DeltaProduct

Improving State-Tracking in Linear RNNs via Householder Products

Paper by Julien Siems, Timur Carstensen, Arber Zela,
Frank Hutter, Massimiliano Pontil, and Riccardo Grazzi

NeurIPS 2025 arXiv:2502.10297v7
Research discussion prepared on 20 October 2026

More memory updates per token enable richer state transitions.

The transition matrix limits what memory can do

$$H_i = A(x_i)H_{i-1} + B(x_i), \qquad H_i \in \mathbb{R}^{n \times d}$$

Structure of A	Main capability	Main trade-off
Diagonal	Coordinate-wise retention	Limited state mixing
$I - \beta kk^\top$	One directional correction	One factor per token
Householder product	Composed transformations	More internal updates
Dense	General state transformations	Expensive recurrence

State tracking requires composing an ordered sequence of operations:

$$y_i = x_i \cdot x_{i-1} \cdots x_1.$$

Paper, Secs. 1–2, PDF pp. 1–3. The group experiments use permutations.

DeltaNet corrects one key–value association

The matrix state is a linear predictor from keys to values:

$$\mathcal{L}_i(H) = \frac{1}{2} \left\| H^\top k_i - v_i \right\|_2^2 .$$

One gradient step gives

$$H_i = H_{i-1} + \beta_i k_i \underbrace{\left(v_i - H_{i-1}^\top k_i\right)^\top}_{\text{prediction error}} .$$

- $k_i \in \mathbb{R}^n$, $\|k_i\|_2 = 1$; $v_i \in \mathbb{R}^d$; $H_i \in \mathbb{R}^{n \times d}$.
- With $\beta_i = 1$, the updated memory satisfies $H_i^\top k_i = v_i$.
- Directions orthogonal to k_i retain their previous contents.

Paper, Sec. 2, Eq. (2) and the online-learning derivation, PDF p. 3.

DeltaProduct generates several associations per token

For each input x_i, generate n_h distinct triples:

$$k_{i,j} = \frac{\psi(W_j x_i)}{\|\psi(W_j x_i)\|_2}, \quad v_{i,j} = V_j x_i, \quad \beta_{i,j} = \phi(U_j x_i).$$

Then update the same memory sequentially:

$$H_{i,j} = (I - \beta_{i,j} k_{i,j} k_{i,j}^\top) H_{i,j-1} + \beta_{i,j} k_{i,j} v_{i,j}^\top.$$

$$H_{i,0} = H_{i-1} \quad \longrightarrow \quad H_{i,n_h} = H_i.$$

- $\psi = \mathrm{SiLU}$; $\phi = \mathrm{sigmoid}$ or $2\,\mathrm{sigmoid}$.
- n_h counts updates per token, not attention heads.
- Read the final state using $H_i^\top q_i$, then decode.

Paper, Sec. 4, PDF p. 4.

Update order changes both transition and writing

Write $F_j = I - \beta_j k_j k_j^\top$ and $C_j = \beta_j k_j v_j^\top$ for a single token. With two updates,

$$H_i = \underbrace{F_2 F_1}_{A_i} H_{i-1} + \underbrace{F_2 C_1 + C_2}_{B_i}.$$

$$F_2 F_1 = I - \beta_1 k_1 k_1^\top - \beta_2 k_2 k_2^\top + \beta_1 \beta_2 (k_2^\top k_1) k_2 k_1^\top.$$

- The cross term can make the transition non-symmetric.
- Later updates transform earlier writes.
- The structural bound is $\mathrm{rank}(A_i - I) \le n_h$.

Key point

The product expands the changed subspace; A_i itself can already be full rank.

Paper, Eq. (3), PDF pp. 4–5. Product order is written explicitly here.

Two reflections can implement a rotation

For $F = I - \beta k k^\top$:

$$Fk = (1 - \beta)k.$$

- $\beta = 0$: identity.
- $\beta = 1$: projection.
- $\beta = 2$: reflection.

Directions orthogonal to k keep eigenvalue 1.

Teaching example: count modulo 3

$$k_1 = \begin{pmatrix} 1 \\ 0 \end{pmatrix}, \quad k_2 = \begin{pmatrix} 1/2 \\ \sqrt{3}/2 \end{pmatrix}, \quad \beta_1 = \beta_2 = 2.$$

$$F_2 F_1 = \begin{pmatrix} -1/2 & -\sqrt{3}/2 \\ \sqrt{3}/2 & -1/2 \end{pmatrix}.$$

A 120° rotation cycles through three distinct states.

A product may have complex eigenvalues, even though each factor is symmetric.

Paper, Fig. 3, PDF p. 5; Appendix A, pp. 17–18. Numerical example derived for this discussion.

Products preserve a useful norm guarantee

With unit keys and $\beta_{i,j} \in [0, 2]$,

$$\|F_{i,j}\|_2 \le 1 \quad \Rightarrow \quad \|A_i\|_2 \le 1 \quad \Rightarrow \quad \|A_t \cdots A_1\|_2 \le 1.$$

- The homogeneous recurrence cannot amplify old-state perturbations.
- With a gate, $A_i = g_i F_{i,n_h} \cdots F_{i,1}$, $g_i \in [0, 1]$.
- Additive inputs can still accumulate; this is a transition-product guarantee.

Why not add the corrections? For identical keys and $\beta_1 = \beta_2 = 2$:

$$(I - 2kk^\top)^2 = I, \qquad I - 4kk^\top \text{ has eigenvalue } -3.$$

Paper, Sec. 4, PDF p. 5; Appendix B.6, pp. 27–28. Last example is explanatory algebra.

More updates reduce the depth needed in constructions

Theorem 1: sufficient configurations for the word problem of S_n.

Layers	Updates per token	Main construction cost
1	$n_h = n - 1$	Direct permutation factorization
3	Any $n_h > 1$	Large lookup MLP
4	$n_h = 1$	Larger lookup MLP

A coordinate swap is a reflection:

$$k = (e_a - e_b)/\sqrt{2}, \qquad \beta = 2.$$

Any permutation of n objects needs at most $n - 1$ swaps.

Paper, Theorem 1, PDF p. 5; proof in Appendix B.2, pp. 21–22.

A small layer count can hide an enormous lookup

Let $m = \lceil (n-1)/n_h \rceil$. The low-depth construction:

1. Computes position modulo $2m$.
2. Stores the last $2m$ tokens and looks up a factorization.
3. Executes delayed transformations; the decoder completes the prefix.

$$\text{Lookup size} = 2m(n!)^{2m}.$$

For S_5 and $n_h = 2$: $4 \cdot 120^4 = 829{,}440{,}000$ combinations.

Key point

Existence uses task-dependent state capacity and sufficiently expressive parameter maps and decoders.

Paper, Theorem 1 and Appendix B.1–B.2, PDF pp. 19–22. Lookup count calculated from the stated formula.

Gates add reset operations for regular languages

$$H_{i,0} = g_i H_{i-1}, \qquad H_{i,j} = F_{i,j} H_{i,j-1} + C_{i,j}.$$

- Householder products implement permutations.
- $g_i = 0$ removes the old state; writing sets a chosen reset state.
- Krohn–Rhodes decomposition connects permutation-reset automata.

Theorem 2

For any $n_h \geq 1$, any regular language can be recognized with a finite, language-dependent number of Gated DeltaProduct layers.

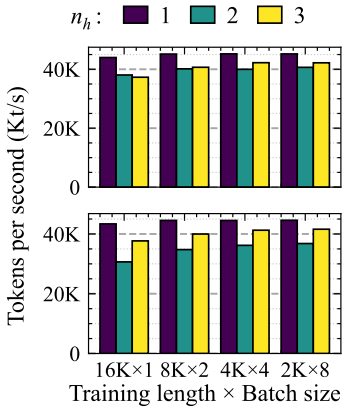
Products of RWKV-7 matrices can also represent copy transitions, at the cost of losing the same universal norm guarantee.

Paper, Theorem 2 and Remark 1, PDF pp. 5–6; Appendix B.3–B.4, pp. 23–24.

Throughput depends on how parameters are matched

- Expand each token into n_h internal key/value updates.
- Use gates $[g_i, 1, \ldots, 1]$ and keep every n_h-th output.
- Reuse a chunkwise DeltaNet-style kernel.
- At fixed head shape, recurrent work scales with n_h; state size stays fixed.

Non-power-of-two head dimensions may be padded by the kernel.



Paper, Sec. 5.1 and Fig. 4, PDF pp. 6–7. 1.3B parameters, H100. Top: head count; bottom: head dimension.

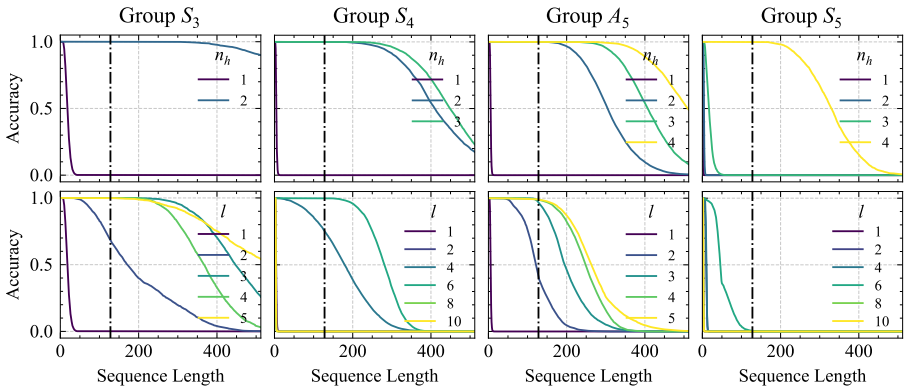
The experiments test four distinct questions

Experiment	Comparison	Outcome measured
Group tracking	More updates vs. more layers	Accuracy by position; train 128, test 512
Formal languages	n_h and coefficient range	Scaled accuracy, length 40–256
Language models	Parameter-matched models	Perplexity and six-task accuracy
Long context	Beyond 4096 training tokens	Token loss; state behavior

Matched parameters do not imply matched FLOPs, state capacity, or wall time.

Paper, Sec. 5 and Appendix C, PDF pp. 7–10 and 28–39.

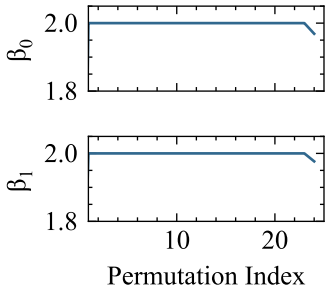
More updates delay failure on group tracking

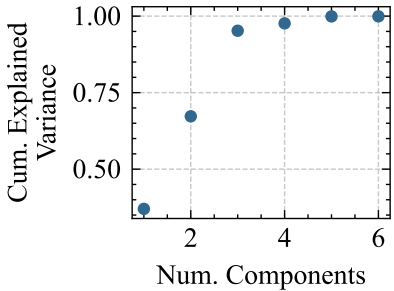


Top: one layer, varying n_h. Bottom: $n_h = 1$, varying depth. Even S_5 with $n_h = 4$ degrades strongly near length 512.

Paper, Fig. 5, PDF p. 7. Dashed vertical line: training length 128.

A learned head resembles a 3D rotation mechanism





- S_4 is the rotation group of the cube.
- One head learns both β values close to 2.
- Three principal components explain over 95% of key variance.

This supports the two-reflection explanation for the selected head.

Paper, Fig. 7, PDF p. 8; Fig. 13, p. 30, identifies head 6 among 12 heads.

Language-model comparisons use three budget levels

Parameters	FineWeb training tokens	Context length
213M	19B	4096
392M	35B	4096
805M	55B	4096

- Table 3 matches parameters through head dimension.
- Table 4 uses a separate set matched through head count.
- Average accuracy: LAMBADA, PIQA, HellaSwag, Winogrande, ARC-e, ARC-c.
- HellaSwag and ARC-c use normalized accuracy; PPL is excluded from the average.

Paper, Tables 3–4, PDF p. 32; training setup in Appendix C.3.1, p. 31.

At 805M, three updates improve language modeling

55B FineWeb tokens, 4096 context, head-dimension matching.

Model	Wiki PPL	LMB PPL	LMB acc.	Avg. acc.
DeltaNet	20.81	20.57	37.8	48.55
DeltaProduct$_2$	20.54	19.56	38.3	49.00
DeltaProduct$_3$	**20.01**	**15.56**	**42.9**	**49.95**

- LAMBADA PPL: 24.36% lower relative to DeltaNet.
- LAMBADA accuracy: +5.1 percentage points.
- Six-task mean: +1.40 percentage points.

Paper, Table 3, PDF p. 32. All models use the extended coefficient range. PPL lower; accuracy higher.

The best update count changes with model scale

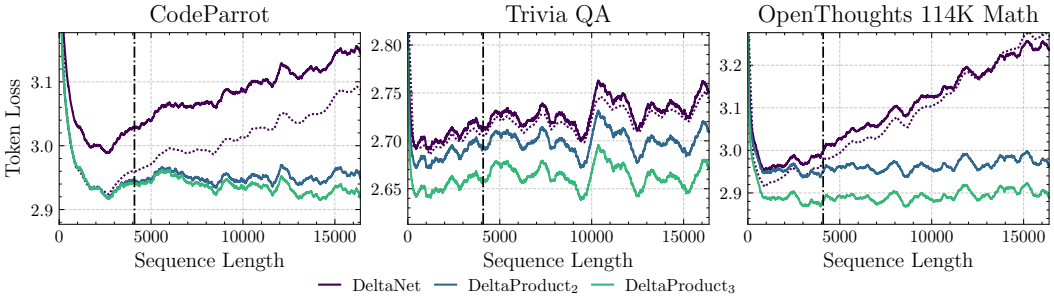
Six-task average accuracy, head-dimension matching.

Parameters / tokens	DeltaNet	DeltaProduct_2	DeltaProduct_3
213M / 19B	40.23	40.88	**41.01**
392M / 35B	43.70	**45.08**	44.10
805M / 55B	48.55	49.00	**49.95**

- At 392M, two updates outperform three.
- Both parameter-matching strategies show this middle-scale pattern.
- Individual tasks can regress even when the average improves.

Paper, Tables 3–4, PDF p. 32. Example: 805M Table 3 Winogrande drops from 55.6 to 53.0 for DP_3.

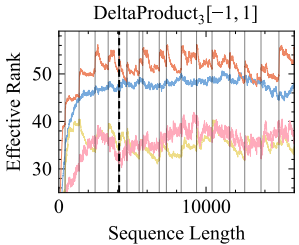
Extra updates suppress loss growth beyond training length

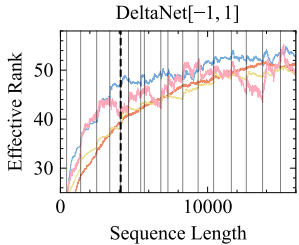


- Train at 4096 tokens; evaluate position-wise loss out to about 16K.
- 8-head DP$_2$ and 12-head DeltaNet both have 392M parameters.
- These are token-prediction losses on the named corpora.

Paper, Fig. 8, PDF p. 9. 501-token moving average. Appendix Figs. 16–18 extend to 32,768 tokens.

Effective rank suggests different memory dynamics





$$\operatorname{erank}(H) = \exp\left(-\sum_k p_k \log p_k\right), \qquad p_k = \frac{\sigma_k}{\sum_j \sigma_j}.$$

DeltaNet's effective rank keeps rising beyond the training context. The authors interpret the comparison as evidence for improved forgetting.

Paper, selected panels from Fig. 9, PDF p. 9. Layer 20/24, four heads. Effective rank is scale invariant, not a direct information measure.

Formal-language gains are task dependent

Extended range; scaled accuracy; best of three runs.

Model	Parity	No brackets	Brackets	Average
DeltaNet	.982	**.915**	.281	.726
DeltaProduct_2	.896	.887	.329	.704
DeltaProduct_3	.932	.736	.330	.666
DeltaProduct_4	**.982**	.893	**.342**	**.739**

- Bracketed modular arithmetic improves with every tested $n_h > 1$.
- Only DP_4 improves the average over the extended-range DeltaNet.
- Training lengths: 3–40; test lengths: 40–256.

Paper, Table 2, PDF p. 31; Appendix C.2, pp. 29–31. The table is narrower than the accompanying all-n_h claim.

Treat update count as a structural capacity parameter

1. **Mechanism:** distinct sequential corrections create richer, potentially rotational transitions.
2. **Theory:** norm-controlled factors support strong finite-state constructions; small depth may require huge width.
3. **Evidence:** better state tracking and language-model extrapolation, with task-dependent accuracy gains.
4. **Cost:** more projections and internal recurrence steps; implementation and parameter allocation matter.

Paper, Secs. 4–6 and Appendix C. Suggested main discussion length: 20–30 minutes.

Paper and discussion references

Julien Siems, Timur Carstensen, Arber Zela, Frank Hutter, Massimiliano Pontil, and Riccardo Grazzi.

DeltaProduct: Improving State-Tracking in Linear RNNs via Householder Products.

NeurIPS 2025. arXiv version 7, revised 22 October 2025.

arxiv.org/abs/2502.10297v7
github.com/automl/DeltaProduct

Figures are reproduced from the paper/source package with figure and PDF-page locators. Algebraic examples and qualifications are reading analysis.

The following slides provide supporting details and corrections.

Backup: head-count matching gives a similar pattern

Parameters / tokens	DeltaNet	DP_2	DP_3
213M / 19B	40.17	40.35	**41.02**
392M / 35B	43.83	**44.98**	44.15
805M / 55B	48.22	49.13	**49.40**

At 392M, Table 6 uses 12, 8, and 6 heads for $n_h = 1, 2, 3$.

The separate 2048-context comparison in Table 5:

- Matched 392M: DeltaNet 44.3 vs. DP_2 45.5.
- Gated: 340M DeltaNet 45.9 vs. 393M DP_2 46.4.
- The 15B SlimPajama rows use a different data budget.

Paper, Tables 4–5, PDF p. 32; Table 6, p. 36. Numbers above are six-task mean accuracy.

Backup: complex eigenvalues require an angle condition

For two factors, let $c = k_1^\top k_2$. On their two-dimensional span:

$$\tau = 2 - \beta_1 - \beta_2 + \beta_1\beta_2 c^2, \qquad \delta = (1-\beta_1)(1-\beta_2).$$

$$\lambda_\pm = \frac{\tau \pm \sqrt{\tau^2 - 4\delta}}{2}.$$

- Complex eigenvalues occur exactly when $\tau^2 < 4\delta$.
- $\beta_1 > 1$ and $\beta_2 > 1$ are necessary, not sufficient alone.
- Equal keys collapse to one effective factor.
- Orthogonal keys give commuting factors and a symmetric matrix.

Paper, Proposition 1 and proof, PDF pp. 17–18. Retain the discriminant condition when reading the final "iff" sentence.

Backup: spectral radius alone does not ensure stability

Appendix B.6 states an overly strong equivalence. A counterexample is

$$J = \begin{pmatrix} 1 & 1 \\ 0 & 1 \end{pmatrix}, \qquad J^t = \begin{pmatrix} 1 & t \\ 0 & 1 \end{pmatrix}.$$

Every $\rho(J^t) = 1$, but $\|J^t\|_2$ grows without bound.

DeltaProduct's valid argument uses the stronger bound

$$\|A_i\|_2 \leq 1 \quad \Rightarrow \quad \|A_t \cdots A_1\|_2 \leq 1.$$

The RWKV-7 example in the paper has $\rho(AA') = (27 + \sqrt{153})/32 \approx 1.2303$. It establishes an allowed unstable configuration.

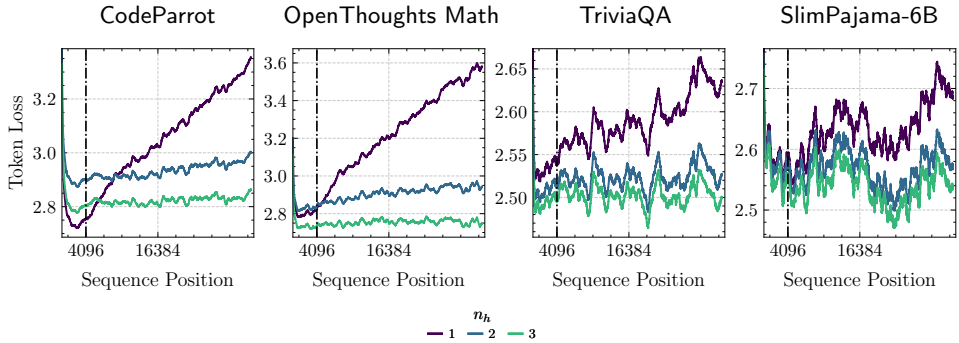
Paper, Appendix B.6 and Theorem 8, PDF pp. 27–28. Jordan example is a correction; the printed RWKV matrices swap the initially named keys.

Backup: geometric and measurement qualifications

- **Dodecahedron:** its rotational symmetry group is A_5; its full symmetry group is $A_5 \times C_2$, not S_5.
- **Why distinguish them?** Central inversion commutes with all symmetries, whereas S_5 has a trivial center.
- **Effective rank:** for nonzero H, $1 \leq \mathrm{erank}(H)$. Low rank does not by itself mean a zero or empty memory.
- **Factor-range labels:** $[-1, 1]$ labels the individual factors; the full product can have complex eigenvalues in the unit disk.

Corrections to Appendix C.1, PDF p. 29, and Secs. 4–5.3, pp. 5–9; supported by the stated algebra and definitions.

Backup: the 805M matched comparison extends to 32K



- Same 805M scale and 55B training-token budget; head-count matching.
- Code and math show pronounced divergence of the one-update model.
- These are smoothed token-loss curves, not answer-success rates.

Paper, Fig. 17 (805M row), PDF p. 35; Table 6, p. 36. Training context: 4096; maximum test context: 32,768.