# DELAYED POSITIONAL ADJUSTMENT FOR CONTEXT EXTENSION

**Anonymous authors**
Paper under double-blind review

## ABSTRACT

Extending a pretrained language model to longer inputs changes two things at once: the training sequences grow longer, and the rotary position embedding (RoPE) frequencies are rescaled to a target such as position interpolation (PI) or YaRN, usually from the first update. We ask when to introduce the target frequencies. We introduce delayed positional adjustment (DEPA), which trains at the new length with the original frequencies, switches once to the target after 90% of the continuation updates, and adapts for the remaining 10%. We compare DEPA with static adjustment, which applies the target from the first update, using pairs matched on the pretrained model, final frequencies, token budget, and optimizer schedule. On 137M-parameter models extended from 1K to 4K tokens on PG19, DEPA lowers perplexity (PPL) beyond the trained window in every tested pair. Under our custom YaRN-family target (a YaRN variant with our own frequency mask), whose static runs already extrapolate well, five constant-learning-rate pairs show a reduction of about 1.9 in 16K-token PPL at a 0.03-PPL cost at 4K. Under PI and published YaRN, DEPA lowers 16K PPL from about 105–190 to below 32. The gain persists on a second pretrained base, on WikiText-103, for a 20-layer model, and at 19M parameters. When fixed target tokens are scored with progressively longer inputs, DEPA reduces added-context harm by 78–80% on PG19. Release timing is a consequential, easily controlled choice that context-extension recipes should set and report.

## 1 INTRODUCTION

Many applications need language models to read inputs far longer than their pretraining sequences (Bai et al., 2023; Hsieh et al., 2024). The common remedy is a short continuation of pretraining that makes two changes together: the training sequences grow longer, and the positional encoding is reconfigured for the new length (Chen et al., 2023b; Rozière et al., 2023; Zhu et al., 2023; Chen et al., 2023a). With rotary position embeddings (RoPE), which rotate each query–key pair by a position-dependent angle (Su et al., 2024), the reconfiguration rescales the rotation frequencies: position interpolation (PI) compresses every frequency by the length ratio, whereas YaRN and LongRoPE rescale frequencies dimension by dimension (Chen et al., 2023b; Peng et al., 2023; Ding et al., 2024). This positional choice shapes how far the model extrapolates beyond its trained window (Press et al., 2022; Sun et al., 2023).

Recipes differ in their target configuration but fix its introduction time: the target frequencies apply from the first update at the new length (Chen et al., 2023b; Rozière et al., 2023), so the model adapts to longer sequences and to new positions at the same time. To our knowledge, no study compares this static schedule with a delayed one at a shared final configuration and token budget (Appendix R). We therefore ask whether delaying the positional change improves the final model when the target and the continuation budget are held fixed.

We isolate the effect of timing with delayed positional adjustment (DEPA; Figure 1a–b). DEPA keeps the source (pretraining) RoPE frequencies during an initial *hold* phase at the longer length, switches all changed frequencies together at a single *release* step, and spends the remaining updates adapting to the target. Our main setting releases after 90% of the continuation updates, leaving 400 of 4000 updates for adaptation. We compare DEPA with *static* adjustment, which applies the target from the first update. The two *arms* of each comparison share the pretrained model, continuation



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.

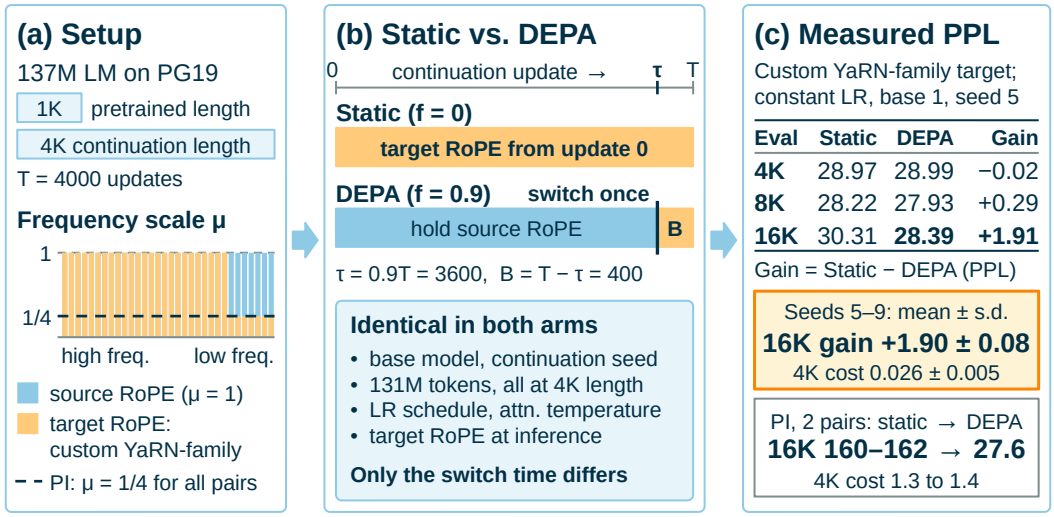



Figure 1: DEPA changes only *when* the target RoPE frequencies enter context-extension continuation. (a) Setup: a 137M PG19 model pretrained at 1K tokens is continued at 4K for $T = 4000$ updates. The bars give the frequency multiplier $\mu$ of each of the 32 RoPE pairs: sky blue is the source ($\mu = 1$) and apricot the custom YaRN-family target, which keeps 25 pairs at $\mu = 1$ and compresses the 7 lowest-frequency pairs to $\mu = 1/4$ (Appendix N.5); PI uses $\mu = 1/4$ for every pair (dark dashed line). (b) Static adjustment ($f = 0$) applies the target from update 0; DEPA with release fraction $f = 0.9$ holds the source frequencies, switches once at $\tau = fT = 3600$, and adapts for the remaining $B = 400$ updates. (c) Measured PPL for one constant-learning-rate pair (custom YaRN-family target, PG19 base 1, continuation seed 5); gain = static − DEPA (positive favors DEPA) and 4K cost = DEPA − static. The upper box gives the mean ± sample s.d. over five such pairs (seeds 5–9; Section 5.3); the lower box summarizes the two decaying-learning-rate PI pairs of Table 1. In Figures 2–4a, blue marks DEPA and dark grey static; Figure 4b colors DEPA runs by target.

seed, token budget, optimizer schedule, final target, and evaluation, so release timing is the only difference and any gap between the arms comes from the training path. DEPA adds one timing parameter, requires no frequency-band ordering, and adds no inference cost. The design has a simple geometric motivation: under a fourfold PI extension, a 16K input at the target frequencies spans the same rotation angles as a 4K input at the source frequencies, so the hold trains on the angle range that 16K inference will use, whereas the static arm must extrapolate fourfold in angle (Section 3).

We test DEPA on 137M-parameter decoder-only transformers pretrained on PG19 at 1K tokens and continued at 4K. The primary outcome is perplexity (PPL) at 8K and 16K, beyond the trained window, reported alongside the trained-length 4K PPL. Under our custom YaRN-family target (YaRN-style dimension-wise scaling with our own frequency mask and attention temperature), whose static arm already extrapolates well, five constant-learning-rate pairs lower 16K PPL by $1.90 \pm 0.08$ at a $0.026$-PPL cost at 4K (Figure 1c). With a constant learning rate, decay alone cannot explain this gain. Under PI and published YaRN, DEPA lowers 16K PPL from about 105–190 to below 32. The gain holds on a second PG19 base, on WikiText-103, with a 20-layer architecture, and at 19M parameters. Same-target diagnostics show where the gain comes from: on identical target tokens, DEPA reduces added-context harm, the loss increase from context beyond the trained window, by 78–80% on PG19.

**Contributions.**

- A single-switch continuation schedule, DEPA, and a matched protocol in which static and DEPA arms share the final positional configuration, token budget, and optimizer schedule, so that release timing is the only difference (Section 3).
- Paired evidence that DEPA improves perplexity beyond the trained window in every tested setting, across three target recipes, two learning-rate regimes, two corpora, five pretrained bases, and models from 19M to 137M parameters (Section 5).

- A finite-budget recovery analysis that motivates reserving adaptation updates after a late release, and the measured reserve response: every tested custom-target reserve from 100 to 2000 updates improves on static, and 100–200 of 4000 updates is best at 16K (Sections 4 and 5.5).
- Same-target diagnostics that score fixed target tokens with progressively longer inputs and trace the long-input gain to reduced added-context harm (78–80% less on PG19 and 86–87% on WikiText-103; Section 5.4).

## 2 Background and related work

**Choosing a positional target.** RoPE rotates query and key pairs at frequency $\omega_d = b^{-2d/D}$ (Su et al., 2024), and a target supplies per-pair multipliers, $\omega_d^* = \mu_d^* \omega_d$, uniform for PI and dimension-dependent for YaRN and LongRoPE (Chen et al., 2023b; Peng et al., 2023; Ding et al., 2024). CLEX learns continuous scaling dynamics, and LongRoPE2 further targets the short/long-context trade-off (Chen et al., 2023a; Shang et al., 2025). Base-frequency and frequency-band studies show why training length and positional configuration must be considered jointly (Liu et al., 2023b; Barbero et al., 2024; Oka et al., 2026). DEPA fixes a target and changes when it enters training, so it applies on top of an existing frequency formula. Because matched arms end at the same frequencies, their difference reflects the training path rather than frequency-band geometry alone.

**Changing the training path.** PoSE changes training position indices, Code Llama uses long-context continuation, and DroPE removes positional embeddings before recalibration at the original training length (Zhu et al., 2023; Rozière et al., 2023; Gelberg et al., 2026). DEPA trains at the new length throughout and retains target RoPE at inference (comparison axes in Appendix R, Table 64). A matched two-step control in the spirit of staged recipes, which moves the multipliers halfway at $0.5T$ and to the full target at $0.9T$, reaches a 16K gain within $0.08$ PPL of the single late switch (Appendix E), so one switch suffices. Curriculum learning and cooldown schedules similarly make update allocation consequential (Bengio et al., 2009; Hu et al., 2024; Hägele et al., 2024; Team OLMo et al., 2025); constant-learning-rate (constant-LR) pairs isolate a timing effect that does not require decay.

**Other routes to longer context.** ALiBi, XPos, KERPLE, functional interpolation, and receptive-field approaches change the positional or attention bias (Press et al., 2022; Sun et al., 2023; Chi et al., 2022a; Li et al., 2023; Chi et al., 2022b). Adaptive spans and contextual positions vary how distance is represented (Sukhbaatar et al., 2019; Golovneva et al., 2024). Transformer-XL and Compressive Transformers instead alter persistent context, while StreamingLLM and Self-Extend modify inference access (Dai et al., 2019; Rae et al., 2019; Xiao et al., 2023; Jin et al., 2024). These address distinct architectural and inference choices. Efficient attention kernels reduce execution cost (Dao et al., 2022; Dao, 2023), but token counts still do not equal FLOPs across sequence lengths, so we match tokens and report wall time separately (Section 5.5).

**What an extended window measures.** Long-input perplexity and the use of distant evidence are distinct properties. LongBench, RULER, position-sensitivity analyses, and retrieval-head studies evaluate complementary aspects of context use (Bai et al., 2023; Hsieh et al., 2024; Liu et al., 2023a; Wu et al., 2024). We therefore report same-target nested-input diagnostics alongside perplexity (Section 5.4) and a counterfactual retrieval probe (Appendix L).

## 3 Method

Let $W_0$ be the pretrained weights and $T$ the continuation budget in optimizer updates. A positional configuration $c = (\mu, a)$ contains per-pair frequency multipliers $\mu$ and an attention-temperature factor $a$ that scales the query–key logits. The hold configuration $c_0$ uses the source frequencies ($\mu = \mathbf{1}$); $c_*$ is the common final target. At optimizer step $s$,

$$c_s = \begin{cases} c_0, & s < \tau, \\ c_*, & s \geq \tau, \end{cases} \qquad \tau = \lfloor fT \rfloor, \quad 0 \leq f < 1. \tag{1}$$

Every update trains on length $L_1$, and all inference lengths use $c_*$. The only timing parameter is the release step $\tau$: updates $s < \tau$ train under $c_0$, the switch at $\tau$ applies every changed frequency at once, and the remaining $B = T - \tau$ updates adapt at $c_*$. We call $f$ the release fraction and $B$ the reserved adaptation budget (the *reserve*). Static adjustment is $f = 0$, and fixed DEPA uses $f = 0.9$ (Figure 1) for every target recipe. Within each pair, the base, continuation seed, token budget, target, evaluation, and optimizer schedule are matched. Weights and optimizer state evolve throughout the hold, so arms with identical final frequencies reach different final weights. Frequency-band probes are diagnostic only; DEPA does not require them.

**Why hold, then switch.** Continuing directly at the target asks optimization to absorb two changes at once — longer sequences and new positional frequencies. Holding the source frequencies first adapts the model to the longer window under a positional encoding it already uses; the single switch then applies one controlled positional displacement, and the reserved updates adapt to the final target. Because both arms terminate at the same $c_*$ and inference uses $c_*$ alone, an endpoint difference between arms reflects the training path, not the final frequencies. Delay creates a trade-off: the switch changes the loss at the weights reached by that point, while a later switch leaves a smaller reserve $B$. The conditional bounds of Section 4 weigh preparedness gained during the hold against the reduced adaptation budget, and the experiments measure the trade-off.

**An elementary angle-coverage view.** At relative offset $m$, pair $d$ rotates by the angle $m\omega_d$ under the source frequencies and by $m\mu_d^*\omega_d$ under the target. PI sets $\mu_d^* = L_0/L_1$ for every pair, so with $L_1 = 4L_0$ a 16K input under the target reaches angles up to about $16384\,\omega_d/4 = 4096\,\omega_d$: the range that a 4K training window reaches under the source frequencies. The DEPA hold therefore trains on the angle range that 16K inference will use, whereas the static arm trains only on angles up to about $1024\,\omega_d$, the pretraining range, and must extrapolate fourfold in angle at 16K. The same holds for the long-wavelength pairs that the custom YaRN-family target compresses (Table 33). This observation offers a geometric explanation for a large PI gain, consistent with the source-hold trajectories in Figure 4a. The custom target, whose static arm already extrapolates well, and published YaRN extend the evidence beyond PI (Section 5.2).

**Frequency timing versus temperature.** PI uses no temperature correction. Our custom YaRN-family target uses a wavelength mask and a sub-unity query–key logit multiplier

$$a_* = \frac{1}{1 + 0.1\ln(L_1/L_0)}, \tag{2}$$

which is $0.87825$ at a fourfold extension. Both arms apply this same temperature from update zero, $c_0 = (\mathbf{1}, a_*)$ and $c_* = (\mu^*, a_*)$, so only frequency timing changes. This custom mask and temperature differ from published YaRN, whose multiplier is $(0.1\ln(L_1/L_0) + 1)^2$. We therefore report the published recipe as a separate target (Appendix D.1). A temperature-timing probe attributes $0.11$ PPL of a $1.98$-PPL custom-target gain at 16K to temperature timing (Appendix D.1).

## 4 Positional change and finite-budget recovery

Write $\ell_L(W, c)$ for mean negative log-likelihood (NLL) and $\mathrm{PPL}_L = \exp \ell_L$. Geometry bounds a positional intervention, and the weights at which it occurs shape the trained endpoint. For offset $m$, pair $d$ changes by $\delta_d = m(\omega_d^* - \omega_d)$, giving the exact identity

$$\|R(m\omega_d^*) - R(m\omega_d)\|_2 = 2|\sin(\delta_d/2)| \le \min\{2, |\delta_d|\}.$$

This yields a layer-local logit bound at fixed input activations (Appendix B, Eq. 4). A full-network loss bound follows under bounded loss derivatives along the multiplier path (Proposition 1 in Appendix B). Early and late switches have identical frequency displacement but occur at different weights and can produce different loss jumps.

**The remaining adaptation budget.** For a fixed reference loss, let $e_u$ denote nonnegative target excess loss after $u$ post-release updates. *Assume locally* that $e_{u+1} \le (1-\kappa)e_u + \epsilon$, with $0 < \kappa \le 1$ and $\epsilon \ge 0$. After $B = T - \tau$ updates,

$$e_B \le (1-\kappa)^B e_0 + \frac{\epsilon}{\kappa}\big[1 - (1-\kappa)^B\big].$$

Table 1: 137M PG19 paired continuations with LR decay; values rounded to three decimals. Each pair shares its pretrained base, continuation seed, final target, and 131.072M continuation tokens; lower PPL is better. Band-curriculum baselines, logged at one decimal, are in Table 11 (Appendix D.3).

| | | | | Validation PPL at length (tokens) | | |
|---|---|---|---|---|---|---|
| Target | Base | Seed | Schedule | 4096 | 8192 | 16384 |
| PI | 0 | 1 | Static | 28.050 | 66.653 | 159.919 |
| | | | DEPA $0.9T$ | 29.385 | 28.343 | 27.568 |
| | 0 | 2 | Static | 28.108 | 71.689 | 162.491 |
| | | | DEPA $0.9T$ | 29.522 | 29.725 | 27.575 |
| Custom YaRN-family | 0 | 2 | Static | 28.239 | 29.459 | 29.007 |
| | | | DEPA $0.9T$ | 28.260 | 28.861 | 27.044 |
| | 1 | 2 | Static | 28.434 | 29.277 | 28.938 |
| | | | DEPA $0.9T$ | 28.465 | 28.978 | 27.148 |

The initial excess $e_0$ includes target quality at the held weights, not only the positive loss jump at release. Delaying can change that state and its recovery rate while reducing $B$. This trade-off motivates reserving adaptation time, and Section 5.5 measures the useful reserve directly. Derivations, a varying-rate version, and recovery traces appear in Appendix B.

# 5 Experiments

## 5.1 Settings and comparison units

Models are decoder-only, pre-layer-normalized transformers with controllable RoPE. The main 137M family on PG19 (Rae et al., 2019) has 14 blocks, embedding width 768, 12 attention heads of dimension 64, tied GPT-2 BPE embeddings, and RoPE with base 10,000 and 32 rotary pairs per head. Each 137M base is pretrained for 25,000 updates at $L_0 = 1024$ (204.8M token positions) and then continued at $L_1 = 4096$ for 4000 updates (131.072M continuation tokens) with a fresh AdamW optimizer (Appendix D). We report validation PPL at 4K, 8K, and 16K on randomly positioned windows (40 batches and 1.31M scored positions per length for the full-precision 137M runs); lower is better, and 8K and 16K test extrapolation beyond training. A *base* is one pretrained checkpoint and a *continuation seed* fixes the randomness of one continuation run. Main pairs share their base and continuation seed; each additional base, recipe, and corpus forms its own paired comparison.

The decaying-LR family uses AdamW, $(\beta_1, \beta_2) = (0.9, 0.95)$, weight decay 0.1, gradient clipping 1.0, and peak LR $3 \times 10^{-4}$. The 137M constant-LR comparisons use $4.96413 \times 10^{-5}$, the mean of the decaying continuation schedule. The separate 19M experiment on WikiText-103 (Merity et al., 2017) uses LR $3.6 \times 10^{-5}$ and 5000 updates at $L_1 = 512$, a temperature-free YaRN-family-style target, and its own base. Most continuations replace 2% of batches with synthetic copy-task batches, as in pretraining; their removal is measured separately. Per-family architecture, data, and evaluation details are in Appendix D.

## 5.2 Replication across targets, bases, and corpora

We first ask whether the DEPA gain survives replication across bases, continuation seeds, target recipes, learning-rate regimes, and corpora.

Table 1 shows gains under both targets, with the largest gains under PI. For PI, the two static 16K endpoints are 159.919 and 162.491 PPL; DEPA reduces them to 27.568 and 27.575 at 4K costs of 1.335 and 1.414 PPL. The custom YaRN-family static endpoints already extrapolate much better, and DEPA improves them further, from 29.007 to 27.044 on base 0 and 28.938 to 27.148 on base 1, at 4K costs of 0.021 and 0.031 PPL.

Table 2 places the three recipes under one constant-LR protocol on base 1. At 16K, the DEPA gain is $84.38 \pm 6.56$ PPL under published YaRN (three seeds) and $155.60 \pm 5.53$ PPL under PI on the two seeds shared with published YaRN (mean $\pm$ sample s.d.). Under PI and published YaRN the static

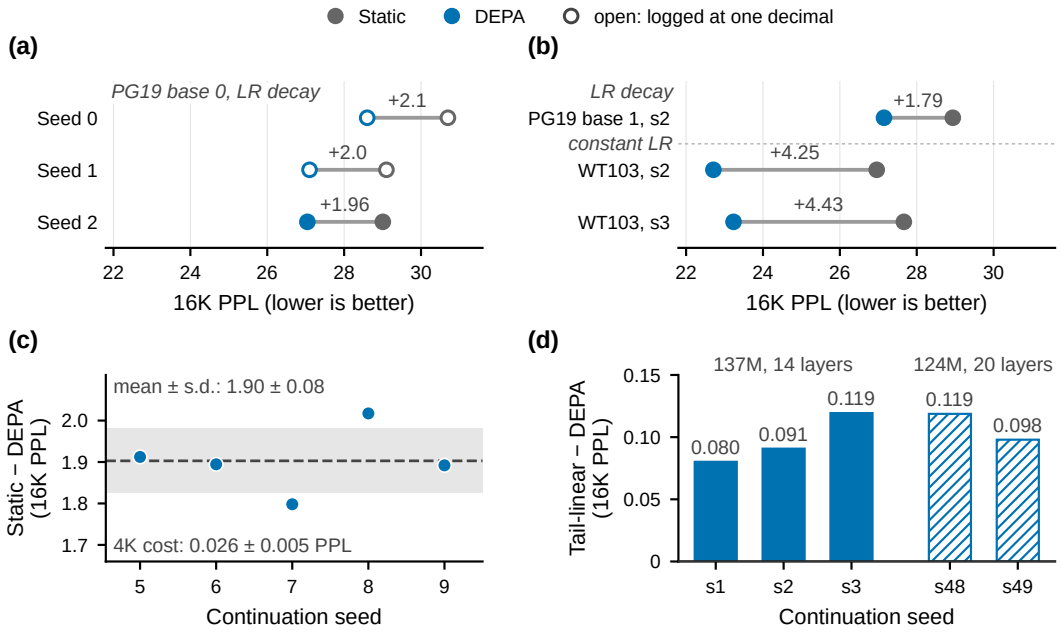



Figure 2: DEPA lowers 16K PPL in every displayed custom YaRN-family pair. Each pair shares base, continuation seed, target, and token budget. (a–b) Paired 16K endpoints; numbers give the gain, static − DEPA. (a) PG19 base 0, LR decay; seeds 0–1 logged at one decimal (open markers), others at full precision. (b) PG19 base 1, LR decay, and WikiText-103 (WT103) pairs at constant LR; s$N$: seed $N$. (c) Five constant-LR pairs on PG19 base 1 (seeds 5–9): paired 16K gains, their mean (dashed line) ± one sample s.d. (band), and the 4K cost; three base-0 constant-LR pairs are in Appendix N.2. (d) Equal-exposure tail-linear ramp minus DEPA, 16K PPL; s1–s3: PG19 base 0, constant LR (Appendix N.2); s48–s49: a 124M, 20-layer architecture (hatched bars; Appendix L).

Table 2: DEPA gains under three positional recipes at a matched continuation protocol (constant LR, base 1; one seed pair per row). Temp.: attention temperature; the interpolation is uniform ($\mu_d = L_0/L_1$) for PI, our band mask for custom YaRN-family, and NTK-by-parts for published YaRN. Gain = static − DEPA PPL (positive favors DEPA); 4K cost = DEPA − static at the trained length; PPL to three decimals, gains and costs to two; same-seed decomposition in Appendix N.4.

| | | | 16K PPL | | Gain and cost (PPL) | | |
|---|---|---|---|---|---|---|---|
| Recipe | Temp. | Seed | Static | DEPA | 16K gain | 8K gain | 4K cost |
| PI | none | 2 | 172.314 | 29.223 | +143.09 | +38.45 | 0.92 |
| | | 44 | 189.587 | 30.076 | +159.51 | +36.81 | 1.10 |
| | | 45 | 183.401 | 31.714 | +151.69 | +35.90 | 0.98 |
| Custom YaRN-family | 0.878 | 5 | 30.307 | 28.395 | +1.91 | +0.29 | 0.02 |
| Published YaRN | 1.296 | 44 | 117.120 | 27.870 | +89.25 | +22.20 | 0.15 |
| | | 45 | 116.397 | 29.427 | +86.97 | +21.76 | 0.16 |
| | | 46 | 104.586 | 27.667 | +76.92 | +22.13 | 0.18 |

endpoints reach about 105–190 PPL at 16K, whereas the DEPA endpoints stay between 27.7 and 31.7 PPL across all three recipes.

Figure 2a–b collects the custom-target pairs with LR decay on PG19 bases 0 and 1, including two base-0 seeds logged at one decimal, and adds a 137M WikiText-103 base. On that base, two constant-LR pairs gain 4.25 and 4.43 PPL at 16K with less than 0.05 PPL trained-length cost. A separate 124M, 20-layer architecture also retains positive gains on two seeds (Appendix L).

Table 3: Separate constant-LR experiment at 19M (temperature-free YaRN-family-style target, WikiText-103, $L_1 = 512$, LR $3.6 \times 10^{-5}$ at every update), evaluated at $8L_1 = 4096$ tokens. Staggered converts frequency bands in index order with 400-step cosine ramps (Appendix D); release $0.5T$ switches all bands at once at mid-budget. Per-seed PPL was logged at two-decimal precision; means are rounded. The lowest PPL in each column is in bold.

| | PPL at 4096 tokens | | | |
|---|---|---|---|---|
| Schedule | Seed 0 | Seed 1 | Seed 2 | Mean |
| Static | 100.39 | 100.61 | 99.84 | 100.28 |
| Staggered | 88.27 | 88.44 | 87.68 | 88.13 |
| Release $0.5T$ | 85.95 | 86.12 | 85.40 | 85.83 |
| DEPA $0.9T$ | **81.21** | **81.30** | **80.66** | **81.06** |

Table 4: Added-context utility $U$ at 16K under the custom YaRN-family target (negative values mean the longer input hurts on identical targets). PG19 rows use nested 4K/16K inputs on fixed targets (base 0, constant LR; Appendix N.2); WikiText-103 rows compare the full 16K input with its final 1K tokens (Appendix Q). Harm reduction is $1 - U_{\mathrm{DEPA}}/U_{\mathrm{static}}$, computed from the rounded values shown.

| | | $U$ (nats/token) | | |
|---|---|---|---|---|
| Corpus (probe) | Seed | Static | DEPA | Harm reduction (%) |
| PG19 (nested 4K→16K) | 1 | $-0.296$ | $-0.059$ | 80 |
| | 3 | $-0.286$ | $-0.062$ | 78 |
| WikiText-103 (1K local vs. full 16K) | 2 | $-0.571$ | $-0.079$ | 86 |
| | 3 | $-0.560$ | $-0.075$ | 87 |

### 5.3 Timing beyond learning-rate decay

Five constant-LR pairs on PG19 base 1 (continuation seeds 5–9) improve 16K PPL by $1.903 \pm 0.078$ (sample s.d. across continuation seeds) at a 4K cost of $0.026 \pm 0.005$ PPL. Figure 2c displays these five pairs.

At 19M, all three seeds preserve the ordering static, staggered, release at $0.5T$, DEPA $0.9T$, with mean PPL 100.28, 88.13, 85.83, and 81.06 (Table 3); DEPA improves on static by 19.22 PPL (19.2%) on average, with paired gains of 19.18–19.31 on every seed. Constant LR rules out decay alone as the explanation and points to the allocation of updates between source and target frequencies.

A tail-linear transition, which ramps the multipliers linearly over the final updates, matches the abrupt switch's cumulative frequency-multiplier exposure and its LR-weighted sum. DEPA retains positive gains over this exposure-matched control in all five displayed pairs (Figure 2d; Appendices N.2 and L; construction in Appendix P.1).

### 5.4 Same-target diagnostics: reduced added-context harm

Lower PPL at 16K can come from using the added context or from being hurt less by it. Because the scored targets of the main PPL sweep change with input length, we separate the two by fixing the final 128 target tokens of 32 maximum-length windows and varying only their available input suffix (Figure 3). Define the added-context utility $U(a, b) = \ell(a) - \ell(b)$ for input lengths $a < b$: positive values mean the longer input helps, and negative values measure added-context harm.

Table 4 summarizes the outcome. On PG19, static utility from 4K to 16K is about $-0.29$ to $-0.30$ nats/token on the two seeds, and DEPA reduces this harm by 78–80%; on the WikiText-103 base, a full-versus-local probe gives an 86–87% reduction of a larger static harm ($-0.56$ to $-0.57$ versus about $-0.32$ for PG19 on that probe; Appendix M.5). The paired decomposition (Appendix N.2) attributes DEPA's 16K advantage on these fixed targets to this reduced harm. The advantage is broad across windows rather than carried by a few: at 16K, DEPA has lower full-input loss than static on all

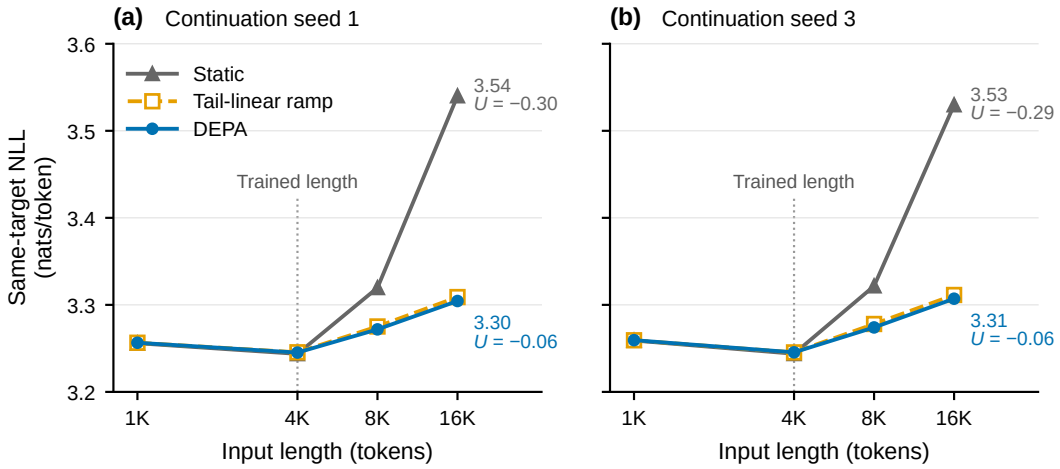



Figure 3: DEPA reduces added-context harm on identical targets (custom YaRN-family target, PG19 base 0, constant LR; the PG19 rows of Table 4). Each panel scores the final 128 targets of 32 windows (4096 targets in total) with nested 1K/4K/8K/16K inputs; lower NLL is better, and the dotted line marks the trained 4K length. Beyond it, DEPA and the tail-linear ramp stay far below static adjustment for both continuation seeds. Numbers at 16K give the window-mean NLL and the added-context utility $U(4\text{K}, 16\text{K})$. Lines connect window means. Full values and paired decompositions are in Appendix N.2.

32 natural-text windows in both constant-LR PI pairs (per-window gains of 1.52–4.39 nats/token) and in the custom-target pair (minimum gain 0.040 nats/token; Appendices M.4 and M.5).

### 5.5 Length staging, reserve size, and release point

All main DEPA updates use $L_1$. Staging the hold across lengths (1K, 2K, then 4K) at the same token budget, with the target phase at 4K, yields a 16K gain of 1.59–1.80 PPL over equally staged static on bases 0 and 1 and runs at $0.831\times$ the fixed-length wall time (Appendix P.1). On base 1, spending 72% instead of 44% of the hold at 4K narrows the 16K gap between staged and fixed-length DEPA from 0.40 to 0.10 PPL, at $0.920\times$ the wall time. Adapting to the released target at 2K instead of 4K keeps the 16K gain ($+2.22$ versus $+2.04$ PPL on one seed), so the adaptation phase can also run at the shorter length (Appendix P.1).

**Learning during the hold.** The hold already improves long-input modeling before any update at the target frequencies. On fixed probe windows of the two decaying-LR PI pairs, the held model's 16K PPL under the source frequencies falls from 458 and 421 at the pretrained base to 53 and 49 after 1001 updates and stays at 49–54 up to the last saved probe before release, whereas the static model under PI stays between 141 and 172 over the same updates (Figure 4a; Appendix P.5). The reserved updates then adapt this prepared state to the final target.

**Measured reserve response.** All main comparisons use $f = 0.9$ ($B = 400$); Figure 4b varies $B$. At $T = 4000$, the custom YaRN-family target is best at 16K with 100–200 reserved updates; at $T = 8000$, it is flat from about 200 to 800 (Appendix E). PI does better with 400 than with 800 reserved updates at 16K for both budgets (Appendix P.3). Every measured custom-target reserve, from 100 to 2000 updates, improves on the static 16K endpoint. Reserving 200 updates is consistently better than releasing at mid-budget (a 2000-update reserve), so the reserve balances hold exposure against target adaptation, as the recovery bound of Section 4 anticipates.

**Trained-length trade-off.** The release point also tunes the 4K difference between the arms. On one PI seed, releasing at $0.8T$ instead of $0.9T$ roughly halves the 4K cost (1.335 to 0.682 PPL) while retaining 98.6% of the 16K gain (Appendix L), so the release point can trade in-window cost against extrapolation.

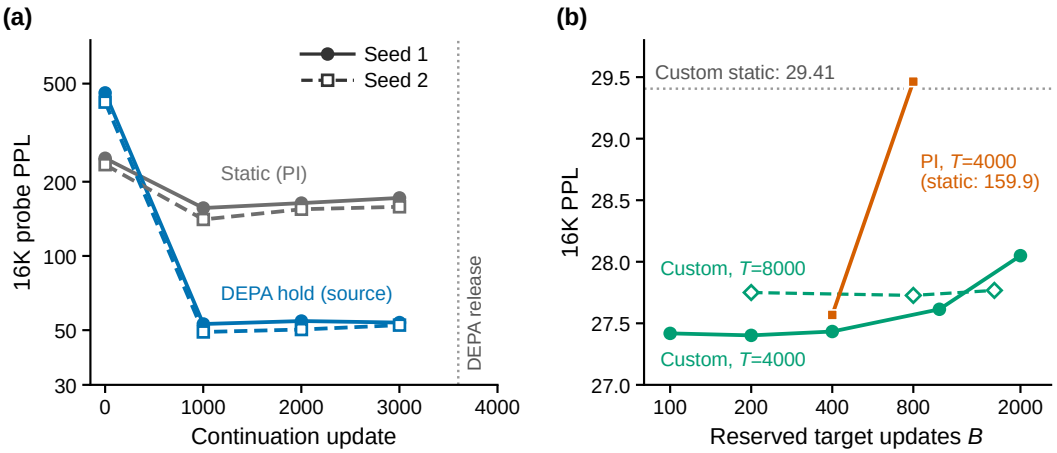



Figure 4: Learning during the hold and the reserve response. (a) 16K probe PPL (log scale) during continuation for the two decaying-LR PI pairs of Table 1 (20 fixed probe batches; lines connect saved observations). Each arm is evaluated at its own training configuration: the DEPA hold at the source frequencies, static at PI. (b) 16K PPL of DEPA against the reserved target updates $B = T(1 - f)$ on a log axis: the custom YaRN-family target ("Custom") at $T = 4000$ (constant LR, base 0, seed 1) and $T = 8000$ (constant LR, base 1, seed 13), and PI at $T = 4000$ (decaying LR, base 0, seed 1); one run per point. The dotted line marks the static endpoint of the $T = 4000$ custom pair (29.41 PPL); the static PI endpoint (159.9 PPL) lies off scale.

Removing continuation copy batches preserves the gain in the PI and custom-target pairs (Appendix N.1). At $T = 8000$ the PI gain grows, and DEPA with the same 400-update reserve improves from 27.6 to 25.6 PPL at 16K (Appendices C.2 and P.3).

## 6 Discussion and Conclusion

Across 19M–137M models, two corpora, and five pretrained bases, delaying the switch to target RoPE frequencies while training at the new length improves perplexity beyond the trained window in every tested setting, including constant-LR continuations, with the largest gains under PI and published YaRN. Same-target evaluations trace this gain to reduced added-context harm, and the measured reserve response follows the finite-budget recovery trade-off of Section 4. Release timing is therefore a simple, controllable variable that context-extension studies should report and match.

### 6.1 Limitations and Future Work

We see four directions for future work; Appendix S discusses each in detail. *Scale and recipes:* testing DEPA on billion-parameter models, larger extension ratios, and production extension recipes, against separately tuned static baselines. *Evaluation:* our counterfactual retrieval probe, run under PI and published YaRN, shows no retrieval gain (Appendix L); extending the evaluation to downstream long-context benchmarks such as LongBench and RULER is the next step. *Release rule:* replacing the fixed release fraction with an adaptive, state-based rule that also balances trained-length and long-input quality. *Mechanism:* turning the conditional recovery model and the angle-coverage view into a tested account of what the hold changes in the network.

### Ethics Statement

This work trains and evaluates language models on publicly available corpora, benchmarks, and model checkpoints. It involves no human-subject experiments. Public corpora can contain personal or sensitive content, and the work shares the data-related and model-use risks common to language-model research.

### Reproducibility Statement

The supplementary material (a zip archive) provides code, configurations, launch scripts, analysis scripts, and result files, together with a README linking reported tables and figures to their producing scripts. Experimental settings are specified in the main text and the appendix.

### AI Use Statement

During the project, vibe coding tools (Codex and Claude Code) were used to help develop the code for the methods and experiments. All such work was conducted under human supervision and underwent human review. During manuscript preparation, AI tools were used to polish and revise the text and figures.

## References


Yushi Bai, Xin Lv, Jiajie Zhang, Hongchang Lyu, Jiankai Tang, Zhidian Huang, Zhengxiao Du, Xiao Liu, et al. LongBench: A bilingual, multitask benchmark for long context understanding. *arXiv preprint arXiv:2308.14508*, 2023. URL https://arxiv.org/abs/2308.14508.

Federico Barbero, Alex Vitvitskyi, Christos Perivolaropoulos, Razvan Pascanu, and Petar Veličković. Round and round we go! What makes rotary positional encodings useful? *arXiv preprint arXiv:2410.06205*, 2024. URL https://arxiv.org/abs/2410.06205.

Yoshua Bengio, Jérôme Louradour, Ronan Collobert, and Jason Weston. Curriculum learning. In *Proceedings of the 26th annual international conference on machine learning*, pp. 41–48, 2009.

Guanzheng Chen, Xin Li, Zaiqiao Meng, Shangsong Liang, and Lidong Bing. CLEX: Continuous length extrapolation for large language models. *arXiv preprint arXiv:2310.16450*, 2023a.

Shouyuan Chen, Sherman Wong, Liangjian Chen, and Yuandong Tian. Extending context window of large language models via positional interpolation. *arXiv preprint arXiv:2306.15595*, 2023b.

Ta-Chung Chi, Ting-Han Fan, Peter J. Ramadge, and Alexander I. Rudnicky. KERPLE: Kernelized relative positional embedding for length extrapolation. *arXiv preprint arXiv:2205.09921*, 2022a. URL https://arxiv.org/abs/2205.09921.

Ta-Chung Chi, Ting-Han Fan, Alexander I. Rudnicky, and Peter J. Ramadge. Dissecting transformer length extrapolation via the lens of receptive field analysis. *arXiv preprint arXiv:2212.10356*, 2022b. URL https://arxiv.org/abs/2212.10356.

Zihang Dai, Zhilin Yang, Yiming Yang, Jaime Carbonell, Quoc V Le, and Ruslan Salakhutdinov. Transformer-XL: Attentive language models beyond a fixed-length context. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pp. 2978–2988, 2019.

Tri Dao. FlashAttention-2: Faster attention with better parallelism and work partitioning. *arXiv preprint arXiv:2307.08691*, 2023. URL https://arxiv.org/abs/2307.08691.

Tri Dao, Daniel Y. Fu, Stefano Ermon, Atri Rudra, and Christopher Ré. FlashAttention: Fast and memory-efficient exact attention with IO-awareness. *arXiv preprint arXiv:2205.14135*, 2022. URL https://arxiv.org/abs/2205.14135.

Yiran Ding, Li Lyna Zhang, Chengruidong Zhang, Yuanyuan Xu, Ning Shang, Jiahang Xu, Fan Yang, and Mao Yang. LongRoPE: Extending LLM context window beyond 2 million tokens. *arXiv preprint arXiv:2402.13753*, 2024.

Yoav Gelberg, Koshi Eguchi, Takuya Akiba, and Edoardo Cetin. Extending the context of pretrained LLMs by dropping their positional embedding. In *International Conference on Learning Representations*, 2026. URL https://proceedings.iclr.cc/paper_files/paper/2026/hash/62133b95efaa6f10a5395eb10ca087cd-Abstract-Conference.html.

Olga Golovneva, Tianlu Wang, Jason Weston, and Sainbayar Sukhbaatar. Contextual position encoding: Learning to count what's important. *arXiv preprint arXiv:2405.18719*, 2024. URL https://arxiv.org/abs/2405.18719.

Alexander Hägele, Elie Bakouch, Atli Kosson, Loubna Ben Allal, Leandro Von Werra, and Martin Jaggi. Scaling laws and compute-optimal training beyond fixed training durations. In *Advances in Neural Information Processing Systems*, 2024.

Cheng-Ping Hsieh, Simeng Sun, Samuel Kriman, Shantanu Acharya, Dima Rekesh, Fei Jia, Yang Zhang, and Boris Ginsburg. RULER: What's the real context size of your long-context language models? *arXiv preprint arXiv:2404.06654*, 2024. URL https://arxiv.org/abs/2404.06654.

Shengding Hu, Yuge Tu, Xu Han, et al. MiniCPM: Unveiling the potential of small language models with scalable training strategies. *arXiv preprint arXiv:2404.06395*, 2024.

Hongye Jin, Xiaotian Han, Jingfeng Yang, Zhimeng Jiang, Zirui Liu, Chia-Yuan Chang, Huiyuan Chen, and Xia Hu. LLM maybe LongLM: Self-Extend LLM context window without tuning. *arXiv preprint arXiv:2401.01325*, 2024.

Shanda Li, Chong You, Guru Guruganesh, Joshua Ainslie, Santiago Ontanon, Manzil Zaheer, Sumit Sanghai, Yiming Yang, et al. Functional interpolation for relative positions improves long context transformers. *arXiv preprint arXiv:2310.04418*, 2023. URL https://arxiv.org/abs/2310.04418.

Nelson F. Liu, Kevin Lin, John Hewitt, Ashwin Paranjape, Michele Bevilacqua, Fabio Petroni, and Percy Liang. Lost in the middle: How language models use long contexts. *arXiv preprint arXiv:2307.03172*, 2023a. URL https://arxiv.org/abs/2307.03172.

Xiaoran Liu, Hang Yan, Shuo Zhang, Chenxin An, Xipeng Qiu, and Dahua Lin. Scaling laws of RoPE-based extrapolation. *arXiv preprint arXiv:2310.05209*, 2023b.

Stephen Merity, Caiming Xiong, James Bradbury, and Richard Socher. Pointer sentinel mixture models. In *International Conference on Learning Representations*, 2017.

Yui Oka, Itsumi Saito, Kyosuke Nishida, and Kuniko Saito. Frequency bands in RoPE: Base frequency and context length shape the interpolation–extrapolation trade-off. In *International Conference on Learning Representations*, 2026. URL https://proceedings.iclr.cc/paper_files/paper/2026/hash/993dbaec0418a6449090f1debbcb8844-Abstract-Conference.html.

Bowen Peng, Jeffrey Quesnelle, Honglu Fan, and Enrico Shippole. YaRN: Efficient context window extension of large language models. *arXiv preprint arXiv:2309.00071*, 2023.

Ofir Press, Noah A Smith, and Mike Lewis. Train short, test long: Attention with linear biases enables input length extrapolation. In *International Conference on Learning Representations*, 2022.

Jack W. Rae, Anna Potapenko, Siddhant M. Jayakumar, and Timothy P. Lillicrap. Compressive transformers for long-range sequence modelling. *arXiv preprint arXiv:1911.05507*, 2019. URL https://arxiv.org/abs/1911.05507.

Baptiste Rozière, Jonas Gehring, Fabian Gloeckle, Sten Sootla, Itai Gat, Xiaoqing Ellen Tan, Yossi Adi, Jingyu Liu, Romain Sauvestre, Tal Remez, Jérémy Rapin, et al. Code Llama: Open foundation models for code. *arXiv preprint arXiv:2308.12950*, 2023.

Ning Shang, Li Lyna Zhang, Siyuan Wang, Gaokai Zhang, Gilsinia Lopez, Fan Yang, Weizhu Chen, and Mao Yang. LongRoPE2: Near-lossless LLM context window scaling. *arXiv preprint arXiv:2502.20082*, 2025. URL https://arxiv.org/abs/2502.20082.

Jianlin Su, Murtadha Ahmed, Yu Lu, Shengfeng Pan, Wen Bo, and Yunfeng Liu. RoFormer: Enhanced transformer with rotary position embedding. *Neurocomputing*, 568:127063, 2024.

Sainbayar Sukhbaatar, Edouard Grave, Piotr Bojanowski, and Armand Joulin. Adaptive attention span in transformers. *arXiv preprint arXiv:1905.07799*, 2019. URL https://arxiv.org/abs/1905.07799.

Yutao Sun, Li Dong, Barun Patra, Shuming Ma, Shaohan Huang, Alon Benhaim, Vishrav Chaudhary, Xia Song, and Furu Wei. A length-extrapolatable transformer. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics*, pp. 14590–14604, 2023.

Team OLMo, Pete Walsh, et al. 2 OLMo 2 Furious. *arXiv preprint arXiv:2501.00656*, 2025.

Wenhao Wu, Yizhong Wang, Guangxuan Xiao, Hao Peng, and Yao Fu. Retrieval head mechanistically explains long-context factuality. *arXiv preprint arXiv:2404.15574*, 2024. URL https://arxiv.org/abs/2404.15574.

Guangxuan Xiao, Yuandong Tian, Beidi Chen, Song Han, and Mike Lewis. Efficient streaming language models with attention sinks. *arXiv preprint arXiv:2309.17453*, 2023. URL https://arxiv.org/abs/2309.17453.

Dawei Zhu, Nan Yang, Liang Wang, Yifan Song, Wenhao Wu, Furu Wei, and Sujian Li. PoSE: Efficient context window extension of LLMs via positional skip-wise training. *arXiv preprint arXiv:2309.10400*, 2023.

## GUIDE TO THE SUPPLEMENTARY EVIDENCE

The opening tables (Appendix A) give the complete token-aligned timing and training-length comparisons, and Table 5 maps each main-text result to its supporting material. Appendix B supplies the conditional rotation and recovery arguments; Appendices C and D specify the continuation controls and the distinct target implementations. The remaining results are grouped by scientific question: frequency geometry and transfer (Appendices F–G), release calibration (Appendices H–K), scale replication (Appendix L), schedule controls (Appendix M), data and target choices (Appendix N), release-time measurements (Appendix O), and matched exposure or continuation horizon (Appendix P). Shared-target perplexity, added-context effects, copy retrieval, and training cost are separate outcomes; a gain in one does not imply a gain in the others. Appendix S gives the extended discussion of limitations.

Table 5: Where the appendix supports each main-text result.

| Main-text result | Supporting appendix material |
|---|---|
| Decaying-LR PI and custom-target pairs (Table 1) | Appendices L and P.4; Table 11 |
| Five constant-LR pairs on base 1 (Section 5.3) | Appendix C; Tables 6 and 7 |
| Three-recipe comparison (Table 2) | Appendices N.4 and N.5 |
| Constant-LR PI triads and equal-exposure ramps | Appendices M.3, M.5, and P.1 |
| WikiText-103 and 124M-architecture replications | Appendices Q and L |
| Added-context utility (Section 5.4) and retrieval probe (Section 6.1) | Appendices N.2, M.4, and L |
| Reserve and release-fraction response (Section 5.5) | Appendices E and N.3 |
| Length staging and wall time (Section 5.5) | Appendices P.1 and D.2 |
| Copy-mixture removal; budget transfer | Appendices N.1 and P.3 |
| Rotation bound and recovery model (Section 4) | Appendix B |
| Setup, target definitions, run ledger | Appendices D and D.1; Table 10 |

## A COMPLETE TIMING AND TRAINING-LENGTH COMPARISONS

Table 6: Complete token-aligned length × timing factorial: main PPL at the extension grid. Every arm trains the same 131.072M prediction targets under the final custom-YaRN-family target and constant LR. Seed-6 base-1 rows record identical pre-segmentation input/target-stream digests across all four cells; the seed-5 fixed-length DEPA cell reuses the fixed-release reference run of Table 55, which has no recorded digest, while the other seed-5 cells record digests. Rows without a seed label use continuation seed 5; s6 and s7 mark continuation seeds 6 and 7. 1K/2K/4K is the default staged hold, which spends a share $S = 0.444$ of the hold's updates at 4K. The seed-7 rows (hold-70) replace it with a compressed schedule, 1K→4K early, whose 1K and 2K phases are shortened so that the hold reaches 4K earlier and spends $S = 0.722$ of its updates there (Table 12). Values are computed from the per-run result files; Table 8 uses the same paired evaluation windows.

| Base | Training length | Target schedule | 4K PPL | 8K PPL | 16K PPL |
|---|---|---|---|---|---|
| base 0 (original) | Fixed 4K | Static | 28.824 | 28.321 | 30.312 |
| base 0 (original) | 1K/2K/4K | Static | 28.904 | 28.482 | 30.298 |
| base 0 (original) | Fixed 4K | DEPA $0.9T$ | 28.862 | 27.809 | 28.299 |
| base 0 (original) | 1K/2K/4K | DEPA $0.9T$ | 28.928 | 27.919 | 28.497 |
| base 1 (independent) | Fixed 4K | Static | 28.967 | 28.217 | 30.307 |
| base 1 (independent) | 1K/2K/4K | Static | 29.053 | 28.622 | 30.384 |
| base 1 (independent) | Fixed 4K | DEPA $0.9T$ | 28.992 | 27.928 | 28.395 |
| base 1 (independent) | 1K/2K/4K | DEPA $0.9T$ | 29.060 | 28.141 | 28.797 |
| base 1 (independent), s6 | Fixed 4K | Static | 28.797 | 29.446 | 30.666 |
| base 1 (independent), s6 | 1K/2K/4K | Static | 28.870 | 29.885 | 30.800 |
| base 1 (independent), s6 | Fixed 4K | DEPA $0.9T$ | 28.830 | 29.142 | 28.771 |
| base 1 (independent), s6 | 1K/2K/4K | DEPA $0.9T$ | 28.895 | 29.367 | 29.167 |
| base 1 (independent), s7 hold-70 | Fixed 4K | Static | 28.184 | 28.678 | 30.966 |
| base 1 (independent), s7 hold-70 | 1K→4K early | Static | 28.199 | 28.852 | 31.106 |
| base 1 (independent), s7 hold-70 | Fixed 4K | DEPA $0.9T$ | 28.214 | 28.412 | 29.168 |
| base 1 (independent), s7 hold-70 | 1K→4K early | DEPA $0.9T$ | 28.218 | 28.473 | 29.275 |

Table 7: Same-seed, same-probe constant-LR static/DEPA comparisons. Every row trains at fixed 4096-token length; staged-length outcomes are reported in Appendix P.1.

| Base | Schedule | 4K PPL | 8K PPL | 16K PPL |
|---|---|---|---|---|
| 0 | Static target | 28.824441 | 28.321316 | 30.311560 |
| 0 | DEPA $0.9T$ | 28.861574 | 27.809344 | 28.299475 |
| 1 | Static target | 28.967136 | 28.216703 | 30.306895 |
| 1 | DEPA $0.9T$ | 28.992129 | 27.928022 | 28.394614 |

Table 8: Paired 16K NLL gains on tail targets, both bases; base 1 rows repeat at continuation seeds 5–6 and the hold-70 seed 7. Positive favors the first recipe. Full, local, and utility all derive from the same raw windows paired by trial, window start, and target start (identity residual 0; 1536 windows across all 16 evaluation runs), computed from the stored per-window losses without further model evaluation.

| Comparison | Base | Seed | Full | Local | Utility |
|---|---|---|---|---|---|
| Fixed DEPA vs fixed static | 0 | 5 | +0.233642 | −0.001581 | +0.235223 |
| Staged DEPA vs staged static | 0 | 5 | +0.201336 | −0.000691 | +0.202028 |
| Staged DEPA vs fixed DEPA | 0 | 5 | −0.020727 | −0.002566 | −0.018162 |
| Fixed DEPA vs fixed static | 1 | 5 | +0.274674 | +0.000187 | +0.274487 |
| Staged DEPA vs staged static | 1 | 5 | +0.178421 | +0.000619 | +0.177802 |
| Staged DEPA vs fixed DEPA | 1 | 5 | −0.033883 | −0.004588 | −0.029294 |
| Fixed DEPA vs fixed static | 1 | 6 | +0.273507 | +0.000135 | +0.273372 |
| Staged DEPA vs staged static | 1 | 6 | +0.185061 | −0.000949 | +0.186009 |
| Staged DEPA vs fixed DEPA | 1 | 6 | −0.029193 | −0.002593 | −0.026599 |
| Fixed DEPA vs fixed static | 1 | 7 | +0.256020 | −0.001906 | +0.257926 |
| Staged DEPA vs staged static | 1 | 7 | +0.215600 | −0.001361 | +0.216962 |
| Staged DEPA vs fixed DEPA | 1 | 7 | −0.009929 | −0.000748 | −0.009181 |

# B THEORY: ROTATION BOUND, RECOVERY MODEL, AND OPTIMIZATION CLOCK

This appendix gives the derivations and empirical scope statements behind Section 4; the main text quotes their conclusions.

## B.1 EXACT ROTATION CHANGE AND A ONE-STEP BOUND

For a relative offset $m$, the target changes pair $d$ by $\delta_d = m(\omega_d^* - \omega_d)$. The two-dimensional rotation identity gives

$$\|R(m\omega_d^*) - R(m\omega_d)\|_2 = 2|\sin(\delta_d/2)| \le \min\{2, |\delta_d|\}. \tag{3}$$

For fixed pre-rotation query and key pairs $q_d, k_d$ in one attention layer, and unchanged attention scale $a$, it follows that

$$|z_* - z_0| \le \frac{|a|}{\sqrt{D}} \sum_d \|q_d\|_2 \|k_d\|_2 \min\{2, |m(\omega_d^* - \omega_d)|\}. \tag{4}$$

This is a layer-local statement at fixed input activations. Deeper-layer inputs can also change. If $a$ changes, the decomposition $a_* u_* - a_0 u_0 = a_*(u_* - u_0) + (a_* - a_0)u_0$ adds an explicit scale-change term; a rotation-only bound does not cover that intervention.

**Proposition 1** (Conditional loss perturbation at fixed weights)**.** *Suppose that, at fixed $W$, the loss is differentiable along the straight path between two multiplier vectors, with $|\partial \ell_L(W, \mu, a)/\partial \mu_d| \le G_d(W, L)$ throughout that path at fixed $a$. Then*

$$|\ell_L(W, \mu', a) - \ell_L(W, \mu, a)| \le \sum_d G_d(W, L)|\mu'_d - \mu_d|. \tag{5}$$

*If attention scale also changes, a corresponding bounded scale derivative contributes $G_a(W, L)|a' - a|$.*

*Proof.* Integrate the directional derivative along the segment and apply the triangle inequality. The constants bound the full network loss, including propagated activation changes. □

Spreading changes can reduce this upper bound per step under common derivative bounds. However, an early and a late abrupt release have the same multiplier displacement. Their bounds may differ because $W$ differs, and the bound alone does not order final losses. This is exactly where recovery and the remaining budget enter.

## B.2 STATE-DEPENDENT RECOVERY WITH A FINITE BUDGET

At release define the positive part of the measured fixed-weight loss change

$$J_\tau = \big[\ell_L(W_\tau, c_*) - \ell_L(W_\tau, c_0)\big]_+. \tag{6}$$

It is state-dependent, even though $c_0$ and $c_*$ are fixed. To reason about adaptation, suppose a nonnegative target excess loss $e_u$ after $u$ target-training updates satisfies, within a local recovery regime,

$$e_{u+1} \le (1 - \kappa_\tau)e_u + \epsilon_\tau, \qquad 0 < \kappa_\tau \le 1, \quad \epsilon_\tau \ge 0. \tag{7}$$

The comparator defining excess loss must be fixed during this interval. This is an explicit modeling assumption, not a global convergence theorem for transformer training. After $B = T - \tau$ updates, induction yields

$$e_B \le (1 - \kappa_\tau)^B e_0 + \frac{\epsilon_\tau}{\kappa_\tau}\big[1 - (1 - \kappa_\tau)^B\big]. \tag{8}$$

The initial excess $e_0$ includes target quality at $W_\tau$; it is not equal to $J_\tau$ in general. More precisely, for a fixed reference loss $\ell_{\mathrm{ref}}$, positive-part subadditivity gives

$$[\ell_L(W_\tau, c_*) - \ell_{\mathrm{ref}}]_+ \le [\ell_L(W_\tau, c_0) - \ell_{\mathrm{ref}}]_+ + J_\tau. \tag{9}$$

This separates preparedness under the hold configuration from the incremental cost of the switch. A late hold can improve that state or its recovery rate, but also reduces $B$. Equation 8 identifies the trade-off: release late enough to benefit from new-length training and early enough to adapt to the target; Proposition 2 states the resulting balance condition.

**Proposition 2** (Reserve as a balance point of the budget bound)**.** *Fix $\kappa \in (0,1)$ and $\varepsilon \geq 0$ in Eq. 8, set $q = 1-\kappa$ and $\lambda = -\ln q$, and let the initial excess $e_0(s)$ at a hypothetical release step $s$ be positive and differentiable. For the bound $F(s) = q^{T-s}e_0(s) + (\varepsilon/\kappa)\big(1-q^{T-s}\big)$, every interior critical point satisfies*

$$e_0'(s_*) = -\lambda\big(e_0(s_*) - \varepsilon/\kappa\big). \tag{10}$$

*Proof.* Differentiate: $F'(s) = q^{T-s}\big(e_0'(s) + \lambda(e_0(s) - \varepsilon/\kappa)\big)$, so $F' = 0$ is equivalent to Eq. 10; the prefactor is positive for $s < T$. A sufficient second-order condition for the crossing to be a local minimum is $e_0''(s_*) + \lambda e_0'(s_*) > 0$, i.e. the preparedness curve bends outward faster than the recovery term steepens. □

When the initial excess at the crossing is still above the floor ($e_0(s_*) > \varepsilon/\kappa$), the condition makes the excess decrease in $s$ within this model ($e_0'(s_*) < 0$), and $|e_0'(s_*)| = \lambda(e_0(s_*) - \varepsilon/\kappa)$ states the balance: the marginal preparedness gained by holding one more update in the source configuration equals the marginal value of the last reserved adaptation update. The location $s_*$ therefore depends on the preparedness curve $e_0(\cdot)$ and on the recovery pair $(\lambda, \varepsilon/\kappa)$; it carries no separate dependence on the per-pair displacement $\delta_d$ beyond what enters through $e_0$ and $\varepsilon$. This is a statement about the bound under the modeling assumptions of Eq. 7, not a proof that a particular schedule attains it.

**Recovery time depends on the optimization clock.** When the local rate varies over target updates, replace $\kappa_\tau$ and $\epsilon_\tau$ in Eq. 7 by $\kappa_u \in (0,1]$ and $\epsilon_u \geq 0$. The same unrolling gives

$$e_B \leq e_0 \prod_{u=0}^{B-1}(1-\kappa_u) + \sum_{j=0}^{B-1}\epsilon_j \prod_{u=j+1}^{B-1}(1-\kappa_u), \tag{11}$$

with an empty product equal to one. Under the additional local approximation $\kappa_u \simeq \lambda\eta_{\tau+u} \ll 1$, the first product is approximately $\exp[-\lambda\sum_u \eta_{\tau+u}]$, with accumulated log-error of order $\sum_u \kappa_u^2$ when the rates stay uniformly small. Even in this simplified model, a time constant expressed in optimizer updates depends on update size as well as the state-dependent coefficient $\lambda$. This approximation is not an identified AdamW law or an estimator fitted to the main model.

The 19M decaying-LR traces give descriptive time constants 633/278 updates at release $0.5T/0.7T$; the separate constant-LR pilot fits 17.78/24.15 updates at $0.9T/0.95T$ (Appendix I). These are effective fits under different release states and measurement regimes, not a controlled estimate of an LR effect. Their ratio cannot be used to attribute the difference to LR alone or to transfer a recovery budget. The constant-LR endpoint comparisons instead answer a separate question: the timing advantage persists when late updates do not receive a smaller nominal learning rate.

Fixed-training-window measurements around the 137M custom-YaRN-family release show a sharp initial target-loss reduction followed by non-monotone changes (Appendix O.1). The $0.975T$ arm can first return to its own pre-release source loss sooner than $0.95T$, yet end with worse main 16K PPL. Because the reference changes with release time, this crossing does not identify a common contraction rate or rank endpoints. A further matched-exposure comparison on base 0 enters the same final 200-update target phase from different model and optimizer states: absolute probe loss rises in both arms while their paired gap shrinks (Appendix P.1). Such finite-budget state measurements inform the model's scope without establishing a universal release condition.

## C Additional continuation and length-staging comparisons

A second corpus at the main scale separates corpus transfer from base variance: a fresh 137M WikiText-103 base (identical recipe and budget, 204.8M tokens, same architecture and 2% copy mixture) continued under the same constant-LR custom-YaRN-family protocol gives a static-to-DEPA 16K improvement of $+4.25$ and $+4.43$ PPL on two fresh continuation seeds (mean $+4.34$, $2.3\times$ the $+1.90$ PG19-base-1 mean): the timing benefit is not specific to PG19 and is larger on the second corpus, whose static added-context harm is also larger (natural-16K utility $-0.57$ nats static versus $-0.08$ under DEPA, an 86–87% harm reduction). The 8K margins are positive ($+0.96$/$+0.98$), the 4K trained-length cost stays below $0.05$ PPL, and continuation cells are digest-matched within each seed pair. The evidence spans three pretrained bases and two corpus families at the main

scale, plus a separate different-architecture base (Appendix L); additional band-curriculum baselines are in Appendices D.3 and G, and the schedule controls (full-budget ramp, equal-exposure tail ramp, end-only conversion) are in Appendices M.1 and P.1; the source-hold and release-loss traces (Appendices P.5 and P.2) do not identify a weight-level cause of the endpoint advantage.

### C.1 Constant-learning-rate comparisons at 19M and 137M

The per-seed 19M values appear in Table 3 in Section 5.3.

All three seeds preserve the ordering static $>$ staggered $>$ middle release $>$ late release, with a mean static-to-DEPA improvement of 19.22 PPL (19.2%; paired gains 19.18/19.31/19.18) and smaller gains over staggered (7.07) and middle (4.77) release. Since the learning rate is constant, these gains cannot be attributed solely to a smaller LR at the switch; they include a different model state and a different allocation of source/target updates. This experiment does not imply that learning-rate choice has no interaction with DEPA.

**Direct main-scale constant-LR evidence.** At 137M PI, constant-LR triads on seeds 1 and 2 retain the large static-to-DEPA 16K improvement (160.058/162.091 to 28.597/28.683), with 0.959/1.012-PPL trained-length costs, and DEPA beats the tail-linear ramp at all lengths despite equal LR-weighted exposure (Appendix M.3). The constant-LR PI triad replicates on the second base, and custom YaRN-family retains its static-to-DEPA gain at constant LR; an additional custom-YaRN-family pair on the second base (seed 5) improves 16K PPL from 30.307 to 28.395 (6.31%) at a 0.025-PPL cost. On matched natural-text targets, most long-input gain is a reduction of added-context harm, not a demonstrated retrieval improvement (Appendix M.4).

Beyond the token-matched constant-LR comparisons, five continuation seeds (5–9) on base 1 compare fixed DEPA with static at constant LR: at 16K the gain is $+1.903 \pm 0.078$ PPL at a $+0.026 \pm 0.005$-PPL 4K trained-length cost (mean$\pm$sample s.d. across five continuation seeds); the effect and its sign are stable across seeds, and the 16K gain is $\sim 70\times$ the 4K cost. Identical-target penalties, matched-window replications, and an exposure-matched nonlinear cosine tail (four paired comparisons on both bases) show no uniform dominance across protocols; the full length-by-timing and hold-share matrices are in Appendix P.1.

Aggregating the natural-text context-utility measurement (full versus 1K-local suffix on identical scored tails) over every evaluated cell—119 cells spanning the token-matched fixed-length and staged families on both budgets, the hold-share, share-reversal, same-share, share-drop and base-hop arms, the post-release cross-$2K$ pair, the binary probe, the temperature probes and the release-fraction sweeps—shows no positive-utility regime: utility is near zero at the trained length ($\pm 0.0049$ nats across the cells trained at $4K$) and negative at 8K and 16K in every one of the 119 cells. The following aggregate is descriptive across heterogeneous families, not a pooled treatment-effect estimate: the mean DEPA-over-static utility gap grows from $0.0010$ at the trained length to $0.0369$ at $8K$ and $0.2639$ at $16K$, where the mean utility is $-0.0778$ for the DEPA-release cells versus $-0.3416$ for the static reference cells (a harm reduction of $77.24\%$), and at the $2\times$ operating point DEPA is near context-neutral ($-0.0171$). These measurements show reduced added-context harm, without establishing positive added-context value. A capability transfer beyond perplexity is not demonstrated (the retrieval probe is null; Appendix L).

A token-aligned gradual-length control at the main 137M scale is complete as a digest-matched two-by-two factorial (length fixed or staged; timing static or DEPA $0.9T$; constant LR) on both bases: the timing benefit transfers (16K gains $1.80/1.59/1.63$) and the interaction is tail-concentrated (positive at 8K, negative only at 16K) (Appendix P.1).

**Continuation-mixture ablation.** Removing the 2% copy-task batches preserves the constant-LR timing gain: PI 16K PPL improves $158.370$ to $28.431$ and custom YaRN-family $29.267$ to $27.346$, with trained-length costs $0.939/0.019$. This rules out the necessity of copy batches during continuation in these pairs, while retaining the original pretraining mixture (Appendix N.1).

**Refining the late release.** Fixed $0.9T$ is our main setting; release refinement was not identified as a robust policy in these comparisons. Nearby release fractions, a cosine tail, and base-refinement/boundary-replication checks produced only checkpoint- and base-specific effects, includ-

ing a 16K sign change across seeds on the independent base and no distinct adaptive advantage for the state-based candidates; the full paired tables are in Appendices N.3 and N.6.

### C.2 Further evidence and schedule controls

The custom target also retains the timing improvement at 19M, and earlier 19M band-order controls support a purposeful hold-and-release schedule over repeated positional variation without exhausting possible regularization mechanisms. The 19M table, the 68M/19M band-curriculum results, and the length-dependent pattern are in Appendix G.

**An efficiency lever with a bounded tail cost.** Because every update keeps 32768 prediction targets, the staged schedule's attention component is sequence-length-weighted while the token-proportional work is unchanged; the measured wall time is $0.831$–$0.832\times$ fixed, and the attention-load arithmetic plus the staged-versus-fixed dilution ($-0.20$/$-0.40$ PPL at 16K at a small $+0.077$–$+0.140$ base-1 staging-alone cost) are in Appendix D.2. Within that bounded dilution the staged curriculum is a defensible compute-saving recipe; it does not free the release schedule of its dependence on the hold's long-context readiness.

**Hold exposure under a compute constraint.** Length-staging experiments measure how the delayed-release margin responds to the hold's long-context exposure. Indexing a tested schedule by the hold's 4K share $S$ (fraction of the release-preceding updates acting on the deployment length; $1.0$ = fixed length), raising the share from $0.444$ to $0.722$ cuts the 16K diluted margin from $-0.40$ to $-0.10$ PPL at $0.92\times$ recorded wall time on the independent base (up from $0.831\times$ for the default staged schedule). The measured points admit a descriptive superlinear interpolation whose shape is sensitive to base and point selection, and a linear alternative is not statistically rejected at the family's residual scale; the schedule points, the interpolation, a dose-matched order contrast, and the target- and budget-dependent reserve response are reported in Appendix E (Eq. 12, Fig. 5 and Table 12). None of these contrasts establishes share sufficiency or a universal share-response law.

The delayed-release margin also depends on the reserve size, and a same-mask mid-strength target separates which schedule property sets that reserve: the temperature-free variant of our YaRN-family target (identical wavelength-mask geometry, without the attention-temperature factor; two fresh base-1 seeds) anchors static 16K at $30.80$/$30.20$ PPL, between custom-YaRN-family's extrapolation and PI's, and its delayed-arm 16K gain grows monotonically as the reserved budget is increased ($1.9325$ to $1.9788$ and $1.8479$ to $1.8987$ PPL), so displaced-pair count alone does not determine the useful reserve in these comparisons. Temperature, target quality, and optimization state remain coupled.

The benefit is concentrated beyond the trained length. On the independent base, custom YaRN-family has a 4K PPL cost of $0.026 \pm 0.005$ versus a 16K gain of $1.90 \pm 0.08$. PI's 4K cost is larger (0.96–1.4 PPL), but its mask and target quality both differ, so this comparison cannot identify a per-pair displacement cost. Across the tested custom-YaRN-family schedules, 8K gains are positive and 4K gains slightly negative; interpolated crossings are approximately $1.04$–$1.10L_1$ across the two budgets. Thus applications near $L_1$ must account for an in-window cost. At the larger budget, custom YaRN-family's margin grows without a deteriorating static endpoint. PI differs: static 16K PPL worsens from $159.9$ to $191.9$, while DEPA improves from $27.6$ to $25.6$. Its enlarged margin therefore cannot be treated as evidence of scalable DEPA capacity (Appendix P.3).

**State and budget in a release rule.** A rule that monitors target-configuration NLL on the hold's fixed 16K training windows and reserves at least 200 target updates chooses a deadline on both tested bases, but its matched fixed controls show no distinct adaptive timing decision; neither a training-loss crossing nor a transferred recovery constant supplies a criterion that improves on fixed schedules (Appendix O.3).

## D Implementation and target definitions

The reported endpoints and plotting inputs are retained in per-run result files and execution logs. In the earliest runs, the saved configuration also lists a nominal release deadline; the release that

was executed, and that is reported here, follows each run's release schedule. The published-YaRN seed-44 static and delayed result files were reconstructed verbatim from their execution logs because the original files were not retained; the files record that provenance. The seed-45 files were written directly at the end of training.

Continuation starts from fixed pretrained checkpoints; each schedule pair shares its base weights and training-data source. The earliest experiments predate the input/target-stream digest instrumentation of the later paired cells; wherever that affects a comparison, the difference is stated next to the numbers. Raw execution records and source-revision hashes are withheld from the anonymous submission. Table 10 is an anonymous run ledger for the headline cells: each row names an arm by experimental family, base, and continuation seed, gives the target formulas it trains under (the schedule identity, the custom mask, and the custom attention temperature (2)), the evaluation target used for the reported number (main 16K PPL, plus natural-context where cited), and the attention scale recorded by the run, so the reported endpoints can be re-derived from the recorded logs even without training new runs.

### D.1 Custom and published YaRN-family target definitions

The frequency multipliers of the custom target follow our wavelength-mask implementation of YaRN's moved-set idea, with the same number of moved pairs as the published threshold rule but a mask of our own construction: $\mu_d^*$ is 1 for the 25 preserved pairs and $1/4$ for the seven longest-wavelength pairs, with the breakpoint set by our implementation's threshold (Table 33; $\mu_d^* > 1$ is not used; the mask interpolates the moved pairs toward their long-range target). The attention temperature is Eq. 2, $a_* = 0.87825$ at $L_1/L_0 = 4$, applied to the QK logits (every paired custom-target arm records this attention scale) from the first continuation update and held fixed across arms, so the schedule, not the temperature, varies. For reference, the published recipe (Peng et al., 2023) defines the moved set and blend through $r(d) = L_0/\lambda_d$, $\lambda_d = 2\pi/\theta_d$, the ramp $\gamma(r)$ with $\alpha = 1$, $\beta = 32$, and the blend $h(\theta_d) = (1 - \gamma(r(d)))\,\theta_d/s + \gamma(r(d))\,\theta_d$ with scale factor $s = L_1/L_0$ (their Eqs. 8, 10, 11, 13), and applies the temperature $\sqrt{1/t} = 0.1 \ln s + 1$ to the QK logits, i.e. the multiplier $(0.1 \ln s + 1)^2 \approx 1.296$ at $s = 4$ (their Eqs. 14–15) — super-unity and opposite in direction to $a_*$. The two definitions are numerically separate; the temperature-timing probe of Section 3 compares temperature schedules, not the two definitions of $a_*$.

**Temperature-timing probe.** A $2 \times 2$ probe crosses the frequency schedule (static or DEPA $0.9T$) with the temperature schedule: $a_*$ from update 0, as in every other custom-target run, or the source temperature 1 until update 3600 and $a_*$ afterward, so that in the DEPA arm the temperature switches together with the frequencies (the joint hold). All four arms use PG19 base 1, continuation seed 12, the custom YaRN-family target, constant LR, $T = 4000$, and 131.072M tokens with identical recorded input/target-stream digests. At 16K they reach 30.210 PPL (static, $a_*$ from update 0), 28.229 (DEPA, $a_*$ from update 0), 30.304 (static, switched temperature), and 28.342 (joint hold). Relative to the first arm, DEPA gains 1.98 PPL with $a_*$ from update 0 and 1.87 PPL under the joint hold, so temperature timing accounts for 0.11 PPL of the gain. The temperature-by-frequency interaction, the joint-hold gain minus the two single-factor gains, is $-0.02$ PPL.

### D.2 Experimental families, run ledger, and training details

Table 9 summarizes the core experimental families and Table 10 lists the headline runs. The paragraphs below give the efficiency of the length curriculum, the release rule, the 137M architecture, and the training and evaluation settings.

**Efficiency inside the length curriculum (supporting detail).** The staged schedule offers an efficiency trade-off while retaining the timing benefit. Because every update keeps 32768 prediction targets, the attention component is sequence-length-weighted while the token-proportional MLP/embedding work is unchanged; the staged attention load is $(1000 \cdot 1024 + 1000 \cdot 2048 + 2000 \cdot 4096)/(4000 \cdot 4096) = 0.69\times$ the fixed-length load (protocol arithmetic, not a timing measurement). The recorded wall times are reproducible across the two bases and the two fresh seeds (1838–1844 s staged versus 2211–2217 s fixed, i.e. a 0.831–0.832$\times$ factor, repeating to within seconds within a seed). That wall factor lies *above* the attention-load ratio ($0.69\times$): the wall *saving* of $\approx$17% is smaller than the attention-ratio saving of 31%, because only the sequence-length-weighted attention cost

Table 9: Core experimental families. Replications on constant-LR base-1 seeds, the WikiText-103 and 124M bases, and published YaRN are specified in their own appendices. Budgets are continuation tokens, rounded. Seeds denote continuation seeds for the static/DEPA comparisons; replication across pretrained bases is in Appendix P.4. S1 denotes the 19M constant-LR WikiText-103 study of Table 3.

| Family | Data | $L_0 \to L_1$ | Budget | Replication |
|---|---|---|---|---|
| 137M PI | PG19 | $1024 \to 4096$ | 131M | seeds 0–2 |
| 137M custom YaRN-family | PG19 | $1024 \to 4096$ | 131M | seeds 0–2 |
| 19M custom YaRN-family | WikiText-103 | $128 \to 512$ | 82M | seed 0 |
| Constant LR (S1) | WikiText-103 | $128 \to 512$ | 82M | seeds 0,1,2 |

Table 10: Anonymous run ledger for the headline cells. Each row gives the arm (family/base/seed; 20L/8H/640E is the 124M architecture with 20 layers, 8 heads, and width 640), the schedule and target identity it trains under (ramp$_{20\%}$: the equal-exposure tail-linear ramp over the final 20% of updates), the evaluation target of the reported number (nat: natural-context utility), and the attention scale recorded by the run: $0.87825$ in every custom-target arm, $1.29648$ in every published-YaRN arm, and none (—) for PI. Published-YaRN arms combine NTK-by-parts frequencies with the published attention multiplier $(0.1\ln(L_1/L_0)+1)^2$ at $L_1/L_0 = 4$ (Appendix D.1). Any row's 16K PPL can be re-derived from the logged endpoint without new training.

| Arm | Schedule | Target | Eval | Attention scale |
|---|---|---|---|---|
| base-0 custom target, s1 | static / DEPA $0.9T$ | custom mask + $a_*(2)$ | 16K PPL/nat | 0.87825 |
| base-1 custom target, s2 | static / DEPA $0.9T$ | custom mask + $a_*(2)$ | 16K PPL/nat | 0.87825 |
| 20L/8H/640E, s48 | static / DEPA / ramp$_{20\%}$ | custom mask + $a_*(2)$ | 16K PPL/nat | 0.87825 |
| 20L/8H/640E, s49 | static / DEPA / ramp$_{20\%}$ | custom mask + $a_*(2)$ | 16K PPL/nat | 0.87825 |
| base-1 PI (no temp), s2 | static / DEPA $0.9T$ | PI ($\mu_d^* = L_0/L_1$, no temp) | 16K PPL | — |
| base-1 PI (no temp), s44 | static / DEPA $0.9T$ | PI ($\mu_d^* = L_0/L_1$, no temp) | 16K PPL | — |
| base-1 PI (no temp), s45 | static / DEPA $0.9T$ | PI ($\mu_d^* = L_0/L_1$, no temp) | 16K PPL | — |
| base-1 published YaRN, s44 | static / DEPA $0.9T$ | NTK-by-parts+$(0.1\ln 4+1)^2$ | 16K PPL | 1.29648 |
| base-1 published YaRN, s45 | static / DEPA $0.9T$ | NTK-by-parts+$(0.1\ln 4+1)^2$ | 16K PPL | 1.29648 |
| base-1 published YaRN, s46 | static / DEPA $0.9T$ | NTK-by-parts+$(0.1\ln 4+1)^2$ | 16K PPL | 1.29648 |

decreases while the token-proportional MLP/embedding work is unchanged, i.e. the recorded wall times bracket the protocol attention load from above. Against this saving, the staged hold dilutes the delayed-release 16K margin by $0.20/0.40$ PPL (10–20% of the $+2.0/+1.9$ PPL benefit) and staging alone adds only a small main-16K cost ($-0.013$ base 0, $+0.077$ to $+0.140$ base 1), while the 8K timing gain survives at or above the fixed-length level. The staged-length curriculum is therefore a defensible compute-saving recipe when the longest-tail margin can absorb a bounded dilution; it does not free the release schedule of its dependence on the hold's long-context readiness.

**Specification of the state-and-budget release rule.** Fixed $0.9T$ is the established baseline. A rule that monitors target-configuration NLL on two fixed 16K training windows during the hold, releases if that signal stops improving at $0.9T$ or $0.925T$, and otherwise releases at $0.95T$ to reserve at least 200 target updates is evaluated in Appendix O.3. The reserve is a design bound, not an estimated recovery time. The rule chooses the deadline on both tested pretrained bases; matched fixed controls show no distinct adaptive timing decision. A useful extension must combine state and a remaining target budget that improves held-out outcomes over fixed schedules. Neither a training-loss crossing nor a transferred recovery constant has yet supplied that criterion.

**137M architecture and continuation specification.** The PG19 model has 14 pre-layer-normalized decoder blocks, embedding width 768, 12 attention heads with head dimension 64, and a four-times-width GELU feed-forward layer. The vocabulary has 50,257 GPT-2 BPE tokens; input and output embeddings are tied. RoPE uses base 10,000, 32 rotary pairs per head, and eight contiguous frequency bands; dropout is zero. The shared source checkpoint uses seed 0, length 1024, batch size 8, and

25,000 updates (204,800,000 token positions). Each continuation uses length 4096, batch size 8, and 4,000 updates (131,072,000 token positions), loading source weights and initializing a fresh AdamW optimizer. A release at $0.9T$ occurs before continuation update 3600, leaving 400 target updates.

The base-model launch arguments use learning-rate parameter $3 \times 10^{-4}$ and 1500 warmup steps. The training implementation uses AdamW with $\beta = (0.9, 0.95)$, weight decay 0.1, and global gradient-norm clipping at 1. Its learning rate warms up linearly, then decays linearly to 10% of the parameter value over the configured stages. Continuation starts at global step 25,000 of 29,000, without restarting warmup; its first rate is approximately $6.93 \times 10^{-5}$ and its last approximately $3.00 \times 10^{-5}$. These values describe the training implementation and the full-precision replication; the launch arguments alone do not determine all source-code defaults.

The base artifact records the default 0.02 probability of substituting a synthetic copy-task batch for a corpus batch, with three copy markers. The primary continuation scripts retain this default; the explicit no-copy ablation in Appendix N.1 sets it to zero. Thus the primary reported budgets count all training token positions, including this mixture, and should not be read as an exclusively PG19-token count. This setting is shared across paired schedules. Disabling the trainer's passkey (copy-retrieval) evaluation does not disable the training mixture.

For the full-precision replication, validation uses 40 batches of randomly positioned contiguous windows per length. Parallel windows per batch equal $\max\{1, \min\{8, \lfloor 32768/L \rfloor\}\}$, giving 1,310,720 scored positions at each of 4096, 8192, and 16384 tokens. Window starts use the same seed within each pair and may overlap; perplexity is the exponential of aggregate token NLL. This is sampled-window evaluation, not a full-corpus sliding-window pass. The custom-YaRN-family attention scale is active throughout continuation in both arms; only frequency timing differs.

The supplementary material accompanying the submission contains this training implementation and its dependencies, the experiment-specific trainers, and the exact paired launch commands of every experiment whose per-run results it includes; it documents the implementation used for the full-precision replications.

The 19M constant-LR WikiText-103 study (S1; Table 3) specifies six transformer layers, eight attention heads, embedding width 256, and head dimension 64. In this implementation query/key/value projections use $n_{\text{head}} \times d_{\text{head}}$ channels, so embedding width need not equal that product. S1 uses batch size 32, 5000 continuation steps at length 512, and the shared pilot base after 12000 steps at length 128. The staggered arm uses index-ascending order, release fraction 0.5, 400-step cosine ramps, and linear spacing. Late arms switch at fractions 0.5 and 0.9. All four use the same final YaRN-style mask and constant LR; the three seeds vary continuation randomness.

PI target comparisons have no temperature change. In custom-YaRN-family experiments, the trainer sets the attention scale by continuation stage, not by release step: the correction is active from the first continuation update in all arms. This holds temperature fixed within the custom target comparison while frequency-release timing changes. Inference uses the target configuration for all schedules.

### D.3 Band-curriculum comparisons at one-decimal precision

The comparisons in Table 11 add band-curriculum baselines and seed coverage. PI improves from static 174.2 and staggered 39.4 to DEPA 29.2; custom YaRN-family also favors DEPA over fixed and adaptive band curricula on seeds 0 and 1. These results keep their logged one-decimal precision and are reported separately from Table 1. For scale, the static PI gap in Table 11 between $4L_1$ and $L_1$ is 146.5 PPL, compared with 2.9 for custom YaRN-family seed 0. This is descriptive headroom, not a universal scaling law.

## E Supporting derivations and residual analyses

This appendix collects the analytical details and diagnostic checks behind the measured share response of Section 5.5.

**Floor-gap calibration and the share-to-attention mapping.** The final-update floor-gap measurement correlates with the endpoint 16K margin ($r^2 = 0.98$ over five cells). Because this measurement is available only after training, it is a retrospective consistency check, not a prospective

Table 11: 137M PG19 continuations with band-curriculum baselines (values logged at one decimal): identical final target and 131M continuation tokens within each target block. The PI block has continuation seed 0 only; custom YaRN-family has paired seeds 0 and 1. The lowest 16K PPL within each target and seed block is in bold. FC denotes a band-wise frequency curriculum with a staggered (fixed) or an adaptive band schedule.

| | | | PPL at length (tokens) | | |
|---|---|---|---|---|---|
| Target | Seed | Schedule | 4096 | 8192 | 16384 |
| PI | 0 | Static | 27.7 | 69.9 | 174.2 |
| | | Staggered FC | 27.8 | 31.4 | 39.4 |
| | | DEPA $0.9T$ | 29.1 | 30.0 | **29.2** |
| Custom YaRN-family | 0 | Static | 27.8 | 29.8 | 30.7 |
| | | Staggered FC | 27.8 | 29.4 | 29.1 |
| | | Adaptive FC | 28.1 | 30.0 | 29.9 |
| | | DEPA $0.9T$ | 27.9 | 29.2 | **28.6** |
| | 1 | Static | 28.2 | 28.0 | 29.1 |
| | | Staggered FC | 28.2 | 27.6 | 27.5 |
| | | Adaptive FC | 28.5 | 28.5 | 28.8 |
| | | DEPA $0.9T$ | 28.2 | 27.5 | **27.1** |

release criterion. The corresponding release-time measurement does not separate the schedules (Appendix O.1); a useful state-based selector remains unestablished. For the equal-short-segment staging sub-family actually run in the decisive schedules (two short phases of equal length, $e_2 = 2e_1$, i.e. $\alpha = n_1/(n_1 + n_2) = 1/2$), the share maps to the attention ratio by $a(S) = 1 - 0.5625(1 - S)$ (protocol arithmetic; the general form with unequal short phases is $a = 1 - (\tau/T)(\frac{1}{2} + \frac{1}{4}\alpha)(1 - S)$, which gives the hold-70 point $\alpha = 0.4$, $S = 2600/3600$, $a = 0.85$, while the equal-family coefficient is exactly $0.5625$).

**Descriptive interpolation of the measured share points.** The four anchor points (default staged at seeds 5/6, hold-70 at seeds 7/8, hold-85 at seed 9) admit the superlinear interpolation

$$D(1 - S) \;\approx\; -0.98\,(1 - S)^{1.66}, \tag{12}$$

which is fit to those points on the independent base and reported here as a descriptive guide, not as an identified scaling law: its shape is sensitive to base and point selection (dropping the default-staged anchor changes the exponent to 1.14; substituting the other base's same-share point gives 1.05), and a linear fit through the same points is not statistically rejected at the family's residual scale ($\chi^2 = 7.63$ on 3 d.f., $p \approx 0.05$; it is rejected at within-cell seed precision, and an equal-parameter AIC comparison prefers the power form by 2.11). Restricting Eq. 12 to the equal-family framing,

$$D(a) \;\approx\; -0.98\left(\tfrac{1-a}{0.5625}\right)^{1.66}, \tag{13}$$

tracks the four measured points to within $0.04$ PPL: at $a = 1.00/0.92/0.85/0.69$ it gives $D = 0/-0.04/-0.11/-0.36$ against the measured $0/-0.05/-0.10/-0.40$, where the hold-70 $a = 0.85$ is the schedule's protocol attention load, not the $\alpha = 1/2$ value ($0.844$). The largest residual exceeds the $0.006$-PPL per-seed spread at that share, so the equation remains a descriptive interpolation rather than a per-point reproduction. The signed diluted margin $D$ is concave in the saved exposure $1 - S$; the *absolute* diluted-margin cost $|D|$ is the convex quantity in the attention savings: with the refit exponent the marginal 16K cost per unit attention saved grows from ≈0 near $a = 1$ to ≈0.80 PPL/unit at $a = 0.92$ and ≈1.95 PPL/unit at $a = 0.69$. These marginal values are operations on the descriptive fit, not an empirically identified curvature law (the exponent's uncertainty and base sensitivity are reported with the fit).

Table 12 lists the measured schedule points $S = 1.000/0.850/0.722/0.444$, with protocol attention ratios $1.00/0.92/0.85/0.69$ respectively; the T= $8000$ rows repeat the same shares at double the continuation budget (fresh seed 14, release still at $0.9T$) within the same $50\%$ per-point tolerance (descriptive; no equivalence or null claim). The four schedule points (default staged at seeds 5/6, hold-70 at seeds 7/8, hold-85 at seed 9) admit the descriptive superlinear interpolation of Eq. 12;

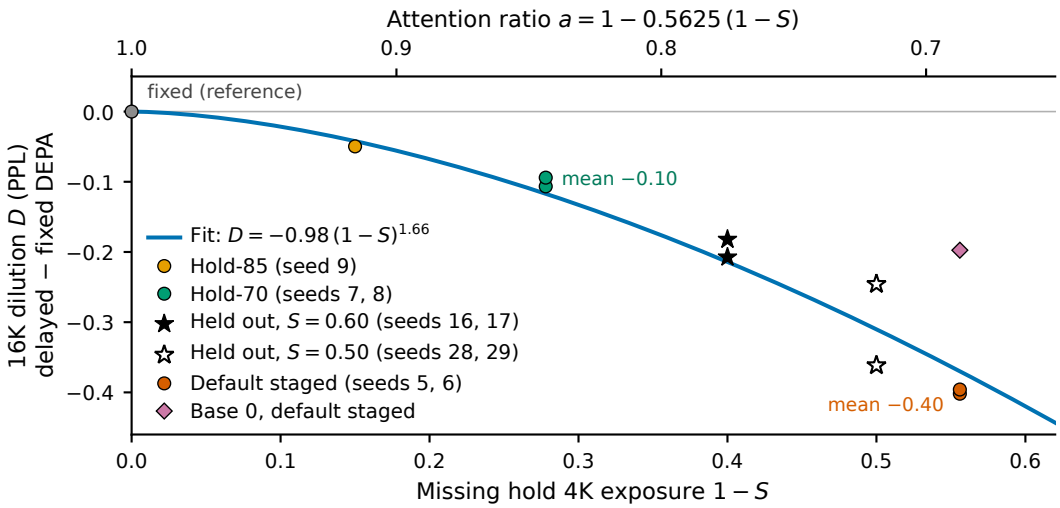



Figure 5: Measured share response on the independent base: the 16K delayed-release diluted margin $D$ as a function of the missing hold 4K exposure $1-S$ (per fresh seed), with a descriptive interpolation shown as a guide and the attention ratio $a$ of the equal-short-segment sub-family on the top axis (ticks evenly spaced in $a$; the measured schedules sit at $a = 0.92/0.85/0.69$). Filled circles are the five fit anchors (hold-85 seed 9, hold-70 seeds 7/8, default staged seeds 5/6); no uncertainty bands are drawn (one value per seed). Restoring the hold's long-context share from $0.444$ to $0.722$ cuts the diluted margin from $\approx -0.40$ to $\approx -0.10$ PPL and returns the 16K length-by-timing interaction to near zero on two seeds. The base-0 default-staged point (diluted margin $\approx -0.20$) is drawn for transfer context at the same share. Two interior schedule points never used in the in-sample fit are compatible with the measured points: $S = 0.600$ (stars, two fresh seeds; diluted margin $-0.19$ vs. interpolated $-0.21$) and $S = 0.500$ (open stars, two fresh seeds; diluted margin $-0.30$ vs. interpolated $-0.31$). Cross-budget points at $T = 8000$ (diluted margin $-0.33$ at $S = 0.444$ and $-0.06$ at $S = 0.722$, open squares in Figure 15; $-0.18$ at $S = 0.600$, reported with that figure) sit within a $50\%$ per-point tolerance fixed before the $T = 8000$ runs, at double the token budget (descriptive; one fresh seed per share, no equivalence or null claim).

Table 12: Measured length-staging schedule points on the independent base. $S$ is the hold's 4K share; dilution is the 16K delayed-release margin relative to the fixed-length DEPA reference (negative = the fixed reference is better); the recorded wall ratio is relative to the fixed schedule on the same continuation run. Each cell holds one value per continuation seed (staged seeds 5/6, hold-70 seeds 7/8, hold-85 seed 9, T=8000 rows seed 14); Section 5.5 and Figure 5 quote the family seed means ($-0.40$ and $-0.10$). The recorded wall ratio of a schedule is the mean recorded wall time of its two length arms (static and $0.9T$-release) over the mean of its two fixed-schedule arms on the same continuation run. — marks a wall ratio that was not recorded. None of these values establishes share sufficiency or an identified share-response law.

| Schedule | $S$ | 16K dilution (PPL) | recorded wall ratio |
|---|---|---|---|
| Fixed | 1.000 | 0.00 | 1.000 |
| Hold-85 | 0.850 | −0.05 | 0.953 |
| Hold-70 (compressed 1K/2K) | 0.722 | −0.11/−0.09 | 0.921/0.919 |
| Default staged | 0.444 | −0.40/−0.40 | 0.831/0.832 |
| T=8000 default | 0.444 | −0.33 | — |
| T=8000 hold-70 | 0.722 | −0.06 | — |

across these points the 16K diluted margin reacts faster than linearly to the hold's short-stage deficit, without rejecting a linear-in-$(1 - S)$ alternative at the family's residual scale (descriptive). Two interior points held out of the fit are compatible with the measured response: $S = 0.600$ (fresh seeds 16/17, diluted margin $-0.19$ vs. interpolated $-0.21$) and $S = 0.500$ (fresh seeds 28/29, diluted

margin $-0.30$ vs. interpolated $-0.31$). These held-out checks tighten the descriptive reading of the curve; they do not turn it into an identified divergence exponent.

**Reserve window, release-step controls, and the copy mixture.** The useful reserve depends on the tested target and budget. At $T = 4000$, custom YaRN-family is best at a 100–200-update reserve at 16K among the measured fractions, whereas PI is best at $400$ updates. PI also favors $400$ reserved updates in its $T = 8000$ comparison (25.64 versus 25.82 PPL). These comparisons do not establish invariance to target displacement: PI and custom YaRN-family differ in both their frequency masks and the quality of their static endpoints. The reserve response is non-monotonic: releasing at the midpoint (a $2000$-update reserve) is consistently worse than reserving $200$, because a long source-frequency hold itself builds readiness; the reserve is the balance of hold exposure against target adaptation, not a resource to maximize (Proposition 2). The window also shrinks with target strength: the strong custom-YaRN-family target (static natural-16K added-context harm $\approx -0.3$ nats) is best served by the smaller 100–200-update reserve, while the weak PI target (harm $\approx -3$ nats, roughly $10\times$) is best served by the larger $400$-update reserve. PI differs from custom-YaRN-family in both the number of displaced pairs and the target strength, so the two axes cannot be separated from these two families alone. A protocol-matched mid-strength target from the same mask family separates them: the temperature-free variant of our YaRN-family target (identical wavelength-mask geometry, without the attention-temperature correction; two fresh seeds, base 1, $T = 4000$) anchors at $30.80$ and $30.20$ PPL static 16K (between custom-YaRN-family's ≈28.9 and PI's ≈160), and its delayed-arm 16K gain over static rises monotonically across the measured 100–400-update window on both seeds ($1.9325$ to $1.9788$ PPL, $0.046$; $1.8479$ to $1.8987$ PPL, $0.051$), leaving the gain largest at the biggest measured reserve instead of forming custom-YaRN-family's flat band. Because the mask geometry is held equal to custom-YaRN-family's, the reserve preference tracks target strength (added-context harm) rather than the displaced-pair count. The prediction of the reserve-in-updates reading—that a custom-YaRN-family fraction sweep at $T = 8000$ keeps the 16K optimum at a few hundred updates—is consistent with the measured response, which is flat from $\approx 200$ to $\approx 800$ reserved updates (27.73–27.75 PPL, spread below seed noise) and only the $1600$-update point trailing (Figure 4b); the optimum is a broad flat band rather than a sharp peak, and the useful operational statement is a reserve of roughly 100–800 target updates. A protocol-matched close-fraction repetition at the original budget on the independent base ($T = 4000$, two fresh seeds) confirms this: the 400–200-update transition is flat at 16K within seed noise (max $0.002$ PPL), the $100$-update reserve trails by $\approx 0.028$ PPL on both seeds in the main-PPL readout and shows a smaller, same-signed cost in 32-shared-window nested NLL, while the separately scored fixed-window natural NLL assigns no cost. These are descriptive two-seed paired contrasts within the flat band, not an equivalence or a universal optimum. A protocol-matched two-step staged release (an intermediate multiplier halfway in $\mu$-space at $0.5T$, the full target at $0.9T$; an abstraction of staged recipes such as Code Llama's, run as a measured arm under the matched configuration) replicates the single late-switch 16K margin within $0.08$ PPL ($1.82$ versus $1.90$, digest-matched, one fresh seed), so additional release steps provide no gain over the single late switch at this budget. The 2% copy-task mixture is part of pretraining but is ablated from continuation without removing the timing gain (Appendix N.1), so the mixture is not the carrier of the hold benefit in the delayed arms. In Figure 4b, the PI points use the decaying-LR schedule and the custom-YaRN-family points use constant LR; target strength and displacement are also confounded across these two families, so that comparison alone does not isolate either cause of the reserve difference (the mid-strength probe above does).

## F Frequency-band geometry and sensitivities

Let $P = D/2$ and $\lambda_d = 2\pi b^{d/P}$, $b > 1$. The first pair with wavelength strictly greater than $x > 0$ is the clipped integer

$$d_>(x) = \min\{P, \max\{0, \lfloor P \log_b(x/(2\pi)) \rfloor + 1\}\}. \tag{14}$$

For a non-strict $\lambda_d \geq x$ threshold, replace floor-plus-one with ceiling. This distinguishes the boundary case exactly. The number of pairs with $\lambda_d \leq x$ is $d_>(x)$; the number newly crossing between $L_0$ and $L_1$ is $d_>(L_1) - d_>(L_0)$. For $P = 32, b = 10000$, the thresholds at 128 and 512 are 11 and 16, so this interval contains five pairs, $11, \ldots, 15$.

The implementation's YaRN-style moved-set threshold is near $2\pi L_0$, and its reported moved fractions are 47%, 38%, and 22% at $L_0 = 128, 256, 1024$. These counts describe this particular mask; they

are not a claim that any smooth positional target has the same threshold. A wavelength census counts geometry, whereas learned query/key weights determine its effect on predictions. Short wavelengths can encode informative local offsets despite wrapping, and multiple pairs jointly encode position.

A band-reset intervention evaluates a continued checkpoint after returning a selected frequency band to its source setting. Such an intervention measures the checkpoint's dependence on that positional change; effects of separate resets need not add because the model combines the bands. Equation 4 bounds a local logit change for small phase displacement and bounded query/key norms. It does not imply that long-wavelength bands are universally irrelevant.

## G SUPPORTING BAND-CURRICULUM TRANSFER RESULTS

Tables 13 and 14 report supporting band-curriculum results at 68M on WikiText-103 and at 19M on PG19. These results support positional timing as a relevant factor across scale and corpus. They do not constitute direct fixed-$0.9T$ replications.

Table 13: Supporting 68M WikiText-103 results, $256 \rightarrow 1024$, 8000 continuation steps (about 196M tokens). YaRN here has no attention-temperature correction; FreqCurv denotes a band-wise frequency curriculum toward YaRN in ascending or descending band order.

| Method | 1024 | 2048 | 4096 |
|---|---|---|---|
| PI | 24.9 | 109.7 | 421.1 |
| Source RoPE | 25.0 | 26.1 | 39.0 |
| Static YaRN | 25.0 | 24.5 | 26.8 |
| FreqCurv YaRN, ascending, seed 0 | 25.0 | 24.3 | 24.7 |
| FreqCurv YaRN, ascending, seed 1 | 25.0 | 24.0 | 24.1 |
| FreqCurv YaRN, descending, seed 0 | 24.9 | 24.3 | 25.1 |

Table 14: Supporting 19M PG19 band-curriculum comparison, $256 \rightarrow 1024$, 6000 equal-budget continuation steps; temperature-free YaRN-family target, single seed.

| Method | 1024 | 2048 | 4096 |
|---|---|---|---|
| PI | 49.2 | 100.2 | 211.1 |
| Source RoPE | 48.9 | 54.6 | 77.3 |
| Static YaRN | 48.8 | 48.4 | 58.9 |
| FreqCurv YaRN, ascending | 48.9 | 47.8 | 50.0 |

Table 15: 19M custom-YaRN-family comparison, WikiText-103, equal 82M continuation tokens, seed 0.

| Schedule | 512 | 1024 | 2048 | 4096 |
|---|---|---|---|---|
| Static | 58.89 | 57.15 | 65.12 | 92.62 |
| Staggered FC | 58.93 | 56.69 | 58.15 | 75.90 |
| DEPA $0.9T$ | 59.00 | 56.83 | **57.37** | **70.49** |

The custom target also retains the timing improvement at 19M (Table 15), with DEPA reducing PPL by 11.9% at $4L_1$ and 23.9% at $8L_1$. In this family staggered release is slightly better at $2L_1$, whereas DEPA is better at the harder lengths; the useful pattern is length-dependent. Temperature-free 19M controls find that one late release outperforms the compared band orders, small per-step jitter gives little improvement, and four source-to-target cycles worsen $4L_1$ PPL to 78.8 versus 65.6 for static. A separate gradual-length control does not reproduce the positional curriculum gain. These observations support a purposeful hold-and-release schedule over repeated positional variation; they do not exhaust possible regularization mechanisms.

## H RECOVERY INTERPRETATION

For $r = 1 - \kappa_\tau$, iterating Eq. 7 gives $e_B \le r^B e_0 + \epsilon_\tau \sum_{j=0}^{B-1} r^j$, which is Eq. 8. If an admissible residual $\eta$ exceeds the recovery floor $\epsilon_\tau/\kappa_\tau$, then, for $0 < \kappa_\tau < 1$ and $e_0 > \eta$, the sufficient budget under this model is

$$B \ge \left\lceil \frac{\log[(\eta - \epsilon_\tau/\kappa_\tau)/(e_0 - \epsilon_\tau/\kappa_\tau)]}{\log(1 - \kappa_\tau)} \right\rceil . \tag{15}$$

This clarifies the meaning of a recovery-aware release constraint. Estimating its quantities from a small probe requires calibration; a source/target forward-loss difference alone cannot identify the contraction rate. A practical adaptive method should therefore retain a conservative deadline near the validated fixed release and compare measured outcomes against that baseline, including any probe overhead.

## I A SMALL ADDITIONAL GAIN NEAR THE LATE-RELEASE BASELINE

We extend the constant-LR setting of Section 5.3 with a lightweight release candidate anchored at $0.9T$. On the same two 512-token training windows, it measures $J = [\ell_{512}(W, c_*) - \ell_{512}(W, c_0)]_+$ once, before update 4500. A heuristic recovery budget

$$\widehat{B} = \lceil 278 \log \max\{1, J/0.02\} \rceil$$

selects release at step 4750 if $\widehat{B} \le 250$, and at 4500 otherwise. The coefficient is the $0.7T$ time constant of the descriptive decaying-LR recovery fit (Appendix B), not a newly identified constant-LR contraction rate. The tolerance 0.02 is in NLL units. This pilot uses the switch increment as a simple proxy; Eq. 9 explains why readiness also depends on the hold-state excess loss.

Each arm trains for 5000 steps at length 512, batch 32, LR $3.6 \times 10^{-5}$, toward the temperature-free YaRN-family-style target from the shared pilot base. Both the fixed baseline and candidate perform the same probe, with an independent sampler and no validation-data selection. The extra measurement processes 2048 forward tokens in total, 0.0025% of the 81,920,000 training tokens; this token-count fraction is not a measured FLOP ratio. The observed probe wall time was approximately 0.04 seconds per run.

Table 16: Constant-LR late-release pilot at 19M on WikiText-103, shared pilot base, 82M continuation tokens per arm. Entries are mean ± sample standard deviation over three continuation seeds. The candidate chose $0.95T$ in every seed; these outcomes therefore support a later fixed release in this setting, without establishing an adaptive-selection advantage.

| Schedule | 512 | 1024 | 2048 | 4096 |
|---|---|---|---|---|
| Fixed $0.9T$ | $66.428 \pm 1.624$ | $64.611 \pm 0.811$ | $66.121 \pm 1.497$ | $81.056 \pm 0.341$ |
| Candidate (all $0.95T$) | $66.450 \pm 1.625$ | $64.687 \pm 0.808$ | $66.207 \pm 1.511$ | $80.775 \pm 0.306$ |

All three candidates had $J = 0$ after positive-part clipping and selected $0.95T$. Mean 4096-token PPL improved from 81.056 to 80.775 (Table 16); paired improvements were 0.282, 0.318, and 0.243 (mean 0.281, sample s.d. 0.037). Mean PPL increased by 0.022, 0.076, and 0.086 at lengths 512, 1024, and 2048, respectively. Thus the additional gain is small and concentrated at the hardest length, while the fixed late-release baseline remains strong. The fixed-$0.9T$ runs reproduce the S1 values of Table 3 to the reported precision.

**What a zero clipped jump means.** The signed differences $\ell_{512}(W, c_*) - \ell_{512}(W, c_0)$ before update 4500 are $-0.002532$, $-0.000775$, and $-0.004338$ NLL for seeds 0, 1, and 2. The target configuration already has slightly lower loss on these particular training windows before target-configured optimization. Thus $J = 0$ reflects the positive-part definition, not a missing measurement or equality of the two losses. These small-window observations do not establish a population-level advantage of target over source positions.

Algebraically, write $\delta = \ell_L(W, c_*) - \ell_L(W, c_0)$ and $a = \ell_L(W, c_0) - \ell_{\mathrm{ref}}$. Then the initial target excess is $e_0 = [a + \delta]_+$, whereas $J = [\delta]_+$. A negative $\delta$ can coexist with positive $e_0$ whenever

$a > -\delta$. Consequently $J = 0$ does not imply that the target state is optimal, that further target updates are unnecessary, or that its gradients equal those under source positions. It measures an incremental loss change rather than the total adaptation still available. This distinction motivates the target-readiness measurement in Appendix J; its fresh-seed results also show why a recovery proxy must be evaluated against fixed release schedules.

Because the candidate chose the same release in every seed, its realized training schedules are fixed-$0.95T$ schedules. The measurements establish a useful late-release option in this setting, without establishing an advantage of state-dependent selection over that fixed option. A separate continuation seed with target-NLL observations after the actual release provides within-setting recovery calibration (Appendix J); its outcomes are kept separate from this three-seed table.

## J Recovery measurements within the constant-LR setting

The remaining budget in Eq. 8 should reflect adaptation under the current optimizer and positional target. We therefore run an additional paired continuation seed (seed 3, the same pretrained base), with fixed releases at $0.9T$ and $0.95T$ and all other settings as in Appendix I. After the actual release, we measure target NLL on the same two 2048-token training windows. Observations are taken before the next update at 0, 25, 50, 100, 200, 250, and 500 completed target updates where the remaining budget permits, including the exact endpoint. There is one actual positional switch per training run; the measurements do not change weights or optimizer state.

Table 17: Additional continuation seed 3, constant LR. Final validation PPL is separate from the fixed training-window NLL used for recovery calibration.

| Release | 512 | 1024 | 2048 | 4096 |
|---|---|---|---|---|
| $0.9T$ | 67.221 | 65.789 | 64.331 | 79.894 |
| $0.95T$ | 67.245 | 65.867 | 64.416 | 79.626 |

The additional seed preserves the small longest-length improvement (0.269 PPL at 4096; Table 17) with costs of 0.024, 0.079, and 0.085 PPL at the other lengths. The temporal refinement therefore has a consistent, length-dependent effect across the four observed continuation seeds. It does not replace the much larger static-to-DEPA effect as the main contribution.

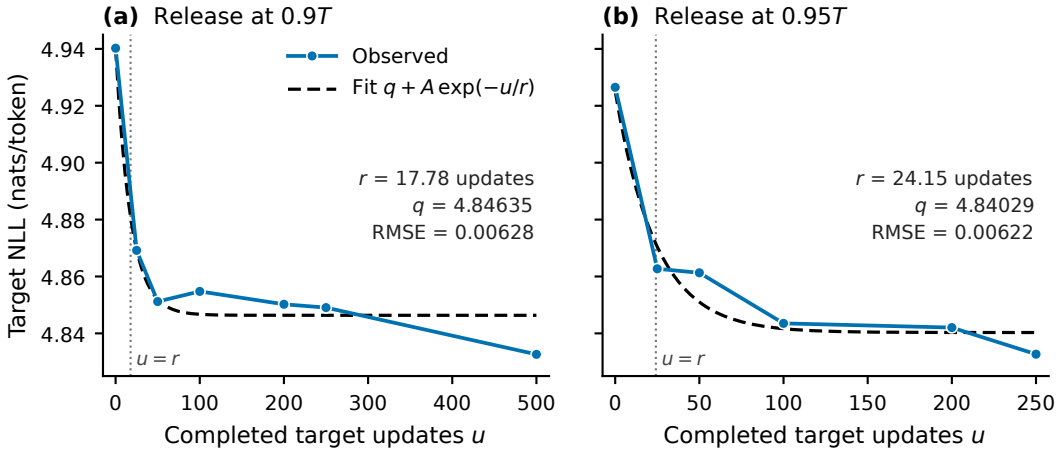



Figure 6: Actual post-release target-NLL observations on fixed training windows, seed 3, for releases at (a) $0.9T$ and (b) $0.95T$. Dashed curves are descriptive floor-plus-exponential fits $\ell(u) = q + A\exp(-u/r)$ (fitted $r$, $q$, and RMSE printed in each panel; the dotted line marks $u = r$). They summarize effective adaptation, including ordinary continued learning, and do not isolate a switch-only convergence process.

Most of the observed target-NLL decrease occurs early. With the $0.9T$ release, NLL changes from 4.94024 at release to 4.85119 after 50 target updates, and ends at 4.83262 after 500. With the $0.95T$ release, it changes from 4.92650 to 4.86130 after 50, and ends at 4.83268 after 250. Later observations fluctuate; the fitted floor is a descriptive parameter rather than a proven terminal loss.

Fitting $\ell(u) = q + A\exp(-u/r)$ with nonnegative floor and amplitude (Figure 6) gives $r = 17.78/24.15$ updates, $q = 4.84635/4.84029$, and fit RMSE 0.00628/0.00622 NLL for the two releases. These within-setting measurements differ from the decaying-LR recovery fits of Appendix B. The value of this calibration is practical: a 250-update target phase is substantially longer than the rapid initial adaptation seen here, motivating a test of a shorter reserved budget. The measured 100–200-update reserve optimum is several times the fitted fast-adaptation time constant ($\approx$ 18–24 updates; ratio ranging from about 4 to 11), so the reserve is set by the hold-versus-adaptation balance (Section 4) rather than by the recovery rate itself—supportive of, though not a mechanistic proof of, the reserve-in-updates reading. Two small calibration windows on one seed do not establish a universal recovery time.

**A calibrated one-measurement candidate.** Before evaluating new continuation seeds, we fix a candidate using these observations: at $0.9T$, measure target NLL $h$ on the same training windows and reserve

$$\widehat{B} = \max\left\{100, \left\lceil 50 \log \max\left\{1, \frac{h - 4.828}{0.01}\right\}\right\rceil\right\}, \qquad \tau = \max\{\lfloor 0.9T \rfloor, T - \widehat{B}\}.$$

Here 50 updates is approximately twice the larger fitted timescale; 4.828 is below both fitted floors by roughly two fit RMSE. These are empirically chosen slack values, not confidence bounds. The minimum 100-update target phase limits how far the hold can extend. This candidate uses a target-readiness proxy rather than the clipped switch increment alone. The fixed-$0.9T$ comparator receives the same measurement. The fresh-seed comparison appears in Appendix K.

## K Fresh-seed release refinements

We evaluate the frozen recovery-budget candidate on continuation seeds 4–6, separate from calibration seed 3. The pretrained base, target mask, constant learning rate, continuation tokens, and validation protocol are shared within each pair. All five arms perform the same training-only anchor measurement: source and target forward passes on two fixed 2048-token training windows, totaling 8192 forward tokens (0.01% of the 81,920,000 continuation training tokens). The probe does not update weights or alter the training-batch random stream. All arms share the S1 temperature-free YaRN-family-style target; these results are separate from the custom-YaRN-family experiments. The candidate releases at steps 4867, 4866, and 4869, leaving 133, 134, and 131 target updates. Thus the state-dependent rule selects a narrow neighborhood of $0.974T$ in this setting.

Table 18: Five-arm release comparison, constant-LR 19M WikiText-103. Mean $\pm$ sample s.d. PPL across continuation seeds 4–6.

| Schedule | 512 | 1024 | 2048 | 4096 |
|---|---|---|---|---|
| Fixed $0.9T$ | $66.251 \pm 0.868$ | $64.605 \pm 2.005$ | $68.112 \pm 0.655$ | $79.658 \pm 0.998$ |
| Calibrated release | $66.298 \pm 0.873$ | $64.766 \pm 2.001$ | $68.351 \pm 0.650$ | $79.482 \pm 0.973$ |
| Fixed $0.95T$ | $66.274 \pm 0.870$ | $64.683 \pm 1.999$ | $68.205 \pm 0.647$ | $79.380 \pm 0.964$ |
| Late cosine | $66.258 \pm 0.867$ | $64.629 \pm 2.001$ | $68.121 \pm 0.651$ | $79.564 \pm 0.988$ |
| Fixed $0.925T$ | $66.259 \pm 0.868$ | $64.636 \pm 2.002$ | $68.141 \pm 0.651$ | $79.476 \pm 0.974$ |

The candidate improves 4096-token PPL in all three pairs (Table 18), with mean gain 0.175 and paired-gain sample s.d. 0.035. The corresponding mean costs are 0.047, 0.161, and 0.239 PPL at lengths 512, 1024, and 2048. This extends the late-release result at the hardest length while preserving the substantially larger static-to-DEPA gain established in the main experiments. The additional timing refinement is not uniformly better across lengths.

The calibrated candidate's numerical recovery budget is an empirical choice, not a guarantee of final generalization. In particular, effective adaptation on its fixed training windows is distinct from

the validation objectives at each length. The same-seed fixed-$0.95T$ comparison below identifies a simpler release with better validation outcomes than this calibrated candidate. The common late-transition comparison below evaluates a different balance between shorter-length retention and extrapolation.

**A simpler matched late release.** On the same continuation seeds 4–6, a fixed release at $0.95T$ improves 4096-token PPL over $0.9T$ by 0.278 on average (Figure 7), with respective costs of 0.023, 0.077, and 0.093 at 512, 1024, and 2048 tokens. It also outperforms the calibrated candidate at every evaluation length in each of the three pairs; mean improvements are 0.024, 0.083, 0.146, and 0.102 PPL. All compared arms use the same training-only anchor probe and training budget. These results favor retaining the simple fixed release in this setting. A rapid initial reduction in anchor NLL does not by itself identify the target-training budget that gives the best held-out perplexity: the 250-update target phase improves on the candidate's 131–134 updates here. The additional refinement concerns the balance across evaluation lengths; the main evidence for delayed adjustment remains the static-to-DEPA result.

**A short common transition preserves more short-length performance.** A common cosine interpolation of all changed frequency multipliers from $0.9T$ to $0.95T$ improves 4096-token PPL over the fixed $0.9T$ release by 0.094 on average across seeds 4–6. Its mean costs at 512, 1024, and 2048 are 0.007, 0.023, and 0.008 PPL, respectively. Relative to fixed $0.95T$, this improves those shorter lengths by 0.016, 0.054, and 0.084 while giving up 0.184 PPL at 4096. Thus it offers a different empirical balance rather than uniform dominance. The final positional configuration, constant learning rate, continuation tokens, and 250 fully target-configured updates are shared with fixed $0.95T$.

The cosine arm also changes average exposure to the target multipliers. Its discrete schedule has 374.5 target-equivalent exposure updates, whereas fixed $0.95T$ has 250. Here exposure means the sum of the scalar interpolation coefficients, not an equivalence of nonlinear training trajectories. A fixed $0.925T$ release with 375 target updates completes this comparison on the same three seeds. It improves 4096-token PPL over fixed $0.9T$ by 0.181 on average, with costs of 0.008, 0.031, and 0.028 at shorter lengths. Against the cosine transition, the midpoint gains 0.087 at 4096, with respective shorter-length costs of 0.001, 0.008, and 0.020. Fixed $0.95T$ retains another 0.096 of longest-length improvement, while the midpoint improves its shorter-length PPL by 0.015, 0.047, and 0.065. The midpoint therefore supplies a simple intermediate balance without a smooth transition. These results do not identify a uniformly best refinement or establish an adaptive-selection advantage. They favor keeping a single release as the practical design and selecting its budget for the intended evaluation lengths.

## L 137M PAIRED REPLICATIONS AND CONTENT EVALUATION

These continuations use the training implementation documented in Appendix D, from the same pretrained base, with fresh continuation seeds. Each paired comparison shares the final target and 131,072,000 training-token budget. These runs replicate the direction and magnitude of the one-decimal comparisons in Table 11; we report their exact measurements separately from those rounded values. These continuation-seed pairs do not themselves measure variation across pretrained bases; Appendix P.4 reports a separate independently initialized custom-YaRN-family replication.

**Different-architecture and different-recipe replication.** An architecturally different base provides an out-of-family check of the custom-YaRN-family schedule claim. The base uses 20 layers, 8 heads, embedding width 640, and head dimension 64 ($\sim$124M parameters), was pretrained on PG19 at context 1024 with a distinct LR schedule, and was continued with the identical constant-LR custom-YaRN-family protocol (4000 updates at 4096, LR $4.96 \times 10^{-5}$, warmup 0, seeds 48/49). All six arms were produced by the same trainer under identical input/target streams, and the final evaluation uses the full target configuration in every arm; the only differences are the schedule and the pretrained base. Per-seed 4K/8K/16K PPL are in Table 19: DEPA beats static at 16K by $+5.36/+4.88$ PPL (5.12 PPL mean) and keeps the small over-ramp margin ($+0.119/+0.098$ PPL). The schedule-shape comparison is therefore not confined to the two 14-layer, width-768 pretrained bases, and its magnitude here is larger: this recipe's static 16K natural-context added-context harm is $-0.66/-0.62$ nats (versus

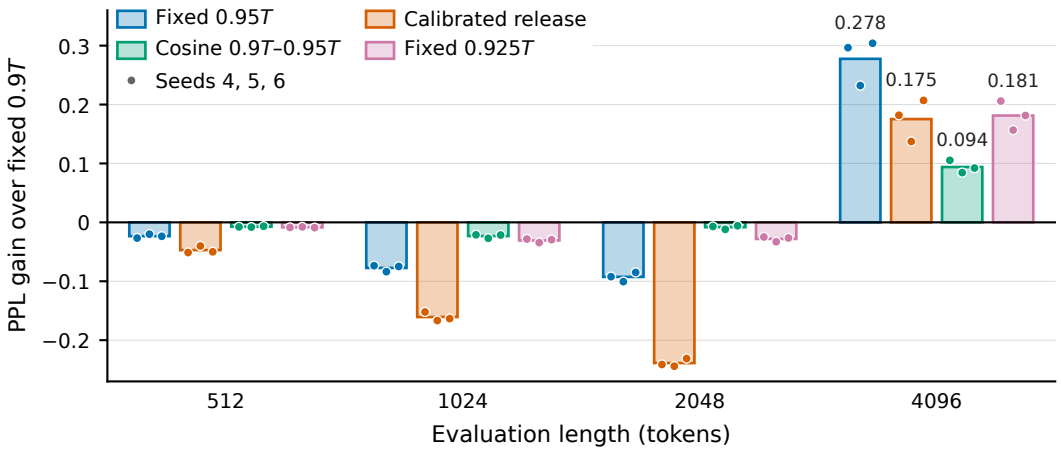



Figure 7: Paired PPL gains over fixed $0.9T$ on continuation seeds 4–6 (constant-LR 19M WikiText-103). Positive values indicate lower PPL. Bars show the three-seed mean (printed above the 4096-token bars); dots show the individual paired gains for seeds 4, 5, and 6 (left to right within each bar); no uncertainty intervals are drawn. The shared vertical scale makes shorter-length costs and longest-length gains directly visible. Every dot uses a run paired with the same continuation seed and pretrained base.

$\approx -0.57$ on the 14-layer WikiText-103 base), and DEPA cuts it to $-0.10/-0.08$ nats (85–87% reduction).

Table 19: Custom-YaRN-family static/DEPA/equal-exposure-ramp triad on the different-architecture base (20L/8H/640E, ∼124M), constant LR, seeds 48/49. Lower PPL is better. Values are computed from the per-run result files.

| Seed | Static | | | DEPA (0.9T) | | | Equal-exposure ramp | | |
|---|---|---|---|---|---|---|---|---|---|
| | 4K | 8K | 16K | 4K | 8K | 16K | 4K | 8K | 16K |
| 48 | 29.27 | 29.50 | 36.29 | 29.31 | 28.68 | 30.94 | 29.32 | 28.72 | 31.05 |
| 49 | 29.40 | 29.90 | 34.25 | 29.45 | 29.04 | 29.38 | 29.45 | 29.08 | 29.47 |

Table 20: Additional 137M PG19 paired results. Lower PPL is better.

| Target/seed | Schedule | 4096 | 8192 | 16384 |
|---|---|---|---|---|
| PI, 1 | Static | 28.050 | 66.653 | 159.919 |
| PI, 1 | DEPA $0.9T$ | 29.385 | 28.343 | 27.568 |
| PI, 2 | Static | 28.108 | 71.689 | 162.491 |
| PI, 2 | DEPA $0.9T$ | 29.522 | 29.725 | 27.575 |
| Custom YaRN-family, 2 | Static | 28.239 | 29.459 | 29.007 |
| Custom YaRN-family, 2 | DEPA $0.9T$ | 28.260 | 28.861 | 27.044 |

In Table 20, the PI seed-1 pair reduces 16384 PPL by 132.351, while increasing 4096 PPL by 1.335. Thus the large extrapolation improvement and its trained-length cost recur on a second continuation seed. The custom-YaRN-family seed improves 16384 PPL by 1.963 (6.77%), with a 0.021 trained-length cost, extending the positive result to a third continuation seed. A full-budget linear ramp on this custom-YaRN-family seed reaches 27.619 PPL at 16384; DEPA improves it by a further 0.575 with a 0.018 trained-length cost (Appendix M.2). The custom target's smaller but nearly costless timing benefit is a useful practical complement to the larger PI effect.

**A third PI continuation seed.** Continuation seed 2 reproduces the PI timing effect from the same pretrained base. Static and DEPA PPL are 28.108/29.522 at 4096, 71.689/29.725 at 8192,

and 162.491/27.575 at 16384. The longest-length reduction is 83.03%, with a 1.414-PPL trained-length cost. Together with seed 0 (Table 11) and seed 1, all three continuation seeds show the same directional tradeoff. This adds replication over continuation randomness, not over independently pretrained bases.

Across the two full-precision pairs (seeds 1 and 2), mean paired NLL gains at 4096, 8192, and 16384 are $-0.047794$, $0.867737$, and $1.765854$. The corresponding ranges are $[-0.049078, -0.046510]$, $[0.855126, 0.880348]$, and $[1.758007, 1.773700]$. These are descriptive two-seed summaries, not precise population uncertainty estimates. The one-decimal Table 11 is kept separate from this full-precision aggregate. Seed 2 has both language-model and content evaluations.

**Counterfactual remote-content evaluation.** To measure whether a final model uses distant content, we construct paired contexts $x^{(a)}$ and $x^{(b)}$ with identical filler, query marker, and candidate tokens $a \neq b$. The source marker is at position zero, its associated answer at position one, and the query marker at position $D$. Only position one changes, from $a$ to $b$. Filler and the original marker–answer pair are sampled from validation tokens; the alternative answer is sampled from the same token marginal, conditional on being different. Natural filler may contain other occurrences of the marker; these are identical within the pair. We use distances $D \in \{4096, 8192, 16384\}$, input length $D + 1$, and 128 context pairs per distance with a fixed evaluation seed shared across training arms.

For model $m$, define the candidate log odds and content response

$$r_m(x) = \log \frac{p_m(a \mid x)}{p_m(b \mid x)}, \qquad C_m = r_m(x^{(a)}) - r_m(x^{(b)}).$$

The log odds equal the corresponding logit difference. A predictor that ignores the replaced remote token has $C_m = 0$, regardless of a fixed lexical preference between $a$ and $b$. Positive $C_m$ indicates a response in the expected direction, but does not require a correct answer in either context. We therefore also report balanced two-candidate accuracy, assigning half credit to ties, and the fraction of pairs for which both predictions are correct: $r_m(x^{(a)}) > 0$ and $r_m(x^{(b)}) < 0$. A context-independent candidate preference yields balanced accuracy $1/2$ and no correct reversal. These are constrained-choice and content-sensitivity measurements, not open-vocabulary passkey accuracy or a general downstream benchmark.

The principal schedule contrast is the trial-paired difference $C_{\mathrm{DEPA}} - C_{\mathrm{static}}$ at each distance, accompanied by paired changes in balanced accuracy and correct reversals. Summary standard errors use the 128 context pairs as sampling units; the two contexts within a pair are not independent observations. Trial-level uncertainty describes the sampled probes conditional on trained weights, not variability across independent training runs. Each training-seed pair is reported separately. Final positional targets and attention corrections are those actually used by the corresponding language-model experiment. This protocol is evaluated on the PI seed-1 pair.

**Observed content-use results.** Balanced two-candidate accuracy for static/DEPA is 0.5000/0.5039 at $D = 4096$ and 0.5000/0.5000 at both larger distances. Only one of the 128 DEPA pairs at 4096 has both contexts correct; no pair has both correct for either model at 8192 or 16384. Trial-paired mean content-response changes (DEPA minus static) are $-0.02643$, $-0.01297$, and $+0.00392$ log-odds units, with respective standard errors 0.01405, 0.00414, and 0.00431. These measurements do not establish a consistent content-use advantage or successful long-range retrieval. The strong PPL improvement therefore remains a language-modeling result, with capability transfer unresolved by this particular zero-shot probe. The training mixture inserts marker–item pairs but does not explicitly train a repeated query to retrieve a previous item; it should not be interpreted as direct supervision for this counterfactual retrieval task.

**Content-probe continuation-seed replication.** On seed 2, static/DEPA balanced candidate accuracy is 0.4961/0.5000, 0.5000/0.5000, and 0.5039/0.4961 at distances 4096, 8192, and 16384. DEPA has zero correct two-context reversals at each distance; static has one at 16384 and zero otherwise. Paired DEPA-minus-static response means are $-0.014488$, $-0.012019$, and $0.001783$, with trial-pair standard errors $0.006801$, $0.002870$, and $0.003395$. These remain conditional on each trained pair. The second probe seed therefore does not establish retrieval gains despite the replicated language-model improvement.

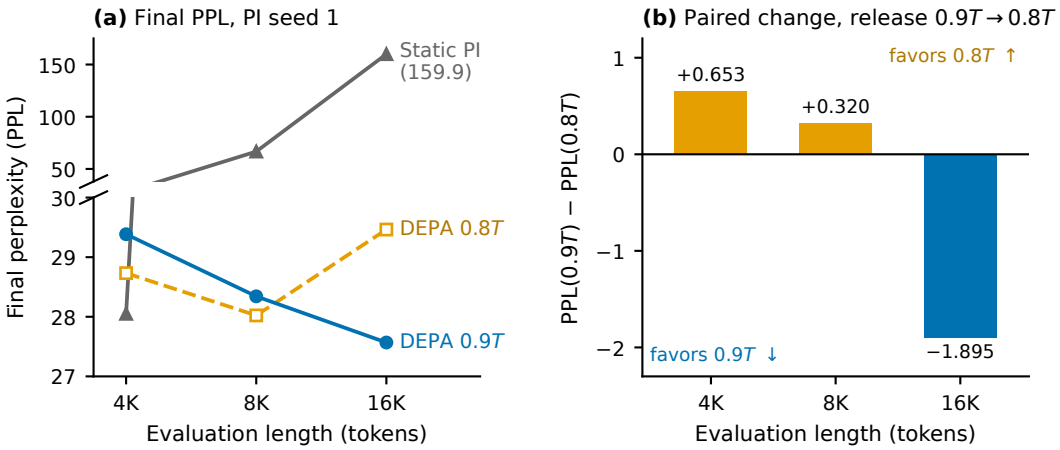



Figure 8: PI seed-1 timing comparison. (a) Final PPL under a shared final target and continuation budget; the y axis is broken (27–30 and 38–175 PPL) so that the two DEPA release curves are resolvable. Lines connect measured lengths, not predicted intermediate values. (b) Paired change PPL($0.9T$) − PPL($0.8T$) from moving release from $0.9T$ to $0.8T$; positive (orange) favors the earlier release, negative (blue) favors $0.9T$. This separate view makes the smaller within-DEPA tradeoff visible beside the much larger static-to-DEPA extrapolation gain. One continuation seed; the figure does not represent unmeasured training variability.

**Content probe on the constant-LR base-1 benchmark families.** The same counterfactual probe, run on the constant-LR base-1 PI benchmark arms whose static-to-DEPA 16K PPL drops are the largest in this PI family ($189.587 \rightarrow 30.076$ on seed 44; $183.401 \rightarrow 31.714$ on seed 45), gives exactly $0.5000$ balanced candidate accuracy for both schedules at all three distances and zero correct two-context reversals in every cell. Trial-paired DEPA-minus-static response means are $-0.01452$, $-0.00451$, and $-0.00071$ on seed 44 and $-0.02772$, $-0.00688$, and $-0.00623$ on seed 45 (standard errors $0.01294$–$0.02022$ at 4096 and $0.00207$–$0.00278$ at the longer distances). This family's dominant language-modeling improvement therefore also shows no measurable differential content use on this probe; if anything, the small response shifts point away from DEPA. These probes do not test trained retrieval, so capability transfer remains unresolved as stated above. The published-recipe base-1 pairs replicate the null on the identical trial set (evaluation seed 57021): on seeds 45 and 46, balanced candidate accuracy stays within $0.5000$–$0.5039$ for both schedules at every distance, with at most one correct two-context reversal per cell, and DEPA-minus-static response means at 8192/16384 are $-0.00500/-0.00386$ (seed 45) and $-0.00496/-0.00394$ (seed 46; 4096 means $+0.00928 \pm 0.01493$ and $+0.00040 \pm 0.01491$). Across six training-seed pairs in three families this probe finds balanced accuracy within one pair of chance in every cell and no consistent content-use advantage for either schedule.

**More target updates reduce the PI trained-length cost.** On the 137M PI continuation seed 1, releasing at $0.8T$ leaves 800 target updates rather than 400, at the same total 4000-update budget. The final PPL values at 4096, 8192, and 16384 are 28.732, 28.023, and 29.464, respectively, compared with 29.385, 28.343, and 27.568 for $0.9T$. Thus earlier release improves the first two lengths by 0.653 and 0.320 PPL but gives up 1.895 PPL at the longest length. Relative to static PI, the trained-length cost falls from 1.335 to 0.682 PPL, a 48.9% reduction. The 16384-token gain remains 130.456 PPL, retaining 98.6% of the absolute static-to-$0.9T$ gain; this retained-gain ratio does not imply equal final perplexity or equal downstream capability.

This same-seed intervention provides a practical way to adjust the trained-length/extrapolation balance: the PI cost is sensitive to the allocation of hold and target updates. It does not vanish at $0.8T$ and does not establish a uniformly superior release. Since moving the release also changes the model state and the learning rates at which target updates occur, the result does not isolate a recovery-rate mechanism. The $0.8T$ content evaluation gives balanced candidate accuracy 0.5 at all three distances and zero pairs with both contexts correct. Thus additional target updates mitigate the trained-length PPL cost here without establishing successful retrieval on this probe.

**Response magnitude versus successful content use.** Pairing the same 128 counterfactual trials for $0.8T$ and $0.9T$ on seed 1, the mean change in remote-answer log-odds response ($0.8T - 0.9T$) is $0.02150$, $0.000569$, and $0.004319$ at distances 4096, 8192, and 16384; paired-trial standard errors are $0.01698$, $0.000339$, and $0.001700$. These quantify trial sampling conditional on the two trained models, not training-seed uncertainty. Balanced candidate accuracy changes by $-0.003906$, $0$, and $0$, respectively. In particular, the higher mean response at 16384 does not produce correct two-context answer reversals. This separates a change in remote-content sensitivity from demonstrated retrieval success; the additional target updates do not establish the latter.

**Choosing a measured operating point.** The appropriate PI release depends on how evaluation lengths are weighted. For nonnegative scored-token weights $w_L$ summing to one, define empirical risk $R(f) = \sum_L w_L \log \mathrm{PPL}_L(f)$; averaging PPL itself would not represent pooled token NLL. On continuation seed 1, the measured difference is

$$R(0.9) - R(0.8) = 0.0224715 w_{4096} + 0.0113519 w_{8192} - 0.0664943 w_{16384}.$$

Positive values favor the earlier release. For a mixture of 4096- and 16384-length evaluations, let $p$ be the fraction of scored tokens assigned to the latter. Among static PI, $0.8T$, and $0.9T$, the empirical minimizer is static for $p < 0.0140$, $0.8T$ for $0.0140 < p < 0.2526$, and $0.9T$ for $p > 0.2526$, with ties at the boundaries. For an 8192/16384 mixture, $0.8T$ is best below $p = 0.1458$ and $0.9T$ above it; static is never best. These are post-hoc summaries of one seed's sampled-window measurements, not independently validated selection thresholds. The weights refer to scored tokens rather than request counts, and the calculation makes no prediction for unmeasured lengths or other data distributions. It shows why neither delayed operating point uniformly replaces the other and gives a concrete interpretation of the observed training-length cost.

# M SCHEDULE CONTROLS UNDER SHARED TARGETS

## M.1 FULL-BUDGET RAMP AND END-ONLY CONVERSION

Two additional seed-1 controls distinguish late release from full-budget interpolation and from postponing conversion until inference. They share the 137M base, 4000 continuation updates at length 4096, batch size 8, optimizer schedule and sampled evaluation windows of the existing PI arms. The linear ramp uses $m_j(t) = 1 + \frac{t}{T-1}(m_j^* - 1)$ for $t = 0, \ldots, T-1$, starting at source frequencies and reaching PI on the last update. The hold-only arm trains at source frequencies throughout; its final weights are evaluated at both source and PI configurations.

Table 21: PI schedule controls, continuation seed 1. Lower PPL is better. All rows except the explicitly separated source endpoint evaluate at the same PI target. Each training run has the same continuation tokens.

| Schedule / evaluation endpoint | 4096 | 8192 | 16384 |
|---|---|---|---|
| Static PI | 28.050 | 66.653 | 159.919 |
| Full-budget linear ramp / PI | 28.379 | 28.317 | 34.756 |
| DEPA $0.8T$ / PI | 28.732 | 28.023 | 29.464 |
| DEPA $0.9T$ / PI | 29.385 | 28.343 | 27.568 |
| Hold-only / end-only PI conversion | 108.044 | 106.759 | 103.272 |
| Hold-only / source (different endpoint) | 28.201 | 30.045 | 51.122 |

**A full-budget ramp helps, but does not match late release at 16K.** Linear interpolation substantially improves on static PI (Table 21), demonstrating that gradual positional adaptation is a useful alternative. DEPA $0.9T$ still improves 16384 PPL by 7.188 (20.68%) over the ramp, but costs 1.006 at 4096 and 0.026 at 8192. DEPA $0.8T$ improves 16384 by 5.293 and 8192 by 0.294, with a 0.353 trained-length cost. Thus this comparison supports a longest-length advantage for late release rather than uniform dominance over gradual adaptation. Both paths share total tokens and endpoint frequency displacement, but the ramp's scalar target exposure is $T/2$, versus approximately $0.1T$ for DEPA $0.9T$. It does not isolate smoothness at equal target exposure.

**Target-phase training has practical value.** At fixed hold-only final weights, converting source frequencies to PI increases NLL by 1.343186, 1.267871, and 0.703159 at the three lengths. This is a direct inference-configuration intervention on the same weights. At a shared PI endpoint, DEPA $0.9T$ improves NLL over end-only conversion by 1.302044, 1.326193, and 1.320709, respectively. Keeping source positions at inference also fails to match DEPA's 16K result (51.122 versus 27.568), although it has lower trained-length PPL. The hold-only baseline therefore does not replace hold-then-adapt in this setting. The trained-path contrast includes replacing the final source updates with target updates, and does not identify a pure recovery rate or unique weight-level mechanism.

These controls use one continuation seed. Their PI-endpoint content probes remain near chance: linear-ramp accuracy is 0.5000/0.5000/0.4961 and hold-only accuracy is 0.5000 at all distances, with zero correct two-context reversals in either arm. These controls establish no retrieval advantage.

**Operating regions after adding the controls.** Including the full-budget linear ramp and end-only PI conversion in the same seed-1 scored-token-weighted NLL comparison changes the 4096/16384 lower envelope. With $p$ the 16384 scored-token share, static minimizes empirical risk below $p = 0.00758$, the linear ramp between $0.00758$ and $0.06970$, DEPA $0.8T$ between $0.06970$ and $0.25259$, and DEPA $0.9T$ above $0.25259$, with adjacent schedules tied at boundaries. This gives the ramp a meaningful operating region when trained-length performance receives most of the weight. For the 8192/16384 mixture, the earlier $0.8T$/$0.9T$ boundary at $p = 0.14582$ is unchanged; neither static nor the ramp lies on that lower envelope. End-only PI conversion minimizes neither mixture at any weight. These comparisons use the same PI endpoint; the source-configuration diagnostic is excluded. They are descriptive single-seed regions, not held-out policy validation or predictions at unmeasured lengths.

These controls characterize the remaining schedule choices. On PI seed 1, $0.8T$ halves the trained-length cost relative to $0.9T$, trading 1.895 PPL at the longest length (Figure 8). A full-budget linear ramp reaches 34.756 at 16384; DEPA improves this by 20.68%, with a 1.006-PPL trained-length cost. Against the equal-scalar-exposure tail ramp, DEPA improves all three lengths on all three seeds (Appendix P.1). End-only PI conversion reaches 103.272, establishing the value of target-phase training in the tested comparison (Appendix M.1). The source-hold and release-loss traces characterize learning before and after conversion; they do not identify a weight-level cause of the endpoint advantage.

### M.2 Full-budget ramp under custom YaRN-family

The same full-budget linear multiplier schedule also improves on the stronger custom-YaRN-family static baseline. On continuation seed 2, with the same base, 4000 updates, target temperature active throughout, and matched final custom-YaRN-family configuration, the three schedules give the endpoints in Table 22. DEPA improves on the ramp by 0.244 PPL at 8192 and 0.575 at 16384 (2.08%

Table 22: Custom-YaRN-family full-budget ramp comparison, paired continuation seed 2.

| Schedule | 4096 | 8192 | 16384 |
|---|---|---|---|
| Static | 28.239 | 29.459 | 29.007 |
| Full-budget linear ramp | 28.242 | 29.104 | 27.619 |
| DEPA $0.9T$ | 28.260 | 28.861 | 27.044 |

at 16384), with a 0.018 trained-length cost. The corresponding paired NLL gains are $-0.000630$, $0.008408$, and $0.021048$. Thus gradual adaptation captures a substantial part of the static-to-DEPA benefit, while late release retains an additional extrapolation advantage in this strong-target pair. Together with the PI control, this shows that the observed advantage over a full-budget ramp is not confined to static PI's weak extrapolation. The ramp comparison has one seed per target; it does not establish a population ranking or isolate smoothness at matched exposure. The linear custom-YaRN-family content probe has accuracy 0.5 at all three distances and no correct two-context reversals; the static and DEPA custom-YaRN-family runs did not receive this added probe, so no within-target paired capability comparison is inferred.

### M.3 CONSTANT LEARNING RATE AT THE MAIN 137M SCALE

Two additional PI continuation seeds compare static, DEPA, and the equal-exposure tail-linear ramp at a constant learning rate of $4.96413 \times 10^{-5}$. This value is the arithmetic mean of the decaying 4000-update linear LR sequence, chosen before observing results. All arms share base 0, the 131.072M-token budget, optimizer reset, and final PI configuration, with no warmup restart. DEPA and the tail ramp match both cumulative scalar multiplier exposure and LR-weighted exposure (400 times the constant LR). DEPA improves static 16K PPL from 160.058/162.091 to 28.597/28.683

Table 23: Main-scale constant-LR PI comparison. Each seed is a matched triad; all values are final full-precision language-model evaluations.

| Seed | Schedule | 4096 | 8192 | 16384 |
|---|---|---|---|---|
| 1 | Static | 28.345 | 66.971 | 160.058 |
| 1 | DEPA $0.9T$ | 29.305 | 28.337 | 28.597 |
| 1 | Tail-linear | 29.414 | 28.412 | 29.880 |
| 2 | Static | 28.454 | 71.904 | 162.091 |
| 2 | DEPA $0.9T$ | 29.467 | 29.838 | 28.683 |
| 2 | Tail-linear | 29.601 | 29.962 | 30.162 |

(Table 23), with trained-length costs of 0.959/1.012 PPL. It also improves over the tail ramp at all three lengths in both seeds; the 16K gains are 1.283/1.478 PPL. Thus neither LR decay nor unequal LR-weighted scalar exposure is necessary for the observed single-release advantage at 137M PI. Across the 12 PI delayed arms with both-length records, 8 of 12 have 8K-minus-16K margins above zero (mean $0.15 \pm 1.33$ PPL, sample s.d., range $-2.64$ to 2.15); the largest offsets in either direction sit in single sampled arms — the seed-2 content-family and no-copy arms above zero, the seed-45 base-1 pair member and the early-release ($f = 0.8$) arm below — and the same-seed recipe arms of Table 2 have offsets of both signs, so the mild non-monotonicity visible at 8K in these tables is within single-arm variation across seeds and release points rather than a systematic length response. The census in Table 24 makes the same point across all recorded delayed arms:

Table 24: Cross-family census of the delayed-arm 8K-minus-16K PPL offset (positive means 16K scores lower). Rows with several arms pool one continuation seed's matched protocol variants — release point (upper custom rows), hold-share level (seeds 20/21), or release rule (seeds 25/26) — never across seeds; single arms are listed individually. The offset's direction flips across families, bases, and adjacent seeds, while the within-seed sample s.d. of the multi-arm matched-protocol clusters shown is at most 0.137 PPL, driven on base-0 seed 1 by the early-release $0.5T$ arm ($-0.017$) versus its $0.95T$ sibling (0.346).

| Target family | base; seed(s) | arms | 8K-minus-16K offset (PPL) |
|---|---|---|---|
| PI | 0/1; all PI delayed arms | 12 | $0.15 \pm 1.33$ (8 positive) |
| Custom | 0; 1 | 7 | $0.276 \pm 0.137$ |
| Custom | 0; 2 | 6 | $1.821 \pm 0.014$ |
| Custom | 1; 2 | 3 | $1.831 \pm 0.009$ |
| Custom (hold-share) | 1; 20/21 | 8/8 | $-1.186 \pm 0.027$ / $1.507 \pm 0.023$ |
| Custom (release rule) | 1; 25/26 | 4/4 | $-0.522 \pm 0.005$ / $0.325 \pm 0.005$ |
| Custom (WikiText-103) | WT103; 2/3 | 1/1 | $-0.015$ / $-0.944$ |
| Custom (20L/8H/640) | ∼124M; 48/49 | 1/1 | $-2.260$ / $-0.331$ |
| Published YaRN | 1; 44/45/46 | 1/1/1 | 0.040 / $-1.274$ / 1.084 |

the sign of the 8K-minus-16K offset is not stable across target families, pretrained bases, or even adjacent continuation seeds of one base (the two hold-share seeds of the same base sit 2.69 PPL apart), while within a seed it is nearly invariant to matched protocol changes and responds to release timing itself. This is the picture expected if the 8K reading is an arm-level property the schedule can move, not a universal length response. This does not imply independence from the LR magnitude or apply automatically to custom YaRN-family. The 16K DEPA-over-ramp NLL gain increases by 0.02443/0.02630 relative to the same-seed decaying-LR pairs, whereas the DEPA-over-static

NLL gain decreases by 0.03577/0.04186. Constant LR changes the entire optimization path; this interaction is not an isolated intervention on LR at the release step.

### M.4 What the natural-context ablation identifies

After training, each constant-LR model scores the same final 128 target tokens of 32 natural validation windows at each full input length. Each window is evaluated both with its full input and with only its final 1024 input tokens, giving 4096 scored targets per length and condition. A private fixed RNG (57036) makes window coordinates identical across schedules and seeds. No parameter updates occur. Define context utility $U_m = \ell_m^{\text{local}} - \ell_m^{\text{full}}$, so positive utility means that providing the added prefix lowers the loss on these same targets; Table 25 lists the utilities. For any reference schedule $s$ and

Table 25: Natural-text context utility in nats/token. Negative values mean that the full input is worse than the local suffix on identical targets.

| Seed | Schedule | 4096 | 8192 | 16384 |
|---|---|---|---|---|
| 1 | Static | 0.0108 | −2.4554 | −2.9191 |
| 1 | DEPA $0.9T$ | 0.0116 | −0.0136 | −0.2445 |
| 1 | Tail-linear | 0.0124 | −0.0520 | −0.4706 |
| 2 | Static | 0.0125 | −2.4354 | −2.9288 |
| 2 | DEPA $0.9T$ | 0.0154 | −0.0219 | −0.2710 |
| 2 | Tail-linear | 0.0147 | −0.0724 | −0.4900 |

DEPA $d$, the full-input gain decomposes exactly as

$$\ell_s^{\text{full}} - \ell_d^{\text{full}} = (\ell_s^{\text{local}} - \ell_d^{\text{local}}) + (U_d - U_s).$$

At 16K, DEPA's full-input gains over static are 2.6350/2.6149 nats/token, while its local-input gains are $-0.0396/-0.0430$. Thus the improvement is not a local-input modeling gain; the context-utility differences are 2.6746/2.6578. Against the tail ramp, full-input gains are 0.2314/0.2301, local-input gains are 0.0053/0.0111, and utility differences are 0.2261/0.2190. However, DEPA's own 16K utility remains negative ($-0.2445/-0.2710$). The directly supported interpretation is reduced harm from the additional context, not successful use of that prefix relative to truncation. This helps explain how a large extrapolation PPL gain can coexist with the null counterfactual retrieval probe. It does not identify a weight-level mechanism or show downstream capability transfer.

These tail-token windows are a distinct evaluation from the main full-window PPL sweep; their numerical gains must not be substituted into the main PPL table. Truncation also changes available context and the sequence boundary, and target positions span different local context amounts within the scored tail. Windows can overlap, and neither their tokens nor the two shared-base continuation seeds provide independent-base confidence intervals.

**Is the mean improvement driven by a few windows?** At 16384, DEPA has lower full-input loss than both static and the tail ramp on all 32 sampled windows in each of the two seeds. Its per-window DEPA-over-static gain ranges from 1.518 to 4.394 nats/token in seed 1 and 1.621 to 4.193 in seed 2 (medians 2.528/2.499). Against the tail ramp, the minimum gain is 0.0698/0.0766 and the median is 0.2086/0.2318. Thus the observed mean schedule advantage is not attributable only to a few unusually favorable windows in this sample.

This broad relative improvement still differs from positive absolute context utility. DEPA's full input beats its local suffix on only 1/32 windows in seed 1 and 0/32 in seed 2; its median utility is $-0.2482/-0.2517$ nats/token. Static and the ramp have positive utility on 0/32 windows in either seed. Both training seeds use the same 32 window coordinates, and windows can overlap. These counts describe the evaluated sample, not 64 independent observations or a population sign test. They strengthen the interpretation of consistently reduced long-input harm while retaining the limitation on net benefit from added context.

### M.5 Constant-LR transfer across pretraining bases and targets

The same constant-LR triad was repeated with continuation seed 2 on the independently pretrained base 1 for PI, and on base 0 for custom YaRN-family. All arms retain the 4000-update, 131.072M-

token budget and the same constant LR; custom YaRN-family keeps its attention-temperature correction active from the first update. The PI base comparison changes the loaded base only, while the custom-YaRN-family comparison tests the stronger target at constant LR. On the second base

Table 26: Constant-LR transfer triads, continuation seed 2. Base 1 is independently pretrained; comparisons are paired within each block.

| Target / base | Schedule | 4096 | 8192 | 16384 |
|---|---|---|---|---|
| PI / 1 | Static | 28.636 | 68.274 | 172.314 |
| PI / 1 | DEPA $0.9T$ | 29.551 | 29.827 | 29.223 |
| PI / 1 | Tail-linear | 29.617 | 29.878 | 31.713 |
| Custom YaRN-family / 0 | Static | 28.533 | 29.831 | 29.386 |
| Custom YaRN-family / 0 | DEPA $0.9T$ | 28.559 | 29.228 | 27.434 |
| Custom YaRN-family / 0 | Tail-linear | 28.562 | 29.289 | 27.524 |

(Table 26), PI DEPA reduces 16K PPL from 172.314 to 29.223, with a 0.916-PPL trained-length cost, and beats the tail ramp at all three lengths (2.490 PPL at 16K). Thus PI timing and the equal-exposure shape advantage both replicate across two pretraining initializations under the constant-LR recipe. The DEPA-over-static 16K NLL gain is 0.04251 larger on base 1 than base 0; the corresponding gain over the ramp increases by 0.03152. These are paired differences between two bases, not population confidence intervals or evidence for every LR recipe on both bases.

For custom YaRN-family, DEPA reduces static 16K PPL from 29.386 to 27.434 with a 0.0258-PPL trained-length cost. It also beats the equal-exposure ramp at every length, although the 16K margin is only 0.0907 PPL and the 4096 margin is 0.0032. The static-to-DEPA 16K NLL gain is 0.06876, only 0.00132 below the same-seed decaying-LR gain. The strong-target benefit therefore persists without LR decay in this pair; the small ramp margin motivates the three-seed replication in Appendix N.2, which supports the ordering within this base.

The same natural-text context ablation accompanies both triads. At 16K, PI base 1 has utilities $-3.0636/-0.3072/-0.6309$ for static/DEPA/ramp. DEPA's full-input gain over static is 2.7227 nats/token despite a local-input gain of $-0.0337$, again indicating reduced added-context harm rather than better local modeling. Custom YaRN-family has utilities $-0.3162/-0.0923/-0.0994$ for static/DEPA/ramp. Its DEPA-over-static full-input gain is 0.2240 and local-input gain 0.0001; against the ramp these are 0.0076 and 0.0006. Absolute full-context utility remains negative at 16K for all three schedules even at the stronger target. The mean full-window PPL and this tail-token ablation measure different scored positions and must remain separate. Neither result establishes retrieval improvement. For the PI base-1 triad, the decomposition replicates on two additional continuation seeds: on seeds 44 and 45 the static/DEPA 16K utilities are $-3.1019/-0.2667$ and $-3.0503/-0.2864$, the DEPA-over-static full-input gains are 2.7956 and 2.7274 nats/token (three-seed mean 2.7486), and the paired local-input gains stay negligible at $-0.0396$ and $-0.0365$. Across all three seeds the decomposition is unchanged; window coordinates are identical across seeds by construction, and these remain descriptive paired measurements without population confidence bounds. The full-input gain is positive on all 32 windows of every PI seed (window minimum $+1.55$ nats/token), so no single window carries the replicated effect. For the published-recipe base-1 pair the same structure holds on seeds 45 and 46: static/DEPA 16K utilities $-2.4968/-0.0305$ and $-2.4274/-0.0311$, DEPA-over-static full-input gains 2.4616 and 2.3855 nats/token, and local-input gains of only $-0.0047$ and $-0.0109$; the full-input gain is positive on all 32 windows on both seeds (minima $+1.2466/+1.3133$; local gains positive on 13/32 and 9/32 windows).

**Matched-seed custom-YaRN-family replication at constant LR.** The fixed-length reference comparisons for the length experiment are complete on both bases, continuation seed 5. Each pair retains 4000 updates, 131.072M tokens, the final target and temperature, constant LR, and the same ten-forward 16K training-probe grid. Across bases, the loaded pretrained checkpoint differs; continuation seed and probe/evaluation recipes match. The static arms of this comparison have identical recorded token-stream digests. The fixed-release reference runs have no such digest, so exact recorded stream equality is not asserted between static and DEPA. The 16K PPL improvements are 2.01208 on base 0 and 1.91228 on base 1, with 4K costs of 0.03713 and 0.02499; both also improve 8K. The corresponding 16K NLL gains are 0.068686 and 0.065176, a base-1-minus-base-0

difference of $-0.003510$. This directly matched comparison retains the strong-target timing benefit on both pretrained checkpoints without LR decay. It remains two-base paired evidence, separate from the seed-1–3 family of Appendix N.2.

At 16K, natural full-input NLL improves by 0.233642 on base 0 and 0.274674 on base 1. Base 0 combines a local cost of 0.001581 with utility gain 0.235223; base 1 combines local gain 0.000187 with utility gain 0.274487. Natural full-input gains are positive in all 32 observed windows on both bases, but absolute DEPA utility remains negative ($-0.073861/-0.072217$). On identical nested targets, the 4K-to-16K penalty falls from 0.295961 to 0.059581 on base 0 and from 0.310041 to 0.057017 on base 1. Both nested full-16K and added-context gains are positive in all 32 observed windows on each base. These potentially overlapping windows are not independent population trials, and neither this comparison nor the retrieval probe of Appendix L establishes a capability gain. The length-curriculum comparison on these bases is reported separately (Table 6).

**Window-level coverage of the strong-target advantage.** The 32-window distribution adds information beyond the mean in this seed-2 pair. At 16K, custom-YaRN-family DEPA beats static on all 32 full-input windows: the median NLL gain is 0.23214 and the minimum is 0.04041 nats/token. Against the equal-exposure ramp, DEPA wins on 31/32 windows, with median gain 0.00491 and range $[-0.00178, 0.03347]$. The context-utility difference also favors DEPA on 31/32 windows against the ramp, with median 0.00527. Thus the small mean shape advantage is accompanied by broad window-level coverage in this pair, while allowing a reversal on one window; these observations do not replace replication across training seeds.

Removing the continuation copy mixture preserves the static comparison on all 32 windows (median gain 0.22027, minimum 0.04426 nats/token), and the context-utility difference favors DEPA on all 32 windows under both mixture recipes. Absolute full-input utility is positive for DEPA on only 2/32 windows with the mixture and 3/32 without it, versus 0/32 for static in both cases; the mixed-recipe ramp is positive on 2/32. Accordingly, the dominant effect remains reduced context harm, with positive net context utility confined to a few sampled windows. All counts reuse the same window coordinates across recipes, which may overlap; they are descriptive observations on one base and one continuation seed, not independent samples for a sign test. The copy-free condition applies only to continuation, and no copy-free ramp was trained.

# N Data mixture, target shape, and temperature

## N.1 Removing the continuation copy-task mixture

We repeated the constant-LR static/DEPA pair at base 0, continuation seed 2, for both targets with the continuation copy-task probability set to zero. Each arm still uses 4000 updates and 131.072M training tokens. Copy batches are replaced by PG19 batches, so the intervention changes the continuation data distribution and sampling stream while preserving total budget. The pretrained checkpoint is unchanged and retains its original copy mixture; this experiment does not remove synthetic supervision from pretraining. In Table 27, the PI 16K improvement persists ($158.370$ to $28.431$), with a $0.939$-PPL trained-length cost. Custom YaRN-family likewise improves $29.267$ to $27.346$, with a $0.019$-PPL trained-length cost. The no-copy minus mixed interaction in the 16K

Table 27: No-copy continuation results, seed 2, constant LR. Positive NLL gains favor DEPA; the mixed-reference rows use the same target/base/seed/LR with the original 2% mixture.

| Target | Quantity | 4096 | 8192 | 16384 |
|---|---|---|---|---|
| PI | Static PPL | 28.409 | 71.259 | 158.370 |
| PI | DEPA PPL | 29.348 | 29.707 | 28.431 |
| | No-copy NLL gain | −0.03252 | 0.87494 | 1.71745 |
| | Mixed NLL gain | −0.03496 | 0.87954 | 1.73184 |
| Custom YaRN-family | Static PPL | 28.509 | 29.781 | 29.267 |
| Custom YaRN-family | DEPA PPL | 28.528 | 29.177 | 27.346 |
| | No-copy NLL gain | −0.00067 | 0.02047 | 0.06788 |
| | Mixed NLL gain | −0.00090 | 0.02041 | 0.06876 |

DEPA-over-static NLL gain is $-0.01439$ for PI and $-0.00088$ for custom YaRN-family. These paired outcomes show that continuation copy batches are not necessary for the timing advantage in either tested target. They do not establish exact effect equivalence, independence from the pretraining mixture, or reliability across new continuation seeds.

The PI no-copy natural-context ablation gives a 16K full-input DEPA-over-static gain of 2.5810 nats/token and a local-input gain of $-0.0317$, yielding a context-utility difference of 2.6127. Thus the relative benefit remains concentrated in reduced long-input degradation on these fixed tail targets, rather than a local-input improvement. The custom-YaRN-family no-copy ablation likewise gives a 16K full-input gain of 0.2190 nats/token and local-input gain of $-0.0006$, for a context-utility difference of 0.2195. This is close to the mixed reference utility difference of 0.2239, without establishing exact equivalence or independence from the pretraining mixture. These outcomes are separate from a retrieval-capability claim.

### N.2 Strong-target shape replication and identical-target length effects

We repeat the custom-YaRN-family constant-LR triad on base 0 with additional continuation seeds. Each arm uses 4000 updates, 131.072M continuation tokens, and the original 2% copy mixture. The attention-temperature correction is active from the first update. DEPA and the tail-linear ramp match both cumulative multiplier exposure and its scalar LR-weighted sum. Across the 3 seeds of

Table 28: Custom-YaRN-family constant-LR triads. All seeds share base 0; values are main-protocol PPL.

| Seed | Schedule | 4096 | 8192 | 16384 |
|---|---|---|---|---|
| 1 | Static | 28.428 | 28.300 | 29.406 |
| 1 | DEPA $0.9T$ | 28.459 | 27.765 | 27.433 |
| 1 | Tail-linear | 28.461 | 27.814 | 27.513 |
| 2 | Static | 28.533 | 29.831 | 29.386 |
| 2 | DEPA $0.9T$ | 28.559 | 29.228 | 27.434 |
| 2 | Tail-linear | 28.562 | 29.289 | 27.524 |
| 3 | Static | 29.351 | 29.171 | 30.549 |
| 3 | DEPA $0.9T$ | 29.386 | 28.648 | 28.546 |
| 3 | Tail-linear | 29.389 | 28.727 | 28.666 |

Table 28, DEPA improves 16K PPL over the equal-exposure ramp by 0.0802, 0.0907, 0.1193 (mean 0.0967, sample s.d. 0.0203). The corresponding paired mean NLL gain is 0.003464, with sample s.d. 0.000643. DEPA beats the ramp at all three evaluation lengths in each completed seed. These small, consistently signed paired gains support the shape effect within this base and recipe; they do not quantify reliability across pretrained models.

**Holding target composition fixed across input lengths.** The preceding full/local ablation used different sampled target positions at different full lengths. We fix 32 maximum-length 16K windows (private RNG 57043) and their final 128 target tokens, and score those same 4096 targets using the nested 1K/4K/8K/16K input suffixes. Target IDs, coordinates, and per-token NLL are retained. This evaluation is available for seeds 1 and 3; the seed-2 model has no nested evaluation. Its targets differ from the main language-model sweep; Table 29 lists the means. Define the paired schedule advantage

Table 29: Same-target nested-input mean NLL. $U(4K, 16K) = \ell(4K) - \ell(16K)$; $n_+$ counts windows with positive $U$ out of 32.

| Seed | Schedule | 1K | 4K | 8K | 16K | $U(4K, 16K)$ | $n_+$ |
|---|---|---|---|---|---|---|---|
| 1 | Static | 3.2559 | 3.2437 | 3.3196 | 3.5400 | $-0.2963$ | 0 |
| 1 | DEPA $0.9T$ | 3.2565 | 3.2452 | 3.2718 | 3.3044 | $-0.0592$ | 3 |
| 1 | Tail-linear | 3.2565 | 3.2455 | 3.2752 | 3.3090 | $-0.0635$ | 1 |
| 3 | Static | 3.2590 | 3.2438 | 3.3216 | 3.5297 | $-0.2859$ | 0 |
| 3 | DEPA $0.9T$ | 3.2595 | 3.2455 | 3.2741 | 3.3071 | $-0.0616$ | 3 |
| 3 | Tail-linear | 3.2593 | 3.2454 | 3.2780 | 3.3114 | $-0.0660$ | 2 |

$G_r(L) = \ell_r(L) - \ell_{\mathrm{DEPA}}(L)$. The exact identity

$$G_r(b) = G_r(a) + [U_{\mathrm{DEPA}}(a, b) - U_r(a, b)]$$

separates the shorter-input advantage from the response to the added prefix. On seed 1, DEPA's advantage over static changes from $-0.00148$ at 4K to 0.23559 nats/token at 16K; the added-context interaction is 0.23707, positive on 32/32 windows. Against the ramp the interaction is 0.00422, positive on 29/32 windows. On seed 3, DEPA's advantage over static changes from $-0.00164$ at 4K to 0.22260 nats/token at 16K; the added-context interaction is 0.22424, positive on 32/32 windows. Against the ramp the interaction is 0.00434, positive on 29/32 windows. All three schedules improve in mean from 1K to 4K, but worsen from 4K to 8K and from 8K to 16K on these identical targets. DEPA reduces the extrapolation penalty relative to both baselines. Consequently, lower main-sweep PPL at a longer length does not itself establish that adding a prefix improves the predictions of fixed target tokens. The interventions change prefix availability and input boundary; they do not isolate semantic relevance or identify an attention circuit. Windows may overlap and are reused across seeds. Absolute utility and schedule differences are descriptive paired measurements, not retrieval scores or population confidence intervals. Figure 3 shows these same measurements in the main text.

### N.3 Release-time response under custom YaRN-family at constant LR

We extend the base-0, continuation-seed-1 constant-LR comparison with fixed release fractions 0.5, 0.75, 0.95, and 0.975, retaining the static and 0.9 controls. Every arm uses the same 4000 updates, 131.072M tokens, 2% continuation copy mixture, base checkpoint, continuation seed, and attention-temperature correction from update zero. Only release fraction changes within the DEPA arms. This varies source/target update allocation while preserving total training budget and nominal LR. All six endpoints and their nested evaluations are reported. Up to release 0.95, later releases improve 8K

Table 30: Measured fixed-budget release response, custom YaRN-family, base 0, continuation seed 1. $B$ counts target-configuration updates. All six endpoints are shown.

| Release | $B$ | 4096 PPL | 8192 PPL | 16384 PPL |
|---|---|---|---|---|
| Static | 4000 | 28.4278 | 28.3004 | 29.4062 |
| $0.5T$ | 2000 | 28.4364 | 28.0328 | 28.0495 |
| $0.75T$ | 1000 | 28.4426 | 27.8330 | 27.6140 |
| $0.9T$ | 400 | 28.4592 | 27.7654 | 27.4333 |
| $0.95T$ | 200 | 28.4683 | 27.7483 | 27.4020 |
| $0.975T$ | 100 | 28.4770 | 27.7636 | 27.4182 |

and 16K PPL while slightly increasing the 4K loss (Figure 9). Moving from 0.9 to 0.95 halves the target phase from 400 to 200 updates and improves 16K PPL by 0.03124 (NLL gain 0.001139), with a 0.00909-PPL trained-length cost. This is a small refinement in one paired seed, not an established optimum or a replicated adaptive-policy improvement. Release 0.975 leaves 100 target updates and gives PPL 28.4770/27.7636/27.4182: worse than 0.95 by 0.00875/0.01526/0.01618 at 4K/8K/16K. Thus 0.95 is the best measured 8K/16K point and strictly dominates 0.975 across the three main evaluation lengths. The 100-update endpoint still improves 16K over 0.9 by 0.01506 PPL and is far better than static; it does not establish adaptation collapse. This is a discrete single-seed response, not a continuous or universal optimum. On the fixed nested targets (Table 31), the 4K-to-16K

Table 31: Identical-target nested evaluation for the same endpoints. $U(4K, 16K) = \ell(4K) - \ell(16K)$; negative values indicate harm from the added prefix.

| Release | 4K NLL | 16K NLL | $U(4K, 16K)$ |
|---|---|---|---|
| Static | 3.24371 | 3.54001 | $-0.29631$ |
| $0.5T$ | 3.24414 | 3.35891 | $-0.11476$ |
| $0.75T$ | 3.24517 | 3.32159 | $-0.07642$ |
| $0.9T$ | 3.24518 | 3.30443 | $-0.05924$ |
| $0.95T$ | 3.24569 | 3.30108 | $-0.05539$ |
| $0.975T$ | 3.24631 | 3.30113 | $-0.05482$ |

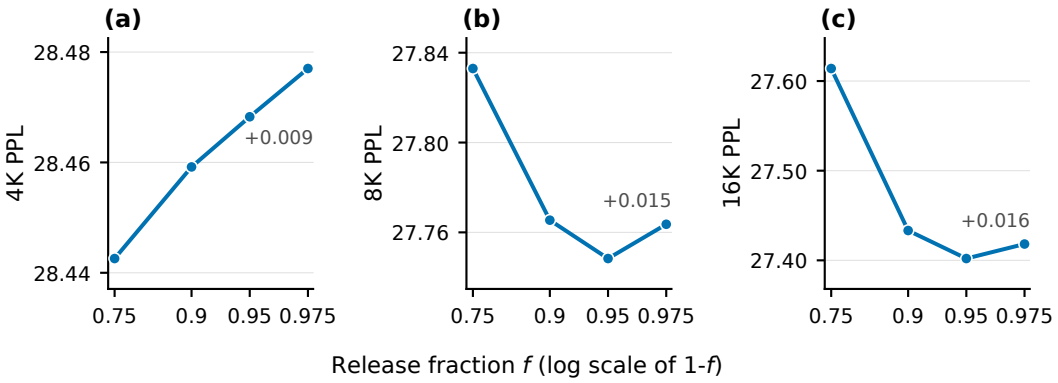



Figure 9: Late-release portion of the measured main-protocol response (custom YaRN-family, base 0, continuation seed 1); (a) 4K, (b) 8K, (c) 16K PPL. The x axis is logarithmic in the target-phase share $1 - f$ (1000/400/200/100 target updates). Lines connect observations; numbers give the PPL change from 0.95 to 0.975, and the small 8K/16K reversal is visible at this scale. One seed, so no error bars; Table 30 reports all six conditions.

loss penalty decreases from 0.11476 at 0.5 to 0.07642 at 0.75, 0.05924 at 0.9, 0.05539 at 0.95, and 0.05482 at 0.975. The 0.95-versus-0.9 16K NLL gain is 0.00335 despite a 4K cost of 0.00050, again locating the relative improvement in reduced added-context harm. Absolute mean utility remains negative at every measured release. This evaluation uses the same targets across arms and input lengths, but different targets from the main PPL sweep. Both protocols support a small long-input refinement; neither identifies a universal recovery time or establishes that 200 updates suffice for other targets, budgets, or base models. All varied fractions use one continuation seed and the same validation windows. Continuation-seed replications of 0.95 appear in Appendix N.6; selection and evaluation on this curve alone do not constitute an independently validated release-selection policy.

**Absolute loss and added-context utility need not select the same release.** Although 0.975 has slightly less negative 4K-to-16K utility than 0.95, its nested 16K NLL is 0.0000566 higher and its 4K NLL is 0.0006260 higher. The apparent utility improvement therefore does not imply a better long-input prediction loss. This is another reason to assess a readiness or recovery signal against final losses at the intended lengths, rather than optimize a prefix-utility difference alone.

**Selecting among the measured fixed schedules.** For scored-token fraction $p$ at 16K, define $R_f(p) = (1-p)\ell_f(L_s) + p\ell_f(16K)$, taking $\ell_f(L)$ as the main full-window validation NLL (log PPL) of schedule $f$ at length $L$. We compute the exact lower envelope over all six schedules under this metric. For $L_s = 4096$, the empirical optimum changes as shown in Table 32; equality boundaries are ties. For 8K/16K mixtures, 0.95 minimizes the measured NLL for every $p \in [0, 1]$. Release 0.975

Table 32: Single-seed empirical risk minimum for 4K/16K scored-token mixtures, main full-window sweep. Percentages are token proportions, not request frequencies.

| 16K scored-token fraction | Selected release |
|---|---|
| 0.000–0.637% | Static |
| 0.637–1.378% | $0.5T$ |
| 1.378–8.162% | $0.75T$ |
| 8.162–21.890% | $0.9T$ |
| 21.890–100.000% | $0.95T$ |

lies outside the three-length Pareto frontier. Boundary positions are probe-dependent: under the identical-target nested probe, the 4K/16K boundaries sit at 0.240%, 1.879%, and 13.052% with no segment owned by 0.75T (0.5T is succeeded directly by 0.9T), while the 8K/16K dominator and the 0.975 Pareto exclusion are unchanged. These are descriptive selection regions on the same validation

observations used to compare schedules; they have no population confidence bounds and are not an independently tested policy.

**Coverage on the separate full/local probe.** On seed 1, 0.95 over 0.9 gives 16K full-input gain 0.00486 nats/token, local gain 0.00038, and utility gain 0.00448 on the separate fixed full/local window set. The full-input gain is positive on 24/32 windows; at 4K the mean full-input gain is $-0.00097$. This supports the average tradeoff without window-wise dominance. These targets differ from the nested-input probe. The continuation-seed 2/3 replications are reported separately from seed 1 in Appendix N.6; all share one pretrained base. The probe traces saved every 1000 updates (Appendix P.5) miss the release, so they cannot identify release-local recovery times or validate an online readiness signal.

### N.4 Temperature-recipe anchors for the pairing gain

Table 2 aligns the three positional recipes under an identical continuation protocol (constant LR $4.964\times10^{-5}$, 25,000 pretraining updates at length 1024 followed by 4000 continuation updates at 4096, base 1, warmup 0), so that the only differences across rows are the frequency interpolation scheme and the attention temperature. DEPA's 16K endpoint moves little across recipes while static's endpoint moves by nearly forty times as much in PPL terms: the pairing gain tracks the recipe's static-side extrapolation weakness rather than any single temperature value, and the custom-temperature main line of this paper is the smallest-gain recipe measured. Each row is one seed pair; all rows are descriptive, not population estimates. On seeds $44/45$ the PI and published rows share warm start, budget, learning rate and protocol, so those rows are paired per seed: within each seed the PI static endpoint and pairing gain both exceed the published ones, while the two DEPA endpoints sit within 2.29 PPL of each other where the static endpoints differ by up to 72.47 PPL; at 8K the same-seed endpoint separation is at most 1.09 PPL (Figure 10). The boost also persists away from the 16K cusp: on the same shared seeds, PI's 8K pairing gain exceeds the published recipe's by $14.38 \pm 0.33$ PPL (range 14.14–14.61 PPL), versus $67.49 \pm 3.92$ PPL at 16K. The same-seed boost is an exact lengthwise decomposition — static-side endpoint gap minus delivered delayed separation — which can be re-derived from the per-run records for every shared seed at both lengths: on seed 44 at 16K the gap is 72.47 PPL against a separation of 2.21 PPL, and at 8K the shared-seed gaps are $15.38 \pm 0.44$ PPL (range 15.07–15.69 PPL) against separations of at most 1.09 PPL. The PI-side improvement is therefore accounted for entirely by the static-side gap: a cross-length property of the static-side weakness, not a 16K-cusp artifact. On the shared-tail natural probe the same-seed recipe contrast decomposes the same way: with window coordinates identical across recipes (seed 45), the PI-minus-published static 16K utility gap is $-0.5535$ nats/token while the DEPA-utility separation is $-0.2559$, and the post-release residual sits in the added-context component (full-input separation $+0.2964$ versus local-input separation $+0.0405$), with both orderings holding on all 32 windows. This tail-probe residual is a separate measurement from the full-window endpoint separation above; with one probe-shared seed it is descriptive. Table 2 itself appears in Section 5.2.

### N.5 Band targets and recovery for the three-recipe comparison

Two decompositions that need no additional training support Table 2. Table 33 gives the exact per-band target frequency multipliers of the three recipes as implemented in the training code ($s$=4, 32 rotary pairs in 8 contiguous bands; values are the multiplier of the first pair of each band). PI compresses every pair by $4\times$ (0/32 pairs kept at unity); published YaRN keeps 6/32 (the finest pairs, then an NTK-by-parts ramp); the custom recipe keeps 25/32 pairs unchanged and scales only the seven longest-wavelength pairs. Trained-length cost therefore tracks fine-band preservation: PI rows pay up to 1.10 PPL at 4K against at most 0.18 PPL for published rows and 0.02 PPL for the custom anchor. Table 34 reads the recorded 16K probes around the switch: the 16K state at update 3000 of 4000 (the last pre-switch probe) versus the endpoint, for every delayed arm in Table 2. Pre-switch states already cluster across recipes, and the 400-update recovery does not track the recipe ordering; the endpoint clustering of DEPA is thus consistent at two points (the state before the switch and the state after recovery), while the static endpoints span about 160 PPL. Table 35 adds the behavioral axis to the same arms: the mean natural-context utility $U = \ell^{\mathrm{local}} - \ell^{\mathrm{full}}$ (positive = added context helps) of every Table 2 run that has a natural-context evaluation, read per evaluation length under the shared probe protocol (probe seed 57036, 32 windows, local input 1024 tokens, 128-token scored

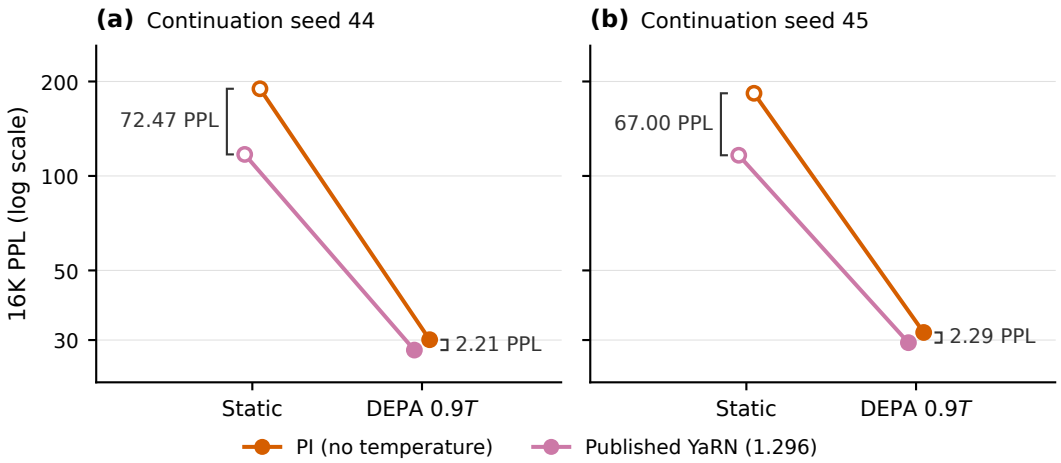



Figure 10: Same-seed pairing of the published-YaRN (purple) and PI (vermillion) arms under the matched continuation protocol of Table 2 (16K PPL, log scale; one panel per shared seed). Open markers: static arms (target frequencies from update 0); filled markers: DEPA $0.9T$ (switch at update 3600 of 4000). Within every seed the delayed endpoints sit within 2.29 PPL of each other while the static endpoints differ by up to 72.47 PPL (brackets give the per-seed PI − published-YaRN endpoint gap in PPL on each side); at 8K the endpoint separation is at most 1.09 PPL. The two recipes are offset slightly along x for visibility. Values are the per-run endpoints of Table 2.

Table 33: Recipe-side band targets at $s$=4 (as implemented in the trainer; target frequency multiplier $\mu$ of the first pair of each band; the custom recipe's band 6 straddles its blend boundary, pairs 25–27 being 0.250).

| | | Target multiplier $\mu$ | | |
|---|---|---|---|---|
| Band | Wavelength (tokens) | PI | Custom YaRN-family | Published YaRN |
| 0 | 6.3–14.9 | 0.250 | 1.000 | 1.000 |
| 1 | 19.9–47.1 | 0.250 | 1.000 | 1.000 |
| 2 | 62.8–149.0 | 0.250 | 1.000 | 0.620 |
| 3 | 198.7–471.2 | 0.250 | 1.000 | 0.350 |
| 4 | 628.3–1490.0 | 0.250 | 1.000 | 0.265 |
| 5 | 1986.9–4711.7 | 0.250 | 1.000 | 0.250 |
| 6 | 6283.2–14899.8 | 0.250 | 1.000 | 0.250 |
| 7 | 19869.2–47117.2 | 0.250 | 0.250 | 0.250 |

tails, evaluation only). At the trained length every row is positive and at most 0.012 nats/token, so the recipes are behaviorally indistinguishable where they were trained. Under the static targets the long-length utility falls in exactly the hardness order of Table 2 — custom stays near neutral ($-0.347$), the published-YaRN rows sit between $-1.836$ and $-1.847$ at 8K and between $-2.427$ and $-2.497$ at 16K, and the PI rows sit between $-3.102$ and $-3.050$ — adding context toward an unreachable target is actively harmful relative to truncating it. Every delayed row returns to a narrow band: $-0.030$ to $-0.307$ at 16K across all three recipes, and the published-recipe delayed rows are the closest to zero. The published-recipe seed-44 pair has no utility rows: its natural-context evaluation files were not retained (Appendix D). The natural-text readouts of the token-level probes elsewhere in this appendix use a different (per-token) window set and are not comparable row-wise with this table.

### N.6 Continuation-seed replication of the $0.95T$ refinement

The seed-1 response curve of Appendix N.3 motivated an unchanged $0.95T$ schedule on continuation seeds 2 and 3, paired with their $0.9T$ runs. Both follow-up seeds are reported. All comparisons retain base 0, the constant LR, 4000 updates, 131.072M continuation tokens, the 2% mixture, and the custom-YaRN-family endpoint. All three seeds in Table 36 favor $0.95T$ at 8K/16K and $0.9T$ at

Table 34: Pre-switch 16K states (update 3000 of 4000) and 400-update recoveries for the delayed arms of Table 2.

| recipe | seed | pre-switch 16K | endpoint 16K | recovery |
|---|---|---|---|---|
| published | 44 | 51.50 | 27.870 | 23.63 |
| published | 45 | 50.64 | 29.427 | 21.21 |
| published | 46 | 54.88 | 27.667 | 27.21 |
| PI | 44 | 48.82 | 30.076 | 18.74 |
| PI | 45 | 48.52 | 31.714 | 16.81 |
| PI | 2 | 47.20 | 29.223 | 17.98 |
| custom | 5 | 49.14 | 28.395 | 20.75 |

Table 35: Natural-context utility $U$ (local minus full NLL, nats/token; positive = added context helps) by evaluation length, for the Table 2 arms. All runs share the same probe protocol (probe seed 57036, local input 1024, 32 windows, scored tail 128).

| recipe | seed | arm | $U$@4K | $U$@8K | $U$@16K |
|---|---|---|---|---|---|
| PI | 2 | static | 0.007 | −2.401 | −3.064 |
| PI | 44 | static | 0.008 | −2.411 | −3.102 |
| PI | 45 | static | 0.008 | −2.438 | −3.050 |
| published | 45 | static | 0.012 | −1.847 | −2.497 |
| published | 46 | static | 0.009 | −1.836 | −2.427 |
| custom | 5 | static | 0.001 | −0.044 | −0.347 |
| PI | 2 | delayed | 0.009 | −0.010 | −0.307 |
| PI | 44 | delayed | 0.010 | 0.001 | −0.267 |
| PI | 45 | delayed | 0.009 | 0.001 | −0.286 |
| published | 45 | delayed | 0.011 | 0.017 | −0.030 |
| published | 46 | delayed | 0.010 | 0.010 | −0.031 |
| custom | 5 | delayed | 0.002 | −0.012 | −0.072 |

4K. The mean 16K PPL improvement is 0.03565 (sample s.d. 0.00454); the corresponding paired NLL gain is 0.001284 (sample s.d. 0.000169). The mean 4K cost is 0.00850 PPL. These are small consistent tradeoffs, not uniform improvements at every length. The initiating and follow-up seeds are kept distinct, and three continuation seeds do not provide independent-base replication. The full-input and utility gains retain their signs on both follow-up seeds (Table 37), with small window-wise reversals. These measurements concern paired available-prefix effects, not retrieval competence.

**Transferring the fixed two-action workload threshold.** The seed-1 threshold was recorded before consuming the follow-up endpoints. Retaining it, the 4K/16K workload selects $0.95T$ above a 16K scored-token fraction of 0.21890 and $0.9T$ below it. We do not refit that choice to the follow-up seeds. The transferred choice (Table 38) disagrees only between each follow-up threshold and 0.21890, where it retains $0.9T$. For 8K/16K mixtures all three seeds prefer $0.95T$ throughout. This is a continuation-seed transfer check of a fixed workload-conditioned choice, restricted to two schedules. It does not validate the full six-action seed-1 envelope or a model-state-dependent online release rule. The separate independent-base application and its transfer limit appear in Appendix O.4; no new dataset is tested.

# O Release-time measurements and transfer

## O.1 Direct fixed-window measurements around release

A separate custom-YaRN-family, base-0, continuation-seed-1 pair measures the $0.95T$ and $0.975T$ schedules directly at constant LR. Both traces and final language-model evaluations are complete. Two fixed PG19 training windows of 4096 tokens (private RNG 57047) are scored under source and target multipliers at the same weights and temperature. Every token is scored; these are training anchors, not held-out examples. Measurements restore model mode and Torch RNG state, use a

Table 36: $0.9T$/$0.95T$ pairs. Seed 1 initiated the refinement; seeds 2 and 3 are follow-up continuation seeds on the same pretrained base.

| Seed | Release | 4096 PPL | 8192 PPL | 16384 PPL |
|---|---|---|---|---|
| 1 | $0.9T$ | 28.4592 | 27.7654 | 27.4333 |
| 1 | $0.95T$ | 28.4683 | 27.7483 | 27.4020 |
| 2 | $0.9T$ | 28.5586 | 29.2285 | 27.4337 |
| 2 | $0.95T$ | 28.5681 | 29.2208 | 27.3934 |
| 3 | $0.9T$ | 29.3862 | 28.6477 | 28.5464 |
| 3 | $0.95T$ | 29.3931 | 28.6352 | 28.5110 |

Table 37: Natural 16K full/local probe: $0.95T$ gains over $0.9T$, nats/token. Positive gains favor $0.95T$. Windows are shared across seeds and may overlap.

| Seed | Full gain | Local gain | Utility gain | Full wins/32 |
|---|---|---|---|---|
| 1 | 0.00486 | 0.00038 | 0.00448 | 24 |
| 2 | 0.00442 | −0.00085 | 0.00527 | 28 |
| 3 | 0.00469 | 0.00106 | 0.00363 | 27 |

private NumPy generator, and neither update weights nor select the release. The training budget is 4000 updates and 131.072M tokens per arm.

Write $S_\tau = \ell_S(W_\tau)$ for the source loss just before release and define

$$E_\tau(n) = \ell_T(W_{\tau+n}) - S_\tau, \qquad J_\tau(n) = \ell_T(W_{\tau+n}) - \ell_S(W_{\tau+n}). \tag{16}$$

Both use the same two windows. $E_\tau$ measures excess over a fixed, pre-release reference; $J_\tau$ compares configurations at current weights. The first sampled return of $E_\tau$ to zero is a descriptive crossing, not a fitted recovery time or an online decision: it is observed only after the release has already occurred. For $0.95T$, immediately before release at $\tau = 3800$, source/target NLL is 3.351130/3.363836, giving an initial increment 0.012705 nats/token. For $0.975T$, immediately before release at $\tau = 3900$, source/target NLL is 3.344025/3.357102, giving an initial increment 0.013077 nats/token.

Both schedules rapidly reduce their initial switch increment (Table 39). The first sampled return of the mean to pre-release source loss occurs at 80 target updates for $0.95T$ and 40 for $0.975T$. Neither trace is monotone: the former has four sampled increases in nine intervals, the latter two in seven. At $0.975T$, the mean excess is $-0.001450$ after 40 updates but rebounds to $+0.006807$ at the final 100-update state. At $0.95T$, the final 200-update excess is $-0.000418$. Both current-weight differences $J_\tau$ become negative after five updates, while both excesses $E_\tau$ remain positive. Current-weight target preference, first return to an earlier source loss, and sustained recovery are distinct (Figure 11).

**Reference changes matter when comparing recovery.** The later arm's pre-release source reference is lower by 0.007105 nats/token. At the common final training step, the difference in excesses is therefore much larger than the difference in absolute target losses:

$$\begin{aligned}\ell_T(W_T^{.975}) - \ell_T(W_T^{.95}) &= (S_{3900}^{.975} - S_{3800}^{.95}) + (E_{3900}^{.975}(100) - E_{3800}^{.95}(200)) \\ &= -0.00710523 + 0.00722563 = 0.00012040.\end{aligned} \tag{17}$$

The displayed terms are rounded; the identity uses full-precision values. A larger residual relative to a better pre-release source state is not the same quantity as a larger absolute final loss. At the common step 3900, the later arm has yet to adapt to the target and its target NLL exceeds the earlier arm's by 0.012737; by step 4000 this difference is only 0.000120. Thus the 100-update target phase largely closes that observed gap. These two training windows do not identify the optimum for held-out 16K prediction.

**Means can conceal window-specific reversals.** At $0.95T$, individual first sampled returns occur at 80 and 20 updates; both windows are below their references at update 80, but their final excesses are $-0.001763$ and $+0.000926$. At $0.975T$, individual first returns are 40 and 20 updates; no sampled state places both windows below their respective references, and final excesses are $+0.005064$ and

Table 38: Follow-up evaluation of the seed-1 two-action threshold. Excess loss is relative to each follow-up seed's descriptive best of $0.9T$/$0.95T$, maximized over the scored-token mixture.

| Follow-up seed | Best-of-two threshold | Maximum excess NLL |
|---|---|---|
| 2 | 0.18469 | 0.00006169 |
| 3 | 0.15916 | 0.00008819 |

Table 39: Complete post-release mean traces on the fixed training anchors. Dashes lie beyond the corresponding training budget. Both excess and signed jump are in nats/token.

| | $0.95T$, $B = 200$ | | $0.975T$, $B = 100$ | |
|---|---|---|---|---|
| Target updates $n$ | $E_\tau(n)$ | $J_\tau(n)$ | $E_\tau(n)$ | $J_\tau(n)$ |
| 0 | 0.012705 | 0.012705 | 0.013077 | 0.013077 |
| 1 | 0.009623 | 0.010276 | 0.010955 | 0.010909 |
| 5 | 0.005450 | −0.004368 | 0.006530 | −0.005025 |
| 10 | 0.000587 | −0.001915 | 0.004661 | −0.001749 |
| 20 | 0.000858 | −0.002380 | 0.000885 | −0.002762 |
| 40 | 0.004006 | −0.001720 | −0.001450 | −0.002160 |
| 80 | −0.000663 | −0.003332 | 0.004696 | −0.002828 |
| 100 | −0.006765 | −0.002243 | 0.006807 | −0.001891 |
| 160 | −0.001196 | −0.003694 | — | — |
| 200 | −0.000418 | −0.002430 | — | — |

$+0.008551$. A mean crossing therefore certifies neither simultaneous nor sustained window-wise recovery. No improvement in held-out loss from a stricter window-wise rule is established.

**Held-out endpoints and execution variation.** The instrumented $0.975T$ arm has a faster first sampled mean return but worse 16K PPL by 0.01056 than its paired $0.95T$ arm. This rules out treating that crossing statistic as a sufficient ranking criterion in this pair. The 4K ranking also favors $0.95T$, whereas 8K slightly favors $0.975T$ (Table 40).

Table 40: Held-out main PPL and recorded probe cost for the instrumented seed-1 pair. Cost covers measured regions and excludes JSON writes.

| Release | 4K | 8K | 16K | Forward calls | Seconds |
|---|---|---|---|---|---|
| $0.95T$ | 28.4738 | 27.7672 | 27.4074 | 56 | 7.17 |
| $0.975T$ | 28.4761 | 27.7607 | 27.4180 | 52 | 6.62 |

The uninstrumented sweep of Appendix N.3 instead favors $0.95T$ at all three lengths, with $0.975T$ penalties 0.00875/0.01526/0.01618 PPL at 4K/8K/16K. The instrumented penalties are 0.00233/$-0.00650$/0.01056. Thus the small 4K/16K ordering recurs in these two executions of seed 1, while the 8K ordering does not; this is not a second independent continuation seed.

Numerical variation also occurs before measurement: the uninstrumented and instrumented $0.95T$ runs re-execute identical arguments and RNG seeds, agree bitwise through the last 1024-token update (25000), and first differ at global update 25100 — the first logged 4096-token update — before the first anchor at 28200 (maximum logged pre-anchor difference 0.00798). Within the instrumented pair, the largest absolute target-NLL difference at common pre-release anchors is 0.0000701. The drift therefore originates in execution nondeterminism of the 4096-token stage; the exact kernel is not isolated, and its presence before measurement does not prove absence of later probe effects. We keep these instrumented endpoints separate from the uninstrumented three-seed refinement estimates and do not claim bitwise-equivalent executions.

During the $0.95T$ hold, signed increments at completed updates 3200/3400/3600/3700/3800 are 0.01142/0.01092/0.00946/0.00991/0.01271; the paired later arm reaches 0.01308 at its step-3900 release. This signal does not decrease monotonically with longer hold. The two schedules use the same

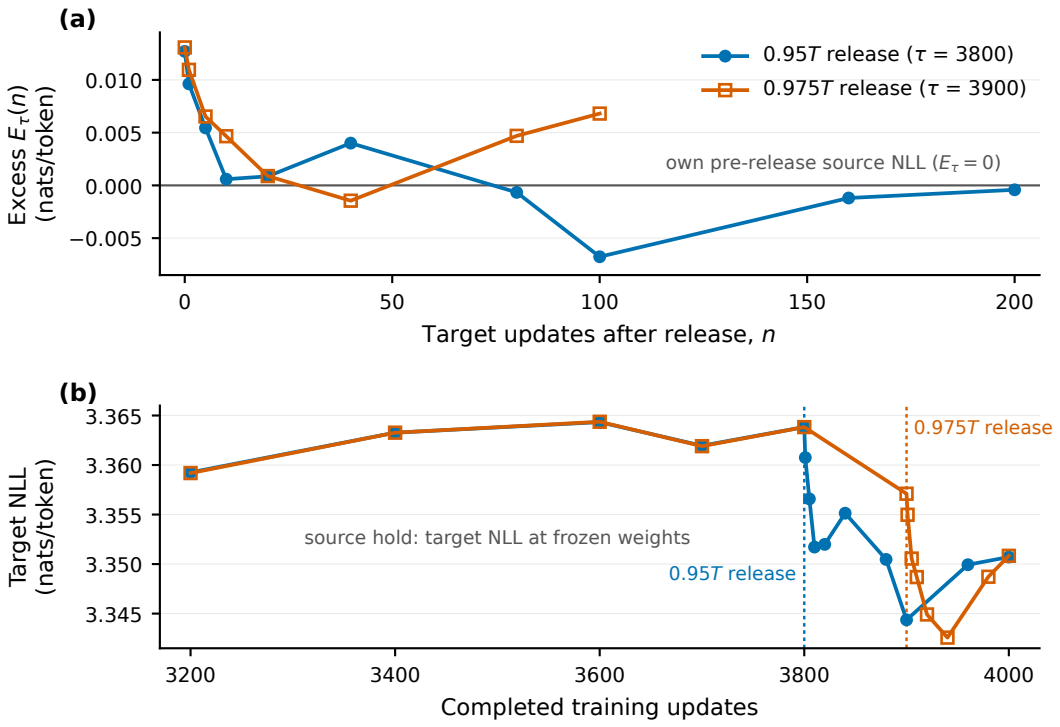



Figure 11: Paired fixed-training-window traces without a recovery fit. (a) Excess $E_\tau(n)$ (Eq. 16) relative to each schedule's own pre-release source loss, aligned by number of target updates. (b) Absolute target-configuration losses at the same completed training updates, including frozen-weight target evaluations during the source hold, where the two arms coincide (filled circles inside open squares). Vertical dotted lines mark the two releases. Lines connect recorded samples; they do not imply monotone interpolation.

measurement procedure, but the release-relative grid gives different forward counts. Training tokens are matched; total measurement compute is not identical. Taken together, the paired measurements motivate preserving absolute losses, individual-window traces, and remaining target budget when assessing a future state-based release rule. They do not validate such a rule or a universal contraction rate.

### O.2 Late-boundary replication and dependence on scored targets

We repeat the instrumented $0.95T/0.975T$ pair on continuation seed 2 of base 0. Both arms have the same constant LR, custom-YaRN-family target, 4000 training updates, and training/evaluation protocols as the instrumented seed-1 pair in Appendix O.1. All four endpoints and traces are complete. We aggregate only these two instrumented pairs; the uninstrumented seed-1 sweep remains a separate execution.

Table 41: Complete instrumented boundary pairs. PPL uses the main length sweep; 4000 total updates leave 200 or 100 target updates at the respective releases.

| Seed | Release | 4K | 8K | 16K |
|---|---|---|---|---|
| 1 | $0.95T$ | 28.47381 | 27.76725 | 27.40740 |
| 1 | $0.975T$ | 28.47614 | 27.76075 | 27.41796 |
| 2 | $0.95T$ | 28.56961 | 29.22087 | 27.39383 |
| 2 | $0.975T$ | 28.57265 | 29.23205 | 27.40331 |

The main 16K preference for $0.95T$ recurs (Table 41): its gains are 0.01056 and 0.00948 PPL, with mean 0.01002 and sample s.d. 0.00077 (mean NLL gain 0.0003657, sample s.d. 0.0000278). Both pairs also favor $0.95T$ at 4K, by 0.00233/0.00304 PPL. At 8K, the signs differ: seed 1 favors $0.975T$ by 0.00650, seed 2 favors $0.95T$ by 0.01118. These are paired observations on one pretrained base with reused evaluation targets, not population uncertainty estimates or a universal optimum over release fractions.

**Better measured recovery need not give the better endpoint.** Seed 2 gives a sharper distinction than seed 1 (Table 42). Its $0.975T$ arm has the smaller initial switch increment, a first sampled mean return at 40 target updates, and both individual windows below their references at the final 100-update state. The $0.95T$ mean has no sampled return within 200 updates, yet its main 16K endpoint is better. Even strengthening a mean crossing to require both final windows below their references would favor the worse main endpoint in this pair.

Table 42: Training-anchor statistics. A dash means no sampled mean return within the available budget, not an inferred return time beyond that budget. Final excess uses each schedule's own pre-release source reference.

| Seed | Release | Initial jump | First return | Final excess | Final windows $\leq 0$ |
|---|---|---|---|---|---|
| 1 | $0.95T$ | 0.012705 | 80 | $-0.000418$ | 1/2 |
| 1 | $0.975T$ | 0.013077 | 40 | 0.006807 | 0/2 |
| 2 | $0.95T$ | 0.014223 | — | 0.007347 | 0/2 |
| 2 | $0.975T$ | 0.011285 | 40 | $-0.001294$ | 2/2 |

The reference-aware decomposition is again essential. For seed 2, the later arm's pre-release source loss is *higher* by 0.00893211 nats/token, and its final excess is lower by 0.00864100. Their sum is a positive absolute final target-loss difference of 0.00029111. Thus a negative final excess under the later arm's easier reference coexists with a slightly worse absolute loss. For seed 1 the reference difference has the opposite sign. A reference-dependent recovery statistic cannot be compared across release times as though it were an absolute endpoint objective. Seed-2 recorded probe costs are 56 calls/7.11 seconds and 52 calls/6.59 seconds, respectively, excluding JSON writes; training tokens remain matched.

**The small boundary effect depends on which targets are scored.** The main sweep scores all positions in length-specific validation windows. The natural full/local protocol scores the final 128-token tails of a separate set of 32 windows; the nested protocol scores another fixed set of 32 target blocks across input lengths. The 16K penalties of moving from $0.95T$ to $0.975T$ are given in Table 43.

Table 43: Late-release NLL penalty at 16K; positive values favor $0.95T$. These are distinct evaluators, each internally paired between schedules.

| Seed | Main full-window | Natural tail | Nested tail |
|---|---|---|---|
| 1 | 0.0003853 | $-0.0025831$ | 0.0002696 |
| 2 | 0.0003460 | $-0.0000802$ | $-0.0001720$ |

The natural tail mean favors $0.975T$ in both pairs, though its seed-2 advantage is only 0.0000802 nats/token. The nested sign differs across seeds. In each nested pair, only 17/32 windows favor $0.95T$ in absolute 16K loss. Accordingly, the main-sweep turning point does not establish an all-protocol 16K optimum. Unlike the larger static-to-DEPA effect, this very small refinement is sensitive to scored targets. These observations also motivate aligning a prospective state signal with the intended objective instead of transferring a 4K training-anchor recovery threshold.

**Prospective long-input state candidate.** The target-trend candidate uses two fixed 16K PG19 *training* windows drawn with private RNG 57053 and scores all 16384 next-token positions under the target configuration at frozen current weights. Let $D_t$ be their mean target NLL after $t$ completed

training updates. It records $D_{3400}$ during the source hold. At $t = 3600$, it releases before the next optimizer update if $D_{3600} \geq D_{3400}$; otherwise it continues holding. If still holding at $t = 3700$, it releases when $D_{3700} \geq D_{3600}$. Any remaining hold ends at $t = 3800$. Thus there is exactly one release in $\{0.9T, 0.925T, 0.95T\}$, with 200–400 target updates remaining. The minimum reserve is a fixed design bound motivated by the main-sweep boundary, not a fitted recovery time.

The paired fixed-$0.95T$ control uses the same five measurement times 3400/3600/3700/3800/4000 and two target-only forward passes per measurement; post-release and final measurements never alter the candidate's decision. Both arms retain constant LR, 131.072M training tokens, the same target and temperature, and identical final evaluators. The probe restores model mode and Torch RNG state and uses no validation examples or optimizer updates. Elapsed measurement cost and the realized release are recorded. Paired tests use continuation seed 4 on bases 0 and 1 and are reported in full (Appendix O.3). The rule is fixed before its held-out outcomes; this remains an exploratory causal decision rule with matched probe counts, not an established adaptive improvement.

### O.3 Realized target-trend decisions on two bases

The prospective target-trend rule from Appendix O.2 has complete paired tests on bases 0 and 1, each with continuation seed 4. The rule is unchanged between bases. Both candidates release at update 3800, just as their paired fixed controls do, leaving 200 target updates. These pairs establish the realized decisions, but do not compare different candidate-versus-control positional schedules.

Table 44: Actual target-trend observations. Both candidates continue holding at 3600 and 3700, release at the fixed deadline 3800, and only observe at 4000.

| Completed updates | Base 0 target NLL | Base 1 target NLL |
|---|---|---|
| 3400 | 3.448718 | 3.449991 |
| 3600 | 3.439810 | 3.439054 |
| 3700 | 3.430738 | 3.432177 |
| 3800 | 3.433636 | 3.434903 |
| 4000 | 3.388089 | 3.386552 |

From 3400 to 3600 and then 3700 updates (Table 44), base-0 mean target NLL decreases by 0.008907/0.009072 nats/token; the base-1 decreases are 0.010938/0.006876. Both satisfy the rule's condition for continued hold. The means then increase by 0.002897/0.002725 at 3800, when the deadline already requires release. All decisions use only the two fixed 16K training windows at current frozen weights, without held-out outcomes or retrospective selection. The rule preserves the fixed deadline in both observed cases; this does not show that its optional early-release branches improve losses.

Table 45: Main endpoints. Candidate and fixed-$0.95T$ realize the same release on each base. Both fixed-$0.9T$ controls are complete, using the same seed, probe grid, and final evaluators.

| Base | Arm | Release | 4K PPL | 8K PPL | 16K PPL |
|---|---|---|---|---|---|
| 0 | Fixed $0.9T$ | 3600 | 28.50710 | 29.45663 | 29.55022 |
| 0 | Fixed $0.95T$ | 3800 | 28.51292 | 29.43297 | 29.48213 |
| 0 | Target trend | 3800 | 28.51340 | 29.42974 | 29.47733 |
| 1 | Fixed $0.9T$ | 3600 | 28.67957 | 29.49646 | 29.48661 |
| 1 | Fixed $0.95T$ | 3800 | 28.70184 | 29.52678 | 29.50995 |
| 1 | Target trend | 3800 | 28.69839 | 29.52133 | 29.50709 |

Candidate-minus-control numerical outcomes (Table 45) do not identify timing gains: base 0 shows a 0.00480-PPL 16K advantage at a 0.00048-PPL 4K cost; base 1 shows numerical gains of 0.00345/0.00545/0.00286 PPL at 4K/8K/16K. Within each pair the actual source/target schedule is identical. Logged training losses first differ at global update 25100 — the same first logged 4096-token update as the instrumented anchor pair — before the first probe at 28400, on both bases. The maximum logged pre-probe differences are 0.003057/0.006734. This shared onset is consistent with

execution nondeterminism of the 4096-token stage rather than a protocol difference; the exact kernel is not isolated, and these observations neither exclude later probe effects nor estimate population execution uncertainty.

Each arm uses ten target-only forward calls: two 16K windows at five predetermined updates. Recorded measurement times are 4.40308/4.40359 seconds for base-0 fixed/candidate and 4.41537/4.41510 for base 1, excluding JSON writes. Each scores 163840 additional training-data next-token positions without gradient updates, separately from 131.072M optimizer-training tokens.

**Comparison with a fixed $0.9T$ reference.** The base-0 seed-4 fixed-$0.9T$ arm uses the same training/evaluation recipe and measurement grid (ten forwards, 4.41780 recorded seconds). Moving to fixed $0.95T$ improves 8K/16K PPL by 0.02366/0.06809, at a 0.00582-PPL 4K cost. Its 16K NLL gain is 0.002307. This reproduces the direction of the base-0 seed-1 refinement (Appendix N.3) on another continuation seed, with measurement compute matched across these arms. The candidate, which also chooses $0.95T$, gains 0.07289 PPL at 16K over $0.9T$; the additional numerical difference from fixed $0.95T$ is not an adaptive timing effect. We keep this instrumented seed separate from the uninstrumented three-seed estimate.

The base-1 fixed-$0.9T$ control is also reported. Here fixed $0.95T$ instead costs 0.02227/0.03032/0.02334 PPL at 4K/8K/16K; the candidate, also choosing $0.95T$, costs 0.01882/0.02487/0.02048. The base-1 control uses ten forwards and 4.41778 recorded seconds. Thus the three-arm comparison exposes a limit of the state signal: similar signs of the two pre-release target-loss trends lead to the same choice on both checkpoints, although the main endpoint preference between fixed $0.9T$ and $0.95T$ differs. The two checkpoint-specific fixed-refinement 16K gains differ by 0.09143 PPL, or 0.003098 nats/token. Continuation seed, probe scheme, and final evaluators match in this comparison; only two checkpoints are observed, so this is not a population interaction estimate.

The final evaluators also expose different aspects of the tradeoff (Table 46): On base 0, the natural

Table 46: Seed-4 NLL gain of fixed $0.95T$ over $0.9T$ at 16K. Positive values favor the later release; the three columns use distinct target sets.

| Base | Main full-window | Natural tail | Nested tail |
|---|---|---|---|
| 0 | 0.002307 | 0.005713 | 0.005345 |
| 1 | −0.000791 | −0.001054 | 0.000437 |

full-input gain occurs despite a 0.001700 local cost, with utility improvement 0.007413 and 23/32 full-input windows improving. Nested 16K loss improves on 29/32 windows; the 4K-to-16K prefix-effect improvement is 0.005394, positive on 30/32. This is a consistent fixed refinement across the three scoring protocols in this seed. On base 1, natural full-input loss worsens by 0.001054 despite a utility improvement of 0.002712, because local-input loss worsens by 0.003766. Twenty of 32 natural windows nevertheless improve, illustrating why a window win count alone does not determine the mean. The nested 16K gain is only 0.000437, with 19/32 windows improving and prefix-effect gain 0.000647. Improved added-context utility therefore remains distinct from improved absolute long-input prediction loss.

Both fixed controls and both state-policy pairs are reported. The rule has not demonstrated an advantage from adapting the release time; its actual choices match $0.95T$ on both bases and lose to $0.9T$ on base 1 under the main evaluator. The fixed late refinement remains useful on base 0, while transfer requires validation on the intended checkpoint and scoring objective. The state rule as specified and all executions are reported; no post-hoc threshold is presented as a newly validated method.

### O.4 Late-refinement transfer to an independently pretrained base

We test the unchanged $0.9T$/$0.95T$ pair on independently pretrained base 1 with continuation seeds 2 and 3, custom YaRN-family, constant LR, and 4000 updates (131.072M tokens) per arm. All four traces and final evaluations are reported. Both schedules use the fixed-training-window measurement procedure from Appendix O.1. This extends the refinement beyond base 0; because the base-0

refinement runs of Appendix N.6 were uninstrumented, differences between paired effects do not isolate a causal effect of the pretrained base.

Table 47: Independent-base refinement at constant LR, main PPL sweep.

| Seed | Release | 4K | 8K | 16K |
|---|---|---|---|---|
| 2 | $0.9T$ | 28.77582 | 29.32281 | 27.50020 |
| 2 | $0.95T$ | 28.78841 | 29.33668 | 27.49592 |
| 3 | $0.9T$ | 29.58105 | 28.66716 | 28.58777 |
| 3 | $0.95T$ | 29.59527 | 28.68473 | 28.59719 |

Both seeds incur trained-length and 8K costs from moving to $0.95T$ (Table 47): mean PPL costs are 0.01340 and 0.01572. The 16K effect changes sign: seed 2 improves by 0.00429 PPL, but seed 3 worsens by 0.00942. The mean 16K gain is consequently $-0.00257$ PPL (sample s.d. 0.00970), or $-0.0000869$ nats/token (sample s.d. 0.0003433). In seed 3, $0.9T$ is better at all three main evaluation lengths. The base-0 three-seed 8K/16K refinement therefore does not establish a universal $0.95T$ recommendation. This small refinement remains distinct from the much larger static-to-DEPA comparisons on the two bases.

**Testing the source choice without refitting.** We retain the base-0, seed-1 two-action rule from Appendix N.6. For a scored-token proportion $p$ at 16K, the evaluated objective is $(1-p)\ell(L) + p\ell(16\text{K})$. The source rule chooses $0.95T$ above $p = 0.21890$ for $L = 4$K and at every $p$ for $L = 8$K. Applying it unchanged gives Table 48.

Table 48: Independent-base application of the fixed source rule. The crossing is descriptive for that pair; above it the pair favors $0.95T$. "None" means $0.9T$ wins throughout. Excess NLL compares the source-selected action with the same pair's best of the two actions, over all $p \in [0, 1]$.

| Seed | Mixture | Pair crossing | Maximum excess NLL |
|---|---|---|---|
| 2 | 4K/16K | 0.73720 | 0.0003074 |
| 2 | 8K/16K | 0.75209 | 0.0004729 |
| 3 | 4K/16K | None | 0.0004476 |
| 3 | 8K/16K | None | 0.0006127 |

In seed 2, the source rule selects $0.95T$ too early in mixture proportion: it disagrees with the observed best of two on $0.21890 < p < 0.73720$ for 4K/16K and on $p < 0.75209$ for 8K/16K. In seed 3, $0.9T$ wins every mixture; the source rule disagrees whenever it selects $0.95T$ (Figure 12). Thus the seed-2 crossings are not universal replacement thresholds either. The absolute costs are small, but the action regions change substantially. $p$ denotes token share, not request frequency; this analysis does not evaluate the full six-action grid or a model-state-dependent online rule.

**Scored-target dependence at the longest length.** The natural full/local evaluation scores final 128-token tails in 32 windows per length, with the same targets under full and local input. At 16K, $0.95T$ improves full-input NLL by 0.002945/0.000635 in seeds 2/3, despite local-input costs of 0.001007/0.001102. The utility improvements are 0.003952/0.001738, with 26/32 and 16/32 full-input windows improving. Absolute mean utility remains negative: $-0.068622$/$-0.070272$ for $0.9T$, versus $-0.064670$/$-0.068535$ for $0.95T$. Both seeds have negative mean full-input gains at 4K and 8K.

The nested evaluator fixes another set of 32 final target blocks across input lengths. These targets differ from the main sweep and natural full/local evaluator; every within-pair and within-length contrast is matched (Table 49).

The nested 16K gains are 0.001588/0.000961 nats/token, with 4K costs 0.000576/0.000457. Hence the 4K-to-16K prefix-effect improvements are 0.002164/0.001418, positive on 23/32 and 18/32 windows; absolute 16K loss improves on 20/32 and 17/32. The two tail evaluators retain a small 16K refinement, while the main full-window evaluator has mixed signs. It is therefore the scoring

Table 49: Independent-base nested mean NLL on identical targets.

| Seed | Release | 1K | 4K | 8K | 16K |
|---|---|---|---|---|---|
| 2 | $0.9T$ | 3.271958 | 3.263673 | 3.287186 | 3.317795 |
| 2 | $0.95T$ | 3.272216 | 3.264249 | 3.287801 | 3.316207 |
| 3 | $0.9T$ | 3.262986 | 3.254763 | 3.280772 | 3.315586 |
| 3 | $0.95T$ | 3.263189 | 3.255220 | 3.281437 | 3.314625 |

target, as well as the continuation seed, that bounds the demonstrated benefit. Reused or overlapping windows are not independent replications, and these results do not establish retrieval gains.

**Recovery statistics also vary across seeds.** The training anchors use two fixed 4K windows, distinct from every final held-out evaluator. Their measured statistics are in Table 50.

Table 50: Independent-base training-anchor dynamics. "None" denotes no sampled mean return within the allocated target updates. Final excess is relative to each arm's own pre-release source loss.

| Seed | Release | Initial jump | First return | Final excess |
|---|---|---|---|---|
| 2 | $0.9T$ | 0.014777 | 160 | 0.003088 |
| 2 | $0.95T$ | 0.015658 | None | 0.008228 |
| 3 | $0.9T$ | 0.015390 | 200 | $-0.001995$ |
| 3 | $0.95T$ | 0.014254 | 20 | $-0.001325$ |

Seed 2 has no sampled mean return for $0.95T$ within 200 target updates, yet a small positive main 16K effect. In seed 3, the same schedule returns much earlier than $0.9T$ (20 versus 200 updates), and both of its windows are below their references at the final state, yet it loses at every main length. Thus a sampled return is neither necessary for a relative main 16K gain in these pairs nor sufficient to select the better schedule. These are sparse-sample observations, not exact first-passage times.

The reference-aware final-loss decomposition also holds. In seed 2, $-0.00515270 + 0.00513959 = -0.00001311$ nats/token; in seed 3, $-0.00070214 + 0.00066972 = -0.00003242$. Absolute final target-anchor losses are nearly equal within both pairs, despite different recovery summaries and held-out outcomes. The recorded probe costs are 60 forward calls/7.71 seconds for each $0.9T$ arm, and 56 calls/7.17 or 7.18 seconds for $0.95T$, excluding JSON writes. Same training tokens and measurement procedure do not imply identical measurement compute.

These two continuation seeds support retaining $0.9T$ as the fixed reference and evaluating any later or state-based choice on its intended scoring objective. The already specified target-trend rule from Appendix O.2 has also run unchanged on base 1, continuation seed 4, with its own matched fixed-$0.95T$ control (Appendix O.3). Both choose the fixed deadline; no adaptive timing gain is claimed.

# P Matched exposure and continuation horizon

## P.1 Matching cumulative multiplier exposure

The full-budget ramp differs from DEPA in both temporal shape and cumulative multiplier exposure. To separate these, an additional seed-1 PI control holds source frequencies through update 3200 and then uses

$$w_t = \begin{cases} 0, & t < 3200, \\ (t - 3200)/799, & 3200 \le t \le 3999, \end{cases} \qquad m_j(t) = 1 + w_t(m_j^* - 1).$$

The 800 equally spaced ramp weights include zero and one and sum to 400. DEPA $0.9T$ likewise has 400 weights equal to one and 3600 equal to zero. Thus both paths have the same training tokens, final PI configuration, and exact per-band integrated multiplier displacement $\sum_t (1 - m_j(t)) = 400(1 - m_j^*)$.

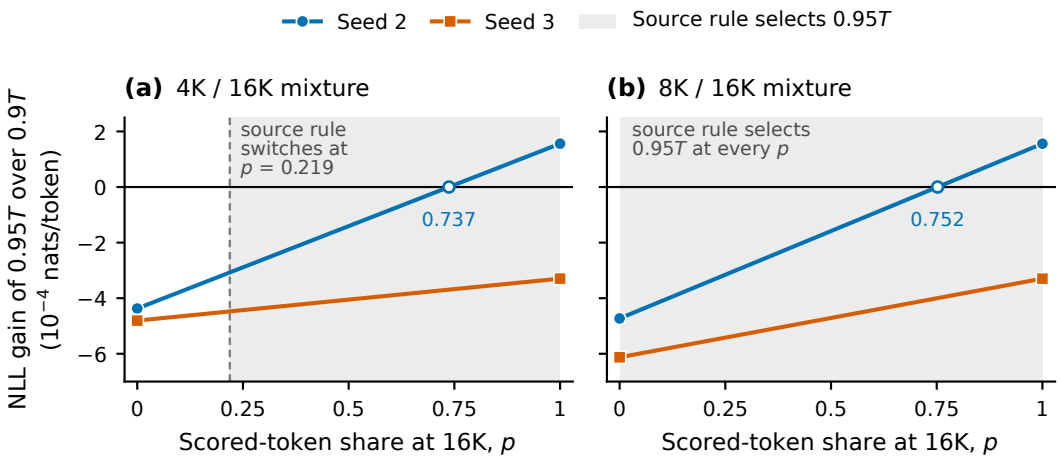



Figure 12: Actual NLL gain of $0.95T$ over $0.9T$ across two scored-token mixtures on base 1, in $10^{-4}$ nats/token. Positive values favor $0.95T$. Markers are the measured endpoints ($p = 0$: 4K or 8K only; $p = 1$: 16K only), and each line is the exact mixture arithmetic between them, not a fitted predictor. Grey shading marks where the fixed base-0 source rule selects $0.95T$: above $p = 0.219$ in (a) and at every $p$ in (b). Open circles mark the seed-2 break-even shares. One run per seed, so no uncertainty is shown.

Table 51: Equal cumulative multiplier exposure at the PI target, seeds 1–3. Both schedules have scalar weight sum 400 over 4000 updates. Positive paired gains favor the single DEPA release.

| Schedule / quantity | 4096 | 8192 | 16384 |
|---|---|---|---|
| Seed 1: tail-linear ramp PPL | 29.501 | 28.414 | 28.110 |
| Seed 1: DEPA $0.9T$ PPL | 29.385 | 28.343 | 27.568 |
| Paired PPL gain | 0.116 | 0.071 | 0.542 |
| Paired NLL gain | 0.003940 | 0.002495 | 0.019470 |
| Seed 2: tail-linear ramp PPL | 29.653 | 29.826 | 28.244 |
| Seed 2: DEPA $0.9T$ PPL | 29.522 | 29.725 | 27.575 |
| Paired PPL gain | 0.131 | 0.101 | 0.669 |
| Paired NLL gain | 0.004432 | 0.003394 | 0.023959 |
| Seed 3: tail-linear ramp PPL | 30.365 | 29.034 | 29.058 |
| Seed 3: DEPA $0.9T$ PPL | 30.291 | 29.009 | 28.640 |
| Paired PPL gain | 0.074 | 0.025 | 0.418 |
| Paired NLL gain | 0.002424 | 0.000864 | 0.014486 |

DEPA improves all three measured lengths in this pair (seed 1 in Table 51); its 16384 PPL is 1.93% lower. Moving most positional adjustment late is already effective: the tail-linear ramp reaches 28.110 at 16384 versus 34.756 for the full-budget ramp. The remaining advantage of the single release therefore cannot be explained solely by a difference in the scalar multiplier sum. A second continuation seed reproduces positive paired gains at every length: 0.131, 0.101, and 0.669 PPL, respectively (2.37% at 16384). All three seeds therefore support the single-switch implementation in this setting, without establishing uniform dominance over other shapes or training seeds. Scalar exposure matching does not match nonlinear rotary attention outputs, learning-rate-weighted exposure, or the time-ordered weight states; the comparison does not isolate a unique mechanism. The tail-linear content probe has accuracy 0.5 at all distances and zero correct two-context reversals in both seeds. The third paired seed also favors DEPA at every length, with gains 0.074/0.025/0.418 PPL. These three paired outcomes are descriptive and do not establish population-level dominance.

**What exposure matching can isolate under a decaying learning rate.** The three observed paired gains at 16384 have mean 0.543 PPL and sample standard deviation 0.125 (NLL: $0.019305 \pm 0.004738$); these summarize three continuation seeds, not uncertainty across pretrained bases. There

is also a structural limit to this control. Let $w_t \in [0,1]$, $t = 0, \ldots, T-1$, with $\sum_t w_t = B$ for integer $B$ (a bounded ramp with the same scalar exposure as DEPA), and define the latest binary release $d_t = \mathbf{1}\{t \geq \tau\}$, $\tau = T - B$. For a strictly decreasing affine learning rate $\eta_t = a - bt$, $b > 0$, equal weight sums give the centroid identity

$$\sum_t \eta_t (w_t - d_t) = b \sum_t t(d_t - w_t) = b\, B \Big[\tau + \tfrac{B-1}{2} - \tfrac{1}{B} \sum_t t\, w_t\Big], \tag{18}$$

so by Eq. 18 the ramp's LR-weighted-exposure excess or deficit over DEPA is proportional to the distance of its exposure centroid $\frac{1}{B}\sum_t t w_t$ above or below DEPA's centroid $\tau + \frac{B-1}{2}$. Because $\eta_t$ is strictly decreasing, DEPA—the schedule that puts all $B$ exposed units as late as possible—is the unique minimizer of LR-weighted exposure over the whole exposure polytope $\{w \in [0,1]^T, \sum_t w_t = B\}$: the map $w \mapsto \sum_t \eta_t w_t$ is linear, so its minimum is attained at an extreme point, and the strictly decreasing coefficients select the last-possible vertex. This holds for every bounded ramp with equal scalar exposure, including a full-budget ramp whose centroid lies below $\tau$; no termwise sign condition on the ramp is involved. This is a schedule identity, not a loss ordering or an AdamW update bound.

For the actual $T = 4000$, $B = 400$ pair, the exposure centroid $\sum_t t w_t / B$ is 3733.0 for the tail ramp and 3799.5 for DEPA (zero-based continuation updates). The continuation LR is $\eta_t = 3\times10^{-5} + 2.7\times10^{-4}[1 - (23500 + t)/27500]$. Consequently $\sum_t \eta_t w_t$ is 0.01304858 versus 0.01278742, a 2.04% excess for the ramp despite both weight sums being 400. The observed endpoint advantage therefore excludes scalar exposure alone as an explanation, but cannot separate shape from LR-weighted exposure. The constant-LR matched-shape comparison in Appendix M.3 removes this scalar difference and retains the DEPA advantage on both additional PI seeds. The separate 19M four-arm study provides additional timing evidence in its own setting.

**Nonlinear transitions at matched exposure.** We compare linear and cosine tails with the fixed-$0.9T$ reference, using custom YaRN-family at constant LR, continuation seed 4. The same profiles were fixed for bases 0 and 1 before either pair's endpoints were available. Both hold source multipliers until step 3200 and use the last 800 updates for a fixed transition. For $j = t - 3200 \in \{0, \ldots, 799\}$, define

$$a_j^{\text{lin}} = j/799, \qquad a_j^{\cos} = \tfrac{1}{2}[1 - \cos(\pi j/799)], \qquad \theta_j = (1 - a_j)\theta_S + a_j \theta_T. \tag{19}$$

Both satisfy $a_j + a_{799-j} = 1$, so each has cumulative weight 400, as does release at step 3600. Their LR-weighted sums also match at constant LR. All arms use the same 131.072M optimizer training tokens, final target, temperature correction, ten-forward 16K training-probe grid, and endpoint evaluators. This does not equate intermediate attention outputs or optimization trajectories. Both base-specific pairs and their same-probe fixed-release references are reported in Table 52. On base 0,

Table 52: Constant-LR custom-YaRN-family shape controls, continuation seed 4. All listed arms have cumulative multiplier weight 400 and matched probe protocols. All six endpoints are shown. These rows are separate from the uninstrumented three-seed family of Table 28.

| Base | Schedule | 4K PPL | 8K PPL | 16K PPL |
|---|---|---|---|---|
| 0 | Linear, 800 updates | 28.48964 | 29.51793 | 29.65257 |
| 0 | Cosine, 800 updates | 28.49882 | 29.50082 | 29.61967 |
| 0 | DEPA $0.9T$ | 28.50710 | 29.45663 | 29.55022 |
| 1 | Linear, 800 updates | 28.69017 | 29.51755 | 29.51605 |
| 1 | Cosine, 800 updates | 28.68455 | 29.50210 | 29.49500 |
| 1 | DEPA $0.9T$ | 28.67957 | 29.49646 | 29.48661 |

cosine improves on linear by 0.01711/0.03290 PPL at 8K/16K, with a 0.00918-PPL cost at 4K. DEPA improves further on cosine by 0.04419/0.06945 PPL at 8K/16K (NLL gains 0.001499/0.002347), with a 0.00828-PPL 4K cost. Thus the longest-length advantage persists against this nonlinear control, whereas the all-length ordering of the three seeds of Table 28 does not extend to this base-0 seed-4 pair. On base 1, DEPA beats the linear control by 0.01060/0.02109/0.02944 PPL. Against cosine on base 1, DEPA improves all three main lengths by 0.00498/0.00565/0.00839 PPL (16K NLL gain 0.000285). Cosine itself beats linear by 0.00563/0.01544/0.02105 PPL on this base. The 8K/16K ordering DEPA, cosine, linear therefore recurs on both checkpoints, but DEPA's residual

16K advantage over cosine is much smaller on base 1. Cosine reduces the linear-to-DEPA 16K gap by 32.1% on base 0 and 71.5% on base 1. These are descriptions of two same-seed checkpoint comparisons, not population reliability or evidence of an optimal ramp.

**Same-target prefix effects.** The nested evaluation fixes the final 128 tokens of each of 32 windows across 1K/4K/8K/16K input suffixes. Define $G(L) = \ell_{\text{reference}}(L) - \ell_{\text{numerator}}(L)$; then $G(16K) - G(4K)$ is the improvement in response to the added prefix; Table 53 lists the contrasts. On base 0,

Table 53: Shape contrasts on identical nested targets on each base. Positive values favor the first schedule named. Counts are positive paired windows out of 32; they do not assume independent windows.

| Base | Contrast | $G(4K)$ | $G(16K)$ | Prefix effect | $n_+(16K)$ |
|---|---|---|---|---|---|
| 0 | Cosine over linear | +0.000199 | +0.002017 | +0.001819 | 25 |
| 0 | DEPA over linear | +0.000033 | +0.004549 | +0.004516 | 27 |
| 0 | DEPA over cosine | −0.000166 | +0.002532 | +0.002697 | 26 |
| 1 | Cosine over linear | −0.000344 | +0.000253 | +0.000596 | 18 |
| 1 | DEPA over linear | −0.000268 | +0.000880 | +0.001148 | 19 |
| 1 | DEPA over cosine | +0.000076 | +0.000628 | +0.000552 | 20 |

relative to cosine, DEPA's small nested 4K cost is offset by a larger prefix-effect improvement; the latter is positive in 29/32 windows. The separate natural-text full/local protocol also favors DEPA over cosine at 16K: full-input gain 0.004893 equals local gain 0.002160 plus utility gain 0.002733, with full-input improvement in 25/32 windows. Absolute 16K utility remains negative ($-0.083695$ for DEPA versus $-0.086428$ for cosine), so this is reduced long-input harm, not demonstrated retrieval. For base 1, DEPA over linear has natural 16K full gain 0.003003 and nested gain 0.000880; its nested prefix effect is 0.001148. Against cosine on base 1, the nested 16K gain is 0.000628 and the prefix effect 0.000552, both positive in 20/32 windows. Its natural 16K full-input gain is only 0.000044: local gain 0.000358 offsets a utility change of $-0.000314$. At 8K, natural full-input NLL slightly favors cosine (by 0.000063), as does nested NLL (by 0.000080), despite DEPA's positive main-sweep PPL gain. Hence the replicated main 8K/16K ordering is not an all-protocol dominance claim. These tail targets differ from the main language-model sweep, and the natural protocol uses different targets across lengths. They are complementary paired measurements, not interchangeable estimates or population intervals.

**Target-state trajectory and a finite adaptation phase.** The base-0 probes (Figure 13) evaluate all three models at the same target RoPE on the same training windows, irrespective of their current training profile. At completed update 3600, target NLL is lower for linear/cosine (3.421342/3.421571) than for the still-held DEPA model (3.439821). After 100 target updates, DEPA reaches 3.387663, below linear/cosine 3.407323/3.404141; at the endpoint it remains lower (3.390018 versus 3.392436/3.391326). On base 1, DEPA also crosses below both ramps after its first 100 target updates. At the final probe, however, cosine and DEPA nearly coincide (3.386119 versus 3.386148), while the held-out main 16K score favors DEPA. These measurements describe adaptation under the different schedules; a smaller pre-release target loss alone does not rank their endpoints. Each arm uses ten additional forward calls, taking 4.397–4.418 recorded seconds excluding JSON writes, with no gradient updates on the probe.

The 800-update ramps first reach the complete target at update 3999, whereas DEPA trains at that target for 400 updates. We therefore define a follow-up with a 400-update cosine transition at steps 3400–3799, followed by a 200-update plateau at steps 3800–3999:

$$a_t = \begin{cases} 0, & t < 3400, \\ \frac{1}{2}[1 - \cos(\pi(t - 3400)/399)], & 3400 \le t < 3800, \\ 1, & 3800 \le t < 4000. \end{cases}$$

The transition contributes 200 cumulative weight and the plateau another 200; the final transition update also has weight one. Total weight and constant-LR-weighted exposure still match the fixed-release controls. This profile preserves a smooth change and an explicit final target phase. It also changes the duration of source holding and the intermediate trajectory, so it is not an isolated causal

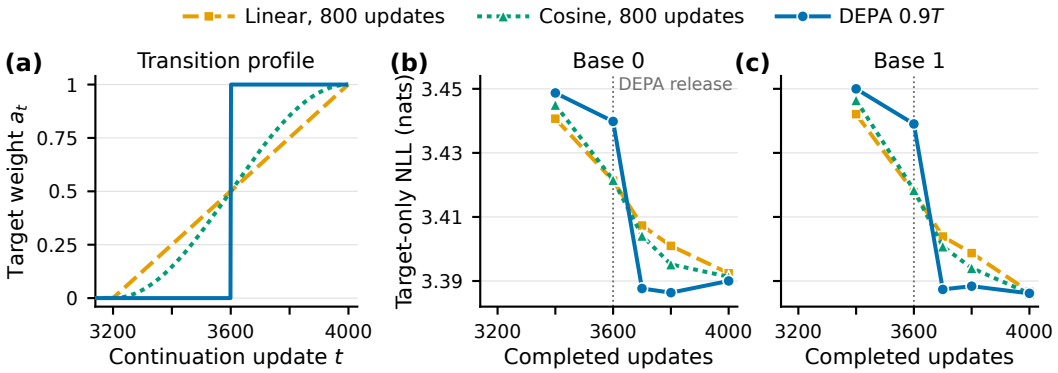



Figure 13: Seed-4 profiles and target-only training-window losses on both pretrained bases. (a) Target multiplier weight $a_t$ per zero-based continuation update $t$; (b, c) target-only training-window NLL on bases 0 and 1; the dotted line marks DEPA's release at update 3600. Symbols are the five measured states (each the mean over two 16K-token windows; no error bars); connecting segments do not resolve the trajectory between them. DEPA's target probe at update 3600 is measured before its first target-training update.

intervention on plateau length. It was chosen after the base-0 seed-4 results above and run unchanged on both bases with the same probe/evaluation protocol. Both seed-4 endpoints are reported below. This is a development follow-up, not an independently validated optimum or a state-adaptive policy. A further unchanged three-arm comparison on base 1, fresh continuation seed 5, pairs fixed release, the 800-update cosine, and the 400-update cosine plus target plateau. This comparison was specified before either compressed seed-4 endpoint was available, to distinguish a useful repeatable refinement from the small seed-4 shape differences. Its three endpoints and tail evaluations are compared below. The same unchanged three-arm protocol was also run on base 0, continuation seed 5, after the seed-4 two-base comparison; its fixed-release reference reaches PPL 28.86157/27.80934/28.29948 at 4K/8K/16K, and all three endpoints and their tail evaluations are reported.

**Long-cosine replication on two bases and two seeds.** The unchanged fixed-release versus 800-update cosine comparison covers continuation seeds 4 and 5 on both pretrained bases. The base-0 seed-5 compressed-cosine endpoint is also complete and included below. All four pairs favor fixed

Table 54: Cosine minus fixed-release PPL: positive favors DEPA $0.9T$. All four comparisons share the constant LR, cumulative weight 400, final target and temperature, token/update budget, and ten-forward probe recipe.

| Base | Seed | 4K gain | 8K gain | 16K gain |
|---|---|---|---|---|
| 0 | 4 | −0.008282 | +0.044190 | +0.069448 |
| 0 | 5 | +0.000961 | +0.016995 | +0.027653 |
| 1 | 4 | +0.004976 | +0.005648 | +0.008394 |
| 1 | 5 | +0.003788 | +0.004325 | +0.003669 |

release at 8K and 16K (Table 54). At 4K, base 0 changes from a 0.00828-PPL cost on seed 4 to a 0.00096 gain on seed 5; both base-1 gains are positive but small. The base-0 seed-5 main NLL gains are 0.000033/0.000611/0.000977 at 4K/8K/16K. The two-seed mean 16K PPL gain is 0.04855 on base 0 and 0.00603 on base 1. These descriptive paired outcomes support a repeated long-length advantage over the long cosine, with checkpoint-dependent magnitude; they do not establish a population guarantee or dominance over every smooth profile. Main validation windows vary with continuation seed.

On base 0, seed 5 also retains positive nested 8K/16K NLL gains (0.000819/0.001576), while its nested 4K gain is negative ($-0.000214$). The resulting added-context effect is 0.001789; 21/32 windows have positive 16K gain and 25/32 have positive prefix effect. This repeats the positive base-0 seed-4 nested 16K and prefix signs. In contrast, base 1 has mixed prefix-effect signs across seeds,

and its nested 8K evaluation favors cosine on both. The fixed nested target sets and seed-dependent main sweep therefore support distinct statements about endpoint PPL and added-context effects.

The base-0 seed-5 natural-text evaluation likewise favors fixed release at 4K/8K/16K by 0.000230/0.001178/0.001910 NLL. At 16K, a utility gain of 0.002042 offsets a local-prediction cost of 0.000132, with positive full-input gains in 25/32 windows. Absolute utility is still negative: $-0.073861$ for fixed release versus $-0.075903$ for cosine. This is reduced harm from the added prefix, not evidence that full input outperforms truncation or establishes retrieval capability. These natural/nested window sets are held fixed across the four base/seed comparisons; their repeated use does not create independent context samples.

**Complete fresh-seed shape comparison on base 1.** All three seed-5 endpoints and their natural/nested evaluations are reported. The profiles were fixed before either compressed seed-4 endpoint was available. The base-0 seed-5 comparison is also complete. Table 55 reports all four complete three-arm families; the following paragraph describes base 1.

Table 55: Complete three-profile comparisons on both bases and continuation seeds 4 and 5, plus the base-1 fresh-seed replication (seeds 6/7). Every profile has cumulative target weight 400. The 800-step cosine reaches the target on its last update; the 400-step cosine is followed by a 200-update plateau. Fixed release reserves 400 target updates. Target, constant LR, token budget and probe grid match within each comparison.

| Base | Seed | Schedule | 4K PPL | 8K PPL | 16K PPL |
|---|---|---|---|---|---|
| 0 | 4 | DEPA $0.9T$ | 28.507101 | 29.456634 | 29.550219 |
| 0 | 4 | Cosine 800 | 28.498819 | 29.500824 | 29.619668 |
| 0 | 4 | Cosine 400 + plateau | 28.501024 | 29.481673 | 29.589608 |
| 0 | 5 | DEPA $0.9T$ | 28.861574 | 27.809344 | 28.299475 |
| 0 | 5 | Cosine 800 | 28.862535 | 27.826339 | 28.327128 |
| 0 | 5 | Cosine 400 + plateau | 28.860834 | 27.816991 | 28.313212 |
| 1 | 4 | DEPA $0.9T$ | 28.679570 | 29.496457 | 29.486611 |
| 1 | 4 | Cosine 800 | 28.684546 | 29.502104 | 29.495005 |
| 1 | 4 | Cosine 400 + plateau | 28.688600 | 29.507324 | 29.495242 |
| 1 | 5 | DEPA $0.9T$ | 28.992129 | 27.928022 | 28.394614 |
| 1 | 5 | Cosine 800 | 28.995916 | 27.932347 | 28.398284 |
| 1 | 5 | Cosine 400 + plateau | 28.994898 | 27.929532 | 28.393375 |
| 1 | 6 | DEPA $0.9T$ | 28.829929 | 29.141639 | 28.770972 |
| 1 | 6 | Cosine 800 | 28.831883 | 29.143842 | 28.768435 |
| 1 | 6 | Cosine 400 + plateau | 28.830878 | 29.140960 | 28.763329 |
| 1 | 7 | DEPA $0.9T$ | 28.213810 | 28.412009 | 29.167640 |
| 1 | 7 | Cosine 800 | 28.213003 | 28.411912 | 29.165473 |
| 1 | 7 | Cosine 400 + plateau | 28.212113 | 28.409061 | 29.161240 |

Against long cosine, fixed release retains positive seed-5 main gains of 0.00379/0.00433/0.00367 PPL, or 0.000131/0.000155/0.000129 NLL. This repeats the seed-4 signs on this checkpoint with smaller margins. Compression itself changes ordering across seeds: it was worse than the long cosine at all three lengths on seed 4, but improves all three on seed 5 by 0.00102/0.00282/0.00491 PPL. Against fixed release, compressed cosine on seed 5 gains only 0.00124 PPL at 16K while costing 0.00277/0.00151 at 4K/8K. Fixed release won all three lengths on seed 4. Thus this refinement has not uniformly improved the fixed baseline; the very small 16K reversal also limits a claim that release dominates every tested smooth profile. Two continuation seeds neither establish population reliability nor estimate numerical-execution uncertainty.

**Base-1 fresh-seed replication (seeds 6/7).** Two further fresh continuation seeds on base 1 were specified before the seed-4 endpoints were available and reproduce the three-arm comparison (Table 55, lower block). On both seeds the plateau profile's 16K PPL is below fixed release ($-0.0076$/$-0.0064$ PPL, plateau minus DEPA; plateau lower), matching the seed-5 sign; the seed-4 endpoint had the opposite sign ($+0.0086$). The four-seed panel ($+0.0086$, $-0.00124$, $-0.0076$, $-0.0064$ at 16K, mean $-0.0017$ PPL, population sd $0.0064$) is therefore mixed with all magnitudes at or below the execution-level variation between repeated runs (Appendix O.1); this reads as a measured tradeoff

rather than a repeatable DEPA 16K advantage over the plateau profile on base 1. The plateau profile also sits below fixed release at 8K on seeds 6/7 ($-0.00068$/$-0.00295$ PPL) and below the long cosine at 16K on every fresh-seed arm ($-0.0051$/$-0.0042$ PPL), so the fresh seeds are not a single-arm spurious shift. These two observations do not establish a plateau advantage, a lower bound, or non-inferiority; they limit the opposite claim (that DEPA dominates the plateau profile at 16K on this base) to a point-difference reading.

Table 56: Compressed cosine over fixed release at 16K. Positive NLL gains favor compression. Natural full gain equals local gain plus utility gain; nested prefix gain subtracts the corresponding 4K gain.

| Seed | Natural full | Local | Utility | Nested | Prefix gain |
|---|---|---|---|---|---|
| 4 | −0.001496 | −0.002439 | +0.000943 | +0.000465 | +0.000029 |
| 5 | +0.000074 | −0.000117 | +0.000191 | +0.000367 | +0.000518 |

On seed 5 (Table 56), the compressed profile also has a small natural 16K full-input gain, with positive difference in 18/32 windows. Its local loss cost is offset by improved prefix utility. The nested 16K and prefix effects are positive in 19/32 and 18/32 windows. Absolute natural 16K utility remains negative ($-0.072026$ compressed versus $-0.072217$ fixed release), so these gains do not establish retrieval ability. For fixed release versus long cosine, seed-5 natural/nested 16K gains are 0.000333/0.000070. However, its nested 4K gain is larger (0.000293), leaving negative added-context effect $-0.000223$; seed 4 instead had positive prefix effect 0.000552. Natural and nested 8K NLL favor long cosine on both seeds, with seed-5 fixed-release costs 0.000213/0.000211. Main windows depend on the continuation seed, whereas the natural/nested window sets are reused across seeds. These small paired effects must therefore be distinguished by target set and length, rather than pooled into an all-protocol shape ordering.

**Fresh-seed compressed-cosine outcomes on base 0.** The remaining base-0 seed-5 endpoint completes all four three-profile comparisons in Table 55. Compression improves the long cosine at 4K/8K/16K by 0.00170/0.00935/0.01392 PPL. Relative to fixed release, however, it gains only 0.00074 PPL at 4K and costs 0.00765/0.01374 at 8K/16K. This repeats the base-0 seed-4 short-versus-long-length tradeoff with smaller margins. Across all four base/seed pairs, fixed release wins 8K over compression in every pair and 16K in three; the exception is the small base-1 seed-5 gain described above. Neither compressed cosine nor the state rule has supplied a uniform improvement over the fixed baseline at the intended main lengths.

On base 0, seed 5, compression improves long-cosine natural 16K NLL by 0.000930 and nested 16K NLL by 0.000783; the nested added-context gain is 0.000819. It still trails fixed release by 0.000980 natural and 0.000793 nested NLL, with nested prefix effect $-0.000970$. The natural comparison comprises local gain 0.000178 and utility cost 0.001158; absolute utility remains negative ($-0.075020$ compressed, $-0.073861$ fixed). The 4K main advantage also does not extend to the natural 4K tail, where compression costs 0.000274 NLL. This preserves the distinction between the main sweep and target-matched prefix effects.

In the common final 200-update target phase, base-0 seed-5 mean probe loss rises by 0.001862 for fixed release and 0.001618 for compression, while their compressed-minus-fixed gap shrinks from 0.000474 to 0.000231. One fixed probe window improves and the other worsens in both arms. Thus the mean increase does not imply that every training window worsens; this finite-interval decomposition does not identify a transferable recovery rate or a unique effect of plateau length.

**Finite-target-phase outcomes on both bases.** The seed-4 comparisons in Table 55 exhibit the following width responses. All profiles have cumulative weight 400.

On base 0, compressing the cosine improves 8K/16K over the long cosine by 0.01915/0.03006 PPL, with a 0.00220-PPL 4K cost. It retains a 0.00608-PPL 4K advantage over DEPA but trails by 0.02504/0.03939 at 8K/16K. Across these three measured profiles, a shorter transition improves the long-length score and slightly raises the trained-length score. This development-seed ordering does not reproduce on base 1: the compressed cosine is worse than the long cosine by 0.00405/0.00522/0.00024 PPL, and worse than DEPA by 0.00903/0.01087/0.00863 at 4K/8K/16K.

Thus DEPA retains the best main 8K/16K scores among both tested cosine profiles on both checkpoints. An explicit target plateau alone does not guarantee that compressing a smooth transition improves its endpoint. The comparison jointly changes source holding, intermediate multipliers, and complete-target adaptation; it does not isolate a unique mechanism of plateau length or establish an optimal continuous profile.

Table 57: Compressed cosine relative to each reference at 16K. Positive NLL gains favor the compressed profile. Natural full and nested targets are different; nested prefix gain is $G(16K) - G(4K)$.

| Base | Reference | Main | Natural full | Nested | Prefix gain |
|---|---|---|---|---|---|
| 0 | Long cosine | +0.001015 | +0.001684 | +0.001065 | +0.001143 |
| 0 | DEPA | −0.001332 | −0.003209 | −0.001466 | −0.001554 |
| 1 | Long cosine | −0.000008 | −0.001451 | +0.001093 | +0.000581 |
| 1 | DEPA | −0.000293 | −0.001496 | +0.000465 | +0.000029 |

**Prediction targets affect the small refinement.** On base 0 (Table 57), compressed versus long cosine has nested 16K and prefix-effect gains in 26/32 and 28/32 windows, respectively; both mean gains remain smaller than DEPA's. On base 1, compression also improves nested 16K loss over long cosine (26/32 windows), despite slightly worse main 16K PPL and worse natural full-input loss. Relative to DEPA on this base, its small nested 16K gain (18/32 windows) accompanies an almost equal 4K gain, leaving only 0.000029 additional-prefix improvement. Natural 16K loss is higher by 0.001496: local loss is higher by 0.002439, which outweighs a utility gain of 0.000943. Absolute natural 16K utility remains negative for both profiles. These results support separating absolute prediction loss, target composition, and prefix response when judging refinements; they do not establish retrieval improvement or independent-window confidence intervals.

**Equal-exposure states before a common target phase.** At completed update 3800, compressed cosine and fixed release have each accumulated weight 200; both then spend the final 200 updates at the complete target with the same constant LR. Their inherited model and optimizer states differ. Let $D_C, D_F$ denote mean target-only NLL on the same two fixed 16K training windows, and $\Delta D_a = D_a(4000) - D_a(3800)$. The exact finite-interval identity is

$$D_C(4000) - D_F(4000) = [D_C(3800) - D_F(3800)] + [\Delta D_C - \Delta D_F].$$

On base 0 (Table 58), the paired gap shrinks while absolute probe loss rises on both individual

Table 58: Measured target-state decomposition. Differences are compressed minus fixed release; the two absolute changes need not be negative.

| Base | Gap at 3800 | $\Delta D_F$ | $\Delta D_C$ | Change difference | Gap at 4000 |
|---|---|---|---|---|---|
| 0 | +0.000691 | +0.003654 | +0.003519 | −0.000135 | +0.000557 |
| 1 | +0.000069 | −0.002240 | −0.002538 | −0.000299 | −0.000229 |

windows in both arms. On base 1, both windows improve in both arms, and compressed cosine ends with lower probe loss, yet has worse main PPL at all three lengths. The common target phase therefore permits different absolute changes and endpoint rankings; entering that phase or minimizing these two training-window losses is not a validated release condition. This is a training-state decomposition, not a held-out causal mediation estimate, a contraction-rate fit, or evidence of bitwise identical execution. The fresh-seed comparison above retains the same profiles and shows which of these small differences repeat across both bases; all fresh-seed endpoints and tail evaluations are complete.

**Fresh-seed common-phase comparison.** On base 1, the seed-5 compressed-minus-fixed target-probe gap changes from $+0.000083$ at update 3800 to $-0.000435$ at update 4000. The exact difference in changes is $-0.000518$ NLL. Both arms' absolute probe NLLs increase over these same 200 full-target updates: by 0.002832 for fixed release and 0.002314 for compressed cosine; both individual probe windows increase in each arm. In contrast, both probe windows improved

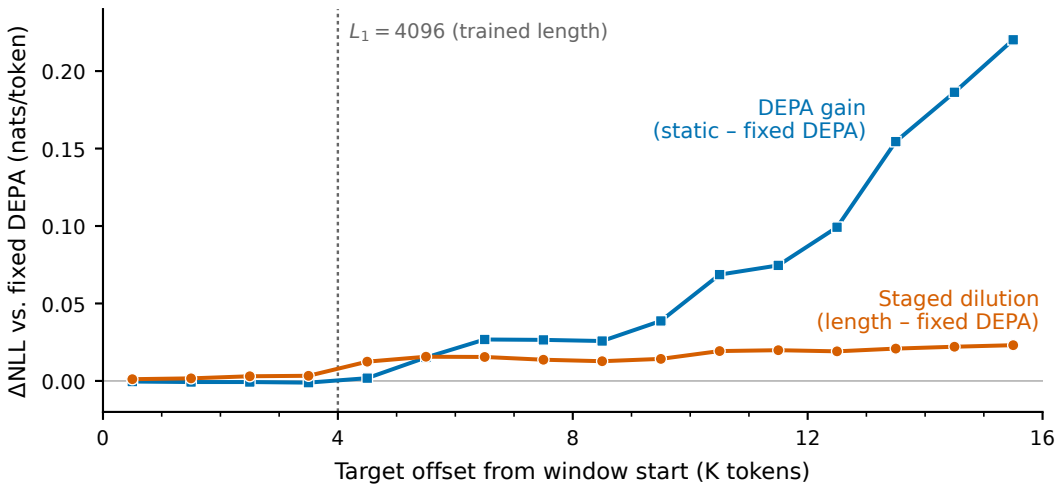



Figure 14: Target-offset structure of the staged dilution and the DEPA gain at 16K (base 1, seed 11, T= 4000, default staged; per-offset-bucket mean NLL, fixed windows, width $L_0$). The staged-hold dilution (length–fixed DEPA) is a step at the trained length $L_1 = 4096$ (dotted): offsets within $L_1$ are unaffected, the extrapolation band is diluted roughly uniformly with a mild deep-tail enhancement. The DEPA-over-static gain is tail-monotone over the same extrapolation buckets, i.e. the late release buys readiness exactly where the added-context harm is largest. Markers sit at bucket midpoints; the y axis is the difference in per-bucket mean NLL (nats/token) relative to fixed-length DEPA (positive = worse than fixed DEPA); one run per curve, so no error bars.

during this phase on seed 4. The final compressed profile has lower probe NLL on both seeds, while its main 16K advantage is positive only on seed 5 and its 4K/8K costs remain on both. Thus a smaller between-profile gap can coexist with increasing absolute probe loss, and the sign of the final training-probe gap alone does not select a uniformly better endpoint recipe. The same fixed training windows are used on both seeds; the main validation sample changes with seed. These observations motivate keeping model state, the measurement target, and the remaining budget explicit in a release rule rather than transferring a single measured recovery rate between runs. No contraction rate is fitted to this sparse grid.

**Token-aligned length and release schedules.** A position-binned evaluation of the same untrained windows (target-offset buckets of width $L_0$, fixed draws, one run per cell) places the hold-state mechanism on the offset axis: the staged-hold dilution is a *step in target offset*—near zero for offsets within the trained length $L_1$, roughly uniform across the extrapolation band $L_1..4L_1$, with only a mild deepest-bucket enhancement—while the DEPA-over-static gain is tail-monotone across those same extrapolation buckets (Fig. 14). The dilution thus shows up exactly where the added prefix spans out-of-training-range offsets, and the late release buys readiness at the same offsets; at 8K the same step structure appears at about nine-tenths of its per-bucket magnitude, so the documented 8K margin survives by amount rather than by a different mechanism. We read this as extrapolation-offset readiness: the hold's long-context share prices the reduction of added-context harm at long offsets, connecting the measured share response to the same-target decomposition of Appendix N.2.

**Hold-state share and composition detail.** The two-base length-by-timing factorial shows the following. Staging alone changes main 16K PPL by only $-0.013$ (base 0) and $+0.077$ to $+0.140$ (base 1), small but consistently non-negative on the independent base, while staging reduces the delayed-release margin against the fixed-length DEPA reference by $-0.20/-0.40$ PPL at 16K, and the interaction (timing gain under staged minus fixed length) is $-0.0074/-0.0115$ NLL, negative on both bases. The dilution is far smaller than the $4/9$ long-context share of the hold would bound linearly, indicating strong transfer of long-context readiness across the $1K/2K$ stages; the theoretical statement is therefore structural (the release exploits long-context readiness accrued during the hold; shorter hold stages dilute it, but not proportionally) rather than an identified rate law. The dilution is tail-concentrated: on both bases the interaction (timing NLL gain under staged minus fixed

length) is positive at 8K ($+0.0017$/$+0.0067$, and $+0.0052$/$+0.0105$ on the natural decomposed gain) and negative only at 16K ($-0.0074$/$-0.0115$), while the staged-versus-fixed DEPA margin shrinks monotonically in target length ($-0.07$/$-0.21$/$-0.40$ PPL at 4K/8K/16K on base 1). Re-sequencing the hold updates therefore leaves the 8K-ready portion of the readiness intact (or gently favors it) and erodes mainly the far-tail readiness that the release would otherwise buy at the longest scored context. A hold-share intervention measures the hold-share channel in isolation on the independent base (without establishing it as the sole channel): raising the staged hold's 4K share from 0.444 (default schedule) to 0.722 ($1K/2K$ phases compressed to updates 0–399/400–999, release still at 3600) cuts the 16K dilution from $-0.40$ to $-0.11$ PPL and brings the length-by-timing interaction at 16K back to $+0.0009$ NLL, near zero, with the same $+1.80$/$+1.83$ PPL fixed-vs-staged DEPA margins and all four cells digest-matched. The measured schedules give a share-response pattern (descriptive interpolation; the staged rows are two fresh seeds each, per-observation log-log fit on the base-1 seed rows): the 16K dilution scales as $D \approx -0.98\,(1-S)^{1.66}$ in the missing hold-share $1-S$, i.e. readiness transfer across stages makes the dilution superlinearly sensitive to the share deficit– restoring hold exposure buys back the tail margin faster than linearly. The exponent's value is unstable across fit perturbations: the leave-one-seed-out range is 1.60–1.99, but removing the $S = 0.444$ schedule point entirely (fit on hold-70 and hold-85 only) flattens it to 1.14 and substituting base 0's weaker same-share point gives 1.05; the value is anchored by the default-staged rows and is base-sensitive, so these few shared-seed observations support a descriptive convex-in-$1-S$ reading, not a statistically identified exponent above 1. A fifth schedule point at $S = 0.600$ (two fresh seeds, never used in the fit) reproduces the curve: 16K dilution $-0.19 \pm 0.02$ (seeds 16/17, $-0.182$/$-0.207$) versus the fitted $-0.21$, and including it raises the anchor-drop refit from 1.14 to 1.40; consistent with a smooth superlinear curve, but not by itself decisive curvature evidence. A same-share composition control (fixed $S = 0.722$; 1K-fraction 0.0/0.2/0.6 variants against a same-seed $e_1 = 400$ reference, two fresh seeds 20/21, token-matched within each seed) leaves the 16K dilution at $-0.110$/$-0.129$/$-0.111$ on seed 20 and $-0.142$/$-0.133$/$-0.126$ on seed 21, with the same-seed references at $-0.139$/$-0.148$: in-family spreads are 0.029/0.021 PPL and the reference is the most-negative cell on both seeds—the 1K/2K partition of the short phase shifts dilution by at most $\approx 0.03$ PPL per seed, all eight cells lie within 0.05 PPL of $-0.100$, a band fixed before these runs, and the reference cell is not a composition extreme relative to the share-response interpolation. These descriptive contrasts bound the composition effect but are not an equivalence or share-sufficiency claim. A matched-composition order replication (fixed $S = 0.722$; forward 1K*600 then 2K*400 versus reversed 2K*400 then 1K*600, the same onset 1000, token- and stream-matched within each of three fresh seeds 22/23/24) gives an order-by-release interaction at 16K of $+0.125/+0.126/+0.130$ PPL (mean +0.127, sample s.d. 0.003) and at 8K of $-0.200/-0.164/-0.204$ (mean $-0.19$, SD 0.02), with no 4K shift. At 8K the gradual (static) arm alone penalizes the reversed order by $+0.34/0.38$ PPL main (and $+0.013/0.017$ natural-NLL) while DEPA absorbs most of that penalty; at 16K static is order-neutral while DEPA retains a $+0.13$ PPL reverse penalty. In the natural decomposition this reverse penalty is an added-context utility effect at both 8K and 16K: the per-arm local (within-window) NLL spread is at most about a fifth of the full-input contrast, so short-phase order changes how the model reads added context rather than its in-window local modeling. On fixed-window evaluations the 16K interaction is $+0.0201/+0.0183/+0.0190$ nested-NLL and the natural full-NLL interaction $+0.023/+0.020/+0.020$, carried by the added-context utility channel (the within-window local interaction is near zero, $+0.0016/-0.0001/+0.0008$ NLL on the same seeds). These are descriptive three-seed paired contrasts; they show the short-phase order measurably shifts the delayed-release margin at fixed share on both length regimes, rather than establishing an order-sufficiency or order-dominance law. A cross-budget test at $T = 8000$ (double token budget, fresh seed, release at $0.9T$; Figure 15) is directionally consistent with the measured response at three shares: 16K dilution $-0.33$ at $S = 0.444$, $-0.06$ at $S = 0.722$, and $-0.18$ at the interior $S = 0.600$, versus $-0.40$, $-0.11$, and $-0.19$ at $T = 4000$, with the pooled fit over both budgets giving $D_0 = 0.92, \gamma = 1.70$ (ten observations)—each measured share falls within the 50% per-point tolerance fixed before these runs (a descriptive single-seed cross-budget comparison per share, not a statistical-equivalence or null confirmation), with a mild level softening from the larger adaptation reserve.

With a static target, staged-length training changes main 16K PPL by $-0.013$ (base 0) and $+0.077$/$+0.135$/$+0.140$ (base 1, seeds 5/6/7), i.e. small but consistently non-negative on the independent base; the stage-first reading of the measured share response already prices this as the hold's missing long-context share, and the attribute decomposition (Table 8) shows the timing gains

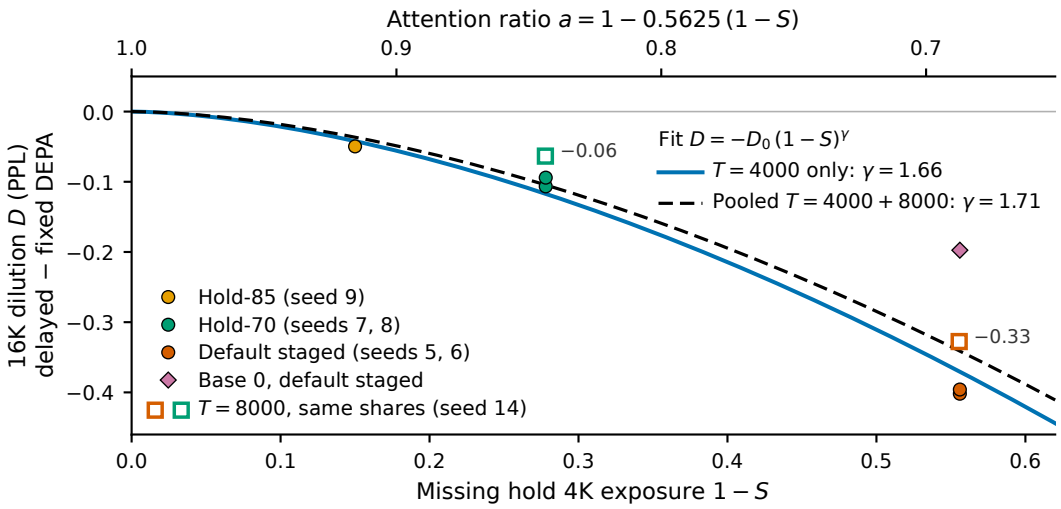



Figure 15: Cross-budget share response on the independent base: the $T = 8000$ measured points (open squares, one fresh seed per share; edge color matches the $T = 4000$ schedule family at the same share) beside the $T = 4000$ per-seed points (filled circles), with the $T = 4000$ descriptive interpolation ($\gamma$=1.66) and a pooled seven-observation fit on the base-1 points of both budgets ($\gamma$=1.71; the ten-observation pooled fit in the text gives $\gamma = 1.70$). No uncertainty bands are drawn (one seed per point). Used only as a descriptive check against the $50\%$ per-point tolerance, not a statistical-equivalence or null claim.

are long-context/utility effects with local components within $5 \times 10^{-3}$ (a mix of signs). Late release transfers into the length curriculum: staged-length DEPA improves on staged-length static by 1.80 (base 0) and 1.59 (base 1) PPL at 16K, with positive gains on all 32 natural 16K windows, and the gain is a prefix (long-context) effect with a small local cost. The fixed-length DEPA reference still wins on the full length grid: staging lengths erodes the delayed-release 16K margin by $-0.20$ (base 0) and $-0.40$ (base 1) PPL relative to fixed-4K DEPA, and the interaction (timing NLL gain under staged minus fixed length) is $-0.0074$ / $-0.0115$ NLL at 16K on the two bases. The release budget is therefore most effective when the pre-release hold itself acts on the deployment length. The staged and fixed protocols perform the same post-release schedule (updates 3600–4000 are all at 4K in both), so the dilution shown by the factorial is produced by the hold history itself: the staged arm's missing long-context share over updates 2000–3599, not by post-release sequence length. The fixed-versus-staged comparison is a recipe comparison (both training length and release timing differ), and the factorial above is the isolated interaction; none of these paired, possibly overlapping windows establish population reliability or retrieval ability.

The 16K timing gain is a long-context prefix effect with essentially zero local component on both bases and in both length settings; the staged-versus-fixed DEPA loss is likewise long-context, not a 4K cost. On base 1 the seed-6 replication reproduces the full contrast structure: fixed delayed over static $+1.895$ versus $+1.912$ (s5), staged delayed over staged static $+1.633$ versus $+1.587$, staged over fixed on DEPA $-0.396$ versus $-0.402$, and the interaction $+0.0071/-0.0093$ at 8K/16K versus $+0.0067/-0.0115$ – every sign and magnitude within 0.05 PPL across seeds. The context decomposition replicates too: the seed-6 timing gains are 16K 32/32-window prefix effects with practically zero local components ($+0.2735/+0.1851$ natural full at 16K, utility $+0.2734/+0.1860$, local $+0.0001/-0.0009$), and the staged-versus-fixed DEPA loss is a far-tail, not a local, cost with 0 of 32 windows positive at 16K in both seeds. The hold-share intervention on the independent base (seed 7, $1K/2K$ phases compressed to 0–399/400–999) reproduces the entire structure with the dilution largely removed: fixed DEPA over fixed static $+1.798$ (vs $+1.895/+1.912$), staged DEPA over staged static $+1.832$, staged over fixed on DEPA $-0.107$ (vs $-0.396/-0.402$), and the interaction $+0.0039/+0.0009$ at 8K/16K with the 16K term near zero. The replication at a second fresh seed gives staged-over-fixed on DEPA $-0.094$, interaction $+0.0034/-0.0010$ at 8K/16K, and wall factor $0.919$, so the dilution reduction ($-0.40$ default to $-0.11/-0.09$) and the vanishing interaction are reproducible across seeds 7/8. All cells (both seeds) record identical pre-segmentation stream digests;

the recorded wall-time factor is $0.921/0.919$ (attention load $0.85\times$ by protocol arithmetic). The rows are seed-wise descriptive; the share response they support is a measured pattern over the tested schedules, not a fitted population curve. Note on the seed-5 cell: its fixed-schedule reference is the fixed-$0.9T$ run of the shape comparisons (Table 55), whereas seeds 6–9 use the fixed arm of the token-matched length family. The seed-5 reference shares the release arm's base checkpoint, probe seed, context/tail length, and window starts (checked against the run records); the run does not record a training-token budget and has no recorded input/target-stream digest, so the seed-5 dilution and wall cells rest on recorded PPL and wall times rather than on the token and digest accounting available for seeds 6–9.

**Length-training versus target-adaptation decomposition.** The same fixed 16K training-window measurements separate the two phases that make up DEPA's endpoint advantage. In these runs the recorded quantity at every probe is the *target-configuration* natural NLL over the fixed 16 384-token training window (a contiguous short prefix of the training stream, scored under the target attention scale; the controller uses the target configuration throughout), so both the $t = 3400$ and $t = 4000$ readings in the paired contrast below are target-configuration losses, not source-configuration ones. Across the five base-1 seeds the static arm's own movement over updates 3400–4000 under this fixed target is $+0.001 \pm 0.008$ NLL – essentially zero on these windows. The DEPA arms move $+0.056 \pm 0.009$ over the same target readings, leaving a target-configuration adaptation component of $+0.055 \pm 0.004$ NLL and a final DEPA-over-static window gap of $+0.053 \pm 0.002$ NLL (5-seed means $\pm$s.d.). On this in-window (target-configuration) measure, essentially all of DEPA's advantage is carried by the release-plus-adaptation phase, not by the concurrent length-training trend. Because $t = 3400$ precedes the $0.9T = 3600$ release and the two arms have been training along different paths all along, subtracting the two trajectories does not by itself isolate the *causal* contribution of the release adaptation; it quantifies the combined phase contrast on a shared fixed target. This is complementary to, and measured on a different quantity than, the far-tail source-configuration PPL collapse reported in Appendix P.5 ($\sim$458$\rightarrow$ 53 at 16K under the source configuration): the two observations are not in conflict because one scores far-tail positions under the held source rotation while the other scores in-window target-configuration NLL under the released target – a length-training far-tail effect coexists with a switch-carried in-window gain.

**Cross-length release: the timing margin is not bound to the deployment length.** A further token-aligned $2 \times 2$ on the independent base (fresh seed 10) crosses the *released-update length* (4K versus 2K from update 3600) against the target schedule, all four cells digest-matched. The DEPA-over-static 16K margin is $+2.035$ PPL when the released updates act on 4K and $+2.222$ when they act on 2K: adapting the released target at a shorter length does not erode the timing benefit. The length-after intervention itself costs absolute 16K level on both arms roughly equally ($-1.65$ PPL static, $-1.47$ PPL DEPA), i.e. the deployment-length requirement is a property of the whole continuation, applied to both schedules alike. Together with the hold-share measurements this sharpens the mechanism: the released margin is gated by the *hold's* long-context share (pre-release readiness) and by the timing of the switch, not by the length of the adaptation phase. Single seed; descriptive.

**State-level witness of the hold-share mechanism.** The release controllers record target NLL on two fixed 16K training windows at a common probe grid, separating the hold-state account from endpoint PPL. Across the five base/seed cells (default staged: base 0 seed 5 and base 1 seeds 5 and 6; hold-70: base 1 seeds 7 and 8) the staged-minus-fixed gap at the release threshold (update 3600) does *not* separate the schedules (0.0023–0.0042 versus 0.0026–0.0037 nats, overlapping). The observed-endpoint probe gap at the final probe (update 4000) does split them (0.0072–0.0124 default versus 0.0032–0.0044 hold-70 nats) and calibrates the 16K margin loss across all five cells (slope $\approx 0.037$ PPL per $10^{-3}$ nats, $r^2 = 0.98$, matching the exponential derivative $\mathrm{d}\ln \mathrm{PPL} = \mathrm{dNLL}$, i.e. $\mathrm{dPPL} \approx \mathrm{PPL}_{16K} \cdot \mathrm{dNLL}$ with $\mathrm{PPL}_{16K} \approx 29$). Because the released updates are the same 4K steps in both schedules after 3600, the probe gap reflects the state geometry left by the 1K/2K hold history: the target adapts less efficiently from that state, so the staged schedule's hold-share deficit $1 - S$ is a worse observed-endpoint *adaptation gap*, not a worse release-time reading. It is the gap between two observed endpoints, not an identified asymptotic floor: high correlation across the five cells does not by itself bound the recovery limit. This is a five-cell, scheduled-comparison calibration consistent with the length-matched hold-excess account, not an identified recovery law. Window counts (32/32

positive at 16K for both timing contrasts on both bases; $\leq$3/32 for the staged-vs-fixed DEPA contrast at 16K) support the directional reading.

The recorded trainer intervals, including scheduled evaluations, are 2213.4/2216.6 seconds (fixed, seeds 5/6) and 1839.5/1842.2 seconds (staged, seeds 5/6) on base 1, a 16.9% wall reduction on each of the two seeds, with the same token budget. They are two sequential executions within a job, not a repeated throughput benchmark or complete job runtimes; equal token counts do not imply equal attention work. This observed time/quality tradeoff does not by itself establish a faster DEPA recipe, and the shorter staged schedule did not retain the full fixed-length DEPA margin.

### P.2 Observed training-loss signatures around release

The training logs record minibatch cross-entropy every 100 updates. The actual multiplier trace places the PI release at continuation update 3600 on both seeds. Table 59 compares the logged pre-update losses against static PI and the exposure-matched tail ramp. Within each seed these runs use the same batch-sampling streams; different rows use different minibatches and different schedules have different weights.

Table 59: Training-loss excess around the PI release (nats/token). Batch types are reconstructed by replaying the trainer's seeded task-mixture RNG; they were not logged during training. The copy batch at seed 1's release makes its raw spike unsuitable as evidence of a positional shock.

| Seed | Update | Batch | DEPA − static | DEPA − ramp |
|---|---|---|---|---|
| 1 | 3500 | PG19 | −0.0050 | −0.0078 |
| 1 | 3600 | Copy | −0.0605 | 0.0125 |
| 1 | 3700 | PG19 | 0.1283 | 0.1172 |
| 1 | 3800 | PG19 | 0.0907 | 0.0681 |
| 1 | 3900 | PG19 | 0.0582 | 0.0284 |
| 2 | 3500 | PG19 | 0.0003 | −0.0034 |
| 2 | 3600 | PG19 | 1.4253 | 1.4214 |
| 2 | 3700 | PG19 | 0.1309 | 0.1209 |
| 2 | 3800 | PG19 | 0.0843 | 0.0650 |
| 2 | 3900 | PG19 | 0.0568 | 0.0250 |

At seed 2's release, DEPA loss is 4.711 versus static 3.285 and ramp 3.289; 100 updates later the excess over static is 0.131. Seed 1 instead has losses 10.808, 10.868, and 10.795 at the release across DEPA, static, and ramp. The trainer draws the task mixture once per update from the seeded global NumPy RNG, with separate generators for data contents. Replaying these mixture draws identifies seed 1's release batch as a copy batch, whereas seed 2's is PG19. Thus a large raw loss at the switch is not by itself evidence of a release-specific shock.

At 100, 200, and 300 updates after release, both seeds show similarly shrinking positive excesses against either comparator. This is consistent with a transient adaptation cost followed by partial recovery, while DEPA ultimately has lower extrapolation PPL. These sparse, changing-minibatch observations do not measure a fixed-weight positional intervention, identify a contraction constant, or establish which weight changes cause the final gain. No gradient norms were retained in these artifacts.

### P.3 Continuation budget and static extrapolation

**Fraction versus absolute target-update count.** All three $T = 8000$ arms are reported. Each uses 262.144M continuation tokens and the same PI target, base 0, and continuation seed 1. The $T = 4000$ comparison uses 131.072M tokens per arm. Target-update counts $B = 400/800$ correspond to release fractions $0.9/0.8$ at $T = 4000$ and $0.95/0.9$ at $T = 8000$.

At $T = 8000$ (Table 60), fixed $f = 0.9$ reaches 25.821 at 16384 versus static 191.945, with a 0.740-PPL trained-length cost. Transferring the absolute count $B = 400$ instead reaches 25.643, a further 0.178-PPL improvement, but costs 0.586 and 0.563 PPL relative to $B = 800$ at 4096 and 8192. The ranking therefore transfers by target-update count across these two budgets: $B = 400$ is

Table 60: Complete PI budget-transfer experiment, one shared-base seed. All within-budget arms share the LR schedule; cross-budget recipes have different planned LR schedules.

| $T$ | Schedule | 4096 | 8192 | 16384 |
|---|---|---|---|---|
| 4000 | Static | 28.050 | 66.653 | 159.919 |
| 4000 | $B = 400$ | 29.385 | 28.343 | 27.568 |
| 4000 | $B = 800$ | 28.732 | 28.023 | 29.464 |
| 8000 | Static | 26.186 | 71.176 | 191.945 |
| 8000 | $B = 400$ | 27.512 | 26.448 | 25.643 |
| 8000 | $B = 800$ | 26.926 | 25.885 | 25.821 |

better at 16384, while $B = 800$ is better at both shorter lengths. The longest-length margin shrinks substantially, from 1.895 to 0.178 PPL; its NLL contrast shrinks from $-0.066494$ to $-0.006915$. This is evidence for a length-dependent allocation choice in this one-seed setting, not a universal absolute-count law or an optimal count outside these candidates. The fixed $0.9T$ default remains effective at the larger budget.

Static trained-length PPL improves from 28.050 to 26.186 as the budget doubles, but its 16384 PPL worsens from 159.919 to 191.945. More static training under this recipe does not repair extrapolation. The continuation starting LR rises from $6.93 \times 10^{-5}$ to $9.86 \times 10^{-5}$ with the larger planned budget. Thus the transfer comparison concerns the entire budget-dependent recipe, not a pure extra-token intervention at fixed LR. Independent seeds remain necessary to assess the small $T = 8000$ margin.

**Which allocation transfers for a mixed-length workload?** Preserving the longest-length ranking does not preserve the preferred allocation under mixed-length use. For a scored-token fraction $p$ at 16384 and $1-p$ at $L_s$, define the empirical risk $R_a(p) = (1-p)\log \mathrm{PPL}_a(L_s) + p \log \mathrm{PPL}_a(16384)$. Exact intersections of these lines give Table 61. At a 50/50 scored-token mix, $B = 400$ minimizes

Table 61: Minimum empirical NLL among static, $B = 800$, and $B = 400$. Intervals describe $p$, the 16K scored-token fraction; boundary points are ties. Only one seed is measured, and other schedules are not in this comparison.

| $T$ | $L_s$ | Static | $B = 800$ | $B = 400$ |
|---|---|---|---|---|
| 4000 | 4096 | $[0, .0140)$ | $(.0140, .2526)$ | $(.2526, 1]$ |
| 8000 | 4096 | $[0, .0137)$ | $(.0137, .7568)$ | $(.7568, 1]$ |
| 4000 | 8192 | none | $[0, .1458)$ | $(.1458, 1]$ |
| 8000 | 8192 | none | $[0, .7569)$ | $(.7569, 1]$ |

risk at $T = 4000$, whereas $B = 800$ minimizes it at $T = 8000$, for either short length. Thus the absolute-count ranking at 16K does not transfer as a workload-independent selection rule. At the larger budget, fixed $f = 0.9$ ($B = 800$) is preferable over a broad range of mixed-length weights, while $f = 0.95$ ($B = 400$) suits predominantly longest-length scoring. These are post-hoc operating regions, not a validated adaptive policy: new training seeds could move the boundaries, request proportions differ from scored-token proportions, and unmeasured lengths are not covered.

### P.4 A SECOND INDEPENDENTLY PRETRAINED 137M BASE

To vary more than continuation randomness, we pretrained a second 137M model from random initialization with seed 1, without loading base 0. It uses the same training implementation and base recipe: 25000 updates at length 1024, batch size 8, PG19 tokens, peak learning rate $3 \times 10^{-4}$, 1500-update warmup and the same 2% synthetic-copy mixture (204.8 million pretraining tokens). Static custom YaRN-family and DEPA $0.9T$ then each continue from this checkpoint for 4000 updates at length 4096, using continuation seed 2, reset optimizer state and target attention temperature throughout (131.072 million tokens per arm).

On base 1 (Table 62), DEPA improves 8192 PPL by 0.299 and 16384 PPL by 1.791 (6.19%), with a 0.031 trained-length cost. Its paired NLL gains are $-0.001101$, $0.010261$, and $0.063878$, compared

Table 62: Custom-YaRN-family timing effects on two independently initialized pretraining runs. Continuation seed is 2 in each pair; lower PPL is better.

| Base | Schedule | 4096 | 8192 | 16384 |
|---|---|---|---|---|
| 0 | Static | 28.239 | 29.459 | 29.007 |
| 0 | DEPA | 28.260 | 28.861 | 27.044 |
| 1 | Static | 28.434 | 29.277 | 28.938 |
| 1 | DEPA | 28.465 | 28.978 | 27.148 |

with $-0.000749$, $0.020520$, and $0.070074$ on base 0 at the same continuation seed. The base-1-minus-base-0 differences in paired NLL gain are $-0.000352$, $-0.010259$, and $-0.006196$. Thus both bases show the same small trained-length cost and positive extrapolation effect; the 8192 benefit is smaller on base 1. This extends custom-YaRN-family evidence beyond a single pretrained checkpoint, without treating two bases as a population confidence interval. The PI decaying-LR family uses base 0; its constant-LR replication also includes base 1. The source-version caveat of Appendix D applies to the base-0 pretraining. This base pair keeps the copy mixture fixed; Appendix N.1 separately removes the continuation mixture. No additional capability evaluation was run for this pair.

### P.5 Learning during the source hold

**What improves before release?** Saved probes that evaluate each run at its current positional configuration show that long-length language-modeling improvement emerges during the source hold, before conversion. On continuation seed 1, source-configuration 16384 PPL changes from 458.473 at the initial base to 52.957 after 1001 updates, then 54.445 and 53.672 after 2001 and 3001 updates. Static PI on the same probe windows gives 248.988 initially and 156.456, 163.807, 171.925 at those updates. Seed 2 shows the same pattern: source-hold PPL 420.611, 49.158, 50.294, 52.402 versus static PI 234.500, 140.597, 154.677, 158.246. The first saved interior observation therefore already favors the source-hold model at its own configuration in both seeds; the exact crossing time is unobserved.

These probes reset the validation sampler to the same seed at every measurement and use 20 batches (655360 scored tokens per length), whereas the final sweeps use 40 batches. The interior probes are logged after the optimizer update at each labeled global step, hence 1001 rather than 1000 completed continuation updates. Source versus PI configurations differ between arms throughout these observations. The pattern shows that the hold phase learns useful long-length language-modeling behavior; it does not show that the held weights already perform well at PI, identify the responsible model components, or demonstrate retrieval. The end-only conversion comparison separately establishes that target-phase training remains important.

The distinction between learning during the hold and adapting to PI is empirically important (Figure 4a): source frequencies are retained while training at four times the pretraining length, so long offsets are already encountered before release. The own-configuration improvement precedes conversion, whereas the fixed-weight end-only PI intervention degrades the final hold-only model. Together these observations support separate roles for new-length training and target-phase adaptation, without assigning the benefit to any particular attention head, frequency band, or capability. In Figure 4a, the crossing time between the base and the first interior probe is not measured, and the final 40-batch endpoint sweeps are not joined to these 20-batch trajectories.

## Q WikiText-103 main-scale replication

A fresh 137M base was pretrained on WikiText-103 with the identical architecture, optimizer, learning-rate schedule, token budget (25000 updates at 1024, 204.8M tokens) and 2% copy mixture as the independent PG19 base, differing only in the corpus (final trained-length PPL 28.17). Two fresh continuation seeds (2 and 3) each ran the custom YaRN-family static and DEPA ($f = 0.9$) arms under the constant-LR token-matched protocol; within-seed input/target-stream digests agree (Table 63). The two static-to-DEPA 16K gains ($+4.25$/$+4.43$) place the timing benefit and the harm-reduction reading on a second corpus; both cells meet the transfer criteria fixed before these

Table 63: WikiText-103 main-scale replication (137M, custom YaRN-family target, constant LR, release at $0.9T$). Natural-16K utility $U$ compares the full 16K input with its final 1K tokens on the same targets (negative values mean the longer input hurts).

| | | PPL at length (tokens) | | | |
|---|---|---|---|---|---|
| Seed | Schedule | 4K | 8K | 16K | $U$ (nats/token) |
| 2 | Static | 23.00 | 23.65 | 26.96 | $-0.571$ |
| | DEPA $0.9T$ | 23.03 | 22.69 | 22.71 | $-0.079$ |
| 3 | Static | 21.80 | 23.27 | 27.67 | $-0.560$ |
| | DEPA $0.9T$ | 21.84 | 22.29 | 23.23 | $-0.075$ |

Table 64: Comparison to prior long-context positional recipes along the axes that define a schedule comparison. "Endpoint" (matched) means the final positional configuration is identical to the comparator's target; "Budget" (matched) means the same continuation token budget; "Band order / probe" means training-specific band curricula or skip-window probes. Entries read each published recipe at the coarse level named in its citation; the table does not implement every variant. — and n/a mark axes that do not apply to a method.

| Method | Holds | Switches | Endpoint | Budget | Band order / probe |
|---|---|---|---|---|---|
| PoSE (Zhu et al., 2023) | no positional hold (skip-window offsets) | offsets exposed throughout | not asserted | not asserted | skip-window schedule |
| Base-value rescaling (Liu et al., 2023b) | source base (pretrain) | one-time base rescale at continuation | endpoint base | no | no |
| CLEX (Chen et al., 2023a) | — | frequency conditioned on length | conditioned family | via training curve | conditioned model |
| Code-Llama-style staged fine-tuning (Rozière et al., 2023) | source base in stage 1 | base raise at the start of long-context fine-tuning | target stage config | stage budget | no |
| SelfExtend (Jin et al., 2024) | — | windowed attention at inference | n/a (inference) | n/a | no |
| XPOS (Sun et al., 2023) | — | blockwise causal attention | n/a (architecture) | n/a | no |
| **DEPA (this paper)** | **source frequencies** | **one binary switch at** $0.9T$ | **yes** | **yes** | **no** |

runs (8K/16K margins non-negative, 4K cost $< 0.1$ PPL). All cells of Table 63 are recomputed from the per-run results (384 natural windows, identity residual 0); 15 of 16 cells equal the logged values, and the seed-2 DEPA utility cell is $-0.079$, logged as $-0.080$. Two fresh seeds are descriptive, not population-level.

## R Extended positioning and comparison boundaries

Several established recipes also decouple new-length training from the target positional configuration, and at a coarse level DEPA is the same two-stage "hold-then-rescale" object; the differences are the schedule and the level of control enforced across the comparison. PoSE exercises large positions during extension by placing sampled segments at non-contiguous offsets, exposing the range of target indices immediately rather than switching a held source vector once at a matched budget (Zhu et al., 2023). Scaling-law studies characterize how the rotary base value sets the extrapolation trade-off and motivate a one-time base rescale at continuation (Liu et al., 2023b); gradual base-frequency ramps instead spread such a change across training. CLEX conditions the frequency vector on the sequence length and trains a family of scales, removing the positional schedule from the stage sequence

altogether (Chen et al., 2023a). Large-model long-context recipes change the RoPE base once for a long-context fine-tuning stage rather than repeatedly per training pass (e.g. Code Llama raises the base to about $10^6$ during its long-context fine-tuning) (Rozière et al., 2023); our two-step midway-rescale control (Appendix E) is an abstraction of such staging, not a faithful implementation of that recipe. Structural methods change which relative rotations are used rather than when the final vector is applied—grouped "extended" attention (SelfExtend) (Jin et al., 2024) and length-extrapolatable rotary constructions that combine rotation with an exponential-decay gating and blockwise causal attention (XPOS) (Sun et al., 2023). DEPA is the degenerate, fully controlled point of this space: one binary switch of the complete final vector at a fixed length and token budget, with the endpoint positional configuration held identical across arms. The contribution is the controlled schedule comparison and its measured schedule-shape contrasts, not the coarse two-stage object.

**Positioning against progressive scaling schedules.** Continuous base-frequency scaling (NTK-aware scaling and gradual base-frequency adjustment) and dynamic schedules vary the effective frequencies over continuation rather than switching once; two-stage "hold-then-rescale" pipelines are the same object as DEPA at a coarse level. The present comparison is deliberately internal to the schedule-shape axis at a matched final configuration: any recipe that reaches the same target can be entered into the same two-arm protocol (equal continuation tokens, equal final RoPE, identical evaluation), and the exposure-matched tail ramps (Appendix P.1, including its nonlinear-shape paragraph) serve as the measured progressive-schedule control — they reach the same target through a gradual multiplier course of equal cumulative exposure, and the single late release beats them at the long lengths. Two boundaries of the claim are stated rather than enforced: dynamic inference-time schedules that rescale inside a single sequence act on a different axis than the training schedule studied here, and a head-to-head against a third-party "searched" configuration (e.g. an optimized LongRoPE multiplier table or a specific rollback recipe) would be a direct application of the same protocol but is outside the evidence we report. The contribution is therefore the schedule-shape result under matched configurations, not a claim of dominance over every externally studied multiplier course.

## S Extended discussion of limitations

This section collects the detailed limitations summarized in Section 6.1.

**Scale.** The largest model has 137M parameters and the main extension is 1K→4K, so transfer to billion-parameter models and to production extension recipes, and an adaptive timing rule, remain open.

**Trained-length cost.** The gain lies beyond the trained window: at 4K, DEPA is consistently slightly worse, by 0.02–0.04 PPL under the custom target and 0.9–1.4 PPL (about 3–5%) under PI (Tables 1 and 2). An application that only uses the trained length therefore gains nothing from DEPA and should weigh this in-window cost.

**Evaluation and context use.** The evidence is perplexity-based; no downstream long-context benchmark such as LongBench or RULER was run. Lower long-input perplexity does not establish retrieval: on identical targets every tested schedule still worsens when context grows beyond the trained window, and utility stays negative in every row of Table 4, so the gain mainly reduces a penalty. The nested intervention changes prefix availability and the input boundary; it does not isolate semantic relevance or an attention circuit. A counterfactual retrieval probe, whose paired contexts differ only in an answer token placed 4K, 8K, or 16K positions before the query, finds no retrieval gain: across six training-seed pairs in three experimental families under PI and published YaRN, balanced two-candidate accuracy stays within one pair of chance (0.5) in every cell, and neither schedule shows a consistent content-use advantage (Appendix L). The probe was not run on the custom target, and extrapolation-PPL gains should not be read as a demonstrated retrieval capability.

**Statistics.** Where reported, uncertainty is measured across two to five continuation seeds on a single base; its unit is the continuation seed, not the pretrained-model population. Single-pair rows are paired outcomes, not population confidence intervals, and the nested-probe windows can overlap and

are reused across seeds, so their means are descriptive rather than independent-sample intervals. The second corpus and architecture broaden the evidence without establishing population-level reliability over arbitrary pretrained models or a universal release rule.

**Baselines and recipes.** Both arms share one optimizer recipe by design, and a separately tuned static recipe could narrow the gaps under PI and published YaRN, whose large gains partly reflect weak static endpoints (about 105–190 PPL at 16K); longer PI budgets likewise enlarge the gain partly by worsening the static endpoint. Because masks and temperatures both differ, Table 2 establishes recipe-dependent outcomes, not a temperature-only causal effect; the custom and published YaRN implementations cannot be interchanged, and the temperature-timing probe does not compare their two numerical temperature definitions. Neither the original searched LongRoPE configurations nor DroPE are matched experimental competitors here.

**Schedule.** Fixed $0.9T$ is a tested default, not a per-recipe optimum or a universal release point. DEPA's margins over the tail-linear transition are small; nonlinear cosine transitions and nearby release fractions show no uniform ordering across protocols; nearby release refinements can change sign across seeds; and the contribution of execution nondeterminism to the smallest refinements remains unresolved (Appendix O.1). These results support the primary timing contrast more strongly than a universal advantage of abrupt over gradual transitions. State-based release candidates do not beat their matched fixed deadlines. Staging the hold across lengths weakens the 16K gain, and the staging results are measured cost–benefit points, not a universal scaling law. Removing continuation copy batches does not remove the pretraining copy mixture.

**Theory.** The recovery model is conditional, not a weight-level mechanism: it neither proves convergence of transformer training nor selects $f = 0.9$, the full-network bound needs bounded loss derivatives that geometry alone does not supply, and recovery to a schedule's own pre-release loss does not reliably rank final extrapolation PPL. The angle-coverage view of Section 3 is an interpretation, not a tested mechanism, and the source-hold trajectories of Figure 4a compare each arm at its own configuration, so they do not show that the held weights already perform well at the target.

**Overall scope.** DEPA changes the training path under a shared final positional configuration. The evidence spans 19M–137M models, two corpora, and five pretrained bases in three architectures, with continuation lengths 512–4096. This is limited coverage, especially for contemporary large-model context extension. Recipe, base, corpus, and optimizer variation should remain separate comparison units. The measured sign of the primary timing gain transfers more broadly than its magnitude. Practical use must weigh extrapolation against in-window cost; fixed $0.9T$ is a tested baseline, not a universally safe or optimal setting. Small schedule-shape differences are less secure than the static-versus-delayed contrast because execution nondeterminism can contribute to them. Reduced added-context harm leaves residual negative utility and does not establish retrieval or copy improvement, and further applications should distinguish the intended evaluation length from the mere availability of a larger input window.