# What Makes Recurrence Effective in Looped Language Models?

## Architecture, state history, and inference budgets

Paper authors: Xinlin Zhuang, Siyuan Wang,
Imran Razzak, and Weiyang Liu

Independent research-paper discussion

arXiv:2609.36636v1 | 15 October 2026

Extra loops help when architecture and conditioning sustain useful computation.

# A fixed training horizon leaves two deployment gaps

$$\underbrace{P}_{\text{input}} \longrightarrow \underbrace{F \circ \cdots \circ F}_{r \text{ shared-core executions}} \longrightarrow \underbrace{C}_{\text{output}}$$

| Inference regime | Loop count | Question |
|---|---|---|
| Under-unrolling | $r < K$ | Can an early state be read out? |
| Training horizon | $r = K$ | Does the trained path work? |
| Loop extrapolation | $r > K$ | Is extra computation still useful? |

## The empirical gap

Training supervises the final output at a fixed $K$. It does not directly optimize either shorter or longer execution.

Paper, Secs. 1–2, PDF pp. 1–4. Diagram is a teaching schematic.

# Physical depth and executed depth must be separated

$$L_{\text{phys}} = p + s + c, \qquad D = L(r) = p + rs + c.$$

- $p$: Prelude layers; $s$: shared-core layers; $c$: Coda layers.
- $K$: training loops; $r$: inference loops.
- We consistently write configurations as $p + s \times K + c$.

| Configuration | Physical depth | Training depth |
|---|---|---|
| BaseLoop $4 \times 5$ | 4 | 20 |
| CoreLoop $0 + 2 \times 9 + 2$ | 4 | 20 |
| NonLoop $20 \times 1$ | 20 | 20 |

Paper, Sec. 2.1 and Table 2, PDF pp. 2–3, 18. Sec. 3 inconsistently swaps $s$ and $K$.

# The study trains small models from scratch

| | Llama-style family | Qwen-style family |
|---|---|---|
| Non-recurrent reference | 20 layers, 1.007B | 28 layers, 0.596B |
| Hidden width $d$ | 1536 | 1024 |
| Effective training depth | 20 | 28 |
| Training tokens | 20.15B | 12.00B |
| Representative BaseLoop | $4 \times 5$, 516.70M | $4 \times 7$, 218.51M |

- FineWeb-Edu; packed 2048-token sequences; next-token loss.
- Fixed $K$, full recurrent backpropagation, final-output supervision.
- Muon + AdamW parameter groups; one random seed (42).

Paper, App. C.1–C.2, Tables 1–3, PDF pp. 17–21. Family names identify architectures, not pretrained weights.

# Separate capability groups before reading the average

| Knowledge: 3 tasks | Reasoning: 8 tasks |
| --- | --- |
| SciQ, ARC-Easy, PIQA | ARC-Challenge, WinoGrande, OpenBookQA, HellaSwag, CommonsenseQA, ProofWriter, CLUTRR, BBH |

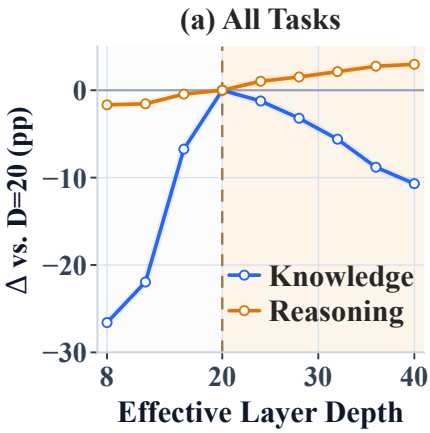
Operational grouping based on the authors' view of computational demand.

$$S_{\text{Overall}} = \frac{3S_{\text{Know}} + 8S_{\text{Reas}}}{11}$$

- Zero-shot accuracy or length-normalized accuracy; higher is better.
- BBH uses 14 selected subtasks, excluding near-chance tasks.

Paper, App. C.3, PDF pp. 21–22. Overall weighting is inferred from and agrees with the result tables.

# Extra loops improve reasoning while knowledge declines




Llama BaseLoop $4 \times 5$

| Score (%) | $D = 20$ | $D = 40$ |
|---|---|---|
| Knowledge | 62.80 | 52.11 |
| Reasoning | 28.52 | 31.49 |
| Overall | 37.87 | 37.11 |

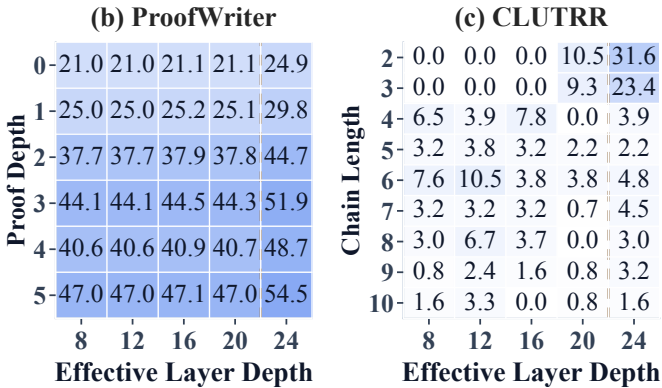
**The trade-off**

Reasoning: $+2.97\,\mathrm{pp}$
Knowledge: $-10.69\,\mathrm{pp}$

No test-time parameter updates.

Paper, Fig. 1(a), PDF p. 3; Table 8, p. 29. The Sec. 3 prose mislabels this case as $2 \times 10$.

# Harder instances do not consistently gain more



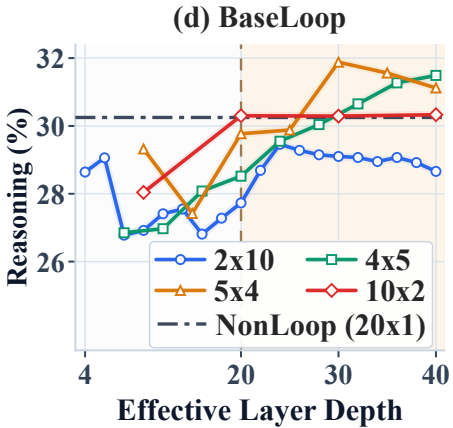

Llama $2 \times 10$

$D = 20 \rightarrow 24$:

- ProofWriter depths 3–5 gain 7.5–8.0 points.
- CLUTRR gains concentrate on short chains.
- Long-chain accuracy remains low and uneven.

Paper, Fig. 1(b,c), PDF p. 3; Sec. 3, p. 4. Cells show absolute accuracy (%).

# Equal training depth produces different scaling curves




- Shallow core: $2 \times 10$ peaks early.
- Deep core: $10 \times 2$ gains little from more loops.
- $5 \times 4$: 31.88 reasoning at $D = 30$.
- NonLoop: 30.25 at its native $D = 20$.

**Budget interpretation**

Training depth is matched. The higher recurrent score uses 50% more inference layer executions.

Paper, Fig. 1(d) and Sec. 3, PDF pp. 3–5. Physical depth and parameter count vary across these curves.

# Boundary layers have different roles at different budgets

| Regime | Observation in the Llama comparisons |
|---|---|
| Fewer loops | Coda often protects knowledge; e.g., $0 + 2 \times 9 + 2$. |
| Training depth | Differences among boundary allocations become smaller. |
| More loops | Prelude-heavy layouts can better sustain reasoning. |

## An informative exception

On Qwen at $D = 56$, BaseLoop has higher Overall and Reasoning scores than all three tested CoreLoop layouts.

Paper, Fig. 2 and Sec. 4, PDF pp. 5–6; App. D.2 and Fig. 9, pp. 26–27.

# Small updates can coexist with continued state drift

## Local movement

Angular distance between loops decreases. Relative update norms also decrease:

$$\frac{\|h^{t+1} - h^t\|_2}{\|h^t\|_2}.$$

## Accumulated change

Normalized state variance continues growing:

$$\widetilde{V}(t) = \frac{V(h^t)}{V(h^K)}.$$

$V$ measures within-token dispersion across hidden channels, then averages over tokens.

## Design implication

Update size alone is insufficient to decide whether another loop is useful.

Paper, Fig. 3 and Sec. 4, PDF pp. 5–6; App. C.4, Eqs. (18)–(21), p. 23. Norm formula shown schematically.

# Repeated static input injection can harm extrapolation

$$h_{\ell+1} = F_\theta(\mathcal{I}(h_\ell, h_0))$$

| Injection | Map | Parameters |
|---|---|---|
| Scalar | $h + \alpha h_0$ | 1 |
| Channel-wise | $a \odot h + b \odot h_0$ | $2d$ |
| Residual channel | $(\mathbf{1} + \delta_a) \odot h + b \odot h_0$ | $2d$ |
| Dense | $W_h h + W_0 h_0$ | $2d^2$ |

**Qwen $4 \times 7$, Dense injection**

Knowledge falls from 58.77 at $D = 28$ to 32.28 at $D = 84$.
BaseLoop retains 52.29 at $D = 84$.

Paper, Eq. (2), PDF p. 7; Table 10, p. 30. Every injection starts as the identity on the current state.

# History injection conditions on recent state differences

$$h_{\ell+1} = F_\theta\left(h_\ell + \sum_{j=1}^{m_\ell} \mathcal{B}_j(h_{\ell-j} - h_\ell)\right), \quad m_\ell = \min\{w, \max(\ell - 1, 0)\}.$$

- $w$: history window; $j$: relative lag; $\mathcal{B}_j$: learned lag-specific map.
- Exclude $h_0$: only completed recurrent states enter the history.
- First history-dependent update:

$$h_3 = F_\theta(h_2 + \mathcal{B}_1(h_1 - h_2)).$$

- Zero-initialize history maps; retain gradients through buffered states.

**Teaching intuition, scalar $w = 1$**

$(1 - \beta)h_\ell + \beta h_{\ell-1}$ can damp or extrapolate the recent trajectory; $\beta$ is unconstrained.

Paper, Eq. (3), PDF p. 7; App. C.1, Eqs. (4)–(5), pp. 18–19.

# The best history window depends on parameterization

Qwen $4 \times 7$, $D = 84$; all scores in percent.

| Method | Overall | Knowledge | Reasoning |
|---|---|---|---|
| BaseLoop | 35.60 | 52.29 | 29.34 |
| Dense initial-state | 29.88 | 32.28 | 28.98 |
| Dense history, $w = 1$ | 37.15 | 54.16 | **30.77** |
| Dense history, $w = 4$ | 30.38 | 32.73 | 29.50 |
| Scalar history, $w = 4$ | **37.24** | **59.18** | 29.01 |
| Channel history, $w = 2$ | 36.14 | 53.86 | 29.49 |

- One-step Dense history rescues static-injection collapse.
- Four-scalar history is a strong knowledge-preserving baseline.

Paper, Table 10, PDF p. 30. Selected rows; bold identifies the best score among the displayed rows.

# Loop gating adds a budget-aware update scale

$$t_\ell = \ell/T, \quad \Delta t = 1/T, \quad \psi_\ell \in \mathbb{R}^8, \qquad g_\ell = 1 + q^\top \psi_\ell$$

$$\boxed{h_{\ell+1} = h_\ell + g_\ell(F_\theta(h_\ell) - h_\ell)}$$

- Train with $T = K$; rescale to $T = r$ at inference.
- Time features include $t$, $\Delta t$, polynomial and sine terms.
- Only 8 learned parameters; initialize $q = 0$ to recover BaseLoop.

## What the formula does

The step size is a conditioning feature. There is no additional explicit $\Delta t$ multiplier in the update.

Paper, App. C.1, Eqs. (7)–(8), PDF pp. 19–20. Gates are unconstrained.

# Finer time modulation trades capacity for control

$$u^{+} = u + g^{b}_{\ell,k} \odot \mathcal{B}^{b}_{k}\Big((\mathbf{1} + s^{b}_{\ell,k}) \odot N^{b}_{k}(u)\Big)$$

| Method | Modulation granularity | Parameters |
|---|---|---|
| LG | One scalar for the entire core update | 8 |
| BG | One scalar per attention/MLP branch | $16s$ |
| AdaLN | Channel gates and norm-output scales | $32sd$ |

- $k$ indexes a core layer; $b$ indexes its attention or MLP branch.
- BG sets the additional normalization scale to zero.
- All start from the unconditioned computation.

Paper, App. C.1, Eqs. (9)–(11), PDF pp. 19–20. Here $s$ consistently denotes shared-core layer count.

# Time conditioning helps some architectures and capabilities

Qwen $7 \times 4$, $D = 56$; all scores in percent.

| Method | Overall | Knowledge | Reasoning |
|---|---|---|---|
| BaseLoop | 36.59 | 60.33 | 27.68 |
| LG | 36.91 | 59.24 | 28.54 |
| BG | **38.46** | 58.67 | **30.88** |
| AdaLN | 38.01 | **61.72** | 29.12 |

- BG gains 1.87 pp Overall, driven by reasoning.
- AdaLN better preserves knowledge in this comparison.
- Qwen $2 \times 14$: none improves Overall at $D = 28$ or 56.

Paper, Table 12, PDF p. 31; Fig. 6, p. 8. The effect depends on core depth and loop count.

# History plus loop gating is the strongest matched pair

I: initial state; H: history; T: time conditioning.

Qwen $4 \times 7$, $D = 84$

| Method | Overall (%) |
|---|---|
| BaseLoop | 35.60 |
| H only | 36.14 |
| I+LG | 36.33 |
| I+H | 36.91 |
| **H+LG** | **37.30** |

**Matched setting**

Channel-wise states, $w = 2$.

**Small adapter**

$wd + 8 = 2056$ parameters.

$+1.16\,\mathrm{pp}$ over H alone;
$+1.70\,\mathrm{pp}$ over BaseLoop.

Knowledge: 56.32 vs. 52.29 for BaseLoop.

Paper, Sec. 6 and Fig. 7, PDF p. 9; Table 10 and App. D.6, pp. 30–31. Parameter total is calculated from the stated maps.

# Adding another conditioning signal need not help

- **I+H can help at a particular budget.** At $D = 56$, channel-wise $w = 2$ reaches 37.74, vs. 36.67 (I) and 36.77 (H).
- **I+T has little complementarity.** At training depth, I+LG reaches 35.99, below LG alone at 36.68.
- **The triple is weaker at $D = 56$.** Four tested I+H+T variants score 35.69–36.60; H+LG reaches 37.49.

## Keep a strong lightweight comparator

Scalar H with $w = 4$ alone reaches 37.24 at $D = 84$, only 0.06 points below the highlighted H+LG result. The study uses one seed.

Paper, App. D.6 and Table 10, PDF pp. 29–31. The triple variants and highlighted pair have their stated distinct parameterizations.

# A skip intervention measures computational dependence

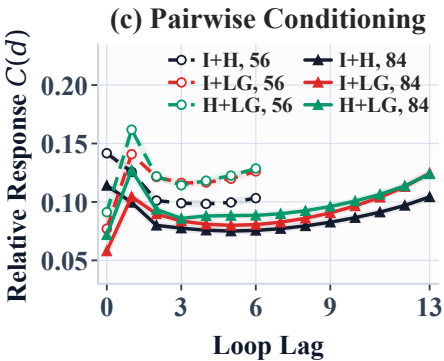
Skip one block occurrence $i$ by replacing its output with its input. Measure how a later block's *update* changes:

$$u_{j,p} = h^{\text{out}}_{j,p} - h^{\text{in}}_{j,p}, \qquad C_{i\to j} = \frac{\mathbb{E}_p \| u^{(-i)}_{j,p} - u_{j,p} \|_2}{\mathbb{E}_p \| u_{j,p} \|_2}.$$

- Only one execution is skipped; other uses of the shared weights remain.
- $C = 0.1$: average update change equals 10% of the baseline update norm.
- Held-out validation text: 49 sequences, 100,000 token positions.
- Aggregate pairs beyond the training horizon by loop lag.

Paper, App. C.4, Eqs. (25)–(32), PDF pp. 22–25. $C$ is a ratio of mean norms.

# H+LG shows stronger interactions across loops




At $D = 84$, weighted mean over lags 1–6:

- H+LG vs. I+H: +18.6% in $C$.
- H+LG vs. I+LG: +10.4% in $C$.
- Raw responses also rise: +6.4% and +10.2%.

Consistent with later loops building on earlier computation; the link to task accuracy remains interpretive.

Paper, Fig. 8(c) and Sec. 6, PDF p. 9; Fig. 23 and App. E.2, p. 36. Dashed: $D = 56$; solid: $D = 84$.

# Design for the intended inference budget

1. Track Knowledge and Reasoning separately as loops increase.
2. Tune physical depth and boundary allocation jointly with loop count.
3. Compare channel-wise H+LG against strong scalar-history baselines.

## Scope of the evidence

Reference models up to about 1B; fixed-budget pretraining; one seed. History combinations mainly use Qwen $4 \times 7$, tested up to $3\times$ training depth. No end-to-end latency measurements or adaptive halting.

Xinlin Zhuang, Siyuan Wang, Imran Razzak, Weiyang Liu.
*What Makes Recurrence Effective in Looped Language Models?*
arXiv:2609.36636v1, 29 September 2026.
Paper, App. A, C.2, and D.4–D.6, PDF pp. 15, 21, 27–35.

# Backup: exact time features and adapter sizes

$$\psi(t, \Delta t) = \begin{pmatrix} t,\ \Delta t,\ t^2,\ (\Delta t)^2,\ t\Delta t, \\ \sin(\pi t),\ \cos(\pi t) - 1,\ \sin(2\pi t) \end{pmatrix}^\top$$

| Qwen $4 \times 7$: $d = 1024$, $s = 4$, $w = 2$ | Added parameters |
|---|---|
| Channel history | 2,048 |
| Dense history | 2,097,152 |
| LG / BG / AdaLN | 8 / 64 / 131,072 |
| Channel history + LG | 2,056 |

Direct history buffering requires $O(wnd)$ state elements. Parameter efficiency alone does not determine serving cost.

Paper, App. C.1, PDF pp. 18–20. Counts are calculated from the stated formulas; buffer cost is an implementation inference.

# Backup: the same function class can train differently

## Channel-wise

$$a \odot h + b \odot h_0$$

Initialize $a = \mathbf{1}$, $b = 0$.

Weight decay on $a$ favors a current-state coefficient of zero.

## Residual channel-wise

$$(\mathbf{1} + \delta_a) \odot h + b \odot h_0$$

Initialize $\delta_a = b = 0$.

Weight decay on $\delta_a$ favors a current-state coefficient of one.

Both add $2d$ parameters and can represent the same maps. Their regularization is different.

Paper, App. C.1, PDF p. 18.

# Backup: the numerical comparisons need three corrections

1. **Configuration convention.** Sec. 3 swaps the meanings of $s$ and $K$. Use Table 2's core-depth $\times$ loop-count convention.
2. **Task divergence example.** The $28.52 \rightarrow 31.49$ reasoning result belongs to $4 \times 5$, as shown by Fig. 1(a) and Table 8, not the $2 \times 10$ label in the prose.
3. **Qwen $14 \times 2$, $D = 56$.** Sec. 5.2 reports LG/AdaLN reasoning as 29.61/28.78. Table 12 reports 29.22/29.36; Fig. 6(d) supports the table's ordering.

These slides use the consistent figure/table evidence for those comparisons.

Paper, Sec. 3, PDF p. 4; Sec. 5.2, p. 8; Tables 2, 8, 12, pp. 18, 29, 31.

# Backup: optimizer and noise ablations support the recipe

| Pilot model | AdamW mean | Muon mean |
|---|---|---|
| NonLoop $12 \times 1$ | 41.95 | 42.04 |
| BaseLoop $3 \times 4$ | 39.98 | 40.74 |
| LoopFormer $3 \times 4$ | 39.96 | 40.05 |
| Huginn $1 + 2 \times 5 + 1$ | Failed to converge | 41.72 |

- The pilot averages ten tasks, distinct from the main eleven-task metric.
- Initialization noise 0.03 gives no consistent benefit over no noise for the tested initial-state injections.

Paper, Table 7 and App. D.1, PDF p. 26; Fig. 22 and App. E.1, pp. 33, 35–36. Selected pilot rows.

# Backup: how the interaction statistic is aggregated

- For Qwen $4 \times 7$, include only source and target occurrences beyond training loop 7.
- For loop lag $d$, average $C_{i \to j}$ over occurrence pairs $\mathcal{P}_d$.
- Cross-lag summaries weight pairs equally:

$$C_{\mathcal{D}} = \frac{\sum_{d \in \mathcal{D}} |\mathcal{P}_d| \, C(d)}{\sum_{d \in \mathcal{D}} |\mathcal{P}_d|}.$$

- At $D = 84$: 21 loops and lags 0–13. Main pair comparison uses lags 1–6.
- Adding I to H+LG lowers mean $C$ over lags 2–6 by 4.2%, and the raw numerator by 6.0%.
- Only 9/16 shared-block pairings show lower $C$ in that triple comparison.

Paper, App. C.4, Eqs. (30)–(33), PDF pp. 24–25; App. E.2 and Fig. 23, p. 36.