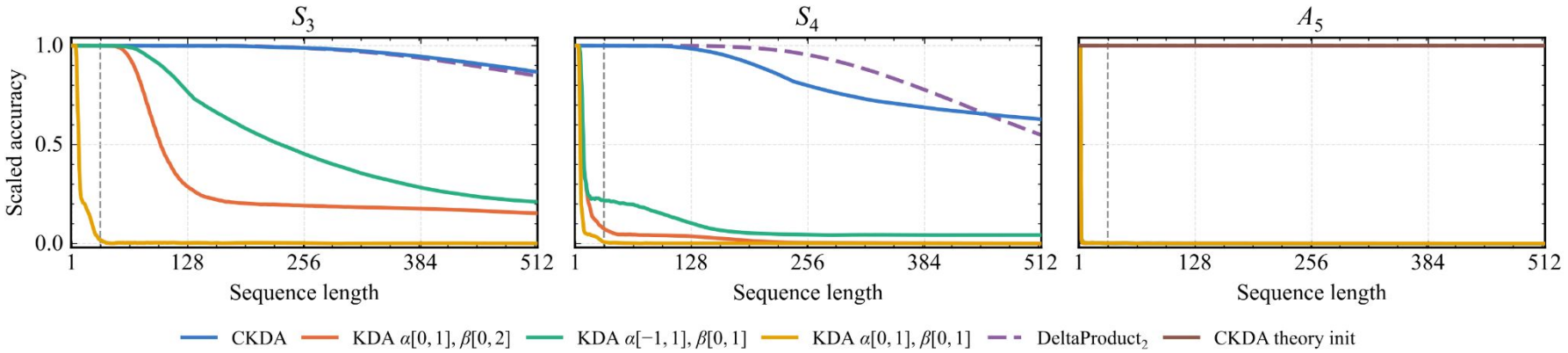


Reference Proofs: Peng, Bo, et al. 2025. “RWKV-7 ‘Goose’ with Expressive Dynamic State Evolution.” arXiv:2503.14456.
Merrill, William, et al. 2026. “Why Are Linear RNNs More Parallelizable?” arXiv:2603.03612.

State Tracking through training

S3, S4: can essentially be solved in one-layer training.

A5: though possible by construction remains difficult to train, initialization from theory works



1) Siems, J. et al 2025. “DeltaProduct: Improving State-Tracking in Linear RNNs via Householder Products.” arXiv:2502.10297.

Language Modeling - 1.3B reference on FineWeb-edu

	ppl ↓		accuracy (%) ↑										recall (%) ↑		
Model	Wiki.	LMB.	LMB.	PIQA	Hella.	Wino.	ARC-e	ARC-c	OBQA	SIQA	BoolQ	Avg.	SQuAD	SWDE	FDA
Recurrent models															
Mamba-2	16.79	12.38	45.24	72.58	55.51	55.33	70.68	35.26	31.00	40.63	60.19	51.82	–	–	–
Gated DeltaNet	16.40	11.89	49.62	72.31	56.50	56.75	68.81	35.15	30.20	40.53	58.78	52.07	–	–	–
KDA	16.81	11.68	48.13	72.09	55.75	55.72	70.83	35.92	30.40	40.99	60.67	52.28	–	–	–
Mamba-3 (SISO)	16.30	12.99	45.06	72.31	55.58	56.20	70.45	34.56	31.00	**41.76**	55.90	51.42	–	–	–
Mamba-3 (MIMO)	16.45	11.66	47.82	72.36	56.49	55.78	72.38	38.07	30.00	40.89	57.74	52.39	–	–	–
Gated DeltaNet-2	15.90	11.41	48.09	72.80	56.84	57.85	72.43	38.23	**31.60**	40.58	59.54	53.11	–	–	–
KDA, bounded gate (ours)	**15.73**	10.53	50.46	72.85	**59.32**	**61.33**	73.23	38.40	28.80	41.35	**61.10**	**54.09**	38.34	**49.59**	30.40
CKDA (ours)	15.78	**10.08**	**51.66**	**73.12**	59.23	58.88	**73.99**	**39.08**	28.60	41.61	60.34	54.06	**39.64**	48.33	**31.76**
Attention or hybrid models															
Transformer (hybrid)	19.22	13.72	48.32	70.21	56.12	55.85	69.23	33.84	25.00	39.74	59.42	50.86	–	–	–
Mamba-2 (hybrid)	17.46	11.29	48.05	71.47	57.52	56.17	70.50	34.73	29.80	40.35	59.31	51.99	–	–	–
Gated DeltaNet (hybrid)	16.00	10.82	48.71	70.06	57.50	56.83	70.41	35.15	30.60	40.97	60.00	52.25	–	–	–
KDA (hybrid)	16.01	10.66	49.21	71.06	56.89	57.77	71.59	35.07	30.00	40.53	62.03	52.68	–	–	–
Mamba-3 (SISO, hybrid)	15.54	10.65	49.19	71.01	58.75	57.30	70.54	36.35	32.00	41.20	57.86	52.69	–	–	–
Mamba-3 (MIMO, hybrid)	15.81	10.92	49.82	71.98	58.19	57.06	70.54	**38.48**	29.40	40.99	57.98	52.72	–	–	–
Gated DeltaNet-2 (hybrid)	15.62	10.43	50.90	72.20	58.46	58.56	71.89	36.69	**33.00**	41.50	62.57	53.97	–	–	–
KDA + attn 3:1 (ours)	**15.04**	9.93	51.60	**73.88**	59.57	**60.62**	**72.69**	37.20	27.00	42.43	60.31	53.92	**48.89**	**75.43**	**79.40**
CKDA + attn 3:1 (ours)	15.34	**9.90**	**52.57**	73.56	**59.68**	60.14	72.56	36.86	28.20	**42.58**	**63.15**	**54.37**	45.21	72.82	75.41

Reference numbers: Hatamizadeh, Ali, et al. 2026. “Gated DeltaNet-2: Decoupling Erase and Write in Linear Attention.” arXiv:2605.22791

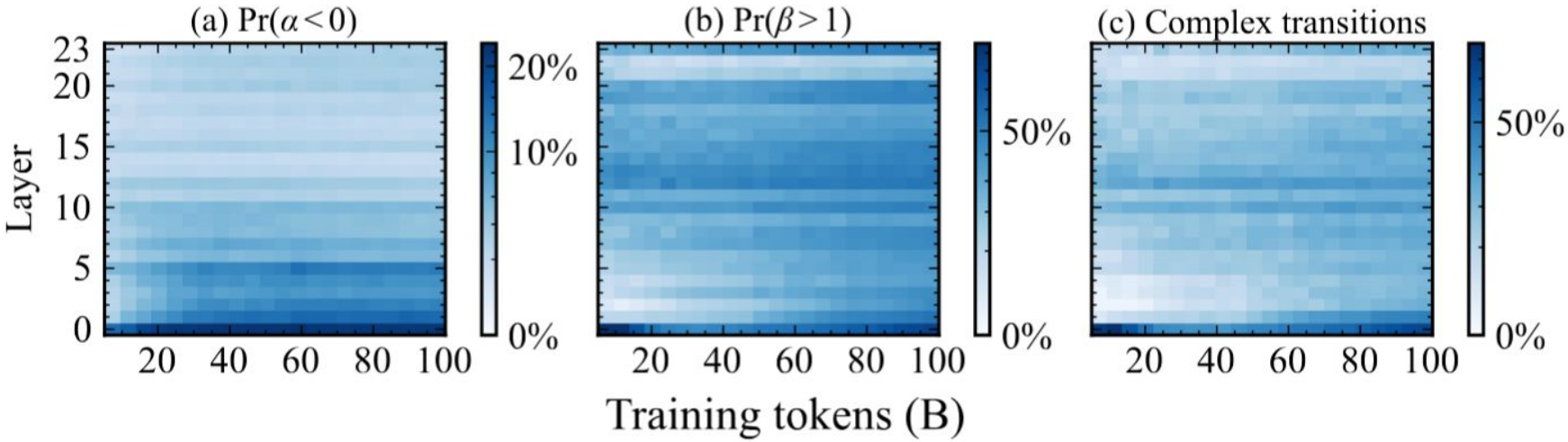
Long Context Retrieval

Model	S-NIAH-1				S-NIAH-2				S-NIAH-3			MK-NIAH-1		
	1K	2K	4K	8K	1K	2K	4K	8K	1K	2K	4K	1K	2K	4K
KDA, bounded gate (ours)	32.2	65.8	94.0	68.8	73.6	100.0	97.6	21.0	7.4	46.6	9.8	48.6	26.2	26.0
CKDA (ours)	100.0	100.0	100.0	87.8	100.0	99.8	88.6	33.2	94.4	88.2	51.0	56.0	46.6	37.0
ours, hybrid - 3:1 full NoPE gated attention														
KDA + attn 3:1 (ours)	100.0	100.0	100.0	100.0	99.8	99.2	99.2	70.2	99.4	98.0	95.8	73.8	60.2	63.8
CKDA + attn 3:1 (ours)	99.8	96.8	81.2	53.4	93.2	98.6	83.2	62.0	97.6	91.0	86.8	95.0	94.4	94.4
Hatamizadeh et al. (2026), recurrent														
Mamba-2	100.0	100.0	97.0	55.8	99.6	99.6	62.6	21.0	59.2	38.6	14.4	29.0	21.2	21.4
Gated DeltaNet	99.8	100.0	100.0	97.6	100.0	100.0	87.2	32.0	89.8	54.2	60.6	58.0	37.0	27.8
KDA	100.0	100.0	99.2	70.6	100.0	100.0	89.0	30.6	77.4	63.2	26.2	54.0	44.2	28.0
Mamba-3 (SISO)	100.0	99.0	63.4	27.8	99.8	99.0	59.4	25.2	60.2	35.6	12.2	44.8	27.4	20.2
Mamba-3 (MIMO)	100.0	99.8	93.0	35.6	99.8	98.8	64.2	27.2	89.2	72.4	29.2	49.4	19.2	18.0
Gated DeltaNet-2	100.0	100.0	100.0	97.8	100.0	100.0	93.0	39.2	92.0	89.8	31.8	72.6	51.4	37.8

Reference numbers: Hatamizadeh, Ali, et al. 2026. “Gated DeltaNet-2: Decoupling Erase and Write in Linear Attention.” arXiv:2605.22791.

Language Modeling - are complex eigenvalues used?

Focused on the first layer, with some in the central layers



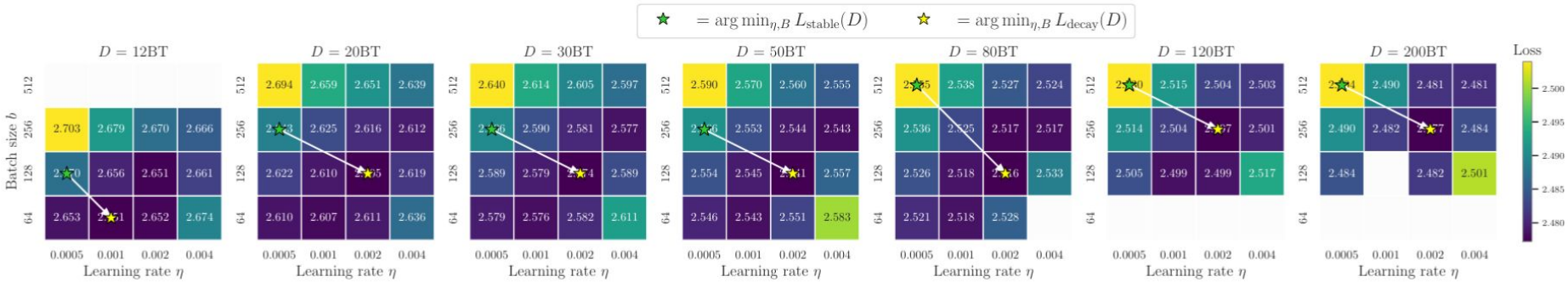
CKDA Models deprived of complex eigenvalues

Post-Restricting α, β activations in pre-trained models:

	ppl ↓		accuracy (%) ↑										recall (%) ↑		
Restriction	Wiki.	LMB.	LMB.	PIQA	Hella.	Wino.	ARC-e	ARC-c	OBQA	SIQA	BoolQ	Avg.	SQuAD	SWDE	FDA
CKDA (recurrent)															
none (as trained)	15.8	10.1	51.7	73.4	59.3	59.1	73.9	39.0	28.4	41.6	60.5	54.1	39.1	48.7	31.7
$\alpha \to \max(0, \alpha)$	+0.1	+1.2	−2.7	+0.1	−0.2	−0.5	−0.2	−0.3	−0.2	−0.2	−1.0	−0.6	−2.4	−4.1	−2.2
β capped at 1	+0.9	+13.9	−16.0	−0.7	−2.5	−2.1	−1.9	−2.5	−0.6	−0.6	+0.5	−2.9	+9.7	+0.5	−5.3
$+ \alpha \to \max(0, \alpha)$	+1.1	+17.0	−18.4	−1.0	−2.6	−2.8	−2.4	−2.4	−0.2	−0.7	+0.4	−3.4	+6.6	−2.6	−8.0
CKDA + attn 3:1															
none (as trained)	15.3	9.9	52.6	73.6	59.7	59.9	72.6	36.9	28.2	42.6	63.1	54.4	44.6	73.0	75.7
$\alpha \to \max(0, \alpha)$	+0.0	+0.9	−2.4	−0.1	−0.1	+0.5	−0.2	−0.4	−0.6	−0.2	−0.3	−0.4	−0.8	−0.4	−0.9
β capped at 1	+0.8	+6.3	−10.6	−0.9	−2.1	−1.4	−2.1	−2.8	−2.2	−2.0	−1.1	−2.8	−0.3	−0.9	−7.1
$+ \alpha \to \max(0, \alpha)$	+1.0	+7.7	−12.4	−0.9	−2.4	−2.4	−2.3	−3.2	−2.0	−1.9	−1.3	−3.2	−1.0	−1.4	−8.7

How does it scale? - reference scaling analysis

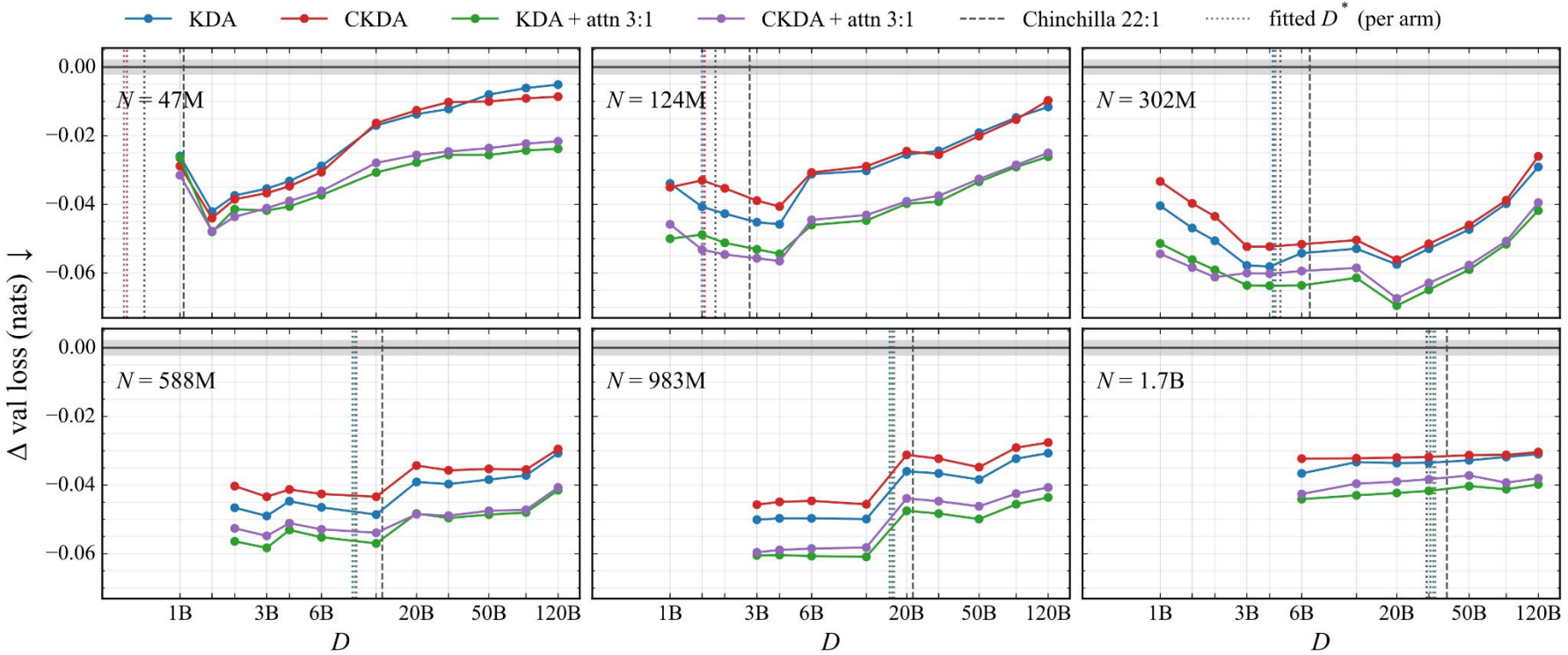
“Deriving Scaling Laws for OpenEuroLLM Models: Learning Rate, Batch Size and Loss.”



(b) $N = 0.13\text{B}$

Ajroldi, Niccolò, et al. 2026. “Deriving Scaling Laws for OpenEuroLLM Models: Learning Rate, Batch Size and Loss.” arXiv:2608.28308

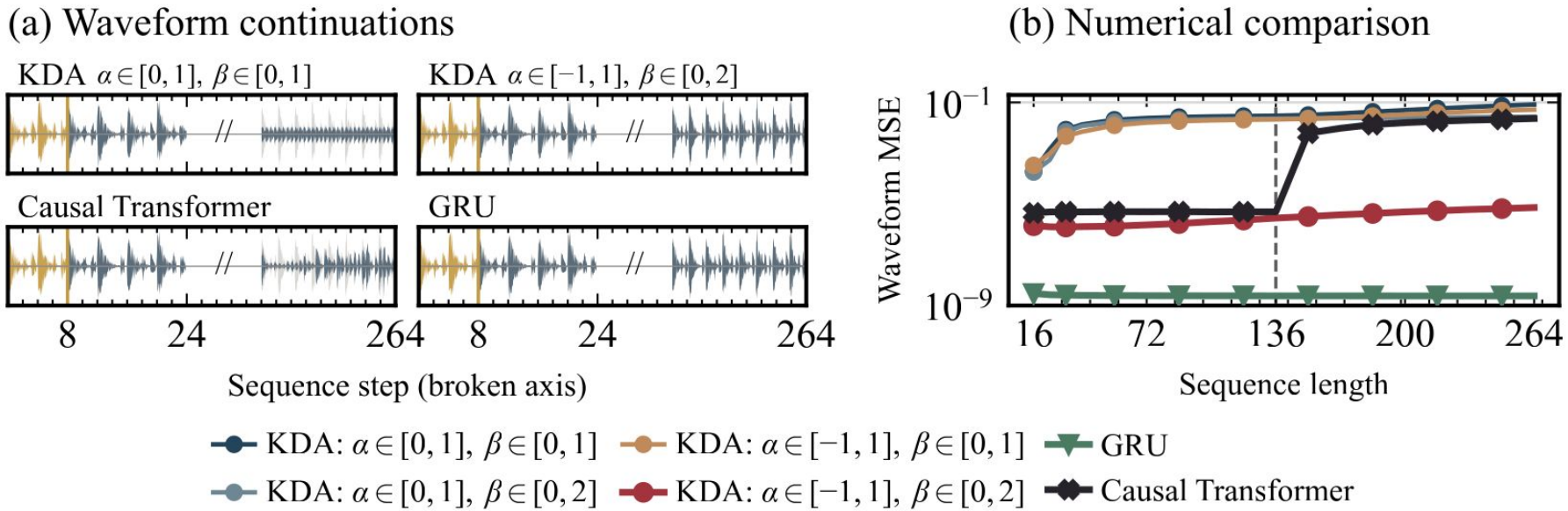
How does it scale? - CKDA and CKDA+attn results



Scaling Reference: Ajroldi, Niccolò, et al. 2026. “Deriving Scaling Laws for OpenEuroLLM Models: Learning Rate, Batch Size and Loss.” arXiv:2608.28308

Audio Modeling

Complex eigenvalues make a difference in extrapolation!



Conclusion

- KDA can be enabled to have complex transitions with up to one complex eigenvalue pair by extending α, β ranges
- CKDA can do state tracking (S3, S4, A5 in one layer) and DFA/MFA recognition in three layers
- Performance in Language Modeling stays on par with KDA (both recurrent and in hybrid) - outperforming a parameter matched Transformer
- Performance benefit holds across scales (in data and model size)
- Strong extrapolation performance for audio modeling