WHEN CLIPPING REVERSES CORRECTION: FAILURE DYNAMICS OF POINTWISE FORWARD-KL ON-POLICY SELF-DISTILLATION

Anonymous authors
Paper under double-blind review

ABSTRACT

On-policy self-distillation (OPSD) trains a student on its own generated responses using feedback from the same model conditioned on privileged information. On mathematical reasoning, the original OPSD study finds that stylistic tokens can dominate the training signal over math-related tokens, and that pointwise clipping of the forward KL objective stabilizes training. Pointwise clipping caps each vocabulary-wise forward KL term at a fixed threshold before summing over the vocabulary. Follow-up studies have adopted this clipping, but its effect on training has not been directly examined. In matched training runs differing only in whether clipping is applied, we observe that clipped runs produce substantially more repetitions that persist to the end of the response than their unclipped counterparts. We trace this failure to the clipped objective. We prove that the clipped objective can fail to correct the student toward the teacher and can instead push clipped and unclipped token probabilities away from its teacher. Our training runs agree with this analysis: inside repetitions, the clipped student places less probability than its teacher on leaving the repetition, and more on continuing it, whereas the unclipped runs stay close to their teachers.

1 INTRODUCTION

Outcome rewards give a reasoning model one scalar for thousands of tokens. On-policy distillation instead provides a per-token target from a teacher evaluated on the student's own responses (Agarwal et al., 2024; Lu & Lab, 2025). Self-distillation uses the same network as both teacher and student, replacing a stronger teacher model with access to additional privilege information that is withheld from the student (Zhao et al., 2026a; Hübotter et al., 2026; Shenfeld et al., 2026). Self-distillation provides dense supervision, but how the student learns from it depends on the training objective.

On-Policy Self-Distillation (OPSD) trains the student with a pointwise-clipped forward KL objective (Zhao et al., 2026a). At each response position, this objective clips each vocabulary-wise forward KL term at a fixed threshold before summing over the vocabulary. The authors introduced this pointwise clipping to keep stylistic tokens from dominating the math-related tokens in the training signal, and many subsequent methods retain it (Chen et al., 2026b; Li et al., 2026; Liu et al., 2026; Shrestha & Tessier, 2026).

Studies using this recipe report response length inflation (Yang et al., 2026), responses that fail to terminate within the generation budget (Chen et al., 2026b; Ichihara et al., 2026), redundant reasoning chain (Gu et al., 2026), and accuracy that declines after early gains (Zhao et al., 2026a; Chen et al., 2026b; Ichihara et al., 2026; Pan et al., 2026; Zhang et al., 2026). Proposed explanations include a frozen teacher that cannot adapt to the student (Chen et al., 2026b), shifts in the teacher's style or reasoning pattern induced by the privileged context (Ichihara et al., 2026; Pan et al., 2026; Yang et al., 2026; Gu et al., 2026). No study reports that pointwise clipped objective contribute to the text degeneration, leaving this possibility open for further investigation.

Prior work offers a partial mathematical description of this clipped objective: a clipped term becomes constant and loses its direct gradient (Chen et al., 2026b), and the clipped sum is no longer a divergence (Chen et al., 2026b; Feng et al., 2026). These observations do not establish which student

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.

distributions the clipped objective favors, in particular whether minimizing it still brings a clipped token's probability closer to the teacher's.

We trained the pointwise-clipped forward KL OPSD recipe with a frozen reference-conditioned teacher and encountered a failure mode of text degeneration: generations ended in periodic tails that never terminated. When we removed only the clipping, nearly all of these loops disappeared. The teacher and its privileged context are identical in the two runs, so neither accounts for the difference. We therefore ask whether, and through what mechanism, this clipped objective contributed to this failure.

Our contributions therefore follow three questions: how the clipped objective changes the logit gradient, where its minimum lies, and what it does to training. **The logit gradient** (Section 4.1). We show that, relative to exact forward KL, the clipped objective reverses the direction in which gradient descent moves the logits, on every clipped token and on unclipped token whose student probability lies within a bounded range above the teacher's. **The minimizer** (Section 4.2). We show that, at a response position with the unclipped and clipped token sets held fixed, the clipped objective is minimized only when the clipped tokens' probability is reduced to zero rather than restored toward the teacher's, and each unclipped token receives more probability than the teacher assigns it. A lower total probability on the clipped tokens always permits a lower objective value. **Training runs** (Sections 4.3). We train four matched pairs of runs that differ only in whether clipping is applied and vary whether the student and the teacher think. The clipped runs produce far more repetitions that continue to the end of the response than their unclipped runs. Inside repetitions, the clipped student places less probability than its teacher on leaving, and more on continuing, whereas the unclipped runs stay close to their teachers. These observations are consistent with the logit-gradient reversals we derived.

2 Related Work

Pointwise-clipped forward KL. OPSD introduces the pointwise clipping with threshold τ (Zhao et al., 2026a) on each vocabulary-wise forward KL term. Later studies default threshold $\tau = 0.05$ with training methods deviations(Tan & Hong, 2026b; Hou et al., 2026; Zhang et al., 2026; Liu et al., 2026; Wang et al., 2026; Chen et al., 2026a; Ichihara et al., 2026), at other values and/or with training methods deviations (Chen et al., 2026b; Ichihara et al., 2026; Shrestha & Tessier, 2026; Tan & Hong, 2026a; Liang et al., 2026), or without stating one (Li et al., 2026; Zhao et al., 2026b). Chen et al. (2026b) observe that a clipped term no longer supplies a gradient, so the cap stops the student from following large, mostly stylistic terms, and that a sum of capped signed terms is not a divergence. Feng et al. (2026) notes that the clipped objective equals exact forward KL wherever no term exceeds the threshold, so the two share their gradient and Hessian around the student that matches the teacher, but that elsewhere it can be negative.

Failures reported under the clipped objective. OPSD reports its best checkpoint, yet its comparison of objectives scores lower at 100 updates than at 50 (Zhao et al., 2026a). Chen et al. (2026b) finds that its OPSD baseline loses accuracy from 100 to 200 updates while truncation doubles, and also stating frozen teacher that cannot respond to what the student rejects as one of the limitations. Zhang et al. (2026) find that their three SmolLM3 students fall below the base model by 100 updates under multiple different configurations. Ichihara et al. (2026) finds that, a teacher with a math question and a solution to a physics question leaves many generations repetitive and non-terminating, as compared to one with another math question's solution instead, attribute this to the context without isolating which property is responsible, and leave deterioration under longer training unexplained. Gu et al. (2026) observe that "OPSD trajectories often recompute the same intermediate quantities or revise earlier steps without new information, leading to long and redundant reasoning chains," and attribute this to a teacher conditioned on a single reference solution.

3 Experimental Setup

Training design. Our goal is to reproduce the training design of OPSD, which we reimplement in the verl framework (Sheng et al., 2024). Each run starts from Qwen3-4B (Yang et al., 2025), with a trainable full-parameter student and a frozen teacher initialized from the same checkpoint. The

Table 1: Paired training families. All use Qwen3-4B, teacher top-128 support, $T = 1.1$, 30 problems per update with one rollout per problem, learning rate of 1×10^{-6}, forward-KL distillation, and one trajectory per run. Clipped runs use $\tau = 0.05$ and their unclipped runs do not.

Family	Reference	Student/teacher thinking	Response cap
F	yes	off/on	18,432
M	yes	on/off	18,432
K	yes	off/off	8,192
L	yes	on/on	18,432

teacher additionally receives a reference solution and provides its next-token distribution at every position of the student's responses.

We train for 200 updates on the same 30k OpenThoughts (Guha et al., 2026) dataset used to train OPSD. We construct four matched clipped/unclipped pairs. Within each pair, the data, sampling setup, and model configuration are identical; only whether pointwise clipping is applied differs and we call the unclipped run the *twin* of the clipped run. Table 1 summarizes the run-specific configurations. The complete training hyperparameters and chat template are given in Appendix C. We refer to each pair by its letter in Table 1. F, M, K and L differ only in which side thinks. The letters index the order in which the configurations entered our run series, each added to isolate one variable from an earlier run.

Distillation objective. OPSD computes its loss over the full vocabulary. Our teacher runs as a separate inference service under SGLang (Zheng et al., 2024), and sending a full-vocabulary distribution for every response position is impractical, so our interface returns the teacher's top-128 token log probabilities at each response position.

At response position t, let support $\mathcal{S}_t$ denote set of the teacher's top-128 tokens, and let $a_{t,i}$ and $b_{t,i}$ be the teacher's and the student's logits for token i. Both distributions are formed by a softmax over $\mathcal{S}_t$ at temperature $T = 1.1$,

$$p_{t,i} = \frac{\exp(a_{t,i}/T)}{\sum_{j\in\mathcal{S}_t}\exp(a_{t,j}/T)}, \qquad q_{t,i} = \frac{\exp(b_{t,i}/T)}{\sum_{j\in\mathcal{S}_t}\exp(b_{t,j}/T)}, \qquad i \in \mathcal{S}_t, \tag{1}$$

so that the teacher distribution p_t and the student distribution q_t each sum to one on $\mathcal{S}_t$. We then define the vocabulary-wise forward KL term of token i:

$$d_{t,i} = p_{t,i} \log \frac{p_{t,i}}{q_{t,i}}. \tag{2}$$

Here $d_{t,i}$ is positive where the student assigns token i less probability than the teacher and negative where it assigns more. Exact forward KL sums $d_{t,i}$ directly, whereas the pointwise-clipped forward KL of OPSD first replaces any $d_{t,i}$ above τ by τ, leaving smaller and negative $d_{t,i}$ unchanged:

$$D_{\mathrm{FKL},t} = D_{\mathrm{KL}}(p_t \| q_t) = \sum_{j\in\mathcal{S}_t} d_{t,j}, \qquad D_{\mathrm{clip},t} = \sum_{j\in\mathcal{S}_t} \min(d_{t,j}, \tau), \qquad \tau = 0.05. \tag{3}$$

Despite the symbol, which follows OPSD's notation, $D_{\mathrm{clip},t}$ is not a divergence, since it can be negative. The training loss is the mean of $D_{\mathrm{clip},t}$ in the clipped runs, or of $D_{\mathrm{FKL},t}$ in the unclipped runs, over all valid response positions in the update. We call the tokens in the teacher support $\mathcal{S}_t$ the *candidate tokens* at position t, to distinguish them from the token the student emits there, and a candidate token with $d_{t,i} > \tau$ an *over-threshold* token. In a clipped run, the over-threshold tokens are the clipped tokens.

Trajectory analysis and evaluation. At every response position of every update, we record the teacher's and the student's log probabilities on the candidate tokens. We can therefore compute both the clipped objective and exact forward KL at the same response positions in both runs of each pair. Full capture details are provided in Appendix C.

We evaluate checkpoints saved every 25 training updates on AIME 2025 (Zhang & Math-AI, 2025) and AIME 2024 (Zhang & Math-AI, 2024). The main text reports AIME 2025 only, with two

metrics: per-sample accuracy (avg@12) and the terminal loop rate (generation ended in periodic tails that never terminated). The complete AIME 2024 and AIME 2025 results, and evaluation configuration are given in Appendix D.

4 Failure Dynamics of Pointwise Clipping

4.1 Pointwise clipping reverses the logit gradient

For an over-threshold candidate token, the exact forward KL and the pointwise-clipped forward KL produce gradients on that token's logit with opposite signs. Exact KL gives a negative gradient, whereas the clipped objective gives a positive gradient. For a candidate token that is not over-threshold, the two gradients again have opposite signs exactly when the student's probability exceeds the teacher's but stays below the teacher's probability divided by the teacher's total probability on the tokens that are not over-threshold: exact KL then gives a positive gradient, whereas the clipped objective gives a negative one.

Fix one response position and suppress the index t. The teacher is frozen, so p and its support $\mathcal{S}$ do not depend on the student's logits. We use $d_i = p_i \log(p_i/q_i)$ from Equation 2. Let $z_i = b_i/T$ be the temperature-scaled student logit, so that $q_i = \exp(z_i)/\sum_{j\in\mathcal{S}} \exp(z_j)$. We call the derivative with respect to z_i as its *logit gradient* for candidate token i. Define

$$A = \{j \in \mathcal{S} : d_j \le \tau\}, \qquad C = \mathcal{S} \setminus A, \qquad P_A = \sum_{j\in A} p_j.$$

Here, A and C are the active and clipped candidate token sets, and P_A is the teacher's total probability mass on the active set. Equation 3 becomes

$$D_{\text{clip}} = \sum_{j\in A} p_j \log \frac{p_j}{q_j} + |C|\tau. \tag{4}$$

Proposition 1 (Logit gradient reversal on clipped candidate tokens)**.** For any threshold $\tau > 0$ and any student distribution with $d_j \neq \tau$ for every $j \in \mathcal{S}$, the gradient of D_{clip} with respect to z_i is

$$\frac{\partial D_{\text{clip}}}{\partial z_i} = P_A q_i - p_i \mathbf{1}\{i \in A\}. \tag{5}$$

Every clipped candidate token $i \in C$ satisfies

$$\frac{\partial D_{\text{FKL}}}{\partial z_i} = q_i - p_i < 0, \qquad \frac{\partial D_{\text{clip}}}{\partial z_i} = P_A q_i > 0. \tag{6}$$

Appendix A gives the proof. Both inequalities in Equation 6 are strict, and clipping is what makes their gradient signs differ. For a clipped candidate token $i \in C$, $d_i > \tau > 0$ implies $p_i > q_i$, so the exact KL logit gradient $q_i - p_i$ is negative. Pointwise clipping instead caps the term d_j of every $j \in C$ at the constant τ, which removes the gradient term $-p_i$ in Equation 5. The remaining gradient $P_A q_i$ is positive because $q_i > 0$ under softmax and $P_A > 0$. Since p and q are both normalized over the same support, there exists some j with $q_j \ge p_j$. Hence $d_j \le 0 < \tau$, so $j \in A$, and therefore $P_A \ge p_j > 0$, since Equation 1 gives $p_j > 0$.

Clipping also changes the logit gradient on the active candidate tokens when the student probability lies within a bounded range above the teacher's. When the clipped set is nonempty, $P_A < 1$, so $p_i/P_A > p_i$, for $i \in A$. By equation 5, if the student's probability in the active set satisfies

$$p_i < q_i < p_i/P_A,$$

the exact KL gradient $q_i - p_i$ is positive while the clipped gradient $P_A q_i - p_i < 0$ is negative.

4.2 Clipped objective moves probability from clipped to active tokens

Proposition 1 shows that, for a clipped candidate token, which already has less student than the teacher probability, the logit gradient of the clipped objective points toward lowering its logit, whereas that of exact KL points toward raising it. This reversal does not by itself determine whether

the token's probability is still restored toward the teacher's, as under exact KL, because softmax probabilities depend on all logits. We therefore consider the clipped candidate tokens jointly and ask what happens at the minimizer of the clipped objective at a fixed response position: does an optimized student favor restoring the probability toward the teacher's, or reduced even further?

At a fixed token position, for fixed nonempty active and clipped sets A and C, we consider the constrained problem:

$$\begin{aligned} \underset{q}{\text{minimize}} \quad & D_{\text{clip}}(q) = \sum_{j \in \mathcal{S}} \min(d_j, \tau) \\ \text{subject to} \quad & q_i \geq 0 \ \text{ for every } i \in \mathcal{S}, \qquad \sum_{j \in \mathcal{S}} q_j = 1, \\ & d_i \leq \tau \ \text{ for every } i \in A, \qquad d_i > \tau \ \text{ for every } i \in C. \end{aligned}$$

The feasible set of student distributions, denoted the region R, is the *fixed clipping region* of (A, C). Unlike in Section 4.1, where q is a softmax of finite logits and every $q_i > 0$, we allow $q_i = 0$ so that a minimizer exists. For such a candidate token i, $d_i = +\infty$, the capped term is still τ, so it stays clipped. Everywhere in R, D_{clip} is given by Equation 4 with these fixed sets.

Proposition 2 (Fixed-region minimizer of the clipped objective)**.** The clipped objective has a unique minimizer over the fixed clipping region R,

$$q_i^\star = \begin{cases} p_i / P_A, & i \in A, \\ 0, & i \in C, \end{cases} \qquad D_{\text{clip}}(q^\star) = P_A \log P_A + |C|\tau. \tag{7}$$

Let $Q_C = \sum_{j \in C} q_j$ be the student's total probability mass on the clipped set. For every value of Q_C attained in R, if its value is further added to the constraints without inconsistencies with the other constraints, the minimum of D_{clip} over the distributions of q's in $R(Q_C)$ with that value is strictly increasing in Q_C:

$$D^\star_{\text{clip}}(Q_C) = P_A \log P_A - P_A \log(1 - Q_C) + |C|\tau, \qquad \frac{dD^\star_{\text{clip}}}{dQ_C} = \frac{P_A}{1 - Q_C} > 0. \tag{8}$$

Appendix B gives the proof. At the minimizer $q^\star$ of the clipped objective, the clipped candidate tokens' total probability is not restored toward the teacher's total $1 - P_A$ on these tokens but reduced further, to zero. All probability then lies on A, and the student's probability on each active candidate token equals the teacher's probability p_i multiplied by the same factor $1/P_A > 1$. Clipping therefore does more than limit the influence of dominant terms: inside a fixed clipping region, its minimizer assigns zero student probability to clipped candidate tokens, even though they already have less student probability than teacher probability, and gives each active candidate token more student probability than the teacher probability.

A softmax of finite logits, however, never attains $q^\star$, because it gives every clipped token positive probability. Hence $Q_C > 0$, and $D_{\text{clip}}(q^\star)$ is an unattained infimum. For such a student, Equation 8 characterizes the minimum attainable objective value for each Q_C. The lower the student's total probability on the clipped candidate tokens, the closer the objective can come to this infimum, and restoring Q_C toward the teacher's $1 - P_A$ only moves this minimum attainable value farther from it.

4.3 Clipped updates turn started repetition into persistent loops

In matched runs that differ only in whether the objective is clipped, clipped training produces substantially more terminal loops than exact KL training (Figure 1; Table 3). We defined the repetition-related terminologies in Table 2 and use them throughout the section. The preceding two sections give a candidate cause: the clipped objective reverses the logit gradient of exact KL on every clipped token and on active tokens within a bounded range above the teacher's probability. Within a fixed clipping region, its minimizer sets the clipped tokens' probability to zero and gives each active token more probability than the teacher, instead of correcting the student toward the teacher. If, in training, the teacher's exit tokens are clipped and the copy token lies in this range, the clipped objective would push the student to leave a repetition less often and to continue it more often than the teacher, so repetitions and terminal loops would be more than in the exact KL twin, although the size of this

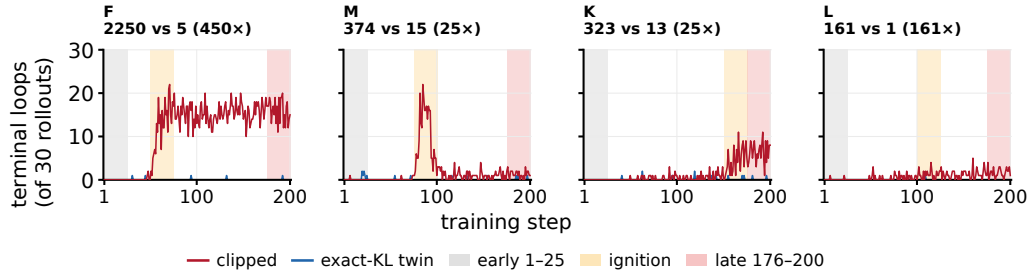


Figure 1: Loop outcome over training, one trajectory per run. Each point is the number of the 30 rollouts sampled at that update that end in a terminal loop. Clipped runs in red, unclipped twins in blue. Shaded windows are three phases: early (updates 1–25, gray), ignition (F 51–75, M 76–100, K 151–175, L 101–125, orange), and late (updates 176–200, pink).

effect may differ across families. Both results describe one response position with a given clipped set, whereas in training the clipped set changes across sampled responses and positions, so whether the student's probabilities move this way must be measured. We compare student and teacher exit mass at the first copy, which every repetition passes through, and the copy token's logit gradient under both objectives before maturity. These comparisons can support the logit gradient reversal as the path from clipping to loops but cannot establish that path.

Table 2: Definitions used to characterize token repetition and analyze its continuation. The upper block defines periodic segments, repetitions, mature repetitions, and terminal loops. The lower block defines the first copy, copy token, and exit mass.

Term	Definition
Periodic segment	A maximal contiguous sequence $x_s, \ldots, x_e$, where s and e are its start and end positions, with period ℓ and length $L = e - s + 1 > \ell$. L satisfies $x_t = x_{t-\ell}$ for $s + \ell \le t \le e$.
Repetition	A periodic segment with at least two complete period $\lfloor L/l \rfloor \ge 2$.
Mature repetition	A repetition $\lfloor L/l \rfloor \ge 3$ with $L \ge 30$ tokens.
Terminal loop	Mature repetition that reaches the last response token.
Copy token	At a position $t \ge s + \ell$ while the periodic continuation remains, the token $c_t = x_{t-\ell}$ that continues the period. Its teacher and student probabilities are $p_{t,\text{copy}}$ and $q_{t,\text{copy}}$, respectively.
First copy	Copy token belong to the second completed occurrence of the period, at positions $s + \ell, \ldots, s + 2\ell - 1$.
Exit mass	At position t, the probability assigned to tokens other than the copy token: $1 - p_{t,\text{copy}}$ for the teacher and $1 - q_{t,\text{copy}}$ for the student.

Outcome. We define three 25-update phases: an early phase at the start of training, an ignition window at each clipped run's loop onset, and a late phase at the end of training. In the early phase, terminal loops are nearly absent from every run. The exact KL runs remain there for the whole run, never exceeding 2 of 30 rollouts at any update. The clipped runs do not. Each run has a family-specific onset after which terminal loops recur, and over the full run it produces 25 to 450 times as many of them as its twin. The result is shown in Figure 1. The thinking-mode configuration sets both the scale and the ignition time.

Trend. Table 3 follows repetitions ($N \ge 2$) through three phases. In the early phase the two runs of every pair are indistinguishable except run L. They generate repetition and mature repetition at the same rate. From ignition on, two amplifications hold. First, a repetition grows into a mature repetition ($N \ge 3$ and $L \ge 30$) more often under clipping than in the twin. The count $N(\text{mat.} \mid N \ge 2)$ is 8.6 to over 400 times the twin's, and the rate $P(\text{mat.} \mid N \ge 2)$ 4.5 to over 150 times. Second, far more repetitions end as terminal loops: 26 to 381 per phase in the clipped runs, against at most two in any twin.

Table 3: Progression of repetition by training phase (period 2–50). Rows: $n(N \geq 2)$, the number of repetitions; $N(\text{mat.} \mid N \geq 2)$, the mature repetitions among repetitions; $P(\text{mat.} \mid N \geq 2)$ the proportion of repetitions that mature; $N(\text{term.} \mid \text{mat.})$, the terminal loops among mature repetition. Each cell reads early → ignition → late, the three phases of Figure 1, 750 rollouts each.

Quantity	Arm	F	M	K	L
$n(N \geq 2)$	clipped	1195→4954→2814	3852→2666→1304	1163→3757→4024	5952→5704→6374
	twin	1237→2395→3154	3288→975→1496	1130→1042→1043	3991→2642→3121
$N(\text{mat.} \mid N \geq 2)$	clipped	1→790→749	6→409→69	4→170→176	16→195→328
	twin	5→3→7	16→1→8	6→4→8	7→20→9
$P(\text{mat.} \mid N \geq 2)$	clipped	.001→.159→.266	.002→.153→.053	.003→.045→.044	.003→.034→.051
	twin	.004→.001→.002	.005→.001→.005	.005→.004→.008	.002→.008→.003
$N(\text{term.} \mid \text{mat.})$	clipped	0→307→381	1→206→26	0→109→155	1→30→35
	twin	0→0→1	6→0→1	0→1→2	0→1→0

Family-specific loop patterns. With student thinking off and teacher thinking on (F), the student writes answer-mode derivations that the teacher scores where its own thinking block would begin. The student chains the repeated "⇒" where the teacher prefers `\text`. This two-token cycle supplies most mature repetitions at ignition, and by the late phase all of them are terminal loops. With both modes off (K), student and teacher share the answer register, and most of its terminal loops are answer-closing formatting such as nested "\boxed{". The teacher prefers a content command such as `\text` or `\frac`. With both modes on (L), most mature repetitions lie inside the thinking block and repeat an equation or a reasoning frame such as "Let me compute:", where the teacher's preferred alternative varies, most often a space, an opening parenthesis, or a different word such as "take". Most of its mature repetitions exit rather than become terminal. With student thinking on and teacher thinking off (M), the teacher's prompt contains a closed, empty thinking block, so the teacher scores the student's thinking as answer text, and both runs stop emitting the closing tag. At ignition, M's terminal loops repeat an answer-closing mark such as a check mark inside the unclosed thinking block, where the teacher prefers the end-of-response token `<|im_end|>`.

Exit mass at the first copy. We dive deep into the training signal in each runs and how clipped objective applied to it. We examine whether clipping suppresses exit mass at the first copy of a repetition, terms defined in Table 2. Let $\mathcal{F}_u$ denotes the window containing first-copy positions of all detected repetitions with periods $2 \leq \ell \leq 50$ across responses generated at update u. For each nonoverlapping five-update block B, we compute the student/teacher exit-mass ratio by summing each distribution's exit mass over these positions:

$$R_B = \frac{\sum_{u \in B} \sum_{t \in \mathcal{F}_u} (1 - q_{u,t,\text{copy}})}{\sum_{u \in B} \sum_{t \in \mathcal{F}_u} (1 - p_{u,t,\text{copy}})}.$$

We then measure the clipped ratio of the teacher's exit mass that lies on clipped tokens. For the unclipped twins, we report the corresponding fraction on over-threshold tokens (Figure 2a). Pooled over the early phase, the student/teacher exit-mass ratio is .79 to 1.04. Under exact KL, the ratio is close to 1 across the whole run, even though 33 to 45 percent of the teacher's exit mass still lies on over-threshold tokens. Under clipping, however, the exit mass ratio falls to .41 to .70 in the late phase, while 71 to 84 percent of the teacher's exit mass is clipped. Thus, at the first copy, the clipped student places less probability than its teacher on leaving the repetition, and most of the teacher's exit mass lies on the tokens whose logit gradient Proposition 1 reverses.

Copy token probability before maturity. We next examine the positions a repetition passes through before it becomes mature: how often the student places more probability than the teacher on the copy token, and whether each objective corrects this excess.

For each 25-update phase ϕ, let $\mathcal{G}_\phi$ denote the window of copied positions after the first copy and before maturity or an observed stop. We record the fraction at which the student assigns greater probability to the copy token than the teacher:

$$R_\phi = \frac{\sum_{t \in \mathcal{G}_\phi} \mathbf{1}\{q_{t,\text{copy}} > p_{t,\text{copy}}\}}{|\mathcal{G}_\phi|},$$

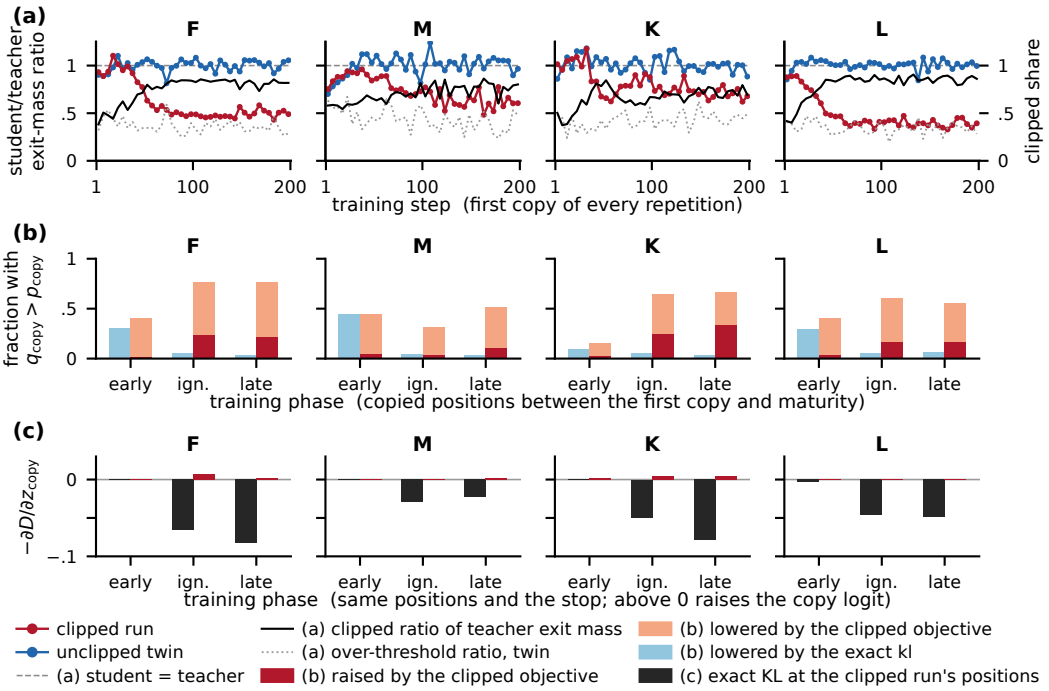


Figure 2: The copy token inside repetitions (period 2–50), with the student's probabilities taken before the sampler's truncation. (a) Student/teacher exit-mass ratio, one point per block of five updates (1–5, . . . , 196–200). Lines without markers give the ratio of the teacher's exit mass on clipped tokens, or on over-threshold tokens in the twins (right axis). (b) Copied positions between the first copy and maturity: fraction with $q_{\text{copy}} > p_{\text{copy}}$. Dark bar: $q_{\text{copy}} < p_{\text{copy}}/P_A$, where the logit gradient of exact KL points toward lowering the copy logit and that of the clipped objective toward raising it. Light bar: both point toward lowering it. (c) The same copied positions, plus observed stopping positions before maturity: mean $-\partial D/\partial z_{\text{copy}}$. Black and red: exact KL (hypothetical) and clipped KL, respectively, evaluated on the same positions and probabilities from the clipped run. Positive values point toward raising the copy logit. Phases as in Table 3.

where $\mathbf{1}\{\cdot\}$ is the indicator function and counts are pooled across repetitions and updates within the phase. The result is in Figure 2b. In the early phase, both runs of F, M, and L have $q_{\text{copy}} > p_{\text{copy}}$ at 29 to 45 percent of these positions. After ignition, this fraction falls to 4 to 6 percent in the exact KL runs but 32 to 77 percent in the clipped runs. In the clipped runs, q_{copy} lies specifically between p_{copy} and p_{copy}/P_A at 4 to 34 percent of positions. The dark part of each bar in Figure 2b shows the fraction of positions at which the copy token's logit gradient is reversed: the clipped objective points toward raising the copy token logit where exact KL points toward lowering it.

We then measure the gradient of each objective $D \in \{D_{\text{FKL}}, D_{\text{clip}}\}$ on the copy token. For each phase ϕ, we record the mean of $-\partial D/\partial z_{\text{copy}}$ over the window $\mathcal{G}_\phi$ plus the observed stop positions, pooled across repetitions and updates. We report the negative of the gradient so that positive values point toward raising the copy token logit. We calculate $\partial D_{\text{FKL}}/\partial z_{\text{copy}} = q_{\text{copy}} - p_{\text{copy}}$ and $\partial D_{\text{clip}}/\partial z_{\text{copy}}$, given by Equation 5. We evaluate both objectives on the positions and probabilities of the clipped run only, so that any difference between the two means comes from the objective alone. Figure 2c shows the result: the net direction of each objective's logit gradient on the copy token over all positions of the window. In the early phase, both values are nearly zero in every family. In the ignition and late phases, the mean of $-\partial D_{\text{FKL}}/\partial z_{\text{copy}}$ is negative, pointing toward lowering the copy token logit, whereas that of D_{clip} is slightly positive in every family, pointing toward raising it. At the positions that decide whether a repetition becomes mature, exact KL would thus correct the student's excess probability on the copy token, and the clipped objective removes this correction.

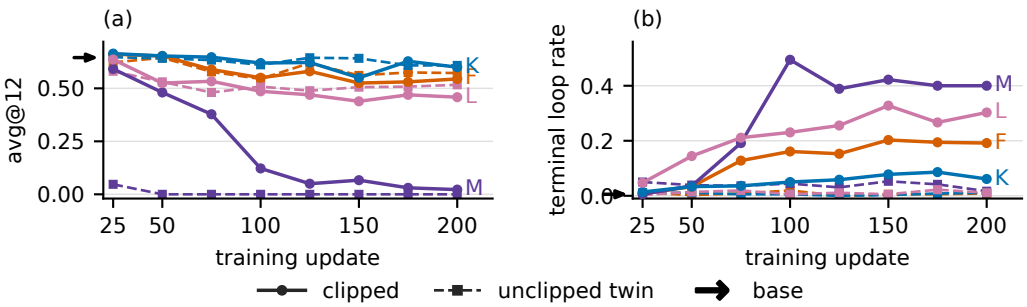


Figure 3: AIME 2025 across training. (a) Avg@12, the mean correct ratio over 12 samples for each of 30 problems at the checkpoint saved after each 25 updates. (b) The ratio of the 360 responses per checkpoint that end in a terminal loop. Solid curves with circles are the clipped runs, dashed curves with squares their unclipped twins. Each trained curve is one realized trajectory. The black arrow on each vertical axis marks the base model, the fixed starting checkpoint averaged over eight evaluation-sampling seeds.

5 Held-Out Evaluation

We evaluate every checkpoint, saved at 25-update intervals, on the AIME 2025 problems. Figure 3 reports accuracy and terminal loop rate at saved checkpoints. It tests whether clipping increases looping and whether the additional loops accompany accuracy losses relative to the unclipped twin.

No run significantly outperforms the base model (Figure 3a). K show only minimal accuracy changes, whereas F, L, and M degrade substantially, and most clipped runs perform worse than their unclipped twins. M is the exception: both of its run collapse, and its twin reads zero only because it stops closing the thinking tag, so the strict scorer finds no answer.

Terminal loop rate separates the runs far more sharply (Figure 3b). Every clipped run ends in a terminal loop far more often than the base model, while every unclipped twin stays near the base rate. Looped responses fail to reach a final answer and are therefore scored as incorrect. Accuracy losses in the clipped runs follow the same ordering as their loop rates: smallest in K, intermediate in F, and largest in L and M.

With 12 samples per problem, no clipped endpoint has lower pass@12 than its twin on either benchmark: every problem solved by the twin is also solved at least once by the clipped arm. We interpret this pattern as evidence that clipped training reduces per-sample reliability. However, the unclipped F and L twins remain below the fixed-base mean despite rarely looping. Thus, clipping-induced looping contributes to the observed degradation, but does not explain all accuracy loss relative to the base model.

Appendix D gives avg@12, terminal loop rate, and pass@12 at every checkpoint on AIME 2024 and AIME 2025 benchmarks.

6 Conclusion

Pointwise clipping was introduced to stabilize training against dominant stylistic tokens, but it also changes the direction in which forward kl corrects the student. In training this change appears as repetitions that continue to the end of the response, a loss of training stability. Because this failure develops over updates, studies that train with the pointwise-clipped objective could report accuracy and a text degeneration measure, across checkpoints, in addition to their best checkpoint. We hope that understanding these clipping dynamics helps make on-policy self-distillation stable enough for its gains to persist over longer training.

AI USE STATEMENT

In this work, we used generative AI tools for helping develop theoretical models or conceptual frameworks, assisting in the writing of proofs, proposing or refining hypotheses, designing or providing feedback on research, methodology or experiments, implementing methods, supporting qualitative and thematic data analysis, interpreting results. We have not used generative AI tools for generate synthetic data sets, formulating mathematical claims, providing critical ingredients for proving mathematical claims, assisting with translation, cleaning and reformatting dataset. and no required disclosure is not applicable to this work. Additionally, we used generative AI tools for creating or modifying scientific figures or images, creating or editing software code, summarizing or analyzing existing literature, and drafting parts of a research paper. We have reviewed all AI-assisted work. LLM-generated code was verified and tested for correctness. We proofread the manuscript to ensure all math propositions and proofs are valid and correct. We manually proofread AI recommended literature and manually add the citation. We take responsibility for the final content of this work, including text, claims or artifacts produced with the aid of generative AI.

REPRODUCIBILITY STATEMENT

Proof of 1 can be found in Appendix A. Proof of 2 can be found in Appendix B

REFERENCES

Rishabh Agarwal, Nino Vieillard, Yongchao Zhou, Piotr Stanczyk, Sabela Ramos Garea, Matthieu Geist, and Olivier Bachem. On-policy distillation of language models: Learning from self-generated mistakes. In *The Twelfth International Conference on Learning Representations*, 2024. URL https://openreview.net/forum?id=3zKtaqxLhW.

Yunmeng Chen, Kunyu Wang, Peihan Li, Yi Wang, Shuyin Xia, Yi Liu, Xinyong Cheng, Dehui Wang, Xiangyong Zhai, Yanxing Liu, et al. Scope-opsd: Fisher-conditioned privileged subspaces for on-policy self-distillation. *arXiv preprint arXiv:2609.12579*, 2026a.

Yutong Chen, Guangfu Guo, Zhichao Xu, and Kunpeng Liu. Dualopsd: Adaptive privileged teachers for on-policy self-distillation. *arXiv preprint arXiv:2608.26019*, 2026b.

Yangyang Feng, Zhuoyan Feng, and Junlan Chen. Past: Privileged adaptation from complete student trajectories for on-policy self-distillation. *arXiv preprint arXiv:2608.08726*, 2026.

Siyi Gu, Jialin Chen, Sophia Zhou, Arman Cohan, and Rex Ying. Rethinking reward supervision: Rubric-conditioned self-distillation. *arXiv preprint arXiv:2606.19327*, 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, pp. 108059–108130, 2026.

ZhiYan Hou, Xinyu Tang, Hongyan An, Jianjin Zhang, Weizhen Wang, Yunyun Han, Gengsheng Li, Xiangzhao Hao, Haiyun Guo, Wenbin Hu, et al. Dash: Divergence-adaptive supervision horizons for on-policy self-distillation of reasoning models. *arXiv preprint arXiv:2608.06243*, 2026.

Jonas Hübotter, Frederike Lübeck, Lejs Behric, Anton Baumann, Marco Bagatella, Daniel Marta, Ido Hakimi, Idan Shenfeld, Thomas Kleine Buening, Carlos Guestrin, et al. Reinforcement learning via self-distillation. *arXiv preprint arXiv:2601.20802*, 2026.

Yuki Ichihara, Naoto Iwase, Mohammad Atif Quamar, and Junpei Komiyama. Privileged solutions or context-induced teacher behavior? dissecting on-policy self-distillation. *arXiv preprint arXiv:2608.09228*, 2026.

Hynek Kydlíček. Math-Verify: Math Verification Library. URL https://github.com/huggingface/math-verify.

Yijiang Li, Bingyang Wang, Yijun Liang, Yunjie Tian, Di Fu, and Nuno Vasconcelos. On-policy self-distillation without any supervision. *arXiv preprint arXiv:2608.06296*, 2026.

Zihan Liang, Yufei Ma, Ben Chen, Zhipeng Qian, Xuxin Zhang, Huangyu Dai, and Lingtao Mao. Search-e1: Self-distillation drives self-evolution in search-augmented reasoning. *arXiv preprint arXiv:2605.22511*, 2026.

Xiaogeng Liu, Xinyan Wang, Yingzi Ma, Yechao Zhang, and Chaowei Xiao. When are teacher tokens reliable? position-weighted on-policy self-distillation for reasoning. *arXiv preprint arXiv:2605.21606*, 2026.

Kevin Lu and Thinking Machines Lab. On-policy distillation. *Thinking Machines Lab: Connectionism*, 2025. doi: 10.64434/tml.20251026. https://thinkingmachines.ai/blog/on-policy-distillation.

Leyi Pan, Shuchang Tao, Yunpeng Zhai, Lingzhe Zhang, Zhaoyang Liu, Bolin Ding, Aiwei Liu, and Lijie Wen. RLCSD: Reinforcement learning with contrastive On-Policy self-distillation. *arXiv preprint arXiv:2606.11709*, 2026. URL https://arxiv.org/abs/2606.11709.

Idan Shenfeld, Mehul Damani, Jonas Hübotter, and Pulkit Agrawal. Self-distillation enables continual learning. In *Forty-third International Conference on Machine Learning*, 2026. URL https://openreview.net/forum?id=qA6FgH0nnZ.

Guangming Sheng, Chi Zhang, Zilingfeng Ye, Xibin Wu, Wang Zhang, Ru Zhang, Yanghua Peng, Haibin Lin, and Chuan Wu. Hybridflow: A flexible and efficient rlhf framework. *arXiv preprint arXiv: 2409.19256*, 2024.

Samyak Shrestha and Alexander Tessier. Rethinking privileged information in on-policy self-distillation. *arXiv preprint arXiv:2608.18271*, 2026.

Zhiquan Tan and Yinrong Hong. Paint: Partial-solution adaptive interpolated training for self-distilled reasoners. *arXiv preprint arXiv:2604.26573*, 2026a.

Zhiquan Tan and Yinrong Hong. Self-supervised on-policy distillation for reasoning language models. *arXiv preprint arXiv:2605.17497*, 2026b.

Jiaxuan Wang, Xuan Ouyang, Zhiyu Chen, Yulan Hu, Zheng Pan, Xin Li, and Lan-Zhe Guo. Trace: Distilling where it matters via token-routed self on-policy alignment. *arXiv preprint arXiv:2605.10194*, 2026.

An Yang, Anfeng Li, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Gao, Chengen Huang, Chenxu Lv, et al. Qwen3 technical report. *arXiv preprint arXiv:2505.09388*, 2025.

Yuxiao Yang, Xiaoyun Wang, and Weitong Zhang. Ogls-sd: On-policy self-distillation with outcome-guided logit steering for llm reasoning. *arXiv preprint arXiv:2605.12400*, 2026.

XiuYu Zhang, Wei Chow, Junfeng Fang, Zhenkai Liang, and Tat-Seng Chua. What does privileged information add to on-policy self-distillation? *arXiv preprint arXiv:2609.20612*, 2026.

Yifan Zhang and Team Math-AI. American invitational mathematics examination (aime) 2024, 2024.

Yifan Zhang and Team Math-AI. American invitational mathematics examination (aime) 2025, 2025.

Siyan Zhao, Zhihui Xie, Mengchen Liu, Jing Huang, Guan Pang, Feiyu Chen, and Aditya Grover. Self-distilled reasoner: On-policy self-distillation for large language models. In *Forty-third International Conference on Machine Learning*, 2026a. URL https://openreview.net/forum?id=Jpxfof0EaS.

Xuyang Zhao, Liting Zhang, Zichen Xu, Zhihu Wang, Xu Caiyue, Shiwan Zhao, and Qicheng Li. Is more privileged information better? From solution traces to problem-solving structure in self-distilled reasoning. *arXiv preprint arXiv:2608.01589*, 2026b. URL https://arxiv.org/abs/2608.01589.

Lianmin Zheng, Liangsheng Yin, Zhiqiang Xie, Chuyue Sun, Jeff Huang, Cody H Yu, Shiyi Cao, Christos Kozyrakis, Ion Stoica, Joseph E Gonzalez, et al. Sglang: Efficient execution of structured language model programs. *Advances in neural information processing systems*, 37:62557–62583, 2024.

A Proof of Proposition 1.

Fix a response position and suppress its index t. Hold the finite support $\mathcal{S}$ and the teacher distribution p fixed. By Equation 1, $p_j, q_j > 0$ for every $j \in \mathcal{S}$, and both distributions sum to one on $\mathcal{S}$. For the temperature-scaled student logits $z_j = b_j/T$, write

$$Z = \sum_{j \in \mathcal{S}} \exp(z_j), \qquad q_j = \frac{\exp(z_j)}{Z}.$$

Consider any logit vector z° satisfying $d_j \neq \tau$ for all $j \in \mathcal{S}$, and let A and C be its active and clipped sets as defined in Section 4.1. Each $d_j = p_j \log(p_j/q_j)$ is smooth in z, since p_j is fixed and q_j is a positive smooth function of z. Because $\mathcal{S}$ is finite and no d_j equals τ at z°, there is an open neighborhood of z° on which A and C, and hence $P_A = \sum_{j \in A} p_j$, remain unchanged. On this neighborhood, Equation 4 gives

$$D_{\text{clip}} = \sum_{j \in A} d_j + |C|\tau,$$

with A and C fixed at their values at z°. This expression is smooth in z, so D_{clip} is differentiable on the neighborhood and its gradient is obtained by differentiating the displayed expression.

For any $i, j \in \mathcal{S}$, the softmax identity $\log q_j = z_j - \log Z$ gives

$$d_j = p_j \log p_j - p_j z_j + p_j \log Z.$$

Because p is fixed and

$$\frac{\partial z_j}{\partial z_i} = \mathbf{1}\{j = i\}, \qquad \frac{\partial \log Z}{\partial z_i} = \frac{1}{Z}\frac{\partial Z}{\partial z_i} = \frac{\exp(z_i)}{Z} = q_i,$$

it follows that

$$\frac{\partial d_j}{\partial z_i} = -p_j \mathbf{1}\{j = i\} + p_j q_i. \tag{9}$$

Since $D_{\text{FKL}} = \sum_{j \in \mathcal{S}} d_j$ by Equation 3, summing Equation 9 over $\mathcal{S}$ gives

$$\begin{aligned} \frac{\partial D_{\text{FKL}}}{\partial z_i} &= \sum_{j \in \mathcal{S}} \big(-p_j \mathbf{1}\{j = i\} + p_j q_i\big) \\ &= -p_i + q_i \sum_{j \in \mathcal{S}} p_j = q_i - p_i, \end{aligned}$$

at every logit vector, including points on the clipping boundaries. For the clipped objective, $|C|\tau$ has zero derivative on the neighborhood under consideration. Summing only over A therefore gives

$$\begin{aligned} \frac{\partial D_{\text{clip}}}{\partial z_i} &= \sum_{j \in A} \big(-p_j \mathbf{1}\{j = i\} + p_j q_i\big) \\ &= -p_i \mathbf{1}\{i \in A\} + q_i \sum_{j \in A} p_j \\ &= P_A q_i - p_i \mathbf{1}\{i \in A\}. \end{aligned} \tag{10}$$

This establishes Equation 5.

Now let $i \in C$. Then $d_i > \tau > 0$, and $p_i > 0$ implies

$$\log \frac{p_i}{q_i} > 0, \qquad \text{hence} \qquad p_i > q_i.$$

Consequently, $\partial D_{\mathrm{FKL}}/\partial z_i = q_i - p_i < 0$. Since $i \notin A$, Equation 10 reduces to $\partial D_{\mathrm{clip}}/\partial z_i = P_A q_i$. To establish strict positivity, first note that $q_i > 0$. Moreover, $\sum_{j\in\mathcal{S}} q_j = \sum_{j\in\mathcal{S}} p_j$ implies the existence of a token j with $q_j \geq p_j$. For this token,

$$d_j = p_j \log \frac{p_j}{q_j} \leq 0 < \tau,$$

so $j \in A$ and $P_A \geq p_j > 0$. Therefore,

$$\frac{\partial D_{\mathrm{FKL}}}{\partial z_i} = q_i - p_i < 0, \qquad \frac{\partial D_{\mathrm{clip}}}{\partial z_i} = P_A q_i > 0,$$

which proves Equation 6. Since z° was arbitrary away from the clipping boundaries, the result holds at every point specified in the proposition. □

A simpler example of updating dynamics To understand the direction of the updates on the logits as suggested by the previous proposition, the example below can give some intuitions. In the most general case, decreasing certain logits on index i does not automatically mean to decrease the corresponding probability q_i–but in the case when other z values are fixed, this holds true. We can construct a continuous-time dynamical system to mimic the optimization process on the objective functions of the two model versions, defining the parameter updates as the negative gradient of the loss with respect to the logits ($\frac{dz_i}{dt} = -\nabla_{z_i} D$). For the exact forward kl objective D_{FKL}, this logit update is defined continuously as:

$$\frac{dz_i}{dt}(t) = p_i - q_i(t)$$

For the clipped objective, the gradient space is instead split into a piecewise system governed by the local divergence threshold d_j:

$$\frac{dz_i}{dt}(t) = \begin{cases} p_i - P_A q_i(t), & \text{if } d_j \leq \tau \\ -P_A q_i(t), & \text{if } d_j > \tau \end{cases}$$

To map these raw logit updates $\frac{dz_i}{dt}$ to the resulting probability trajectories $\frac{dq_i}{dt}$, we must account for the coupling effect of the softmax function, where the evolution of a single probability q_i depends on the updates to all logits in the vocabulary. However, to isolate the localized forces acting on a single prediction and visualize them within a simplified 2D direction field diagram, we introduce a specific framing assumption where all other logits $z_j, j \neq i$ are temporarily held constant ($\frac{dz_j}{dt} = 0$). Under this isolated assumption, because the softmax function is strictly monotonically increasing with respect to its own logit, $\frac{dq_i}{dt}$ acts as a positive monotonic scaling of $\frac{dz_i}{dt}$. This strict sign preservation guarantees that the direction of the probability update exactly mirrors the direction of the logit update, providing the mathematical justification to map the negative gradients directly onto a 2D phase portrait to evaluate their critical points and structural stability. The direction field, null-cline, contour for the clipping region and the diagonal (for the correct alignment p=q) is shown in figure 4.

Applying this localized framework to the exact forward kl divergence (left) demonstrates that the gradient acts as a proportional restorative force across the entire probability space, driving the system toward a single, globally stable equilibrium along the critical boundary where $p_i = q_i$ (the diagonal). Because the directional derivative relies entirely on the unaltered residual, the vector field smoothly and pushes the predictions directly toward perfect calibration. This ensures the parameter space remains entirely free of dead zones, structural barriers, or competing attractors.

Conversely, the clipped objective (right) shatters the global stability of the forward kl. In the active set (unshaded area on bottom-right corner) where the divergence is low ($d_j < \tau$), scaling the predicted term by P_A artificially shifts the fixed point away from true alignment, forcing the vectors toward a false attractor along the line $p_i = P_A q_i$ (the blue line with slope that is less steep). When the local divergence exceeds the τ threshold, the system crosses into the clipped set (green shaded area on the top left), completely erasing the target probability q_i from the derivative calculation. Inside this boundary, the dynamic devolves into pure probability decay. Because the gradient equation yields a negative update ($-P_A q_i$), it relentlessly pushes the predictions until 0 is reached,

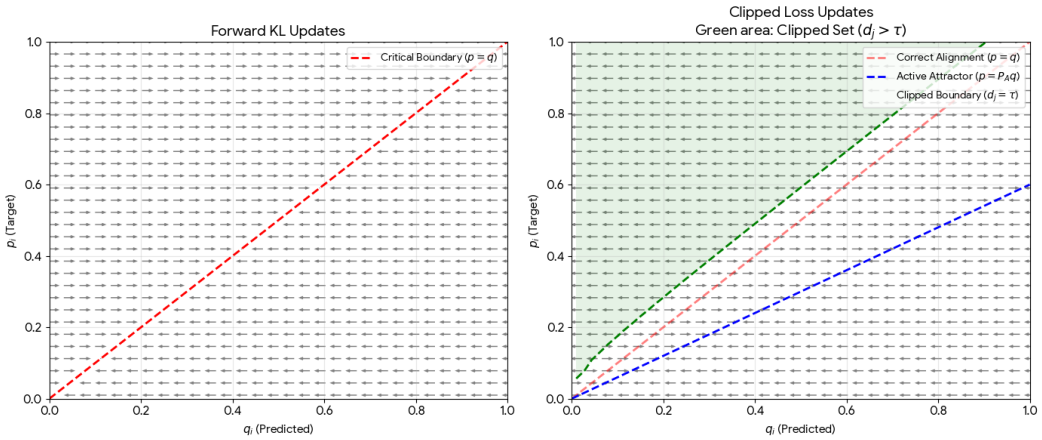


Figure 4: A simple example of updating direction of q under different losses(Forward KL v.s. Clipped Loss Updates.

structurally preventing the system from ever recovering true calibration when initialized or driven too far from the target. To sum up, the green shaded area along with the area between the red and blue lines on the right are where the arrows for the clipped objectives are in a different direction as compared with the ones for forward KL. One might observe that if you trace a phase portrait alone the arrows, then if you start with a point in either clipped or active set, the trajectory never goes into the other–this may give some intuition for boundary check for the next proposition as well.

B Proof of Proposition 2

Setup. Fix one response position. Let p be the teacher distribution on the finite candidate set $\mathcal{S}$. It is a softmax output, so $p_i > 0$ for every $i \in \mathcal{S}$, and $\sum_{i\in\mathcal{S}} p_i = 1$. Let $\tau > 0$. Student distributions are the points of the closed simplex,

$$q_i \geq 0 \quad (i \in \mathcal{S}), \qquad \sum_{i\in\mathcal{S}} q_i = 1,$$

so probabilities equal to zero are allowed. We use the convention $p_j \log(p_j/0) = +\infty$. The capped term of a zero-probability token is then $\min\{+\infty, \tau\} = \tau$, so

$$D_{\text{clip}}(q) = \sum_{j\in\mathcal{S}} \min\Big\{p_j \log \frac{p_j}{q_j},\ \tau\Big\}$$

is finite on the whole simplex.

Let A and C be fixed nonempty disjoint sets with $A \cup C = \mathcal{S}$. The *fixed clipping region* R is the set of student distributions q with

$$p_i \log \frac{p_i}{q_i} \leq \tau \quad (i \in A), \qquad p_j \log \frac{p_j}{q_j} > \tau \quad (j \in C).$$

Write $P_A = \sum_{i\in A} p_i$ and $Q_C = \sum_{j\in C} q_j$. Since A and C are nonempty and every $p_i > 0$, we have $0 < P_A < 1$. For $q \in R$, every active term is at most τ, so its minimum with τ is the term itself, and every clipped term exceeds τ, so its minimum with τ is τ. Hence, for $q \in R$ (Equation 4),

$$D_{\text{clip}}(q) = \sum_{i\in A} p_i \log \frac{p_i}{q_i} + \sum_{j\in C} \tau = F(q_A) + |C|\tau, \qquad F(q_A) = \sum_{i\in A} p_i \log \frac{p_i}{q_i}, \qquad q_A = (q_i)_{i\in A}.$$

This identity is valid on R. Outside R it fails in general, because a term that exceeds τ is capped.

Claim. The distribution $q^\star$ with $q_i^\star = p_i/P_A$ for $i \in A$ and $q_j^\star = 0$ for $j \in C$ is the unique minimizer of D_{clip} over R, and $D_{\text{clip}}(q^\star) = P_A \log P_A + |C|\tau$ (Equation 7).

Overview. The proof has two steps. *Step 1* settles how much mass sits on the clipped set: any such mass can be moved onto an active token without leaving R, and doing so strictly lowers the loss, so only distributions with $Q_C = 0$ can compete. *Step 2* settles how that mass is split within the active set. Dropping the active-token clipping inequalities leaves a larger, smooth problem: minimize the strictly convex function F over the positive allocations with $\sum_{i\in A} q_i = 1$. Its unique global minimizer is the Lagrange point $q_i = p_i/P_A$, which exceeds p_i because $P_A < 1$. *The crucial implication is that the unique global minimizer of the larger, smooth problem satisfies the original clipping constraints.* It is therefore also the unique minimizer of the original, restricted problem. The larger problem is only an auxiliary device: outside R the formula F is not the clipped objective, and nothing is claimed there.

Membership facts (M1)–(M4). Steps 1 and 2 use four facts about when a token is active or clipped. These are the only places where the threshold enters the proof. Fix a token i and regard its term $p_i \log(p_i/x)$ as a function of its student probability $x \geq 0$, with value $+\infty$ at $x = 0$. For $x > 0$,

$$\frac{d}{dx}\Big[p_i \log \frac{p_i}{x}\Big] = \frac{d}{dx}\big[p_i \log p_i - p_i \log x\big] = -\frac{p_i}{x} < 0,$$

so the term is strictly decreasing in x on $(0, \infty)$, and at $x = p_i$ it equals $p_i \log 1 = 0$.

(M1) *Every active token has $q_i > 0$.* If $q_i = 0$, the term is $+\infty > \tau$, so token i would be clipped, not active. Consequently F is finite on R.

(M2) *An active token that gains probability stays active.* If $q_i' \geq q_i > 0$, then, because the term is decreasing, $p_i \log(p_i/q_i') \leq p_i \log(p_i/q_i) \leq \tau$.

(M3) *A clipped token that loses probability stays clipped, including at zero.* Let $0 \leq q_j' \leq q_j$. If $q_j' = 0$, the term is $+\infty > \tau$. If $q_j' > 0$, then $q_j > 0$ as well and $p_j \log(p_j/q_j') \geq p_j \log(p_j/q_j) > \tau$.

(M4) *A token with $q_i \geq p_i$ is active for every $\tau > 0$.* Because the term is decreasing, $p_i \log(p_i/q_i) \leq p_i \log(p_i/p_i) = 0 < \tau$.

Step 1: mass on the clipped set prevents optimality. Let $q \in R$ with $Q_C > 0$. Pick any active token $k \in A$ and move all clipped mass onto it:

$$q_k' = q_k + Q_C, \qquad q_j' = 0 \ \ (j \in C), \qquad q_i' = q_i \ \ (i \in A,\ i \neq k).$$

Normalization. $\sum_{i\in\mathcal{S}} q_i' = \sum_{i\in A,\, i\neq k} q_i + (q_k + Q_C) + 0 = \sum_{i\in A} q_i + Q_C = 1$.

Same region. Token k gains probability, so it stays active by (M2). Every clipped token drops to zero, so it stays clipped by (M3). The other active tokens are unchanged. Hence $q' \in R$, and the identity $D_{\text{clip}} = F + |C|\tau$ applies to both q and q'.

Difference. The constants $|C|\tau$ cancel, and so do all active terms with $i \neq k$. What remains is the k-th term:

$$\begin{aligned} D_{\text{clip}}(q') - D_{\text{clip}}(q) &= p_k \log \frac{p_k}{q_k + Q_C} - p_k \log \frac{p_k}{q_k} \\ &= p_k\big[\log q_k - \log(q_k + Q_C)\big] = p_k \log \frac{q_k}{q_k + Q_C}. \end{aligned}$$

Here p_k and q_k are the teacher and student probabilities of the single receiving token k, not the set masses. We have $p_k > 0$, $q_k > 0$ by (M1), and $Q_C > 0$, so $0 < q_k/(q_k + Q_C) < 1$, the logarithm is negative, and

$$D_{\text{clip}}(q') < D_{\text{clip}}(q).$$

Hence every distribution in R with $Q_C > 0$ has strictly larger loss than some distribution in R with $Q_C = 0$.

Step 2: optimal allocation on the active set. Throughout this step $p_i > 0$ for every i, because the teacher is a softmax output, and $q_i > 0$ for every $i \in A$ by (M1). Neither is an extra assumption. Consequently F is finite and smooth wherever it is evaluated, and every Hessian entry p_i/q_i^2 below is strictly positive.

(a) The restricted problem. After Step 1 the competitors are the distributions $q \in R$ with $Q_C = 0$. For such q we have $q_j = 0$ on C, $q_i > 0$ on A, $\sum_{i\in A} q_i = 1$, and $D_{\text{clip}}(q) = F(q_A) + |C|\tau$. So we must minimize F over

$$K_\tau = \Big\{ q_A \in \mathbb{R}^A : \; q_i > 0, \; \sum_{i\in A} q_i = 1, \;\; p_i \log \frac{p_i}{q_i} \le \tau \;\; \forall i \in A \Big\}.$$

Conversely, every $q_A \in K_\tau$, extended by zeros on C, is a distribution in R with $Q_C = 0$.

(b) The relaxed problem. Drop the threshold inequalities and let

$$K = \Big\{ q_A \in \mathbb{R}^A : \; q_i > 0, \; \sum_{i\in A} q_i = 1 \Big\} \;\supseteq\; K_\tau.$$

We minimize the same formula F over K. On $K \setminus K_\tau$ the formula F is no longer the clipped objective, because there some term exceeds τ and would be capped. The relaxed problem is an auxiliary smooth problem. It enters the argument only through the inclusion $K_\tau \subseteq K$, and nothing is claimed about D_{clip} outside R.

(c) Derivatives of F. We differentiate on the positive orthant $\{q_A : \; q_i > 0\}$, which is open, convex, and contains K. Write

$$F(q_A) = \sum_{i\in A} \big[p_i \log p_i - p_i \log q_i \big].$$

Only the i-th summand depends on q_i, and $p_i \log p_i$ is a constant, so

$$\frac{\partial F}{\partial q_i} = \frac{\partial}{\partial q_i} \big[p_i \log p_i - p_i \log q_i \big] = 0 - p_i \cdot \frac{1}{q_i} = -\frac{p_i}{q_i}.$$

For the second derivatives, differentiate $-p_i/q_i = -p_i\, q_i^{-1}$ with respect to q_m. If $m = i$,

$$\frac{\partial^2 F}{\partial q_i^2} = \frac{\partial}{\partial q_i} \big[-p_i\, q_i^{-1} \big] = -p_i \cdot (-1)\, q_i^{-2} = \frac{p_i}{q_i^2}.$$

If $m \neq i$, the expression $-p_i/q_i$ does not depend on q_m, so $\partial^2 F / \partial q_i\, \partial q_m = 0$. The Hessian is therefore diagonal, $\nabla^2 F(q_A) = \operatorname{diag}\big(p_i/q_i^2\big)_{i\in A}$, and for every vector $v \in \mathbb{R}^A$,

$$v^\top \nabla^2 F(q_A)\, v = \sum_{i\in A} \sum_{m\in A} v_i \, \frac{\partial^2 F}{\partial q_i\, \partial q_m} \, v_m = \sum_{i\in A} \frac{p_i}{q_i^2}\, v_i^2.$$

Every coefficient p_i/q_i^2 is strictly positive, because $p_i > 0$ and $q_i > 0$. If $v \neq 0$, some $v_i \neq 0$, so the sum is strictly positive: the Hessian is positive definite at every point of the orthant.

(d) Strict convexity. A function f on a convex set is *strictly convex* if $f(\theta x + (1-\theta)y) < \theta f(x) + (1-\theta) f(y)$ for all $x \neq y$ in the set and all $\theta \in (0,1)$. We use two standard facts. (F1) A twice differentiable function on an open convex set whose Hessian is positive definite at every point is strictly convex. (F2) A differentiable strictly convex function lies strictly above each of its tangent planes: $f(y) > f(x) + \nabla f(x)^\top (y - x)$ for all $x \neq y$. By (c) and (F1), F is strictly convex on the orthant, independently of normalization, and hence on its convex subset K.

(e) Lagrange point. Introduce a multiplier λ for the normalization constraint:

$$\mathcal{L}(q_A, \lambda) = F(q_A) + \lambda \Big(\sum_{m\in A} q_m - 1 \Big).$$

Using (c) and $\frac{\partial}{\partial q_i}\big(\sum_{m\in A} q_m - 1\big) = 1$,

$$\frac{\partial \mathcal{L}}{\partial q_i} = -\frac{p_i}{q_i} + \lambda = 0 \quad\Longrightarrow\quad \lambda = \frac{p_i}{q_i} \quad\Longrightarrow\quad q_i = \frac{p_i}{\lambda} \qquad \text{for every } i \in A.$$

So the ratio p_i/q_i is the same for every active token. Substituting into the normalization constraint $\partial \mathcal{L}/\partial \lambda = \sum_{i\in A} q_i - 1 = 0$,

$$1 = \sum_{i\in A} \frac{p_i}{\lambda} = \frac{1}{\lambda} \sum_{i\in A} p_i = \frac{P_A}{\lambda} \quad\Longrightarrow\quad \lambda = P_A \quad\Longrightarrow\quad q_i^\star = \frac{p_i}{P_A} \qquad (i \in A).$$

This is the only stationary point, since the equations force $q_i = p_i/\lambda$ and then $\lambda = P_A$. It lies in K: $q_i^\star > 0$ and $\sum_{i\in A} q_i^\star = P_A/P_A = 1$.

(f) Global minimality in the relaxed problem. The gradient at the Lagrange point is

$$\frac{\partial F}{\partial q_i}\Big|_{q_A^\star} = -\frac{p_i}{q_i^\star} = -\frac{p_i}{p_i/P_A} = -P_A \quad \text{for every } i \in A, \qquad \text{that is,} \qquad \nabla F(q_A^\star) = -P_A \mathbf{1}.$$

Let $q_A \in K$ with $q_A \neq q_A^\star$. By (F2) with $x = q_A^\star$ and $y = q_A$,

$$F(q_A) > F(q_A^\star) + \nabla F(q_A^\star)^\top (q_A - q_A^\star),$$

and the last term vanishes because both allocations sum to one:

$$\nabla F(q_A^\star)^\top (q_A - q_A^\star) = \sum_{i\in A} (-P_A)(q_i - q_i^\star) = -P_A\Big(\sum_{i\in A} q_i - \sum_{i\in A} q_i^\star\Big) = -P_A(1-1) = 0.$$

Hence $F(q_A) > F(q_A^\star)$ for every $q_A \in K$ other than $q_A^\star$: the Lagrange point is the unique global minimizer of the relaxed problem.

(g) Membership of the candidate. Since $0 < P_A < 1$, we have $q_i^\star = p_i/P_A > p_i$ for every $i \in A$, and

$$p_i \log \frac{p_i}{q_i^\star} = p_i \log \frac{p_i}{p_i/P_A} = p_i \log P_A < 0 < \tau,$$

in agreement with (M4). Every active term lies strictly below the threshold, for every $\tau > 0$, so the candidate cannot cross into the clipped set. Thus $q_A^\star \in K_\tau$, and with $q_j^\star = 0$ on C we get $q^\star \in R$.

(h) Back to the restricted problem. Let $q_A \in K_\tau$ with $q_A \neq q_A^\star$. Since $K_\tau \subseteq K$, part (f) gives $F(q_A) > F(q_A^\star)$, and $q_A^\star \in K_\tau$ by (g). So $q_A^\star$ is also the unique global minimizer of the restricted problem: a minimizer over the larger set that lies in the smaller set is the minimizer over the smaller set. Its value is

$$F(q_A^\star) = \sum_{i\in A} p_i \log \frac{p_i}{p_i/P_A} = \sum_{i\in A} p_i \log P_A = \Big(\sum_{i\in A} p_i\Big) \log P_A = P_A \log P_A.$$

Conclusion. Let $q \in R$ with $q \neq q^\star$. *Case* $Q_C > 0$. Step 1 gives $q' \in R$ with $Q_C = 0$ and $D_{\text{clip}}(q) > D_{\text{clip}}(q')$, and Step 2 gives $D_{\text{clip}}(q') = F(q'_A) + |C|\tau \geq F(q_A^\star) + |C|\tau = D_{\text{clip}}(q^\star)$. So $D_{\text{clip}}(q) > D_{\text{clip}}(q^\star)$. *Case* $Q_C = 0$. Then $q_j = 0 = q_j^\star$ on C, so $q \neq q^\star$ means $q_A \neq q_A^\star$, and Step 2 gives $D_{\text{clip}}(q) = F(q_A) + |C|\tau > F(q_A^\star) + |C|\tau = D_{\text{clip}}(q^\star)$.

In both cases $D_{\text{clip}}(q) > D_{\text{clip}}(q^\star)$. Hence $q^\star$ is the unique minimizer of D_{clip} over R, with $D_{\text{clip}}(q^\star) = P_A \log P_A + |C|\tau$. Existence is not assumed: the minimizer is exhibited. □

Fixed Q_C (Equation 8). Let c be a value of Q_C attained in R. For $q \in R$ with $Q_C = c$, the loss is $F(q_A) + |C|\tau$ with $q_i > 0$ and $\sum_{i\in A} q_i = 1 - c$, and it does not depend on how c is split among the clipped tokens as long as it does not make any one of them active by going over the $d > \tau$ restriction. Repeat Step 2 with the constraint $\sum_{i\in A} q_i = 1 - c$. Stationarity again gives $q_i = p_i/\lambda$, and now

$$1 - c = \sum_{i\in A} \frac{p_i}{\lambda} = \frac{P_A}{\lambda} \quad \Longrightarrow \quad \lambda = \frac{P_A}{1-c} \quad \Longrightarrow \quad q_i^\star(c) = \frac{1-c}{P_A}\, p_i.$$

The gradient there is $-p_i/q_i^\star(c) = -P_A/(1-c)$ in every coordinate, so for any other positive allocation with the same sum, $\nabla F(q_A^\star(c))^\top (q_A - q_A^\star(c)) = -\frac{P_A}{1-c}\big((1-c) - (1-c)\big) = 0$, and (F2) gives $F(q_A) > F(q_A^\star(c))$: this is the unique global minimizer of the relaxed problem at level c.

Membership. Every clipped token has a term above $\tau > 0$, so $\log(p_j/q_j) > 0$ and $q_j < p_j$. Summing over C gives $c < \sum_{j\in C} p_j = 1 - P_A$, hence $(1-c)/P_A > 1$ and $q_i^\star(c) > p_i$. By (M4) every active term is below τ, so the relaxed minimizer satisfies the threshold inequalities and is also the minimizer of the restricted problem at level c. Its loss is

$$\begin{aligned} D_{\text{clip}}^\star(c) &= \sum_{i\in A} p_i \log \frac{p_i}{(1-c)\, p_i/P_A} + |C|\tau \\ &= P_A \log \frac{P_A}{1-c} + |C|\tau = P_A \log P_A - P_A \log(1-c) + |C|\tau, \end{aligned}$$

Table 4: Training Configuration.

Component	Executed value
Base checkpoint	`Qwen/Qwen3-4B`
Optimizer	`torch.optim.AdamW`; learning rate 10^{-6}; $\beta = (0.9, 0.999)$; $\epsilon = 10^{-8}$; weight decay 0.01
Schedule	Zero warmup. 10^{-6} constant.
Updates and checkpoints	200 updates; a checkpoint is saved every 25 updates
Seed	Training and data-loader seeds are 42
Global update	30 problems, one rollout each
Prompt length cap	2,048 tokens; the response caps are given in Table 1
Training rollout sampler	Temperature 1.1, top-$p = 0.8$, top-$k = 20$
Teacher sampler	The teacher returns its top-128 token log probabilities at temperature 1.0
Loss	Forward KL on the teacher's top-128 tokens, both distributions renormalized on that support at temperature $T = 1.1$; clipped runs use $\tau = 0.05$, unclipped runs apply no clip
Hardware	Four H200 GPUs per run: three for the student's training and rollouts, one for the dedicated SGLang teacher
Software	Transformers 4.57.1; PyTorch 2.9.1+cu128; Ray 2.54.1; SGLang 0.5.9; CUDA toolkit 12.8.2

and

$$\frac{dD^{\star}_{\text{clip}}}{dc} = -P_A \cdot \frac{-1}{1-c} = \frac{P_A}{1-c} > 0.$$

The level-c minimizer is unique in its active coordinates only.

Remarks. *(i) F is not a KL divergence.* Its arguments p_A and q_A are sub-probability vectors, and F can be negative: $F(q^{\star}_A) = P_A \log P_A < 0$. The proof uses only the strict convexity of F.

(ii) Scope. The statement concerns one fixed clipping region; no comparison across clipping patterns is claimed. A softmax over finite logits assigns every token positive probability, so such a student never equals $q^{\star}$; by the fixed-Q_C statement the optimal loss at level c decreases to $D_{\text{clip}}(q^{\star})$ as $c \downarrow 0$, so $q^{\star}$ is approached but not attained.

C Training and Evaluation Configuration, Capture

Training configuration In all eight runs, the student and the teacher are initialized from the same Qwen3-4B checkpoint. Each clipped/unclipped pair receives the same per-update problems. Table 4 lists the detailed training configuration used in our experiment.

Message templates. We give the user messages and the chat template used to serialize the student and teacher prompts. The student's user message is identical across all eight runs. Its serialized prompt differs only by the empty thinking block described below. Before chat serialization, the student's single user message is:

```
Problem: {problem}

Please reason step by step, and put your final answer within
\boxed{}.
```

The teacher's single user message is:

```
Problem: {problem} Here is a reference solution to this
problem: === Reference Solution Begin === {answer} ===
Reference Solution End === After reading the reference
solution above, make sure you truly understand the reasoning
behind each step - do not copy or paraphrase it. Now, using
```

```
your own words and independent reasoning, derive the same
final answer to the problem above.  Please reason step by
step, and put your final answer within \boxed{}.
```

The placeholder {answer} is filled with the full reference solution from the dataset, not the final answer alone. Qwen serializes the one-message prompt as

```
<|im_start|>user
{content}<|im_end|>
<|im_start|>assistant
```

with no system or tool message. When thinking is disabled, the serialization additionally appends an empty <think> block followed by </think> and a blank line. The student's sampled response token IDs are appended to the teacher prompt before teacher scoring.

Capture setting. At every response position of all 200 updates of all eight runs, the trainer stores the token the student emitted, the IDs of the teacher's top-128 candidate tokens, the teacher's log probabilities on those tokens, and the student's log probabilities on the same tokens, gathered from its full-vocabulary log-softmax in the forward pass that computes the loss.

D Complete AIME2024 and AIME2025 Evaluation

Evaluation configuration. Checkpoints 25, 50, 75, 100, 125, 150, 175, and 200 are evaluated on AIME 2024 and AIME 2025. Each benchmark contains 30 problems.

Each saved checkpoint is evaluated with thinking enabled, drawing 12 samples per problem at temperature .6, top-$p = .95$, top-$k = 20$, and minimum-$p = 0$, with a 38,912-token output budget. Scoring uses only the text after the last closing thinking tag: Math-Verify (Kydlíček) compares it with the reference answer, and if that comparison fails, the last \boxed{} expression is compared with the reference answer as a string or a number. We evaluate the same base-model checkpoint eight times for both benchmarks to measure variation from response sampling.

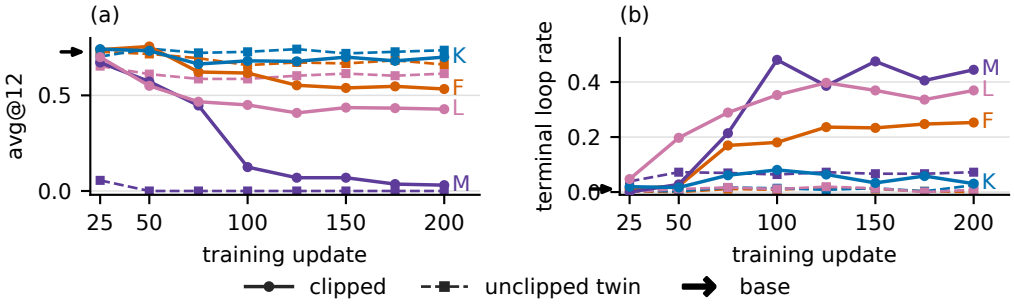


Figure 5: AIME 2024 across training, in the layout of Figure 3. (a) Avg@12, the mean correct ratio over 12 samples for each of 30 problems at the checkpoint saved after each 25 updates. (b) The ratio of the 360 responses per checkpoint that end in a terminal periodic loop. Solid curves with circles are the clipped runs, dashed curves with squares their unclipped twins. Each trained curve is one realized trajectory. The black arrow on each vertical axis marks the base model, averaged over eight evaluation-sampling seeds. The values appear in Table 6.

Figure 5 visualize AIME 2024 evaluation result: (a) avg@12 and, (b) terminal loop rate.

Table 6 and 5 shows the numerical value of per checkpoint evaluated on AIME 2024 and AIME 2025, respectively: (a) avg@12, (b) terminal loop rate and (c) pass@12.

Table 5: Complete AIME 2025 evaluation at every checkpoint (columns: training update). (a) Avg@12, the mean correct ratio over 12 samples for each of 30 problems. (b) Terminal loop rate, the ratio of the 360 responses that end in a terminal periodic loop. (c) Pass@12, the ratio of the 30 problems solved by at least one of the 12 samples. Each trained row is one realized trajectory. The base row gives the mean and, in parentheses, the standard deviation over eight evaluation-sampling seeds of the fixed starting checkpoint.

(a) avg@12

Run	25	50	75	100	125	150	175	200
F	.661	.653	.589	.550	.581	.525	.531	.544
F-no-clip	.622	.644	.578	.544	.619	.561	.575	.572
M	.592	.481	.378	.122	.050	.067	.031	.022
M-no-clip	.047	.000	.000	.000	.000	.000	.000	.000
K	.664	.653	.647	.619	.622	.550	.628	.600
K-no-clip	.647	.642	.633	.611	.644	.642	.608	.608
L	.636	.525	.533	.486	.469	.439	.469	.458
L-no-clip	.581	.531	.481	.508	.489	.506	.508	.517
Base	.645 (.010)							

(b) terminal loop rate

Run	25	50	75	100	125	150	175	200
F	.008	.033	.128	.161	.153	.203	.194	.192
F-no-clip	.011	.003	.008	.019	.000	.006	.006	.008
M	.006	.036	.192	.494	.389	.422	.400	.400
M-no-clip	.050	.039	.036	.044	.031	.053	.042	.017
K	.014	.033	.036	.050	.058	.078	.086	.061
K-no-clip	.011	.011	.006	.006	.000	.003	.008	.011
L	.047	.144	.211	.231	.256	.328	.267	.303
L-no-clip	.003	.014	.019	.006	.011	.006	.022	.011
Base	.006 (.003)							

(c) pass@12

Run	25	50	75	100	125	150	175	200
F	.800	.800	.867	.800	.833	.833	.867	.800
F-no-clip	.867	.800	.800	.733	.833	.833	.800	.800
M	.867	.700	.667	.367	.233	.267	.167	.133
M-no-clip	.133	.000	.000	.000	.000	.000	.000	.000
K	.833	.800	.833	.833	.867	.800	.833	.833
K-no-clip	.833	.800	.833	.767	.867	.833	.767	.800
L	.833	.800	.767	.767	.833	.767	.767	.767
L-no-clip	.767	.767	.733	.767	.767	.767	.700	.767
Base	.817 (.018)							

Table 6: Complete AIME 2024 evaluation at every checkpoint (columns: training update). (a) Avg@12, the mean correct ratio over 12 samples for each of 30 problems. (b) Terminal loop rate, the ratio of the 360 responses that end in a terminal periodic loop. (c) Pass@12, the ratio of the 30 problems solved by at least one of the 12 samples. Each trained row is one realized trajectory. The base row gives the mean and, in parentheses, the standard deviation over eight evaluation-sampling seeds of the fixed starting checkpoint.

(a) avg@12

Run	25	50	75	100	125	150	175	200
F	.739	.756	.622	.617	.553	.539	.547	.533
F-no-clip	.728	.717	.694	.658	.672	.667	.683	.661
M	.672	.572	.447	.125	.069	.069	.036	.031
M-no-clip	.056	.000	.000	.000	.000	.000	.000	.000
K	.742	.733	.664	.681	.678	.700	.681	.700
K-no-clip	.703	.744	.722	.728	.742	.719	.728	.736
L	.700	.550	.467	.450	.408	.436	.433	.428
L-no-clip	.653	.611	.586	.586	.603	.614	.603	.614
Base	.727 (.013)							

(b) terminal loop rate

Run	25	50	75	100	125	150	175	200
F	.011	.022	.169	.181	.236	.233	.247	.253
F-no-clip	.003	.000	.011	.008	.017	.011	.003	.000
M	.000	.028	.214	.481	.386	.475	.406	.444
M-no-clip	.039	.072	.069	.064	.072	.067	.067	.072
K	.019	.017	.061	.081	.064	.033	.058	.031
K-no-clip	.006	.003	.017	.014	.008	.014	.003	.025
L	.047	.197	.289	.353	.397	.369	.336	.369
L-no-clip	.000	.011	.017	.011	.019	.014	.000	.008
Base	.011 (.003)							

(c) pass@12

Run	25	50	75	100	125	150	175	200
F	.867	.833	.900	.867	.900	.867	.867	.833
F-no-clip	.833	.833	.833	.867	.800	.833	.800	.833
M	.833	.800	.700	.467	.333	.333	.167	.100
M-no-clip	.233	.000	.000	.000	.000	.000	.000	.000
K	.833	.867	.833	.833	.867	.867	.833	.867
K-no-clip	.833	.833	.867	.833	.867	.833	.833	.833
L	.867	.833	.800	.767	.767	.800	.833	.833
L-no-clip	.833	.833	.800	.800	.800	.800	.800	.800
Base	.838 (.033)							