WHEN REASONING TRACES OVERRUN THE CANVAS OF DIFFUSION LLMS

Anonymous authors
Paper under double-blind review

ABSTRACT

A diffusion language model (dLLM) decodes into a fixed-length canvas, and supervised fine-tuning (SFT) on teacher traces shapes how much it writes before answering. We show that the SFT corpus affects such a student largely through *overrun*: reaching the end of the canvas without an answer, which we measure directly on saved generations of LLaDA-8B students. Holding teacher, problems and correctness fixed, prompting the teacher for terse rather than detailed traces raises GSM8K accuracy by 15.8 points at a 512-token canvas, and most of the gap (14.3 points) is attributable to overrun. A statistic computed before training, the share of training responses longer than the canvas, orders student overrun across corpora. A second check exposes a defect in the standard d1 pipeline: its 4,096-token SFT cap truncates about 90% of s1K examples and leaves only about one in ten with its answer, which teaches the student not to terminate. On LLaDA-8B-Base, where s1K SFT is no better than no SFT, replacing it with a corpus that fits the canvas improves accuracy by 8.8 GSM8K points, attributable to overrun, and by 13.7 MATH500 points, about half attributable to truncation. On LLaDA-8B-Instruct, which rarely overruns in-domain, s1K remains better in-domain; but on Countdown the s1K-trained student overruns on most problems, before and after identical GSM8K RL, and this accounts for the budget-fitting pipeline's advantage there.

1 INTRODUCTION

Diffusion language models (Austin et al., 2021; Nie et al., 2026; Ye et al., 2025a) generate by iteratively unmasking a block of positions instead of emitting tokens left to right. They are taught to reason with the recipe of autoregressive (AR) models (Wei et al., 2022; Ouyang et al., 2022): SFT on teacher reasoning traces, then reinforcement learning (RL). d1 (Zhao et al., 2026a) established this pipeline for dLLMs, with s1K (Muennighoff et al., 2025) as the SFT corpus, and subsequent work has concentrated on the RL objective (Rojas et al., 2026; Zhao et al., 2025; Zhu et al., 2025; Tang et al., 2026; Wang et al., 2026a). IGPO (Zhao et al., 2026b) proposed SFT on rewritten, concise traces aligned with dLLM generation as part of its RL recipe. This paper asks *why* trace length matters for a dLLM, how to tell in advance whether a corpus will hurt, and when fixing it will and will not help.

Our answer is a failure we call *overrun*. A dLLM decodes a canvas of L positions and pays for all of them at every denoising step, so it is run with small budgets (128 to 512 tokens in d1, 1,024 here), whereas an AR decoder stops paying at the end-of-sequence token and is routinely given tens of thousands. A student fine-tuned on traces that are long relative to L, or that never end, learns to write past the end of the canvas and runs out of room before it answers (Figure 1). Overrun is directly observable in saved generations, which lets us attribute every accuracy difference between two students to overrun, to stopping without an answer, or to wrong answers, problem by problem. Figure 2 shows the scale of the mismatch for the corpora we study.

Findings. *Trace length causes overrun.* With one 14B teacher prompted to be terse or to show all its work on the same problems, overrun (3.3% vs 25.5% of problems) is attributable for 14.3 of the 15.8-point GSM8K gap. *It is predictable before training*: the share of training responses longer than L orders student overrun in 16 of 19 corpus pairs within a student and budget. *The standard*

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.

Problem (GSM8K test). John has 2 houses with 3 bedrooms each. Each bedroom has 2 windows each. There are an additional 4 windows in each house not connected to bedrooms. How many total windows are there between the houses?

Student trained on the 7B corpus ($\rho = 44.9\%$), $L = 1024$: *Hmm, let me break this down step by step. First, [John] has 2 houses, and each house has 3 bedrooms. So, the total number of bedrooms is* $2 * 3 = 6$. *Now, each bedroom has 2 windows* ... $6 * 2 = 12$. *Additionally* ... $2 * 4 = 8$. ... $12 + 8 = 20$. *So, I think the total number of windows between the houses is 20. But [let] me, I want to make sure I didn't make a mistake. Let me double-check my calculations* ... [the check repeats until all 1,024 positions are filled; no answer is written] **overrun**

Student trained on the budget-fit 32B corpus ($\rho = 4.9\%$), same budget, 127 tokens: *First, calculate the number of windows in one house. Each house has 3 bedrooms with 2 windows each, so* $3*2 = 6$ *windows. Additionally, each house has 4 more windows* ... $6+4 = 10$ *windows. Since John has 2 houses, the total number of windows is* $2 * 10 = 20$ *windows.* `<answer>\boxed{20}</answer>` **correct**

Figure 1: **What overrun looks like.** Both students reach the right value; only one writes it down. Verbatim from saved outputs (LLaDA-8B-Base, seed 42), elided for space. About half of such overruns already contain the correct value (Section 5.1).

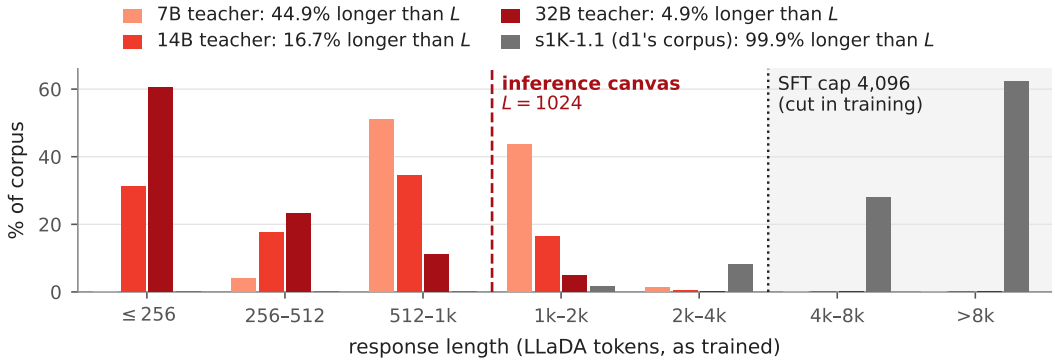


Figure 2: **SFT corpora against the two lengths that matter**, on the responses as they are trained. The inference canvas ($L = 1024$) cuts through the teacher corpora in order of verbosity; s1K-1.1 lies almost entirely beyond d1's SFT cap, so 91% of its examples are truncated in training (Table 1); its bars show lengths before truncation.

corpus fails a second check: after d1's 4,096-token cap, about nine in ten s1K examples have lost their answer and their end, and the resulting student overruns through repetitive non-termination rather than long reasoning. *On the base student the corpus effect is overrun*: replacing s1K with a canvas-fitting corpus gains 8.8 GSM8K points, with the overrun term positive in every seed pair and no detectable difference in answer quality. *Where the student does not overrun, the lever does not act*: on LLaDA-8B-Instruct, s1K barely raises in-domain overrun and remains better in-domain, while on Countdown its student overruns on 62% of problems before RL and 59% after identical diffu-GRPO, against under 1% for ours.

Contributions.

1. We identify overrun as the main channel through which the SFT corpus shapes a base dLLM's reasoning and its cross-task behaviour, distinguish its two forms (over-long reasoning and non-termination), and establish the first with a same-teacher length intervention (Section 5.1).

2. We give two checks computable before training, the over-budget rate and the post-cap answer rate, show that the first orders measured student overrun at a given budget, and use the second to expose the answer-stripping truncation in d1's SFT stage (Section 5.2).

3. We attribute each comparison problem by problem to overrun, stopping or wrong answers, with bootstrap intervals, per-seed values and common-support tests, and explain the base-model gain, the instruction-tuned in-domain null and the cross-task gap after RL within one account (Sections 5.3–5.5).

2 Related Work

Masked diffusion (Austin et al., 2021; Lou et al., 2023; Shi et al., 2024; Sahoo et al., 2024) was scaled to 8B parameters by LLaDA (Nie et al., 2026) and Dream (Ye et al., 2025a), which decode a fixed-length canvas. d1 (Zhao et al., 2026a) introduced SFT on s1K followed by diffu-GRPO; later work refines the RL objective (Rojas et al., 2026; Zhao et al., 2025; Zhu et al., 2025; Wang et al., 2026b; Tang et al., 2026; Wang et al., 2026a), some starting RL directly from an instruction-tuned model. Closest to us, IGPO (Zhao et al., 2026b) proposes SFT on rewritten concise traces to align SFT with dLLM generation before RL; we supply the mechanism behind that recipe, diagnostics that predict when it is needed, and evidence for when it does not: on our instruction-tuned student a length-fitting but single-domain corpus loses in-domain, which suggests that length alignment and domain coverage are separate requirements. Rainbow Padding (Kim et al., 2026) shows that end-of-sequence padding distorts the length behaviour of instruction-tuned dLLMs, and Prophet (Li et al., 2026b) that dLLMs often fix their answer early in decoding, which suggests decoding-side remedies for overrun. Inference-time methods relax the fixed canvas by expanding it, cropping it or editing its length during denoising (Li et al., 2026a; Rossi et al., 2026; Yang et al., 2026; Wu et al., 2026); they act on a trained model whose length habit is already set. In AR distillation, trace selection matters more than volume (Ye et al., 2025b; Guha et al., 2026), small students learn better from shorter chains (Li et al., 2025b), and conciseness is pursued for latency (Sui et al., 2025; Aggarwal and Welleck, 2025). Under a binding canvas a chain that does not fit costs the answer outright. Appendix A extends these threads.

3 Overrun: How the Corpus Meets the Canvas

3.1 Masked diffusion SFT

We fine-tune LLaDA-8B with the masked-diffusion objective of LLaDA and d1 (Nie et al., 2026; Zhao et al., 2026a): draw $t \sim \mathcal{U}(0,1)$, mask each non-prompt position of the sequence $\mathbf{y}$ (response and padding, N positions) with probability t, and minimize

$$\mathcal{L} = \mathbb{E}_{t,\mathbf{m}}\left[\frac{1}{t\,N}\sum_{j\in\mathbf{m}} -\log p_\theta\big(y_j \mid \mathbf{x}, \mathbf{y}_{\setminus\mathbf{m}}\big)\right]. \tag{1}$$

Responses are padded to the SFT sequence cap with end-of-sequence tokens, which are among the targets. At inference the model unmasks a canvas of L positions, 32 at a time from left to right, with low-confidence remasking inside each block.

3.2 Overrun, and an exact accounting of accuracy

A generation *answers* if the benchmark's parser extracts an answer from it. An unanswered generation *overruns* if its text reaches the final 32-token block of the canvas, and *stops* otherwise. Every generation is thus correct, wrong, overrun or stopped, and accuracy factors exactly as

$$\Pr[\text{correct}] \;=\; \Pr[\neg\text{overrun}] \;\times\; \Pr[\text{answer} \mid \neg\text{overrun}] \;\times\; \Pr[\text{correct} \mid \text{answer}]. \tag{2}$$

The identity is bookkeeping, not evidence. To attribute a gap between students A and B on the same test set we count problems: the gap equals the share of problems A solves and B fails minus the converse, and each discordant problem is assigned to the way the failing student failed (overrun, stopped or wrong). The three terms sum exactly to the gap, need no assumption about how an overrun problem would otherwise have been answered, and admit bootstrap intervals by resampling problems. (A log-ratio split of Equation 2 in the style of LMDI (Ang, 2005) gives similar numbers but implicitly assumes overrun problems are of average difficulty, which they are not; Appendix C.1.) Conditional accuracy compares students on different, self-selected subsets, so we also test it on the common set of problems both students answer (Appendix C.2).

3.3 What the student inherits

SFT shapes how much the student writes before its answer, and a student can fail to answer within the canvas in two ways. It can reason at the length of long training traces, producing derivations that run

past L; or, if its training sequences were cut before their end, it can fail to terminate at all, producing repetitive text until the canvas is full. Both are overrun, and saved generations distinguish them: the first kind is non-repetitive and often already contains the correct value, the second is dominated by repeated text (Sections 5.1 and 5.2). The student does not simply copy corpus lengths, however. Its behaviour also depends on the canvas it is given: some students place their answer near the end of whatever canvas they receive, so enlarging the canvas can raise overrun (Section 5.6), consistent with the end-of-sequence padding effects reported for instruction-tuned dLLMs (Kim et al., 2026). We therefore treat corpus length relative to a budget as an ordinal screen, not as a calibrated estimate of the student's overrun. The same logic applies to an AR student decoded under the same cap; what makes it bind for dLLMs is that the cap is small in practice.

3.4 Two diagnostics and a screen

For a corpus $\mathcal{D} = \{(x_i, y_i)\}$ of retained responses, inference budget L and SFT cap C, define the *over-budget rate* and the *post-cap answer rate*

$$\rho(\mathcal{D}, L) = \frac{1}{|\mathcal{D}|} \sum_i \mathbf{1}[|y_i| > L], \qquad \tilde{\varphi}(\mathcal{D}, C) = \frac{1}{|\mathcal{D}|} \sum_i \mathbf{1}\big[\,[x_i; y_i]^{\le C} \text{ keeps its answer block}\big], \tag{3}$$

where $|y_i|$ counts tokens of the response as it enters training and $[x_i; y_i]^{\le C}$ is the templated prompt and response truncated at C. Both cost one tokenization pass. ρ is a length screen: on our corpora mean response length orders corpora exactly as ρ does, and ρ ranks corpora at a given budget without predicting how overrun changes when the budget changes (it can rise with L; Section 5.6). $\tilde{\varphi}$ catches the separate failure of training on sequences that lose their answer and end. The practical use is a screen, not an optimization: at the deployment budget, discard corpora with high ρ or with $\tilde{\varphi} < 0.95$ and choose among the rest on quality, since minimizing ρ alone would favour answer-only corpora. Among our three teacher corpora the screen retains the 32B teacher's, which we call *budget-fit*; teacher size and length are collinear across these teachers, so the causal claim rests on the same-teacher experiment.

4 Experimental Setup

Students, teachers, corpora. Students are LLaDA-8B-Base and LLaDA-8B-Instruct (Nie et al., 2026). Teachers are DeepSeek-R1-Distill-Qwen 7B, 14B and 32B (Guo et al., 2025), sampled once per GSM8K training problem (Cobbe et al., 2021) (temperature 0.6, top-p 0.95, at most 2,048 tokens) and filtered by a strict answer match. Each response is converted to d1's format, the teacher's thinking inside `<reasoning>` and the answer inside `<answer>`. The baseline is s1K-1.1, the DeepSeek-R1 re-release of s1K; d1's code trains on the original s1K, which we analyse under the same cap (Appendix D). s1K is trained with d1's generic system prompt, as in d1, and the teacher corpora with d1's math prompt. We use d1's SFT and RL hyperparameters except that LoRA (Hu et al., 2022) adapts all attention and MLP projections rather than only query, key and value, identically for every corpus (Appendices K –L).

Evaluation. GSM8K test (1,319 problems), MATH500 (Hendrycks et al., 2021) and Countdown (Gandhi et al., 2024) (256 problems) at $L = 1024$, twice d1's largest budget so that the canvas is not artificially tight (results at $L = 512$ are in Section 5.6), with 512 denoising steps, block length 32 and greedy decoding, using d1's evaluation code, parsers and per-benchmark prompts for every model. All generations are stored.

Seeds and checkpoints. Every seeded comparison uses seeds $\{42, 1, 2\}$ in both pipelines, end to end (the length intervention and the Countdown SFT evaluations are single runs); each RL run starts from the SFT checkpoint of its own seed. All seeds of a configuration are evaluated at one reference step, the peak of the seed-42 screening curve at $L = 512$ (for s1K on Base this is every seed's own best step; for s1K on Instruct and all RL runs it is the final step). Choosing the step on one seed flatters that seed (Appendix E), so we report all three seeds and check the key orderings under selection-free protocols (Table 2). Single runs are marked †.

Table 1: **Corpus diagnostics**, computed before training on the responses as trained (LLaDA tokenizer). The isolation corpora come from one 14B teacher on the same 5,661 problems, both variants correct. Only s1K is affected by the SFT cap; $\tilde{\varphi}$ counts examples whose closed answer block survives the cap (11.3% of s1K-1.1 still contain some boxed expression). Raw-output statistics and the original s1K are in Appendix D.

	7B	14B	32B	s1K-1.1	14B terse	14B verbose
Responses retained	6,363	5,841	6,110	1,000	5,661	5,661
Mean / median tokens	1,042 / 970	609 / 530	335 / 230	10,220 / 9,830	292 / 225	943 / 878
$\rho(\mathcal{D}, 512)$	95.9%	51.1%	16.2%	100%	11.7%	94.2%
$\rho(\mathcal{D}, 1024)$	44.9%	16.7%	4.9%	99.9%	0.7%	33.0%
Exceeds SFT cap $C = 4{,}096$	0%	0%	0%	91.3%	0%	0%
Post-cap answer rate $\tilde{\varphi}$	100%	100%	100%	9.0%	100%	100%

Statistics. We report sample s.d. and $[\min, \max]$ over seeds, computed from unrounded values. At $n = 3$ per arm the exact one-sided rank-sum floor is $p = 0.05$, attained only when seed sets separate, and none of our seven two-sample tests survives a Holm correction (Appendix C). Mechanism claims rest instead on per-problem accounting over 1,319 GSM8K problems per run, with bootstrap intervals and exact McNemar tests.

5 Results

5.1 Longer traces from the same teacher produce overrun

The isolation corpora of Table 1 share teacher, problems and correctness and differ in length (292 vs 943 mean tokens) and in whatever stylistic changes the two prompts induce. Fine-tuning LLaDA-8B-Base on each with identical hyperparameters and evaluating at $L = 512$, where the two corpora sit on either side of the budget, gives 72.2 vs 56.4 on GSM8K and 28.6 vs 13.8 on MATH500. On GSM8K the verbose-trained student overruns on 25.5% of problems against 3.3%; problem by problem, overrun accounts for 14.3 of the 15.8 points (95% CI 12.2–16.4), stopping for 0.0, and wrong answers for 1.5 (CI -0.5 to 3.7; Figure 3c). On the 951 problems both students answer, the terse student leads by 2.1 points (McNemar $p = 0.19$). The prompts may change style as well as length. The verbose student's overruns are mostly long, non-repetitive derivations (median share of repeated 8-grams 0.04) and 52% of them already contain the correct value, which argues against a self-checking loop as the main cause, but only a prompt-free length split can rule style out (Appendix O). This is one training run per arm, both evaluated at the same step, and the gap lies between 13.6 and 23.8 points at every one of the five screened checkpoints (Appendix B). With one teacher, trace length moves the student over a wider range (56.4 to 72.2) than the three teachers do at this budget (61.9 to 73.4).

5.2 The over-budget rate orders student overrun

Figure 3a–b places every trained corpus at its ρ against the overrun of the resulting student. Within each student and budget, the higher-ρ corpus yields the student that overruns more in 16 of 19 pairs. The three exceptions involve corpora whose ρ differs by under five points or whose students both overrun on under 0.2% of problems. The evidence is modest in size: most pairs repeat the ordering of the three teacher corpora, and the one panel with non-teacher corpora (Base, $L = 512$) has Kendall $\tau = 0.6$ over five corpora ($p = 0.23$). The same-teacher corpora matter most, because they move student overrun from 3.3% to 25.5% with the teacher held fixed. The statistic is ordinal and student-specific: overrun is a small fraction of corpus ρ on the instruction-tuned student, whose prior habits compete with the inherited one. Including s1K, which the cap truncates, 25 of 33 pairs are concordant; mean corpus length gives identical counts, so ρ is best read as a budget-referenced length screen.

Student accuracy follows where overrun is the dominant failure. On Base, accuracy agrees with ρ across the three teacher corpora (seed 42: $56.4 < 70.5 < 74.0$). On Instruct, the three teacher corpora fall in ρ order under both selection-free protocols (Table 2). At the reference step the 32B corpus's three-seed mean (75.8) falls below the single 14B run (78.2); seed 42 alone gives 79.3, and seeds 1 and 2 give 73.8 and 74.3.

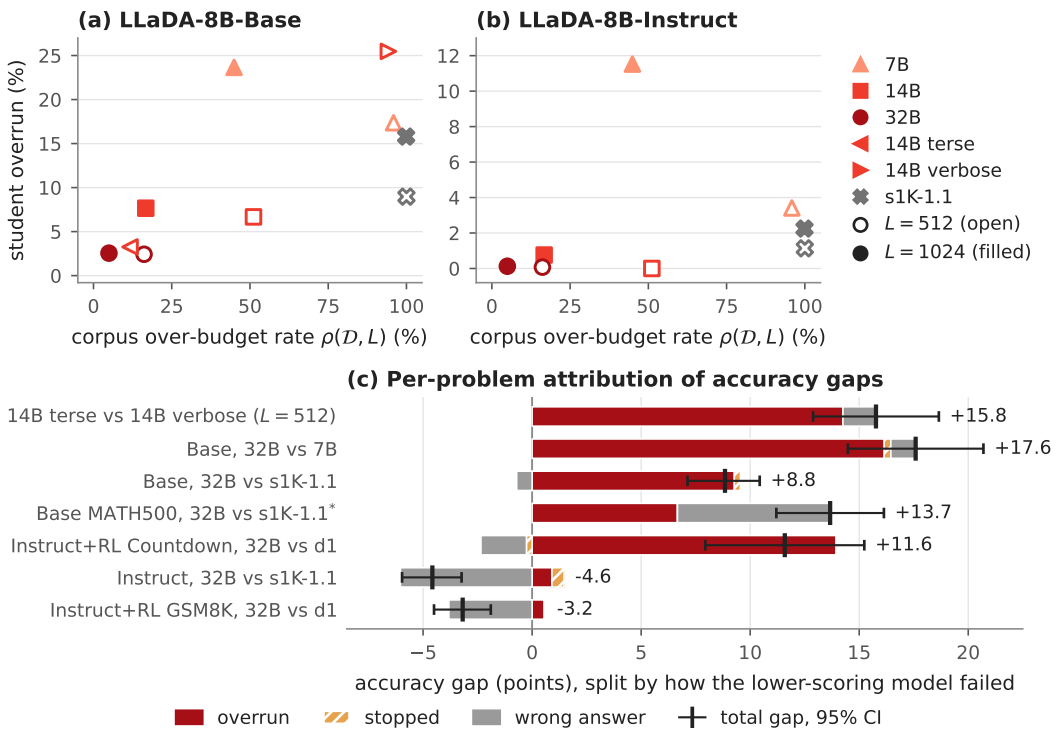


Figure 3: **(a, b)** Corpus over-budget rate against the student's measured GSM8K overrun, for every trained corpus at both budgets (open markers L=512, filled L=1024; the same-teacher corpora were evaluated at L=512 only). **(c)** Per-problem attribution of accuracy gaps: each problem one student solves and the other fails is assigned to how the failing student failed. Bars sum to the gap; whiskers are 95% bootstrap intervals for the gap. Rows against s1K-1.1 or d1 pool three seed pairs, the others are single runs; all rows at L=1024 unless marked. *On MATH500, where the parser also scores truncated text, the budget term counts responses cut off by the canvas.

Table 2: **The teacher-corpus ordering does not depend on checkpoint selection.** GSM8K at $L = 512$ (full test, seed 42) at the reference step (selected on this curve), the final step, and the mean over all saved checkpoints (ten; five for s1K). The last two involve no selection.

	LLaDA-8B-Base			LLaDA-8B-Instruct		
Corpus	Reference	Final	Mean	Reference	Final	Mean
7B	61.9	51.8	53.4	76.5	75.8	74.7
14B	70.9	68.8	67.9	80.7	78.5	78.0
32B (budget-fit)	73.4	73.1	71.1	81.9	80.2	79.3
s1K-1.1	69.9	63.6	64.3	81.9	81.9	80.4

The default corpus loses its answers during SFT. s1K-1.1 responses average 10,220 tokens. d1's pipeline truncates each templated sequence at 4,096 tokens, which cuts 91.3% of examples mid-derivation; only 9.0% keep their closed answer block, and truncated examples carry no end-of-sequence token. The original s1K that d1's code loads is shorter (mean 5,440 tokens) but fares similarly: 88.0% are cut and 12.1% keep the answer block. d1 notes that it truncates the dataset at 4,096 tokens; the consequence that nearly nine in ten examples lose their target and their end has not, to our knowledge, been reported. Before the cap, 93.9% of s1K-1.1 responses contain a boxed answer, so a format check on the raw corpus does not reveal the problem. This is not a criticism of s1K, which was built for AR models trained with long contexts, but of transplanting it under a short cap. The s1K-trained student sits off the ρ relation (Figure 3a) and overruns differently from the others: its median answer position is the shortest of any student (212 tokens), but on $15.8 \pm 4.9\%$ of GSM8K problems at $L = 1024$ it never terminates, and 76–89% of these overruns (by seed) are

Table 3: **Results at $L = 1024$.** Seeded entries are mean ± s.d. over three seeds at a fixed reference step, with $[\min, \max]$ where space allows (all per-seed values in Table 10); † single run. Overrun (share of problems) and $\Pr[\text{c} \mid \text{a}]$ (conditional accuracy) are measured on saved generations. Within a panel, rows differ only in the SFT corpus, except that the s1K arm uses d1's generic system prompt and step budget (Appendix K).

	GSM8K			MATH500	Countdown	
SFT corpus	Accuracy	Overrun	$\Pr[\text{c} \mid \text{a}]$	Accuracy	Accuracy	Overrun
LLaDA-8B-Base, SFT only						
None†	66.6	2.2	68.3	8.8	8.6	34.8
s1K-1.1 (d1)	64.2 ± 4.5 [60.6, 69.3]	15.8 ± 4.9	76.8 ± 0.8	13.9 ± 4.8	1.2†	90.2†
7B†	56.4	23.7	74.4	12.2	2.0	94.1
14B†	70.5	7.7	76.9	23.0	1.6	96.1
32B (budget-fit)	73.1 ± 2.6 [70.1, 75.1]	2.6 ± 1.1	75.2 ± 1.7	27.5 ± 1.7	4.7†	73.4†
LLaDA-8B-Instruct, SFT only						
None†	78.9	0.4	80.6	35.2	13.7	8.6
s1K-1.1 (d1)	80.4 ± 0.3 [80.1, 80.7]	2.2 ± 0.1	83.4 ± 0.3	34.6 ± 1.0	10.5†	61.7†
7B†	68.1	11.5	77.3	25.2	22.3	16.8
14B†	78.2	0.8	79.6	30.6	27.0	0.8
32B (budget-fit)	75.8 ± 3.0 [73.8, 79.3]	0.1 ± 0.1	76.3 ± 2.9	34.9 ± 2.2	28.1†	0.4†
LLaDA-8B-Instruct, SFT then diffu-GRPO on GSM8K (each RL seed from its own SFT seed)						
d1 (s1K-1.1 SFT)	81.8 ± 0.2 [81.6, 81.9]	1.3 ± 0.1	83.2 ± 0.2	34.5 ± 1.0	10.3 ± 0.2	59.4 ± 5.2
32B SFT	78.6 ± 3.0 [76.6, 82.0]	0.2 ± 0.1	79.0 ± 2.9	34.8 ± 3.0	21.9 ± 4.0	0.5 ± 0.6

dominated by repeated text, with the correct value present in only 18–22%. This is the failure $\tilde{\varphi}$ predicts, not the one ρ predicts.

5.3 On the base student, the corpus effect is overrun on GSM8K and half overrun on MATH500

Replacing s1K-1.1 with the budget-fit corpus raises LLaDA-8B-Base from 64.2 to 73.1 on GSM8K (+8.8) and from 13.9 to 27.5 on MATH500 (+13.7; Table 3). Our worst seed exceeds s1K's best on both ($70.1 > 69.3$, $25.6 > 19.0$), Cohen's $d = 2.4$ and 3.8, Welch $p = 0.057$ and 0.029. Relative to no SFT, the budget-fit corpus adds 6.5 and 18.7 points, whereas s1K SFT loses 2.4 on GSM8K.

The budget-fit corpus is also in-domain and has six times as many examples (though, after truncation, s1K supplies about twice as many response tokens: 3.8M vs 2.1M), so it might simply teach better GSM8K reasoning; that would appear as fewer wrong answers. It does not. Overrun is attributable for +9.3 points (CI 8.1–10.4, bootstrap clustered by problem; positive in each seed pair: 5.5, 10.9, 11.4) and wrong answers for −0.7 (CI −2.0 to 0.7). On the problems both students answer, conditional accuracy differs by +0.9 points in s1K's favour (90% interval $[-0.4, 2.2]$ over problems, $[-1.5, 3.2]$ over seeds; Appendix C.2): the in-domain corpus confers no detectable reasoning advantage here. What separates the two students is s1K's non-termination, the failure $\tilde{\varphi}$ flags (Section 5.2). The over-length form, which ρ flags, separates the teacher corpora: overrun is attributable for 16.1 of the 17.6 points between the budget-fit and 7B students (seed 42). Fine-tuning itself does teach reasoning: against the un-finetuned model, the budget-fit gain is almost entirely wrong answers turned right (+7.3 of +7.4, seed 42). The budget decides between corpora, not between fine-tuning and none.

MATH500 is different. Its parser takes the last boxed expression, which also scores truncated responses, so we count as budget failures the responses the canvas cuts off: 53.5% for s1K against 26.3% for the budget-fit student. Truncation is attributable for 6.7 of the 13.7 points and wrong answers for 7.0; on MATH500 the budget-fit corpus also teaches better answers.

5.4 On the instruction-tuned student, the habit is dormant in-domain

On LLaDA-8B-Instruct, s1K reaches 80.4 ± 0.3 on GSM8K against 75.8 ± 3.0, seed sets disjoint in s1K's favour, and MATH500 is tied. The accounting shows why corpus selection has little to offer

here: s1K raises this student's GSM8K overrun only to 2.2%, overrun accounts for just +0.9 points, and the comparison is decided by wrong answers (−6.0, CI −7.4 to −4.8); on the common support, s1K is significantly better on all three seeds (McNemar $p \leq 0.012$). A GSM8K-only corpus narrows a generalist. The diagnostics cannot foresee that a student will resist a length habit in-domain; the accounting states the condition under which a corpus can help, overrun in the comparison student, and that condition fails here.

It does not fail everywhere. Countdown asks the model to search over arithmetic expressions, and there the s1K-trained student overruns on 61.7% of problems against 0.4% for the budget-fit student, which scores 28.1 against 10.5 with no Countdown training data (single runs); the 17.6-point gap is attributable to overrun. The non-terminating habit is present in the s1K-trained Instruct student but expressed only where the task invites long exploration. Countdown overrun is not purely inherited from the corpus, however: on the base student, math-only SFT with every corpus, the budget-fit one included, overruns on 73–96% of Countdown problems (Appendix I).

5.5 Through RL: the Countdown gap survives

We apply d1's diffu-GRPO on GSM8K to each SFT seed, so the initialization is the only difference (bottom panel of Table 3). Our seeded d1 arm reaches 81.8 ± 0.2 on GSM8K, close to the 82.1 that d1 reports, and is highly reproducible. In-domain, RL narrows the gap without closing it (78.6 ± 3.0 vs 81.8 ± 0.2), and the residual is wrong answers (−3.8 of −3.2 points); our arm is unstable, and its best seed (82.0) exceeds every d1 seed, so the deficit reflects that instability as much as its mean. On Countdown the ordering is reversed: 21.9 ± 4.0 against 10.3 ± 0.2, our weakest seed above d1's strongest. d1's policies overrun on $59.4 \pm 5.2\%$ of Countdown problems and ours on $0.5 \pm 0.6\%$; overrun is attributable for 13.9 points (CI 11.5–16.5; 12.5 to 14.8 per seed pair) of the 11.6-point gap, partly offset by more wrong answers on our side (−2.1, CI −4.6 to 0.5). The gap is created by SFT (17.6 points before RL) and survives GSM8K RL at d1's 256-token training budget. d1 trains a separate RL model per task; these numbers measure transfer from GSM8K RL and are not comparable to d1's Countdown-specific results.

5.6 A larger canvas does not undo the habit

If overrun were only a matter of room, a larger canvas would remove it. Every checkpoint was screened at $L = 512$ and evaluated at $L = 1024$ with the same tokens per denoising step, so the same weights can be compared under two budgets (Appendix G). On Base, the median answer position is nearly unchanged (240 → 256 for the 7B student, 234 → 238 for the budget-fit one), but the long tail grows with the canvas, and overrun rises from 17.4% to 23.7% for the 7B student and from $9.0 \pm 2.2\%$ to $15.8 \pm 4.9\%$ for s1K over three seeds, while the budget-fit student stays near 2.5%. On Instruct, the 7B-trained student moves its answer to the end of the larger canvas and overruns on 11.5% instead of 3.4%. The size of the corpus effect therefore depends on the budget: at $L = 512$ the Base gap to s1K is smaller (seed 42: 73.4 vs 69.9; s1K's three-seed mean is 69.0 ± 2.8) because s1K's student overruns less on a smaller canvas, and ρ, which can only fall as L grows, does not capture this; it ranks corpora at a fixed budget. Accuracy changes between budgets are biased against the larger canvas, since the reference step was chosen at $L = 512$, so we lead with overrun, which was not selected on. We did not test adaptive-length decoders (Li et al., 2026a; Yang et al., 2026), early answer commitment (Li et al., 2026b) or answer forcing near the end of the canvas; they act at inference and may compose with corpus selection.

6 Discussion and Conclusion

Recommendations. For a dLLM decoded at budget L: compute ρ of each candidate corpus at L on the responses as trained, and $\tilde{\varphi}$ after the SFT cap with the actual template; screen out corpora that fail either check, then choose on quality. Measure overrun on saved generations and attribute differences problem by problem. Report seeds at a step not chosen on one of them.

Implications for evaluating dLLM post-training. Countdown and Sudoku are standard benchmarks in dLLM RL work (Zhao et al., 2026a; Tang et al., 2026). When a model trained on one task is evaluated on another, our results show that the comparison can be decided by overrun inherited from

SFT, invisible in-domain and measurable at no cost from saved generations. We suggest reporting overrun alongside accuracy and stating the SFT corpus's ρ at the evaluation budget and its $\tilde{\varphi}$ under the sequence cap used.

Scope and limitations. All students are LLaDA-8B (Base and Instruct), decoded with fixed canvases of 512 and 1,024 tokens; block-diffusion and variable-length models relax the fixed canvas and may be less affected. The teachers come from one family whose size is collinear with length, which only the same-teacher corpora break; the length intervention is prompted, single-run and at one budget; the s1K comparison also differs in domain, corpus size, system prompt and step budget; and we used s1K-1.1 rather than d1's original s1K. The seeded comparisons have three seeds per arm and none survives a multiplicity correction. ρ is an ordinal screen, equivalent to mean length on our corpora, and cannot foresee that an instruction-tuned student will resist a habit in-domain. Math-only SFT degrades code on both students (Appendix I). The per-problem attribution is descriptive: it records how the failing student failed, not what it would have scored without overrun. Answer forcing at inference, a repaired s1K, a second dLLM family and an AR student under the same cap are the experiments we would add first (Appendix O).

Conclusion. On a fixed canvas, the length and completeness of a distillation corpus are not matters of style: they decide whether the student writes an answer. Measured directly, overrun is attributable for most of the effect of trace length at a fixed teacher, for the GSM8K gain from replacing d1's corpus on a base student, and for the cross-task gap on an instruction-tuned one, and two checks on the corpus flag it before training. Where the student does not overrun, as on the instruction-tuned student in-domain, the corpus acts through answer quality instead. SFT data for dLLMs should be checked against the canvas and the sequence cap they will be used with.

Use of Large Language Models

We used large language models (LLMs) in three ways. *Writing:* an LLM assistant revised our drafts of the main text, including the abstract, introduction and results, and polished the language throughout; parts of the appendix was drafted by LLMs from our results and notes and then checked against them by us. *Research execution:* LLM coding assistants helped write and debug code for our experiments and analyses, including the scripts that measure overrun, compute the per-problem attribution, bootstrap intervals and corpus statistics from our saved generations and SFT corpora, and produce the figures. The SFT corpora themselves are reasoning traces sampled from the open-weight teacher models of Section 4; this LLM-generated training data is the object of study. *Ideation:* LLM feedback on drafts informed which analyses, controls and caveats to include. Every reported number was produced by code run on our own model outputs, and the authors reviewed the analyses, checked the final text, and take full responsibility for the content.

References

Rishabh Agarwal, Max Schwarzer, Pablo Samuel Castro, Aaron C Courville, and Marc Bellemare. Deep reinforcement learning at the edge of the statistical precipice. *Advances in neural information processing systems*, 34:29304–29320, 2021.

Pranjal Aggarwal and Sean Welleck. L1: Controlling how long a reasoning model thinks with reinforcement learning. In *Second Conference on Language Modeling*, 2025.

Beng W Ang. The lmdi approach to decomposition analysis: a practical guide. *Energy policy*, 33 (7):867–871, 2005.

Jacob Austin, Daniel D Johnson, Jonathan Ho, Daniel Tarlow, and Rianne Van Den Berg. Structured denoising diffusion models in discrete state-spaces. *Advances in neural information processing systems*, 34:17981–17993, 2021.

Xavier Bouthillier, Pierre Delaunay, Mirko Bronzi, Assya Trofimov, Brennan Nichyporuk, Justin Szeto, Nazanin Mohammadi Sepahvand, Edward Raff, Kanika Madan, Vikram Voleti, et al. Accounting for variance in machine learning benchmarks. *Proceedings of machine learning and systems*, 3:747–769, 2021.

Hongzhan Chen, Siyue Wu, Xiaojun Quan, Rui Wang, Ming Yan, and Ji Zhang. MCC-KD: Multi-CoT consistent knowledge distillation. In *Findings of the association for computational linguistics: EMNLP 2023*, pages 6805–6820, 2023.

Shirui Chen, Jiantao Jiao, Lillian J Ratliff, and Banghua Zhu. dUltra: Ultra-fast diffusion language models via reinforcement learning. *arXiv preprint arXiv:2512.21446*, 2025a.

Xingyu Chen, Jiahao Xu, Tian Liang, Zhiwei He, Jianhui Pang, Dian Yu, Linfeng Song, Qiuzhi Liu, Mengfei Zhou, Zhuosheng Zhang, Rui Wang, Zhaopeng Tu, Haitao Mi, and Dong Yu. Do NOT think that much for 2+3=? on the overthinking of long reasoning models. In *Forty-second International Conference on Machine Learning*, 2025b. URL https://openreview.net/forum?id=MSbU3L7V00.

Karl Cobbe, Vineet Kosaraju, Mohammad Bavarian, Mark Chen, Heewoo Jun, Lukasz Kaiser, Matthias Plappert, Jerry Tworek, Jacob Hilton, Reiichiro Nakano, et al. Training verifiers to solve math word problems. *arXiv preprint arXiv:2110.14168*, 2021.

Jesse Dodge, Gabriel Ilharco, Roy Schwartz, Ali Farhadi, Hannaneh Hajishirzi, and Noah Smith. Fine-tuning pretrained language models: Weight initializations, data orders, and early stopping. *arXiv preprint arXiv:2002.06305*, 2020.

Kaituo Feng, Changsheng Li, Xiaolu Zhang, Jun Zhou, Ye Yuan, and Guoren Wang. Keypoint-based progressive chain-of-thought distillation for LLMs. *arXiv preprint arXiv:2405.16064*, 2024.

Yao Fu, Hao Peng, Litu Ou, Ashish Sabharwal, and Tushar Khot. Specializing smaller language models towards multi-step reasoning. In *International Conference on Machine Learning*, pages 10421–10430. PMLR, 2023.

Kanishk Gandhi, Denise Lee, Gabriel Grand, Muxin Liu, Winson Cheng, Archit Sharma, and Noah D Goodman. Stream of search (SoS): Learning to search in language. *arXiv preprint arXiv:2404.03683*, 2024.

Shansan Gong, Ruixiang Zhang, Huangjie Zheng, Jiatao Gu, Navdeep Jaitly, Lingpeng Kong, and Yizhe Zhang. DiffuCoder: Understanding and improving masked diffusion models for code generation. In *International Conference on Learning Representations*, volume 2026, pages 15723–15750, 2026.

Etash Guha, Ryan Marten, Sedrick Keh, Negin Raoof, Georgios Smyrnis, Hritik Bansal, Marianna Nezhurina, Jean Mercat, Trung Vu, Zayne Sprague, et al. OpenThoughts: Data recipes for reasoning models. In *International Conference on Learning Representations*, volume 2026, pages 108059–108130, 2026.

Daya Guo, Dejian Yang, Haowei Zhang, Junxiao Song, Peiyi Wang, Qihao Zhu, Runxin Xu, Ruoyu Zhang, Shirong Ma, Xiao Bi, et al. DeepSeek-R1 incentivizes reasoning in LLMs through reinforcement learning. *Nature*, 645(8081):633–638, 2025.

Tingxu Han, Zhenting Wang, Chunrong Fang, Shiyu Zhao, Shiqing Ma, and Zhenyu Chen. Token-budget-aware LLM reasoning. In *Findings of the Association for Computational Linguistics: ACL 2025*, pages 24842–24855, 2025.

Peter Henderson, Riashat Islam, Philip Bachman, Joelle Pineau, Doina Precup, and David Meger. Deep reinforcement learning that matters. In *Proceedings of the AAAI conference on artificial intelligence*, volume 32, 2018.

Dan Hendrycks, Collin Burns, Saurav Kadavath, Akul Arora, Steven Basart, Eric Tang, Dawn Song, and Jacob Steinhardt. Measuring mathematical problem solving with the MATH dataset. *arXiv preprint arXiv:2103.03874*, 2021.

Geoffrey Hinton, Oriol Vinyals, and Jeff Dean. Distilling the knowledge in a neural network. *arXiv preprint arXiv:1503.02531*, 2015.

Namgyu Ho, Laura Schmid, and Se-Young Yun. Large language models are reasoning teachers. In *Proceedings of the 61st annual meeting of the association for computational linguistics (volume 1: long papers)*, pages 14852–14882, 2023.

Cheng-Yu Hsieh, Chun-Liang Li, Chih-Kuan Yeh, Hootan Nakhost, Yasuhisa Fujii, Alex Ratner, Ranjay Krishna, Chen-Yu Lee, and Tomas Pfister. Distilling step-by-step! outperforming larger language models with less training data and smaller model sizes. In *Findings of the association for computational linguistics: ACL 2023*, pages 8003–8017, 2023.

Edward J Hu, yelong shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. LoRA: Low-rank adaptation of large language models. In *International Conference on Learning Representations*, 2022. URL https://openreview.net/forum?id=nZeVKeeFYf9.

Zemin Huang, Zhiyang Chen, Zijun Wang, Tiancheng Li, and Guo-Jun Qi. Reinforcing the diffusion chain of lateral thought with diffusion language models. *Advances in Neural Information Processing Systems*, 38:152677–152710, 2026.

Bumjun Kim, Dongjae Jeon, Dueun Kim, Wonje Jeung, and Albert No. Rainbow padding: Mitigating early termination in instruction-tuned diffusion LLMs. In *International Conference on Learning Representations*, volume 2026, pages 18977–19001, 2026.

Yoon Kim and Alexander M Rush. Sequence-level knowledge distillation. In *Proceedings of the 2016 conference on empirical methods in natural language processing*, pages 1317–1327, 2016.

Takeshi Kojima, Shixiang Shane Gu, Machel Reid, Yutaka Matsuo, and Yusuke Iwasawa. Large language models are zero-shot reasoners. *Advances in neural information processing systems*, 35:22199–22213, 2022.

Jinsong Li, Xiaoyi Dong, Yuhang Zang, Yuhang Cao, Jiaqi Wang, and Dahua Lin. Beyond fixed: Training-free variable-length denoising for diffusion large language models. In *International Conference on Learning Representations*, volume 2026, pages 91715–91731, 2026a.

Jinsong Li, Xiaoyi Dong, Yuhang Zang, Yuhang Cao, Jiaqi Wang, and Dahua Lin. Beyond fixed: Training-free variable-length denoising for diffusion large language models. In *International Conference on Learning Representations*, volume 2026, pages 91715–91731, 2026b.

Xiang Li, John Thickstun, Ishaan Gulrajani, Percy S Liang, and Tatsunori B Hashimoto. Diffusion-LM improves controllable text generation. *Advances in neural information processing systems*, 35:4328–4343, 2022.

Yang Li, Youssef Emad, Karthik Padthe, Jack Lanchantin, Weizhe Yuan, Thao Nguyen, Jason Weston, Shang-Wen Li, Dong Wang, Ilia Kulikov, et al. NaturalThoughts: Selecting and distilling reasoning traces for general reasoning tasks. *arXiv preprint arXiv:2507.01921*, 2025a.

Yuetai Li, Xiang Yue, Zhangchen Xu, Fengqing Jiang, Luyao Niu, Bill Yuchen Lin, Bhaskar Ramasubramanian, and Radha Poovendran. Small models struggle to learn from strong reasoners. In *Findings of the Association for Computational Linguistics: ACL 2025*, pages 25366–25394, 2025b.

Aaron Lou, Chenlin Meng, and Stefano Ermon. Discrete diffusion modeling by estimating the ratios of the data distribution. *arXiv preprint arXiv:2310.16834*, 2023.

Lucie Charlotte Magister, Jonathan Mallinson, Jakub Adamek, Eric Malmi, and Aliaksei Severyn. Teaching small language models to reason. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 1773–1781, 2023.

Gábor Melis, Chris Dyer, and Phil Blunsom. On the state of the art of evaluation in neural language models. In *International Conference on Learning Representations*, 2018. URL https://openreview.net/forum?id=ByJHuTgA-.

Niklas Muennighoff, Zitong Yang, Weijia Shi, Xiang Lisa Li, Li Fei-Fei, Hannaneh Hajishirzi, Luke Zettlemoyer, Percy Liang, Emmanuel Candès, and Tatsunori B Hashimoto. s1: Simple test-time scaling. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pages 20286–20332, 2025.

Sania Nayab, Giulio Rossolini, Marco Simoni, Andrea Saracino, Giorgio Buttazzo, Nicolamaria Manes, and Fabrizio Giacomelli. Concise thoughts: Impact of output length on llm reasoning and cost. *Information Sciences*, page 123939, 2026.

Shen Nie, Fengqi Zhu, Zebin You, Xiaolu Zhang, Jingyang Ou, Jun Hu, Jun Zhou, Yankai Lin, Ji-Rong Wen, and Chongxuan Li. Large language diffusion models. *Advances in Neural Information Processing Systems*, 38:50608–50646, 2026.

Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, et al. Training language models to follow instructions with human feedback. *Advances in neural information processing systems*, 35: 27730–27744, 2022.

Nils Reimers and Iryna Gurevych. Reporting score distributions makes a difference: Performance study of LSTM-networks for sequence tagging. In *Proceedings of the 2017 conference on empirical methods in natural language processing*, pages 338–348, 2017.

Kevin Rojas, Jiahe Lin, Kashif Rasul, Anderson Schneider, Yuriy Nevmyvaka, Molei Tao, and Wei Deng. Improving reasoning for diffusion language models via group diffusion policy optimization. In *International Conference on Learning Representations*, volume 2026, pages 1938–1973, 2026.

Vittorio Rossi, Giacomo Cirò, Davide Beltrame, Luca Gandolfi, Paul Röttger, and Dirk Hovy. Diffusion language models are natively length-aware. *arXiv preprint arXiv:2603.06123*, 2026.

Subham S Sahoo, Marianne Arriola, Yair Schiff, Aaron Gokaslan, Edgar Marroquin, Justin T Chiu, Alexander Rush, and Volodymyr Kuleshov. Simple and effective masked diffusion language models. *Advances in Neural Information Processing Systems*, 37:130136–130184, 2024.

Zhihong Shao, Peiyi Wang, Qihao Zhu, Runxin Xu, Junxiao Song, Xiao Bi, Haowei Zhang, Mingchuan Zhang, YK Li, Yang Wu, et al. DeepSeekMath: Pushing the limits of mathematical reasoning in open language models. *arXiv preprint arXiv:2402.03300*, 2024.

Jiaxin Shi, Kehang Han, Zhe Wang, Arnaud Doucet, and Michalis Titsias. Simplified and generalized masked diffusion for discrete data. In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*, 2024.

Charlie Snell, Jaehoon Lee, Kelvin Xu, and Aviral Kumar. Scaling LLM test-time compute optimally can be more effective than scaling parameters for reasoning. *arXiv preprint arXiv:2408.03314*, 2024.

Yang Sui, Yu-Neng Chuang, Guanchu Wang, Jiamu Zhang, Tianyi Zhang, Jiayi Yuan, Hongyi Liu, Andrew Wen, Shaochen Zhong, Na Zou, et al. Stop overthinking: A survey on efficient reasoning for large language models. *arXiv preprint arXiv:2503.16419*, 2025.

Xiaohang Tang, Rares Dolga, Sangwoong Yoon, and Ilija Bogunovic. wd1: Weighted policy optimization for reasoning in diffusion language models. In *International Conference on Learning Representations*, volume 2026, pages 150357–150385, 2026.

Chenyu Wang, Paria Rashidinejad, Andy DiJia Su, Song Jiang, Sid Wang, Siyan Zhao, Cai Zhou, Shannon Shen, Feiyu Chen, Tommi Jaakkola, et al. SPG: Sandwiched policy gradient for masked diffusion language models. In *International Conference on Learning Representations*, volume 2026, pages 24623–24660, 2026a.

Guanghan Wang, Gilad Turok, Yair Schiff, Marianne Arriola, and Volodymyr Kuleshov. d2: Improving reasoning in diffusion language models via trajectory likelihood estimation. In *Forty-third International Conference on Machine Learning*, 2026b.

Peifeng Wang, Zhengyang Wang, Zheng Li, Yifan Gao, Bing Yin, and Xiang Ren. SCOTT: Self-consistent chain-of-thought distillation. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 5546–5558, 2023.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Fei Xia, Ed Chi, Quoc V Le, Denny Zhou, et al. Chain-of-thought prompting elicits reasoning in large language models. *Advances in neural information processing systems*, 35:24824–24837, 2022.

Zirui Wu, Lin Zheng, Zhihui Xie, Jiacheng Ye, Jiahui Gao, Shansan Gong, Yansong Feng, Zhenguo Li, Wei Bi, Guorui Zhou, et al. DreamOn: Diffusion language models for code infilling beyond fixed-size canvas. In *International Conference on Learning Representations*, volume 2026, pages 76626–76643, 2026.

Silei Xu, Wenhao Xie, Lingxiao Zhao, and Pengcheng He. Chain of draft: Thinking faster by writing less. *arXiv preprint arXiv:2502.18600*, 2025.

Jingyi Yang, Yuxian Jiang, and Jing Shao. ρ-EOS: Training-free bidirectional variable-length control for masked diffusion LLMs. *arXiv preprint arXiv:2601.22527*, 2026.

Jiacheng Ye, Shansan Gong, Liheng Chen, Lin Zheng, Jiahui Gao, Han Shi, Chuan Wu, Xin Jiang, Zhenguo Li, Wei Bi, et al. Diffusion of thought: Chain-of-thought reasoning in diffusion language models. *Advances in Neural Information Processing Systems*, 37:105345–105374, 2024.

Jiacheng Ye, Zhihui Xie, Lin Zheng, Jiahui Gao, Zirui Wu, Xin Jiang, Zhenguo Li, and Lingpeng Kong. Dream 7b: Diffusion large language models. *arXiv preprint arXiv:2508.15487*, 2025a.

Yixin Ye, Zhen Huang, Yang Xiao, Ethan Chern, Shijie Xia, and Pengfei Liu. LIMO: Less is more for reasoning. In *Second Conference on Language Modeling*, 2025b. URL https://openreview.net/forum?id=T2TZ0RY4Zk.

Hanyang Zhao, Dawen Liang, Wenpin Tang, David Yao, and Nathan Kallus. DiFFPO: Training diffusion LLMs to reason fast and furious via reinforcement learning. *arXiv preprint arXiv:2510.02212*, 2025.

Siyan Zhao, Devaansh Gupta, Qinqing Zheng, and Aditya Grover. d1: Scaling reasoning in diffusion large language models via reinforcement learning. *Advances in Neural Information Processing Systems*, 38:56729–56762, 2026a.

Siyan Zhao, Mengchen Liu, Jing Huang, Miao Liu, Chenyu Wang, Bo Liu, Yuandong Tian, Guan Pang, Sean Bell, Aditya Grover, et al. Inpainting-guided policy optimization for diffusion large language models. In *International Conference on Learning Representations*, volume 2026, pages 6017–6048, 2026b.

Yuchen Zhu, Wei Guo, Jaemoo Choi, Petr Molodyk, Bo Yuan, Molei Tao, and Yongxin Chen. Enhancing reasoning for diffusion LLMs via distribution matching policy optimization. *arXiv preprint arXiv:2510.08233*, 2025.

Xianwei Zhuang, Zhihong Zhu, Zhichang Wang, Xuxin Cheng, and Yuexian Zou. UniCoTT: A unified framework for structural chain-of-thought distillation. In *The Thirteenth International Conference on Learning Representations*, 2025.

Appendix Contents

A Extended related work. B The length intervention in detail. C Attribution formula, common-support checks, transfer data and tests. D Corpus statistics. E Per-seed results and the checkpoint protocol. F Selection-free checks. G Budget sensitivity and denoising steps. H RL training dynamics. I General ability. J Per-checkpoint SFT results and validation-based selection. K–M Hyperparameters and evaluation. N Compute and reproducibility. O Extended limitations and the controls we would run next.

A Extended Related Work

Diffusion language models. Discrete diffusion adapts denoising to categorical data (Austin et al., 2021; Li et al., 2022) and was simplified into masked-token formulations (Lou et al., 2023; Shi et al., 2024; Sahoo et al., 2024). LLaDA (Nie et al., 2026) scaled masked diffusion to 8B parameters trained from scratch; Dream (Ye et al., 2025a) reaches the same scale by initializing from an AR model. Both decode a canvas whose length is a decoding hyperparameter, and both learn where responses end through end-of-sequence padding, the property this paper builds on.

Reasoning in dLLMs. Diffusion-of-Thought (Ye et al., 2024) first trained chain-of-thought reasoning into diffusion models. d1 (Zhao et al., 2026a) established SFT on s1K followed by diffu-GRPO, an adaptation of GRPO (Shao et al., 2024) with a one-step likelihood estimator and random prompt masking. Subsequent work refines the RL stage: DCoLT (Huang et al., 2026), d2 (Wang et al., 2026b), GDPO (Rojas et al., 2026), DiFFPO (Zhao et al., 2025), DMPO (Zhu et al., 2025), wd1 (Tang et al., 2026) and SPG (Wang et al., 2026a); DiffuCoder (Gong et al., 2026) studies code, and dUltra (Chen et al., 2025a) targets decoding speed. Some of these start RL from an SFT checkpoint as d1 does, others directly from the instruction-tuned model. IGPO (Zhao et al., 2026b) adds SFT on synthetically rewritten concise traces aligned with dLLM generation before RL; our analysis gives the mechanism behind such length alignment and the conditions under which it helps.

Variable-length generation. DAEDAL (Li et al., 2026a) expands the canvas per problem before and during decoding. SmartCrop (Rossi et al., 2026) observes that LLaDA trained with EOS padding is natively length-aware and crops the unused tail without training; this is the same learned length awareness through which, in our setting, a corpus's length becomes the student's. ρ-EOS (Yang et al., 2026) provides bidirectional variable-length control, and DreamOn (Wu et al., 2026) adds expand and delete states so the sequence length is editable during denoising. These methods operate on a trained model; Section 5.6 shows that naive canvas enlargement does not reduce the overrun of a student trained on over-budget traces, and we regard their composition with corpus selection as an open experiment.

Reasoning distillation and data curation. Distilling chain-of-thought (Wei et al., 2022; Kojima et al., 2022) into smaller students is well established for AR models (Hinton et al., 2015; Kim and Rush, 2016; Ho et al., 2023; Magister et al., 2023; Hsieh et al., 2023; Fu et al., 2023), including variants that transfer rationales or key steps (Wang et al., 2023; Zhuang et al., 2025; Feng et al., 2024; Chen et al., 2023). Curation dominates volume (Ye et al., 2025b; Muennighoff et al., 2025; Li et al., 2025a), OpenThoughts (Guha et al., 2026) finds that the best teacher is not always the strongest, and Li et al. (2025b) show that small AR students learn better from shorter chains. Our setting differs in the mechanism and in what it predicts: the failure we measure is not that the student cannot absorb a long chain, but that it reproduces the chain's length and runs out of canvas, which is visible directly as overrun and predictable from the corpus and the budget.

Concise reasoning in AR models. Work on efficient reasoning (Sui et al., 2025; Chen et al., 2025b; Han et al., 2025; Nayab et al., 2026; Xu et al., 2025; Aggarwal and Welleck, 2025) shortens chains for latency and cost, trading against accuracy. Under a binding canvas the trade is different: a chain that does not fit costs the answer outright. This also separates our motivation from test-time-compute scaling (Snell et al., 2024; Muennighoff et al., 2025), which spends more tokens to buy accuracy.

Variance and reproducibility. Score distributions rather than single scores are needed to compare models (Reimers and Gurevych, 2017; Melis et al., 2018; Dodge et al., 2020; Henderson et al., 2018; Agarwal et al., 2021; Bouthillier et al., 2021). Appendix E documents a concrete instance: selecting a checkpoint on one seed and reporting that seed inflates results by an amount that grows with the pipeline's variance.

B The Length Intervention in Detail

Construction. The 14B teacher was sampled twice on every GSM8K training problem with the same settings as the main corpora (temperature 0.6, top-p 0.95, at most 2,048 tokens) and one of two instructions appended to the problem: *terse*, "Solve as concisely as possible: give only the essential steps and the final answer within \boxed{}. Be brief and do not over-explain."; *verbose*, "Please reason step by step in detail, showing all your work, and put your final answer within \boxed{}." We kept the 5,661 problems on which both variants are correct. The verbose arm is therefore an explicitly prompted style, longer than the default-prompt 14B corpus (943 vs 609 mean tokens as trained). The default corpus sits between the two arms in $\rho(\cdot, 512)$ (11.7, 51.1, 94.2%) and its student sits between them in accuracy at $L = 512$ (72.2, 70.9, 56.4).

Checkpoint robustness. Both arms were trained with identical hyperparameters and screened on a fixed 256-problem subset of GSM8K test at $L = 512$ (Table 4). Both peak at step 4,000, where we report full-test accuracy. The terse arm leads at every checkpoint, by 13.6 to 23.8 points, so the effect does not depend on which checkpoint is reported. (The verbose arm's full-test GSM8K, 56.4, coincides with the 7B-corpus student at $L = 1024$; both equal $744/1319$ and were verified from saved generations.)

Table 4: Isolation arms on the 256-problem screening subset, GSM8K, $L = 512$ (one run per arm).

Step	1,000	2,000	3,000	4,000	5,000
14B terse	70.3	73.4	71.9	73.8	72.3
14B verbose	46.5	53.5	55.1	60.2	55.1
Gap	23.8	19.9	16.8	13.6	17.2

Mechanism. On the full test set at step 4,000: overrun 3.3% (terse) vs 25.5% (verbose); stopping without an answer 0.2% and 0.3%; conditional accuracy 74.8 vs 76.0; median answer position 215 vs 401 tokens. Per problem, the 15.8-point gap splits into overrun +14.3 (95% CI 12.2–16.4), stopping 0.0 and wrong answers +1.5 (CI −0.5 to 3.7). On the 951 problems both arms answer, accuracy is 78.4 (terse) vs 76.3 (verbose), exact McNemar $p = 0.19$. The verbose student's overruns are mostly long, coherent derivations: the median share of repeated 8-grams in their text is 0.04, only 14% are dominated by repetition, and 52% already contain the correct value. The s1K students' overruns are the opposite (76–89% dominated by repetition, 18–22% containing the correct value), and the 7B-corpus student is in between (71%, 46%).

Threats. *Correctness* and *problem coverage* are excluded by construction. *Formatting* is excluded by measurement: no response in either arm reaches the SFT cap, both have $\bar{\varphi} = 100\%$, and stopping without an answer is 0.2% and 0.3%. *Reasoning style* is not excluded by design. A change in reasoning quality would appear as wrong answers, which account for at most a few points; but a stylistic change that acts through non-termination, such as a habit of re-checking, is indistinguishable from length in this design. *Statistical resolution* is limited to one run per arm, bounded by the checkpoint robustness above. *Budget*: the arms were evaluated at $L = 512$ only, where they straddle the budget; at $L = 1024$ the terse corpus has $\rho = 0.7\%$ and the verbose 33.0%, and we would expect a smaller but same-signed effect. A length-filtered split of a single-prompt corpus, which removes the prompt intervention at the cost of confounding length with problem difficulty, would bracket the effect from the other side (Appendix O).

C Attribution, Common Supports, Transfer Data and Tests

C.1 Per-problem attribution

For students A and B evaluated on the same problems (seed s of A paired with seed s of B), classify each generation as correct, wrong, overrun or stopped (Section 3.2). The accuracy gap is $\frac{1}{n}\big(\#\{A \text{ correct}, B \text{ not}\} - \#\{B \text{ correct}, A \text{ not}\}\big)$, and we assign each discordant problem to the failing student's failure type:

$$\text{gap} = \sum_{k \in \{\text{overrun, stopped, wrong}\}} \frac{1}{n}\Big(\#\{A \text{ correct}, B\ k\} - \#\{B \text{ correct}, A\ k\}\Big). \tag{4}$$

The terms sum exactly to the gap, and 95% intervals come from 2,000 bootstrap resamples of problems. An alternative splits $\log(\text{acc}_A/\text{acc}_B)$ into the log-ratios of the three factors of Equation 2 and rescales them to points (the logarithmic-mean weighting of LMDI, Ang, 2005). That split implicitly assumes an overrun problem would otherwise have been answered at the student's average conditional accuracy; overrun problems are harder than average (on the problems the verbose student overruns, the terse student scores 60.7% against 78.4% on the common set), so it overstates the overrun term (e.g. $+16.8$ rather than $+14.3$ for the isolation pair). We report the per-problem split throughout.

Table 5: Per-problem attribution of accuracy gaps ($A - B$, points; GSM8K at $L = 1024$ unless noted) with 95% bootstrap intervals. n counts paired problems (pooled over seed pairs). *On MATH500 the budget term counts responses the canvas cut off (no end of turn), because the last-boxed parser also scores truncated text.

Comparison (A **vs** B)	n	**Gap**	**Overrun**	**Stopped**	**Wrong answers**
Same 14B teacher: terse vs verbose (L=512)	1319	+15.8 [+12.9, +18.7]	+14.3 [+12.2, +16.4]	+0.0	+1.5 [-0.5, +3.7]
Base: 32B vs 7B (seed 42)	1319	+17.6 [+14.5, +20.7]	+16.1 [+13.9, +18.3]	+0.3	+1.1 [-1.1, +3.4]
Base: 32B vs 14B (seed 42)	1319	+3.5 [+0.8, +6.2]	+3.0 [+1.7, +4.3]	+0.1	+0.4 [-2.0, +2.7]
Base: 32B vs s1K-1.1 (3 seeds)	3957	+8.8 [+7.1, +10.4]	+9.3 [+8.2, +10.3]	+0.3	-0.7 [-2.0, +0.6]
Base MATH500: 32B vs s1K-1.1*	1500	+13.7 [+11.2, +16.1]	+6.7 [+5.2, +8.3]	+0.0	+7.0 [+5.0, +9.0]
Base: 32B (seed 42) vs no SFT	1319	+7.4 [+4.4, +10.2]	+0.1 [-0.7, +0.9]	+0.1	+7.3 [+4.4, +9.9]
Instruct: 32B vs s1K-1.1	3957	-4.6 [-6.0, -3.2]	+0.9 [+0.6, +1.2]	+0.6	-6.0 [-7.4, -4.8]
Instruct Countdown: 32B vs s1K-1.1 (seed 42)	256	+17.6 [+11.3, +23.8]	+17.6 [+13.3, +22.3]	+0.0	+0.0 [-4.3, +4.3]
After RL, GSM8K: 32B vs d1	3957	-3.2 [-4.5, -1.9]	+0.6 [+0.3, +0.8]	+0.1	-3.8 [-5.1, -2.6]
After RL, Countdown: 32B vs d1	768	+11.6 [+7.9, +15.2]	+13.9 [+11.5, +16.4]	-0.3	-2.1 [-4.8, +0.5]

On LLaDA-8B-Base, unanswered generations are almost all overruns (stopping is at most 0.8% in any configuration). On the instruction-tuned student, stopping reaches 1.7% on GSM8K (3.1% on Countdown) and wrong answers dominate every in-domain comparison. On Countdown after RL the overrun term exceeds the gap because our policies make somewhat more wrong answers.

Uncertainty and robustness. The intervals above resample problems with the trained checkpoints fixed; resampling problems as clusters across seed pairs gives nearly identical intervals. Training-seed variance is larger. Per seed pair, the Base overrun term against s1K is 5.5, 10.9 and 11.4 points (seed-level 95% t-interval 1.2 to 17.3), and the Countdown-after-RL term is 14.8, 14.5 and 12.5 (10.8 to 17.1). Pairing seed s with seed s is arbitrary; averaging over all nine cross-seed pairings gives 9.4 (range 3.9–13.2) and 13.0 (range 8.2–17.2). Defining overrun by the final 64 or 128 tokens instead of 32 changes no term by more than 0.1 points. The wrong-answer term equals the common-support accuracy difference scaled by the share of problems both students answer, so the two are not independent confirmations; both condition on answering, a post-treatment event, and cannot reveal a reasoning advantage confined to problems one student overruns.

C.2 Conditional accuracy on a common support

Conditional accuracy is computed on the problems a model answers, which is post-treatment: a model that overruns on hard problems is scored on an easier subset. Wherever an argument compares q across models we therefore also score both models on the intersection of problems both answer, and for the central Base comparison additionally post-stratify by a difficulty proxy independent of both arms (how many of the three students *not* in the comparison, un-finetuned Base, 7B and 14B, solve the problem).

Table 6: Conditional accuracy on the common set of problems both students answer (GSM8K). $\pm$ is the s.d. of the per-seed difference; p is an exact McNemar test, per seed.

Comparison (A vs B)	Protocol	A	B	A − B	McNemar p
s1K vs 32B, Base, 3 seeds	marginal	76.8	75.2	$+1.6$	—
	common ($\sim$1,073)	77.3	76.4	$+0.9 \pm 1.4$	0.48, 0.17, 0.73
	+ difficulty-stratified	76.4	75.5	$+0.8 \pm 1.4$	—
terse vs verbose, Base, L=512	common (951)	78.4	76.3	$+2.1$	0.19
32B vs 7B, Base, seed 42	common (979)	76.0	74.5	$+1.5$	0.36
s1K vs 32B, Instruct, 3 seeds	common ($\sim$1,263)	83.4	77.1	$+6.3 \pm 3.1$	0.012, <0.001, <0.001
d1 vs 32B, after RL, 3 seeds	common ($\sim$1,289)	83.2	79.3	$+3.9 \pm 2.7$	0.51, <0.001, <0.001

About half of s1K's nominal conditional-accuracy margin on Base is selection, and the remainder reverses sign in one seed and is insignificant in all three. Pooling the three seed pairs, the difference is $+0.87$ points, with a 90% interval of $[-0.37, 2.17]$ over problems and $[-1.45, 3.21]$ over seeds; we therefore claim no detectable advantage, not equivalence. On Instruct the common-support comparison confirms the main text: s1K's student is significantly more accurate on every seed. We do not claim that the budget-fit corpus reasons better than s1K, and the argument of Section 5.3 does not need it: an in-domain corpus six times larger confers no detectable conditional-accuracy advantage, while the overrun difference (13.2 points) is an order of magnitude larger than anything resolvable in the quality term. On Countdown after RL, conditional accuracy is slightly higher for d1's policies (25.6 vs 22.2), which is why the per-problem wrong-answer term is negative there; the gap is carried by overrun, whose difference (59.4% vs 0.5%) is resolved on every seed.

C.3 Transfer from corpus to student

Table 7: Every (student, budget) panel of Figure 3a–b: corpus ρ against the student's overrun on GSM8K and its accuracy. Reference checkpoints; seeded entries are three-seed means. Isolation corpora were evaluated at $L = 512$ only.

Panel	Corpus	ρ	Overrun	Acc.	Panel	Corpus	ρ	Overrun	Acc.
Base, 512	14B terse	11.7	3.3	72.2	Instruct, 512	32B	16.2	0.1	81.9
	32B	16.2	2.4	73.4		14B	51.1	0.0	80.7
	14B	51.1	6.7	70.9		7B	95.9	3.4	76.5
	14B verbose	94.2	25.5	56.4		s1K	100	1.1	81.9
	7B	95.9	17.4	61.9	Instruct, 1024	32B	4.9	0.1	75.8
	s1K	100	9.0	69.0		14B	16.7	0.8	78.2
Base, 1024	32B	4.9	2.6	73.1		7B	44.9	11.5	68.1
	14B	16.7	7.7	70.5		s1K	99.9	2.2	80.4
	7B	44.9	23.7	56.4					
	s1K	99.9	15.8	64.2					

Excluding s1K, whose training signal the SFT cap truncates, the four panels contain 19 corpus pairs, of which 16 are concordant (higher ρ, higher student overrun). The discordant pairs are terse vs 32B ($\Delta\rho = 4.5$), verbose vs 7B ($\Delta\rho = 1.7$), and 14B vs 32B on Instruct at $L = 512$, where both students overrun on at most 0.1%. Including s1K, 25 of 33 pairs are concordant. Mean corpus length produces exactly the same counts, so on these corpora ρ carries no ordinal information beyond mean length; we prefer it because it is stated relative to the budget. The pairs are not independent: most repeat the ordering of the three teacher corpora across two students and two budgets, and the one panel with non-teacher corpora (Base, $L = 512$) has Kendall $\tau = 0.6$ over five corpora (two-sided $p = 0.23$). Student overrun is a student-specific fraction of corpus ρ (0.13–0.53 on Base, 0–0.26 on Instruct), and it is not comparable across budgets because reasoning length responds to the canvas (Appendix G). ρ saturates when most of a candidate set exceeds L (the verbose and 7B corpora at $L = 512$ are both above 94%); length percentiles should then be compared instead.

C.4 TWO-SAMPLE TESTS

Table 8: Seeded two-sample comparisons (budget-fit arm minus comparison arm), from the per-seed values of Table 10 (differences of unrounded means). Welch tests are two-sided with Welch–Satterthwaite df; "Separated" means the seed sets do not overlap, which at $n = 3$ attains the exact one-sided rank-sum floor $p = 0.05$.

Comparison	Bench.	Δ	Welch t	df	Welch p	Cohen's d	Separated
32B vs s1K, Base	GSM8K	$+8.8$	2.93	3.18	0.057	2.39	yes
32B vs s1K, Base	MATH500	$+13.7$	4.63	2.47	0.029	3.78	yes
32B vs s1K, Instruct	GSM8K	-4.6	-2.58	2.04	0.121	-2.11	yes (s1K)
32B vs s1K, Instruct	MATH500	$+0.3$	0.24	2.85	0.828	0.19	no
32B+RL vs d1	GSM8K	-3.2	-1.83	2.01	0.208	-1.50	no
32B+RL vs d1	MATH500	$+0.3$	0.15	2.43	0.895	0.12	no
32B+RL vs d1	Countdown	$+11.6$	5.05	2.01	0.036	4.13	yes

With seven tests, a Holm correction requires $p < 0.0071$ for the smallest; none qualifies. One-sample tests of the budget-fit arm against single-run references: Base GSM8K vs un-finetuned 66.6, $+6.5$, $t = 4.34$, $p = 0.049$; Base MATH500 vs 8.8, $+18.7$, $t = 19.3$, $p = 0.003$; Instruct GSM8K vs 78.9, -3.1, $t = -1.78$, $p = 0.22$. s1K SFT on Base against the un-finetuned model: -2.4, $p = 0.46$. All tests use unrounded per-seed accuracies. Test-set sampling error is separate from seed variance: at these accuracies the binomial standard error is about 1.2 points on GSM8K, 2.1 on MATH500 and 2.6 on Countdown.

D CORPUS STATISTICS

All statistics in the main text are computed on the responses exactly as they enter SFT: the teacher's thinking inside `<reasoning>` followed by `<answer>\boxed{·}</answer>`, tokenized with the LLaDA tokenizer. The conversion drops the summary that R1-distilled teachers write after their thinking, which is why these responses are shorter than the raw teacher outputs (Table 9); orderings are the same on either basis. Retained teacher responses carry an answer by construction, so $\tilde{\varphi} = 100\%$ for every teacher corpus; the raw teacher outputs, before filtering and conversion, are not always well formed (last row).

Table 9: Corpus statistics on both bases (LLaDA tokenizer).

	7B	14B	32B	s1K
Generated / retained	7,473 / 6,363	7,473 / 5,841	7,473 / 6,110	1,000 / 1,000
As trained: mean / median	1,042 / 970	609 / 530	335 / 230	10,220 / 9,830
As trained: ρ at 256/512/1024	99.9/95.9/44.9	68.8/51.1/16.7	39.4/16.2/4.9	100/100/99.9
As trained: over cap / $\tilde{\varphi}$	0 / 100	0 / 100	0 / 100	91.3 / 9.0
Raw: mean tokens (correct)	1,235	838	576	—
Raw: ρ at 1024	69.7	27.9	9.2	—
Raw: well-formed `boxed`	88.7	96.4	99.0	93.9
Teacher accuracy, strict match (filter)	85.2	78.2	81.8	—
Teacher accuracy, normalized match	86.1	91.9	94.6	—

Teacher accuracy and the correctness filter. Traces were retained if the extracted answer matched the reference under a strict string-to-number comparison. That comparison rejects correct answers written with currency symbols or thousands separators (e.g. `\$990.00`, `1,\!920`), which the 14B and 32B teachers use often. Under a normalized numeric match, teacher accuracy is 86.1% (7B), 91.9% (14B) and 94.6% (32B), increasing with size; the strict rates were 85.2%, 78.2% and 81.8%. The filter therefore has false negatives, not false positives: every retained trace is correct, but roughly 1,000 correct traces were discarded from each of the 14B and 32B corpora, many of them with monetary answers. We report the corpora as trained; a normalized filter would enlarge the 14B and 32B corpora and shift their topic mix slightly.

s1K under the cap. We train on s1K-1.1, the release with DeepSeek-R1 traces; d1's released code loads the original s1K (Gemini traces). With d1's generic system prompt and chat template, 91.3% of s1K-1.1 sequences exceed 4,096 tokens. After truncation, 11.3% contain a `boxed` answer and 9.0% the closed `</answer>` block; before truncation, 93.9% contain a `boxed` answer. Truncated sequences also carry no end-of-sequence padding, so the student is shown reasoning that neither answers nor ends. The original s1K is about half as long (mean 5,440, median 5,679 tokens; $\rho(\cdot, 1024) = 99.5\%$), but under the same preprocessing 88.0% of its sequences are truncated and only 6.8% keep a boxed answer (12.1% keep the closed answer block), so the defect applies to d1's corpus as released.

E Per-Seed Results and the Checkpoint Protocol

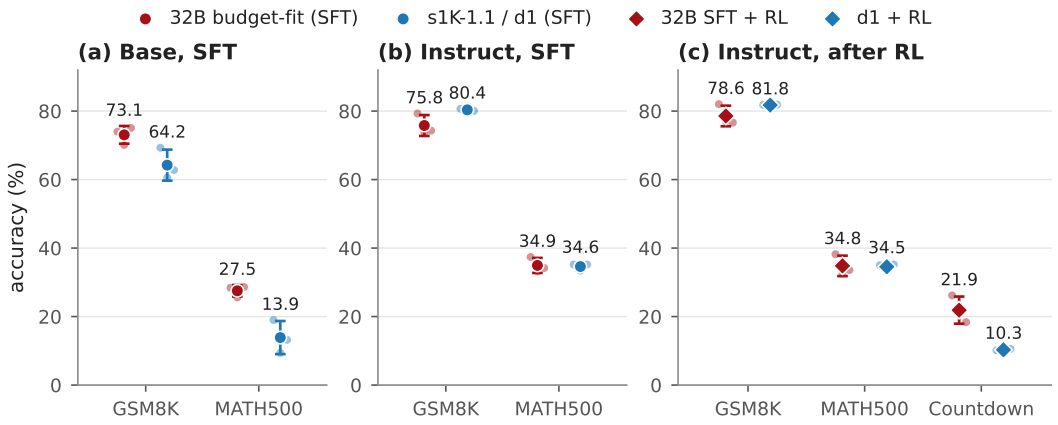


Figure 4: Per-seed accuracy (small dots) with mean ± s.d. (large marker, labelled with the mean) for every seeded configuration at $L = 1024$; red is the budget-fit (32B) pipeline, blue the s1K-1.1/d1 pipeline.

Table 10: Per-seed accuracy at $L = 1024$ (means and s.d. from unrounded values). "Ref. step" is the step at which all seeds are evaluated, chosen on seed 42's screening curve at $L = 512$. "42 − mean" is seed 42 minus the three-seed mean.

Configuration	Benchmark	Ref. step	Seed 42	Seed 1	Seed 2	Mean ± s.d.	42 − mean
SFT, LLaDA-8B-Base							
32B (budget-fit)	GSM8K	3500	74.0	70.1	75.1	73.1 ± 2.6	+0.9
32B (budget-fit)	MATH500	3500	28.4	25.6	28.6	27.5 ± 1.7	+0.9
s1K	GSM8K	500	69.3	60.6	62.8	64.2 ± 4.5	+5.1
s1K	MATH500	500	19.0	9.4	13.2	13.9 ± 4.8	+5.1
SFT, LLaDA-8B-Instruct							
32B (budget-fit)	GSM8K	1000	79.3	73.8	74.3	75.8 ± 3.0	+3.5
32B (budget-fit)	MATH500	1000	37.4	33.2	34.2	34.9 ± 2.2	+2.5
s1K	GSM8K	2460	80.7	80.4	80.1	80.4 ± 0.3	+0.3
s1K	MATH500	2460	35.2	33.4	35.2	34.6 ± 1.0	+0.6
SFT + diffu-GRPO, LLaDA-8B-Instruct							
32B + RL	GSM8K	7700	82.0	77.0	76.6	78.6 ± 3.0	+3.5
32B + RL	MATH500	7700	38.2	32.6	33.6	34.8 ± 3.0	+3.4
32B + RL	Countdown	7700	26.2	21.1	18.4	21.9 ± 4.0	+4.3
d1 (s1K) + RL	GSM8K	7700	81.8	81.6	81.9	81.8 ± 0.2	+0.0
d1 (s1K) + RL	MATH500	7700	35.0	33.4	35.2	34.5 ± 1.0	+0.5
d1 (s1K) + RL	Countdown	7700	10.2	10.2	10.5	10.3 ± 0.2	−0.1

What the seed-42 column measures. Seed 42 is strictly the highest of the three in 8 of the 14 rows and tied for highest in one, and exceeds the three-seed mean by +2.2 points on average. This is what the protocol predicts, not a property of the seed. The reference step is the peak of seed

42's own screening curve, so seed 42 is evaluated at a step selected for it and seeds 1 and 2 are not (a winner's-curse effect). The pattern largely matches that reading: the excess is large where seed 42's curve has a sharp peak at the reference step (the budget-fit corpus on Instruct at step 1,000, a spike between 75.9 and 79.8, +3.5, inherited by its RL descendants, +3.4 to +4.3) and near zero where no step was selected (d1's RL runs, final step from a final-step SFT initialization). The largest excess is an exception: for s1K on Base, step 500 is every seed's own best at $L = 512$, where seed 2 (71.3) beats seed 42 (69.9), so seed 42's +5.1 at $L = 1024$ is not a selection effect; it reflects the large variability of s1K students at the larger canvas (overrun 10.2%, 19.3% and 17.9% by seed). The rows are also not independent (benchmarks share checkpoints, RL rows inherit SFT seeds), so we attach no significance test. The practical content is the size of the effect: a single-seed result reported at a checkpoint selected on that seed can overstate the seed mean by several points when the pipeline is unstable, and by almost nothing when it is stable, as d1's is.

Consequences for the comparisons. Both arms of each comparison in Table 3 are evaluated under the same protocol. For s1K on Base the reference step is also each seed's own best (Appendix F), so the baseline is not handicapped. For the budget-fit corpus on Instruct the step is a seed-42 spike, which may depress seed 1 (Appendix F) and hence our SFT and RL means and variances on that student. We have not run per-seed held-out selection for every configuration (Appendix O).

F Selection-Free Checks

Table 2 (Section 5.2) compares three checkpoint protocols for every corpus at $L = 512$.

Per-seed selection for s1K on Base. Full-test GSM8K grids at $L = 512$ for all three s1K seeds on Base (seed 2: steps 500–2,000) peak at step 500 (seed 42: 69.9; seed 1: 65.9; seed 2: 71.3), the reference step; held-out selection on one half of the test set, scored on the other, also selects step 500 for all three. The baseline in Table 3 is therefore at its best saved checkpoint on every seed. Because that is the first saved checkpoint, earlier steps were not examined; we have not run the matching per-seed grid for the budget-fit corpus. On a 256-problem screen its seeds 1 and 2 peak at step 2,000; seed 1 evaluated there at $L = 1024$ scores 70.7, against 70.1 at the reference step.

The budget-fit corpus on Instruct. At $L = 512$ and seed 42 the budget-fit and s1K corpora are tied at their reference steps (81.9 each), and s1K leads by 1.7 at the final steps (81.9 vs 80.2). The larger deficit of Table 3 (-4.6 at $L = 1024$) comes mostly from seeds 1 and 2 evaluated at seed 42's step. On a 256-problem screen, seed 1 scores 76.2 at that step and 80.9 at the final step; seed 2 scores 79.3 and 78.9, so the step handicaps seed 1 but not seed 2. We report the reference-step numbers in the main text because they are the protocol fixed before these checks, and because the conclusion, no in-domain gain on Instruct, is the same under both.

G Budget Sensitivity and Denoising Steps

Table 11: The same checkpoints at two budgets (GSM8K; seed 42 except s1K on Base, three seeds). Both budgets use two tokens per denoising step and 16 steps per 32-token block. Length is the mean number of tokens up to the first end of turn; answer position is the median token index of the answer among answered generations.

		Accuracy			Overrun (%)		Length		Answer pos.	
Student	**Corpus**	512	1024	Δ	512	1024	512	1024	512	1024
Base	7B	61.9	56.4	−5.5	17.4	23.7	371	594	240	256
	14B	70.9	70.5	−0.4	6.7	7.7	305	351	262	267
	32B (budget-fit)	73.4	74.0	+0.6	2.4	2.0	265	278	234	238
	s1K-1.1 (3 seeds)	69.0	64.2	−4.8	9.0	15.8	269	379	212	212
Instruct	7B	76.5	68.1	−8.4	3.4	11.5	494	990	486	1010
	14B	80.7	78.2	−2.5	0.0	0.8	335	515	301	330
	32B (budget-fit)	81.9	79.3	−2.6	0.1	0.2	279	276	252	251
	s1K-1.1	81.9	80.7	−1.2	1.1	2.1	331	334	299	289

Base students often close their answer and then continue with invented chat turns until the canvas is full, so we measure length up to the first end of turn. Two patterns follow. On Base, the median answer position barely moves with the canvas, but the tail lengthens and overrun rises for the students of over-budget corpora (7B; s1K, from $9.0 \pm 2.2\%$ to $15.8 \pm 4.9\%$ over three seeds), while the budget-fit student is unaffected. On Instruct, the 7B-trained student places its answer near the end of whatever canvas it is given, doubling its length and tripling its overrun; the other Instruct students barely change. The accuracy columns must be read with a caveat: the $L = 512$ entries are at steps selected on $L = 512$ curves, so re-evaluating at $L = 1024$ regresses toward the curve mean, most for noisy curves (7B on Base ranges from 41.0 to 61.9 across checkpoints) and spikes (32B on Instruct at step 1,000). Overrun, length and answer position were not selected on. Doubling the canvas also doubles the number of denoising steps at constant tokens per step, so compute per token is unchanged.

Denoising steps. Fixing the budget-fit Base checkpoint and $L = 512$ and varying only the number of denoising steps (i.e. tokens committed per step), GSM8K accuracy is 0.1, 13.4, 60.0, 73.4 and 78.8 at 32, 64, 128, 256 and 512 steps (single runs). Reducing steps is far from free for reasoning, and the common default ($L/2$) is a compute compromise rather than an accuracy optimum. This varies tokens per step, which the budget comparison above holds fixed.

H RL Training Dynamics

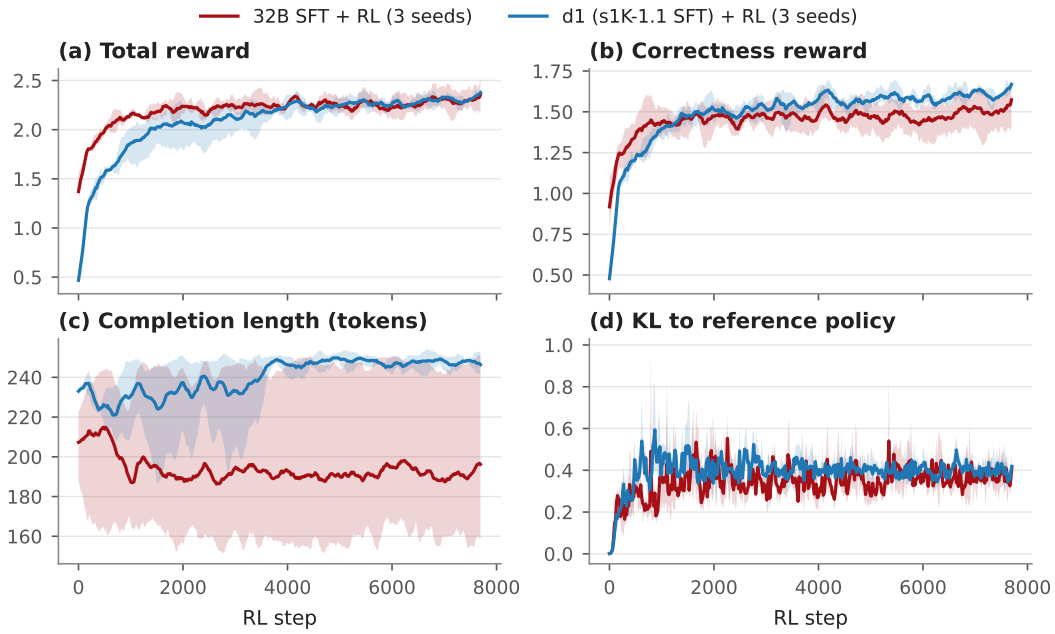


Figure 5: diffu-GRPO on GSM8K at d1's training budget (256 tokens). Lines are means across seeds; bands span min–max ($n = 3$ per arm). The budget-fit initialization starts with higher reward and both converge to similar reward. Completion length separates the pipelines on two of three seeds.

Table 12: diffu-GRPO training statistics, mean over the last 500 steps of each run.

Quantity	32B per-seed	mean	d1 per-seed	mean
Completion length (tok)	248.0/168.7/160.3	192.3 ± 48.4	244.9/246.9/250.8	247.5 ± 3.0
Total reward	2.3/2.2/2.4	2.3 ± 0.1	2.3/2.3/2.3	2.3 ± 0.0
Correctness reward	1.6/1.4/1.5	1.5 ± 0.1	1.6/1.6/1.6	1.6 ± 0.0

RL at a 256-token budget quickly raises the s1K-initialized policy's reward to the level of ours: on GSM8K, at the RL budget, whatever length habit s1K instilled is either absent or repaired, consistent

with its low GSM8K overrun after RL (1.3% at $L = 1024$). It is not repaired where GSM8K does not elicit it: the same policies overrun on 59% of Countdown problems. Among our three RL seeds the one whose completion length converged to d1's (seed 42) is best on all three benchmarks; with three seeds we do not read a trend into this. RL at the evaluation budget, or on a task that elicits long search, is the obvious test of whether RL can repair the habit where it is expressed.

I General Ability

Table 13: Countdown and HumanEval at $L = 1024$ (single runs, seed 42 at the reference step). The unfinetuned Base HumanEval value comes from an earlier evaluation run with the same code; the Base Countdown value is rescored with the parser used for every other row.

	LLaDA-8B-Base			LLaDA-8B-Instruct		
Corpus	Countdown	overrun	HumanEval	Countdown	overrun	HumanEval
None	8.6	34.8	26.8	13.7	8.6	34.1
s1K-1.1	1.2	90.2	7.3	10.5	61.7	31.7
7B	2.0	94.1	9.8	22.3	16.8	11.6
14B	1.6	96.1	4.9	27.0	0.8	23.2
32B (budget-fit)	4.7	73.4	17.1	28.1	0.4	28.7

Math-only SFT on the base student collapses Countdown for every corpus, and the collapse is overrun: every fine-tuned Base student reaches the end of the canvas without an answer on 73–96% of Countdown problems. HumanEval also degrades on both students for every corpus. On the instruction-tuned student Countdown moves the other way for the teacher corpora, which raise its conditional accuracy from 15.2% to 27–28% without adding overrun; s1K raises conditional accuracy as much (27.8%) but overruns on 61.7% of problems. After RL, HumanEval is 34.1 (budget-fit) and 34.8 (d1), single runs. We report these as a limitation of single-domain SFT, not in the budget-fit corpus's favour: no corpus preserves the base model's general ability, and a practitioner who needs breadth should mix domains.

J Per-Checkpoint SFT Results and Validation-Based Selection

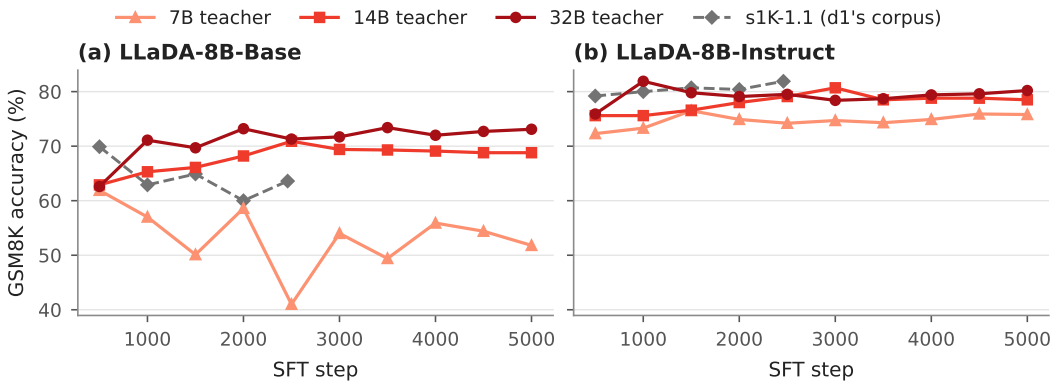


Figure 6: SFT dynamics: GSM8K accuracy at every saved checkpoint (screening at $L = 512$, full test, seed 42). Teacher colour encodes conciseness (light → dark = verbose → concise); s1K-1.1 in gray.

Table 14: Per-checkpoint GSM8K accuracy, LLaDA-8B-Base (L=512, full test, seed 42). Bold: reference step.

Corpus	500	1000	1500	2000	2460	2500	3000	3500	4000	4500	5000
s1K	**69.9**	62.9	64.9	60.0	63.6	—	—	—	—	—	—
7B	**61.9**	57.0	50.1	58.6	—	41.0	54.0	49.4	55.9	54.4	51.8
14B	62.9	65.3	66.1	68.2	—	**70.9**	69.4	69.3	69.1	68.8	68.8
32B	62.6	71.1	69.7	73.2	—	71.3	71.7	**73.4**	72.0	72.7	73.1

Table 15: Per-checkpoint GSM8K accuracy, LLaDA-8B-Instruct (L=512, full test, seed 42). Bold: reference step.

Corpus	500	1000	1500	2000	2460	2500	3000	3500	4000	4500	5000
s1K	79.2	80.0	80.7	80.4	**81.9**	—	—	—	—	—	—
7B	72.3	73.3	**76.5**	74.9	—	74.2	74.7	74.3	74.9	75.9	75.8
14B	75.6	75.6	76.6	78.0	—	79.1	**80.7**	78.5	78.8	78.8	78.5
32B	75.9	**81.9**	79.8	79.1	—	79.5	78.4	78.7	79.4	79.6	80.2

Concise corpora peak later on the base student (32B at step 3,500, s1K at 500), consistent with shorter responses supplying fewer content tokens per step.

Validation-based selection. Splitting GSM8K test into interleaved validation and test halves, selecting the step on the validation half and scoring the other half ($L = 512$, seed 42) changes accuracy by 0.65 points on average (3.7 at most, Instruct 7B) and selects the reference step in 6 of 8 runs (Table 16). This bounds test-set selection for the seed-42 runs; it does not address the asymmetry between seeds discussed in Appendix E.

Table 16: Validation-based against best-test selection (GSM8K, L=512, seed 42). "Held-out": test-half accuracy at the validation-selected step; "Oracle": best test-half checkpoint.

Run	Val-sel step	Held-out	Oracle	Gap	Matches ref.?
Base s1K	500	68.4	68.4	0.0	yes
Base 7B	500	60.4	60.4	0.0	yes
Base 14B	2500	69.7	69.7	0.0	yes
Base 32B	5000	71.9	73.4	1.5	no
Inst. s1K	2460	80.7	80.7	0.0	yes
Inst. 7B	4000	71.3	75.0	3.7	no
Inst. 14B	3000	78.5	78.5	0.0	yes
Inst. 32B	1000	81.0	81.0	0.0	yes
Mean	—	—	—	**0.65**	**6/8**

K SFT HYPERPARAMETERS

We use d1's SFT configuration with one departure: LoRA adapts all seven attention and MLP projections (d1's released code adapts query, key and value only), identically for every corpus. Sequences are padded to the 4,096-token cap with end-of-sequence tokens, which are supervised (Equation 1); prompt positions are never masked. The maximum step count follows d1 for s1K (2,460) and is 5,000 for the larger teacher corpora, with linear decay to zero in both cases, so the learning-rate schedules differ in length. s1K is trained with d1's generic system prompt, as in d1's code, and the teacher corpora with d1's math prompt, which is also the evaluation prompt; both request the same `<reasoning>`/`<answer>` format. We use s1K-1.1 (DeepSeek-R1 traces); d1's code loads the original s1K (Appendix D).

Table 17: SFT hyperparameters.

Parameter	Value
LoRA rank / alpha / dropout	128 / 256 / 0.05
LoRA targets	q/k/v/o/gate/up/down_proj
Learning rate	1×10^{-5} (linear decay, no warmup)
Effective batch size	8 (1 per device $\times$ 4 accumulation $\times$ 2 GPUs)
Max sequence length	4096 (pad with EOS and truncate)
Optimizer	AdamW (β_1=0.9, β_2=0.99, wd=0.1)
Gradient clipping	max_norm = 1.0
Precision	bf16
Max steps	2,460 (s1K) / 5,000 (trace corpora)
Checkpoints	every 500 steps
Hardware	2$\times$ NVIDIA RTX A6000 (48 GB), DDP
Seeds	42, 1, 2

L RL HYPERPARAMETERS

We use d1's diffu-GRPO algorithm, reward and hyperparameters without modification; the SFT checkpoint that initializes the policy (and serves as the frozen reference) is the only difference.

Table 18: diffu-GRPO hyperparameters (d1's configuration).

Parameter	Value
LoRA rank / alpha	128 / 64
Learning rate	3×10^{-6}
Optimizer	AdamW (β_1=0.9, β_2=0.99, wd=0.1)
Gradient clipping	max_norm = 0.2
Effective batch size	96
Generation length / denoising steps	256 / 128 (block length 32)
Prompt masking p / inner updates μ	0.15 / 12
Training steps / data	7,700 / GSM8K train
Reward	d1's composite GSM8K reward (format terms + correctness)
Reference policy	the initializing SFT checkpoint (frozen)
Seeds	42, 1, 2 (each from the matching SFT seed)

M EVALUATION DETAILS

Prompts. Every model is evaluated with d1's per-benchmark prompt, regardless of its SFT corpus; each prompt asks for reasoning in `<reasoning>` and the answer in `<answer>\boxed{}</answer>`.

Answer extraction. GSM8K: first well-formed `boxed` numeric match, else the `<answer>` block; float comparison. MATH500: last `boxed` match, else the `<answer>` block, with symbolic equivalence. Countdown: last `boxed` expression, which must use each given number exactly once and evaluate to the target. These are d1's parsers.

Overrun and stopping. A generation *answers* if d1's parser extracts an answer (for Countdown, if an answer expression is present or the parser finds a valid expression in the text). An unanswered generation *overruns* if its text reaches the final 32-token block of the canvas, and *stops* otherwise. On MATH500, whose last-boxed parser also extracts intermediate expressions from truncated text, we treat a response as cut off by the budget if it contains no end of turn. Lengths are measured up to the first end of turn, because Base students often continue with invented chat turns after answering. With these definitions, every incorrect generation is wrong, overrun or stopped, which is what the per-problem attribution uses.

Generation. Denoising steps $= L/2$, block length 32, greedy decoding with low-confidence remasking. Decoding is deterministic, so reported variance is training-seed variance. Test sets: GSM8K 1,319, MATH500 500, Countdown 256 (d1's split), HumanEval 164.

N Compute and Reproducibility

Table 19: Approximate compute on 2× A6000 (48 GB); per-run figures are wall-clock hours.

Experiment	Per run	Runs
Trace generation (7B, 14B, 32B; 14B terse and verbose)	1–3 h	5
dLLM SFT (8 seed-42 runs, 8 seed replicates, 2 isolation runs)	6–16 h	18
diffu-GRPO (seed 42 and replicates, both pipelines)	∼138 h	6
Evaluation at $L = 1024$ (per benchmark)	10–25 h	∼70
Evaluation at $L = 512$ (screening and grids)	∼6–8 h	∼100

Evaluation dominates: every GSM8K example at $L = 1024$ costs 512 sequential denoising steps over the full canvas, which is the cost structure that keeps dLLM budgets small. By contrast, computing ρ and $\tilde{\varphi}$ for a candidate corpus is one tokenization pass, while training a student to rank it would cost an SFT run and an evaluation. Seed replication roughly tripled total compute (∼4,000 GPU-hours).

We build on the d1 codebase (commit `6f5abf5`) with custom dataset loading and LoRA integration. Software: Python 3.10, PyTorch 2.2.0, Transformers 4.57.6, PEFT 0.15.1, CUDA 11.8; the RL environment pins Transformers 4.49.0 and the trl commit required by d1. All models are public. We will release code, trace corpora as trained, checkpoints, every per-example generation, and the scripts that compute every statistic and figure in this paper from those generations.

O Extended Limitations and the Controls We Would Run Next

- **Answer forcing at inference.** Inpainting `</reasoning><answer>` into the final block of overrun generations would measure directly how much accuracy overrun costs, and whether an inference-time fix recovers it. It requires only evaluation.
- **Repairing s1K.** Truncating the middle of each s1K reasoning trace while keeping its answer and end of sequence would set $\tilde{\varphi}$ to 100% at unchanged ρ and separate the two diagnostics; raising the cap is not feasible on our hardware (LLaDA-8B already fills 48 GB at 4,096 tokens, and 62% of s1K exceeds 8,192). A 1,000-example subset of the budget-fit corpus on s1K's step budget and prompt would separate length from size, domain and prompt. The original s1K should replace s1K-1.1.
- **Length without prompting.** A median-length split of one single-prompt corpus, and a seeded isolation experiment at both budgets.
- **A second dLLM family and an AR control.** Dream-7B (Ye et al., 2025a) decodes a fixed canvas and is initialized from an AR model; an 8B AR student under the same cap would show how much of the effect is the cap rather than the architecture. Block-diffusion and variable-length models relax the fixed canvas and would test the premise directly.
- **Checkpoints and budgets.** Per-seed held-out selection for every seeded configuration, s1K checkpoints before step 500, and seeded results at $L = 256$ and 512.
- **The literature's protocol.** Task-specific Countdown RL as in d1, RL at the evaluation budget, and a newer RL algorithm such as wd1 (Tang et al., 2026).
- **The correctness filter.** Rebuilding the teacher corpora with a normalized answer match (Appendix D).