WHERE SHOULD SUPPRESSED PROBABILITY GO? ONE-STEP LOOKAHEAD KL ROUTING FOR ON-POLICY DISTILLATION

Anonymous authors
Paper under double-blind review

ABSTRACT

On-policy distillation (OPD) supervises a student on its own trajectories, but its sampled-token form leaves the recipients of negative credit implicit. When a sampled token is suppressed, the compensating logit update follows the student's current probabilities rather than explicitly selecting recipients using teacher evidence, even among alternatives that both models rank highly. We formulate this limitation as *recipient routing* and introduce Lookahead KL Routing (LKR). LKR first defines a dense routing operator. It combines current teacher support with agreement between the student and teacher after appending each candidate, then solves a constrained reverse-KL projection to reallocate a bounded portion of the update within this shared candidate set. Its unique water-filling target defines a routing correction that preserves the sampled token's local logit gradient. Dense routing would otherwise require successor evaluation at every eligible event. LKR makes training practical by sampling a small subset of events and adapting the projection budget to keep each routing update bounded. Across six public mathematical reasoning benchmarks, LKR improves step-matched OPD by 2.23 and 4.54 percentage points in Mean@8 and Cons@8, respectively. The gains persist across three seeds on a 1,000-problem held-out set with a 1.5B student and a 7B teacher. The pattern holds with a 7B student and a 32B teacher.

1 INTRODUCTION

Reasoning distillation requires feedback at the decisions the student actually encounters. Reinforcement learning with verifiable rewards trains on student-generated trajectories, but terminal correctness does not identify which intermediate decisions need correction (Uesato et al., 2022; Lightman et al., 2024; DeepSeek-AI, 2025). Offline distillation supplies dense targets on fixed or teacher-generated histories that may differ from the student's own (Kim & Rush, 2016). **On-policy distillation** (OPD) brings teacher supervision to student-generated histories (Agarwal et al., 2024; Gu et al., 2024). Its sampled-token form scores the tokens the student actually emits, making it a lightweight approach to reasoning transfer (Lu & Thinking Machines Lab, 2025; Song & Zheng, 2026). This raises a basic question: *when the teacher rejects a token, what should the student learn to choose instead?*

OPD can tell the student what *not* to do, but it does not tell the student what to do *instead*. Concretely, when the student assigns a sampled token higher probability than the teacher, a **negative-credit event** suppresses that token (Lu & Thinking Machines Lab, 2025). The compensating logit update is distributed among all remaining tokens in proportion to the student's current probabilities, so a low-probability alternative receives little of this update and is rarely sampled for its own feedback. The teacher controls *what to suppress*; the student's existing preferences control *what to promote*. Deciding where the compensating update should go is the **recipient-routing problem** (Fu et al., 2026; Zhang, 2026). This gap motivates an explicit routing mechanism for improving both response and consensus accuracy.

Recent methods broaden token-level supervision or use successor evidence and pairwise preferences to guide alternatives (Fu et al., 2026; Jia et al., 2026; Wang et al., 2026; Liu et al., 2026; Jiang & Ferraro, 2026; Zhang, 2026). They leave open how current teacher preference and candidate-induced

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.

successor states should jointly allocate a bounded negative-update budget while holding the base donor component fixed.

We introduce **Lookahead KL Routing** (LKR). LKR first constructs a dense routing target by decomposing the sampled-token gradient into a donor budget and allocating a bounded portion to candidate alternatives. Each candidate is scored by two signals: its current teacher probability and the student–teacher KL divergence at the prefix obtained by appending it. A constrained reverse-KL projection combines these signals into a unique water-filling allocation, which assigns increments according to the scores and existing student probabilities within the candidate set. A difference of fixed-target distillation losses converts this allocation into a gradient correction that preserves the base donor coordinate. Figure 1 illustrates this construction.

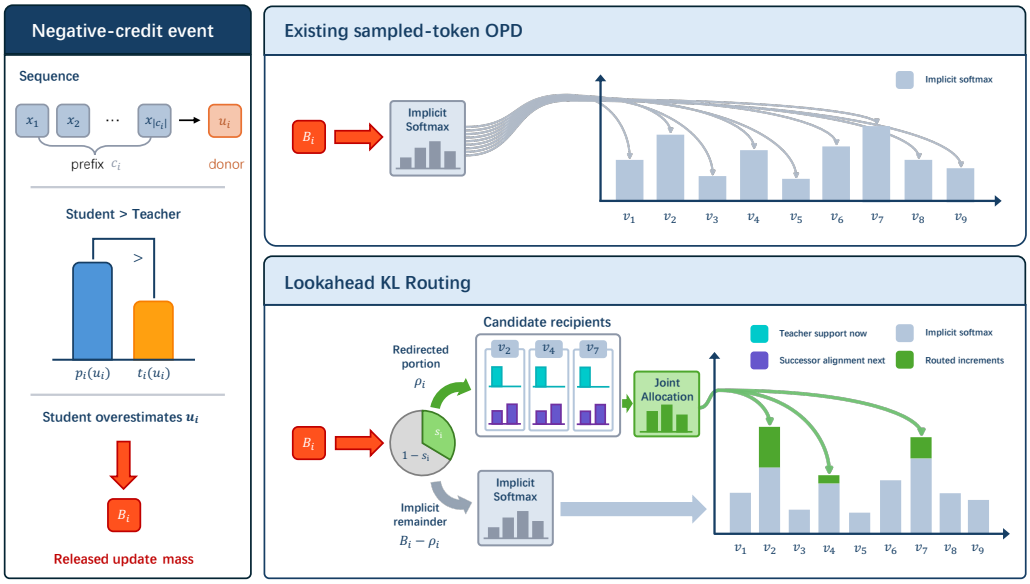

Figure 1: **From implicit redistribution to lookahead routing. Left:** a negative-credit event creates a compensating non-donor update. **Top:** OPD distributes this update implicitly according to the student's softmax. **Bottom:** the dense LKR operator uses current teacher support and candidate-induced successor agreement to redirect a bounded portion jointly; the remainder follows the original implicit allocation. Blue-gray denotes implicit increments, and green denotes LKR-routed increments.

Our contributions are as follows:

- We formulate **donor-preserving recipient routing** and derive a unique optimal auxiliary allocation from a constrained reverse-KL projection. A difference of fixed-target distillation losses connects its water-filling solution to a donor-preserving logit correction (Propositions 1 and 2).
- We design a **training pipeline** that estimates the dense budget-dependent routing target through event sampling. Clipping the routing budget before projection preserves a bounded recipient mixture in every sample and yields a conditionally unbiased target estimator (Proposition 3; Section 3.3).
- We demonstrate **empirical gains**. At the primary 1.5B/7B scale, LKR improves OPD across three training seeds, with 2.41/2.27-point gains in held-out Mean@8/Cons@8 and 2.23/4.54-point gains on the public macro-average. A single-seed 7B/32B comparison provides supporting evidence at a larger scale.
- We provide **controlled ablations** that probe the effects of teacher support, successor agreement, water-filling, and bounded redirection.

Code: `https://anonymous.4open.science/r/lkr-openreview-DE6A`.

2 PRELIMINARIES AND THE ROUTING GAP

2.1 SAMPLED-TOKEN REVERSE-KL OPD

Let π_θ and π_T denote the student and teacher policies. At position i of a student-generated trajectory, let c_i be the prefix and $u_i \sim \pi_\theta(\cdot \mid c_i)$ the sampled token. We write

$$p_i = \pi_\theta(\cdot \mid c_i), \qquad t_i = \pi_T(\cdot \mid c_i).$$

Both are distributions over a shared vocabulary $\mathcal{V}$, parameterized through finite-dimensional softmax logits. Hereafter, OPD refers to the sampled-token objective unless stated otherwise. It uses the detached credit

$$A_i = \log \frac{t_i(u_i)}{p_i(u_i)}. \tag{1}$$

The local loss $\ell_i^{\text{base}} = -\operatorname{sg}(A_i) \log p_i(u_i)$ suppresses u_i when $A_i < 0$. We call this a *negative-credit event* and the sampled token its *donor*.

Let z_i be the student logits at prefix c_i, with $p_i = \operatorname{softmax}(z_i)$, and let e_{u_i} be the donor one-hot vector. Including a nonnegative scalar weight, the local loss gradient at a negative-credit event has the form

$$G_i^{\text{base}} := \nabla_{z_i} \ell_i^{\text{base}} = \gamma_i (e_{u_i} - p_i), \qquad \gamma_i \geq 0, \tag{2}$$

where $\gamma_i = -A_i$ for the unweighted loss. Importance weighting or clipping can rescale this coefficient in policy-gradient implementations (Williams, 1992; Schulman et al., 2017; Lu & Thinking Machines Lab, 2025). The donor coordinate has magnitude $B_i := \gamma_i(1 - p_i(u_i))$, equal to the total magnitude assigned to non-donors. We call B_i the *base routing budget*. When $\gamma_i > 0$, the local descent direction lowers the donor logit and raises every other logit in proportion to its student probability. All budgets and preservation statements below concern this local logit gradient.

2.2 IMPLICIT RECIPIENTS AND THE ROUTING GAP

To see where the base budget goes, normalize the student probabilities over non-donor tokens:

$$b_i(v) = \begin{cases} p_i(v)/(1 - p_i(u_i)), & v \neq u_i, \\ 0, & v = u_i \end{cases}. \tag{3}$$

Because $b_i(u_i) = 0$ and $\sum_v b_i(v) = 1$, b_i is the student distribution conditioned on not choosing the donor. Equation 2 then factorizes exactly as

$$G_i^{\text{base}} = B_i(e_{u_i} - b_i). \tag{4}$$

The *implicit recipient distribution* b_i fixes the budget's relative shares. Thus softmax already supplies a routing rule, with the student's own probabilities determining where the update goes.

This rule assigns little weight to alternatives with low student probability, even when the teacher favors them. The same attenuation appears in the full-vocabulary reverse-KL gradient. At a fixed prefix, let $p = \operatorname{softmax}(z)$ and let t be fixed. Then

$$\frac{\partial D_{\text{KL}}(p\|t)}{\partial z_v} = p(v) \left(\log \frac{p(v)}{t(v)} - D_{\text{KL}}(p\|t) \right). \tag{5}$$

For fixed t, this gradient vanishes as $p(v) \to 0$. In other words, the student's current distribution acts as a gatekeeper: alternatives to which it assigns little probability receive vanishing gradient regardless of teacher preference. Under sampled-token feedback, v receives its own credit signal only when sampled, which also becomes rare as $p(v)$ shrinks. Appendix B.1 derives the gradient identity.

The *routing gap* is therefore a choice of shares: the teacher supplies the scalar correction, while b_i alone determines its destinations. Dense LKR replaces b_i with an explicit recipient distribution a_i derived from teacher support and successor evidence, redirecting a bounded portion of B_i accordingly.

3 LOOKAHEAD KL ROUTING

When OPD suppresses a sampled token, the student's current probabilities decide which other tokens benefit. LKR keeps that suppression but redirects part of the compensating update within the shared candidate set. It favors candidates supported by the teacher whose successors better align the two models, while accounting for how much probability the student already assigns them. We first set a budget for this transfer, then score and allocate it; the budget also keeps each sampled update bounded. All routing quantities are held fixed during local differentiation.

3.1 SUCCESSOR-AWARE RECIPIENT ALLOCATION

Let $S_i \subseteq \mathcal{V} \setminus \{u_i\}$ be a nonempty candidate set. Our implementation first intersects the teacher's and student's top-20 lists and removes the donor, then retains up to k candidates by teacher rank. We let $q_i > 0$ denote the event inclusion probability (Section 3.3) and set the clipped routing budget to

$$\delta_i = \alpha[p_i(u_i) - t_i(u_i)]_+, \qquad \rho_i = \min\{q_i B_i, \delta_i\}, \qquad s_i = \rho_i / B_i, \tag{6}$$

where $0 < \alpha < 1$, with $s_i = 0$ when $B_i = 0$. Here δ_i is the desired surplus-based dose, ρ_i is the amount permitted by the query budget, and s_i is its fraction of the base budget. The cap $\rho_i \le q_i B_i$ ensures $s_i/q_i \le 1$ for each sampled event. We solve the projection at ρ_i.

For $v \in S_i$, appending v gives the successor prefix $c_i \oplus v$. Define

$$d_i(v) = D_{\mathrm{KL}}(\pi_\theta(\cdot \mid c_i \oplus v) \,\|\, \pi_T(\cdot \mid c_i \oplus v)), \qquad g_i(v) = t_i(v) e^{-\beta d_i(v)}, \tag{7}$$

where $\beta \ge 0$ controls lookahead strength. The score $g_i(v)$ rewards current teacher support and discounts candidates that lead to greater successor disagreement.

To derive the shares, introduce an auxiliary distribution $\widetilde{p}_i$ that removes ρ_i from the donor, adds $y(v) \ge 0$ to each candidate, and leaves other coordinates fixed. This distribution is a device for choosing recipients, not the student's post-update distribution. With $\sum_{v \in S_i} y(v) = \rho_i$, it is normalized and nonnegative because $\rho_i \le \delta_i < p_i(u_i)$. Holding the successor conditional distributions fixed, the two-token reverse KL has the following candidate-dependent objective at $\beta = 1$; varying β weights its successor term:

$$\sum_{v \in S_i} \widetilde{p}_i(v) \log \frac{\widetilde{p}_i(v)}{t_i(v)} + \beta \sum_{v \in S_i} \widetilde{p}_i(v) d_i(v) = \sum_{v \in S_i} \widetilde{p}_i(v) \log \frac{\widetilde{p}_i(v)}{g_i(v)}. \tag{8}$$

The KL chain-rule derivation is given in Appendix B.2. Substituting $\widetilde{p}_i(v) = p_i(v) + y(v)$ gives the allocation problem

$$y_i^\star = \arg \min_{\substack{y(v) \ge 0 \\ \sum_{v \in S_i} y(v) = \rho_i}} \sum_{v \in S_i} \big(p_i(v) + y(v)\big) \log \frac{p_i(v) + y(v)}{g_i(v)}. \tag{9}$$

This constrained information projection (Csiszár, 1975; Bregman, 1967) determines the recipient shares. These shares define how the logit gradient is redirected in the next section.

Proposition 1 (Water-filling allocation)**.** *For nonempty S_i, $g_i(v) > 0$ on S_i, and $\rho_i > 0$, Problem 9 has the unique solution*

$$y_i^\star(v) = [\kappa_i g_i(v) - p_i(v)]_+, \qquad \sum_{v \in S_i} y_i^\star(v) = \rho_i, \tag{10}$$

where $\kappa_i > 0$ is uniquely determined by the budget constraint.

The water-filling threshold allocates new mass where teacher and successor evidence is strong relative to existing student mass: a candidate already favored by the student needs less of the added budget. Precisely, a candidate receives mass when $\kappa_i g_i(v) > p_i(v)$. We normalize the increments to obtain

$$a_i(v) = y_i^\star(v)/\rho_i, \qquad \sum_{v \in S_i} a_i(v) = 1, \tag{11}$$

with $a_i(v) = 0$ outside S_i. An event with $\rho_i = 0$ contributes no routing residual. Setting $\beta = 0$ reduces LKR to a current-prefix teacher projection. The KKT proof and a scalar threshold solver appear in Appendix B.3, following the classical water-filling structure (Cover & Thomas, 2006).

For each candidate, successor KL uses singleton bins for the intersection of the student's and teacher's successor top-20 lists, plus a tail bin for all remaining tokens. This partition is distinct from the current-prefix candidate set S_i. The coarse KL retains the projection form (Appendices B.7 and D.2).

3.2 DONOR-PRESERVING ROUTING

Dense LKR uses the optimal auxiliary allocation to replace a fraction s_i of the implicit recipient distribution b_i with a_i, giving the local gradient

$$G_i^{\mathrm{LKR}} = B_i(e_{u_i} - b_i) + \rho_i(b_i - a_i) \tag{12}$$

$$= B_i \left[e_{u_i} - (1 - s_i)b_i - s_i a_i\right]. \tag{13}$$

A difference of fixed-target KL losses converts the optimal auxiliary allocation into the exact logit correction $\rho_i(b_i - a_i)$, retaining the base shares for the remaining budget (Appendix B.6).

Proposition 2 (Donor preservation)**.** *For every routed event, the LKR residual $\rho_i(b_i - a_i)$ has zero donor coordinate and zero row sum. Consequently,*

$$[G_i^{\mathrm{LKR}}]_{u_i} = [G_i^{\mathrm{base}}]_{u_i} = B_i, \qquad \mathbf{1}^\top G_i^{\mathrm{LKR}} = 0. \tag{14}$$

This preserves the donor gradient with respect to local logits, before parameter sharing and optimizer processing; it does not fix the donor probability change after an update.

The gradient is realized by adding a directed candidate-loss correction to the base OPD loss, with routing quantities detached. Appendix B.5 gives the exact loss and derivation. Sampling uses coefficient $s_i/q_i \leq 1$.

3.3 SQ-LKR: BUDGETED LOOKAHEAD

SQ-LKR limits successor queries by sampling eligible events and retaining the base objective at every position. Let $\mathcal{I}$ contain all rollout positions and $\mathcal{E} \subseteq \mathcal{I}$ the eligible events. For $i \in \mathcal{E}$, let I_i indicate inclusion and $q_i = \Pr(I_i = 1) > 0$. Inverse-probability weighting is applied only to the routing residual:

$$\widehat{G}^{\mathrm{SQ}} = \sum_{i \in \mathcal{I}} G_i^{\mathrm{base}} + \sum_{i \in \mathcal{E}} \frac{I_i}{q_i} \rho_i(b_i - a_i). \tag{15}$$

Proposition 3 (Bounded sampling and conditional unbiasedness)**.** *For $\rho_i = \min\{q_i B_i, \delta_i\}$ and shares computed at this budget, each sampled event has recipient mixture $(1 - \tau_i)b_i + \tau_i a_i$, where $\tau_i = I_i s_i / q_i \in [0, 1]$. Its donor gradient remains B_i and its row sum is zero. Conditioned on the rollout and potential routing annotations,*

$$\mathbb{E}_I[\widehat{G}^{\mathrm{SQ}}] = \sum_{i \in \mathcal{I}} G_i^{\mathrm{base}} + \sum_{i \in \mathcal{E}} \rho_i(b_i - a_i). \tag{16}$$

The expectation is the budget-dependent projected target defined above. The result follows from $s_i \leq q_i$ and $\mathbb{E}[I_i/q_i] = 1$ (Horvitz & Thompson, 1952), before global gradient clipping and optimizer processing. We sample up to m events uniformly without replacement, so $q_i = \min\{1, m/|\mathcal{E}|\}$ for $m > 0$. The query budget controls both the projection budget and sampling variability. Appendix C gives the proof and covariance; Algorithm 1 specifies training.

4 EXPERIMENTS

We evaluate effectiveness against OPD baselines (RQ1), the components of recipient routing (RQ2), and transfer to a larger model pair (RQ3).

4.1 EXPERIMENTAL SETUP

Models and data. We distill the publicly released DeepSeek-R1-Distill-Qwen-1.5B student (DeepSeek-AI, 2025) from the publicly released Skywork-OR1-Math-7B teacher (He et al., 2025). We train on 47,909 prompts derived from POLARIS-53K (An et al., 2025), after filtering and held-out removal; no matches to the six public benchmarks remain (Appendix E.1). On-policy arms share the student initialization, prompt order, rollout budget, and checkpoint rule.

Evaluation. The main evaluation covers the 1,000-problem held-out POLARIS split and six public benchmarks: AIME 2024, AIME 2025, AIME 2026, MATH-500, AMC 2023, and HMMT February 2025 (Hendrycks et al., 2021; Lightman et al., 2024; Dekoninck et al., 2026). We sample eight independent responses per problem using the same decoding configuration. Mean@8 is the average response accuracy. Cons@8 selects the most frequent normalized final answer and scores it against the reference. Appendix E.2 gives the normalization, tie-breaking, and aggregation details. We treat the 1,000-problem Held-out split as the primary comparison. Public Avg. is the unweighted macro-average over the six public benchmarks and excludes Held-out. We interpret it as a secondary aggregate because it gives equal weight to several small benchmarks. For checkpoint-wise Held-out comparisons, Figure 2 reports pointwise 95% confidence intervals for paired OPD–LKR gains in both metrics.

Baselines. Reference rows include the teacher and the student initialization. Training baselines include SFT on teacher traces (SeqKD), OPD, top-K OPD (truncated reverse-KL matching), EOPD (Jin et al., 2026), and AOPD (Jia et al., 2026). We compare with LGR as the closest successor-aware method (Liu et al., 2026). Appendix E.6 specifies their implemented objectives and settings.

Implementation. On-policy methods use 250 steps, a global batch of 128 prompts, and learning rate 10^{-5}. SQ-LKR defaults are $m = 20$ events per rollout, up to $k = 4$ candidates, $\alpha = 0.25$, and $\beta = 1$. All LKR runs cap the sampled coefficient at one and compute shares at the clipped dose. Appendices E.2 and E.5 detail the on-policy and separate SFT protocols.

4.2 Overall effectiveness (RQ1)

Table 1: **Main reasoning results.** Trained methods report three-seed means (%) with sample standard deviations as subscripts. Bold marks the best trained result, including ties. On-policy arms match data, rollouts, and step-250 checkpoints; SFT uses three epochs (final step 240).

Method	Held-out	AIME24	AIME25	AIME26	MATH-500	AMC23	HMMT25	Public Avg.
Mean@8 (%)								
Teacher	64.28	64.17	52.50	62.92	95.10	88.10	32.08	65.81
Student init.	20.21	29.58	23.33	20.42	84.20	64.76	13.75	39.34
SFT	$18.95_{\pm 0.31}$	$27.36_{\pm 0.48}$	$25.14_{\pm 0.64}$	$23.06_{\pm 0.64}$	$84.57_{\pm 1.29}$	$65.71_{\pm 0.53}$	$13.89_{\pm 0.64}$	$39.95_{\pm 0.28}$
OPD	$23.46_{\pm 0.56}$	$36.53_{\pm 0.64}$	$28.75_{\pm 0.42}$	$24.72_{\pm 0.64}$	$87.56_{\pm 0.81}$	$68.52_{\pm 0.45}$	$19.72_{\pm 0.64}$	$44.30_{\pm 0.56}$
Top-K OPD	$24.75_{\pm 0.50}$	$37.36_{\pm 0.64}$	$27.08_{\pm 0.83}$	$25.69_{\pm 0.24}$	$87.46_{\pm 0.79}$	$67.57_{\pm 0.23}$	$\mathbf{20.28}_{\pm 0.48}$	$44.24_{\pm 0.31}$
AOPD	$24.79_{\pm 0.49}$	$36.39_{\pm 0.48}$	$28.89_{\pm 0.64}$	$26.94_{\pm 0.24}$	$87.52_{\pm 0.93}$	$\mathbf{72.24}_{\pm 0.38}$	$18.47_{\pm 0.24}$	$45.08_{\pm 0.34}$
EOPD	$24.74_{\pm 0.49}$	$37.64_{\pm 0.24}$	$30.14_{\pm 0.24}$	$24.03_{\pm 0.24}$	$87.56_{\pm 0.87}$	$68.72_{\pm 0.17}$	$16.53_{\pm 0.48}$	$44.10_{\pm 0.06}$
LGR	$23.39_{\pm 0.43}$	$\mathbf{39.44}_{\pm 0.64}$	$30.28_{\pm 0.24}$	$24.44_{\pm 0.48}$	$87.74_{\pm 0.64}$	$68.27_{\pm 0.09}$	$20.14_{\pm 0.24}$	$45.05_{\pm 0.14}$
LKR (ours)	$\mathbf{25.87}_{\pm 0.56}$	$39.03_{\pm 0.24}$	$\mathbf{30.83}_{\pm 0.42}$	$\mathbf{29.03}_{\pm 0.24}$	$\mathbf{88.43}_{\pm 0.28}$	$71.74_{\pm 0.53}$	$20.14_{\pm 0.64}$	$\mathbf{46.53}_{\pm 0.31}$
Cons@8 (%)								
Teacher	77.00	83.33	56.67	73.33	96.60	90.36	43.33	73.94
Student init.	28.60	56.67	33.33	30.00	90.40	75.90	20.00	51.05
SFT	$33.53_{\pm 0.25}$	$43.33_{\pm 3.33}$	$32.22_{\pm 1.92}$	$38.89_{\pm 1.92}$	$90.87_{\pm 0.31}$	$80.72_{\pm 1.20}$	$17.78_{\pm 1.92}$	$50.64_{\pm 1.65}$
OPD	$34.13_{\pm 0.21}$	$62.22_{\pm 1.92}$	$34.44_{\pm 1.92}$	$31.11_{\pm 1.92}$	$91.27_{\pm 0.12}$	$80.32_{\pm 1.39}$	$21.11_{\pm 1.92}$	$53.41_{\pm 0.55}$
Top-K OPD	$34.97_{\pm 0.12}$	$57.78_{\pm 1.92}$	$35.56_{\pm 1.92}$	$32.22_{\pm 1.92}$	$91.87_{\pm 0.12}$	$76.31_{\pm 0.70}$	$22.22_{\pm 1.92}$	$52.66_{\pm 0.43}$
AOPD	$32.37_{\pm 0.25}$	$47.78_{\pm 1.92}$	$34.44_{\pm 1.92}$	$40.00_{\pm 3.33}$	$\mathbf{92.27}_{\pm 0.31}$	$79.12_{\pm 1.84}$	$22.22_{\pm 1.92}$	$52.64_{\pm 1.11}$
EOPD	$35.13_{\pm 0.35}$	$52.22_{\pm 1.92}$	$36.67_{\pm 3.33}$	$36.67_{\pm 3.33}$	$91.53_{\pm 0.31}$	$79.92_{\pm 0.70}$	$22.22_{\pm 1.92}$	$53.21_{\pm 1.26}$
LGR	$35.23_{\pm 0.06}$	$63.33_{\pm 3.33}$	$35.56_{\pm 1.92}$	$38.89_{\pm 1.92}$	$\mathbf{92.27}_{\pm 0.12}$	$76.71_{\pm 0.70}$	$23.33_{\pm 3.33}$	$55.01_{\pm 0.38}$
LKR (ours)	$\mathbf{36.40}_{\pm 0.17}$	$\mathbf{68.89}_{\pm 1.92}$	$\mathbf{37.78}_{\pm 1.92}$	$\mathbf{42.22}_{\pm 1.92}$	$92.07_{\pm 0.31}$	$\mathbf{82.33}_{\pm 0.70}$	$\mathbf{24.44}_{\pm 1.92}$	$\mathbf{57.95}_{\pm 0.71}$

LKR improves OPD by 2.41/2.27 points in Held-out Mean@8/Cons@8 and 2.23/4.54 points on Public Avg. (Table 1). It has the best public macro-average among trained methods and is numerically above OPD on both metrics across all six public benchmarks.

Extending OPD to 450 steps approximates LKR's estimated training-time budget; LKR at step 250 still gains 1.23/1.43 points in Held-out Mean@8/Cons@8 (Appendix E.2).

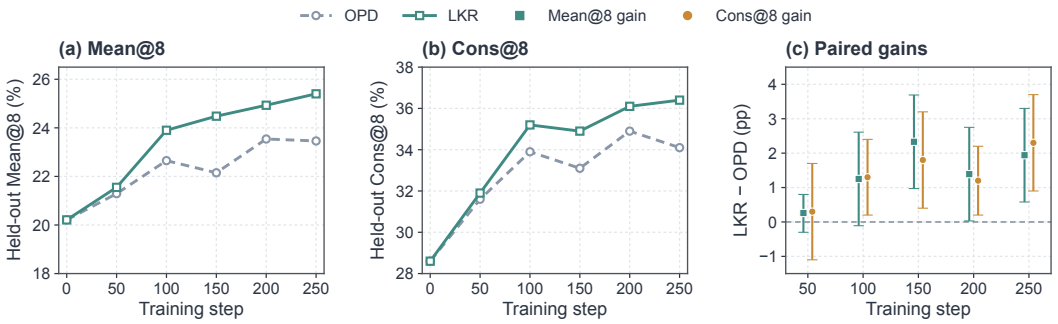

Figure 2: **Held-out training dynamics (single run).** (a,b) Mean@8 and Cons@8 at fixed checkpoints. (c) Paired LKR–OPD gains with pointwise 95% confidence intervals over the 1,000 held-out problems. Error bars measure evaluation uncertainty, not variation across training runs. Figure 5 provides complete per-step traces for the logged metrics.

In this single-run comparison, LKR exceeds OPD in both metrics at every evaluated checkpoint after initialization (Figure 2). Pointwise paired 95% intervals exclude zero at steps 150, 200, and 250 for Mean@8, and at steps 100–250 for Cons@8.

4.3 What makes recipient routing effective? (RQ2)

Redirect coupling. The dense operator in Equation 13 interpolates between OPD ($s = 0$), surplus-capped LKR ($s = \min(q, \delta/B)$), and full redirect at queried events ($s = q$). All preserve the base donor coordinate and cap the sampled coefficient at one; they test the value of explicit routing and the donor-surplus cap.

Routing evidence and joint allocation. A 2×2 ablation of Equation 7 holds donor events, candidate support, redirected budget, and water-filling fixed, varying only the scores: **Neutral** ($g = 1$), **Teacher-only** ($g = t$), **Successor-only** ($g = e^{-\beta d}$), and **Combined (LKR)** ($g = te^{-\beta d}$). Neutral need not yield uniform shares, and Successor-only still uses the teacher to compute KL. A matched-budget **LKR w/o water-filling** control uses normalized scores $a(v) \propto g(v)$ to isolate joint allocation (Table 2).

Table 2: **Routing ablation (single run).** Entries report Mean@8/Cons@8 (%). Public Avg. follows Table 1. Bold marks the best student-training result in each column and metric.

Method / Benchmark	Held-out	AIME24	AIME25	AIME26	MATH-500	AMC23	HMMT25	Public Avg.
Student initialization	20.21/28.60	29.58/56.67	23.33/33.33	20.42/30.00	84.20/90.40	64.76/75.90	13.75/20.00	39.34/51.05
Neutral ($g = 1$)	22.98/33.80	36.25/53.33	27.08/30.00	25.42/40.00	87.28/91.60	67.62/77.11	18.75/20.00	43.73/52.01
Teacher-only	24.49/34.50	38.33/63.33	28.75/**36.67**	26.67/36.67	87.70/91.60	70.18/78.31	17.92/20.00	44.93/54.43
Successor-only	24.39/34.70	**42.50**/60.00	27.92/33.33	25.83/**46.67**	87.40/91.00	67.32/79.52	16.25/23.33	44.54/55.64
LKR w/o water-filling	24.69/34.80	39.17/63.33	27.50/33.33	27.50/**46.67**	**87.75**/91.80	68.52/78.31	15.42/20.00	44.31/55.57
Full redirect	24.68/34.70	39.58/60.00	29.58/**36.67**	24.58/36.67	87.12/92.20	68.22/80.72	17.08/16.67	44.36/53.82
LKR (ours)	**25.40/36.40**	39.17/**66.67**	**30.42/36.67**	**29.17**/43.33	**87.75/92.40**	**70.78/81.93**	**20.00/26.67**	**46.22/57.95**

LKR leads on Held-out and Public Avg. for both metrics. Teacher-only and Successor-only are close in Public Avg. Mean@8, but differ across tasks: Successor-only exceeds Teacher-only by 4.17 points in AIME24 Mean@8 and 10.00 in AIME26 Cons@8, whereas Teacher-only leads by 2.86 in AMC23 Mean@8. These single-run differences suggest complementary signals, not reliable benchmark-specific rankings. Combining them improves Public Avg. Mean@8/Cons@8 over Teacher-only by 1.29/3.52 points and Successor-only by 1.68/2.31. Replacing water-filling with normalized scores lowers Public Avg. by 1.91/2.38 points: the projection accounts for existing student mass (Equation 10). Full redirect loses 1.86/4.13 points relative to LKR, supporting partial redirection. We emphasize Held-out and the predefined macro-average because several public sets are small.

4.4 TRANSFER TO A LARGER MODEL CONFIGURATION (RQ3)

We repeat the student-initialization, OPD, and LKR comparison with a DeepSeek-R1-Distill-Qwen-7B student and Skywork-OR1-32B teacher, retaining the data, training and evaluation protocol, and checkpoint rule. LKR gains 1.31/1.50 points over OPD in held-out Mean@8/Cons@8 (Table 3) and 0.75/5.14 points on Public Avg. Relative to initialization, its held-out gains are 1.84/0.30 points. Because this comparison uses one training seed and several public benchmarks are small, we view it as preliminary evidence for scale transfer.

Table 3: **Scale transfer: 7B student and 32B teacher.** Entries report Mean@8/Cons@8 (%). Public Avg. follows Table 1. Bold marks the best student-training result in each column and metric. All arms use one seed.

Method / Benchmark	Held-out	AIME24	AIME25	AIME26	MATH-500	AMC23	HMMT25	Public Avg.
Teacher	82.71/88.90	81.67/90.00	72.92/80.00	72.08/73.33	96.00/97.00	94.88/96.39	52.08/63.33	78.27/83.34
Student initialization	44.75/65.70	55.00/66.67	40.83/56.67	39.17/53.33	92.23/95.40	79.82/87.95	23.75/30.00	55.13/65.00
OPD	45.28/64.50	**55.42**/73.33	40.83/56.67	42.92/50.00	92.15/95.60	81.48/**91.57**	26.67/**33.33**	56.58/66.75
LKR (ours)	**46.59/66.00**	54.17/**76.67**	**41.25/66.67**	**45.83/66.67**	**92.35/96.40**	**83.28/91.57**	**27.08/33.33**	**57.33/71.89**

4.5 HYPERPARAMETER SENSITIVITY

Figure 3 reports step-250 sensitivity on the 1,000 held-out problems for the 1.5B student. Each panel varies one parameter and shows the Mean@8 and Cons@8 gains over the single-run OPD reference. Defaults are $m = 20$, $\alpha = 0.25$, $\beta = 1$, and $k = 4$, fixed before the sweeps.

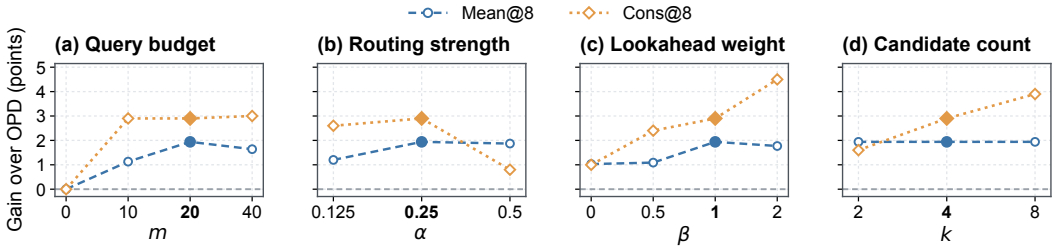

Figure 3: **Hyperparameter sensitivity.** Each panel varies one parameter and reports held-out Mean@8 and Cons@8 gains over the single-run OPD reference (23.46%/33.50%). The darker dash-dotted line marks zero gain; light dashed grid lines aid comparison. Filled markers and bold ticks identify defaults. Lines connect tested settings. These are single-run results, not confidence intervals.

Query budget and candidate count. Increasing m from 10 to 20 raises Mean@8 from 24.59 to 25.40, with Cons@8 fixed at 36.40; $m = 40$ gives no further Mean@8 improvement. Through q_i, m controls both the projection budget ρ_i and sampling variability, so this sweep measures their combined effect. For $k \in \{2, 4, 8\}$, Mean@8 stays at 25.40 while Cons@8 increases from 35.10 to 36.40 and 37.40.

Routing strength and lookahead weight. Increasing α from 0.25 to 0.5 leaves Mean@8 near 25.40 but lowers Cons@8 from 36.40 to 34.30, consistent with the bounded-redirection ablation. At $\beta = 0$, Teacher-only routing reaches 24.49/34.50 Mean@8/Cons@8, matching Table 2; $\beta = 1$ reaches 25.40/36.40 and $\beta = 2$ reaches 25.23/38.00. This supports complementary successor evidence. Figure 4 shows the cost and routing trade-offs. At $m = 20$, step time is 1.77× OPD; the main comparisons match steps, not wall time. Additional query-count and timing measurements are in Appendix E.3.

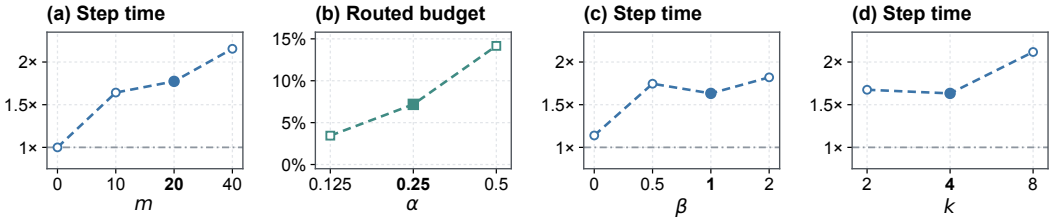

Figure 4: **Cost and routing sensitivity.** Panels follow the parameter order of Figure 3: step time relative to OPD in (a,c,d), and routed donor budget in (b). Filled markers identify defaults. Measurements come from separate sweeps; additional measurements are reported in Appendix E.3.

5 Background and Related Work

On-policy distillation. MiniLLM (Gu et al., 2024) and GKD (Agarwal et al., 2024) train on student-generated histories with reverse-KL or mixed divergences. Sampled-token OPD reduces this to token-level policy gradients (Lu & Thinking Machines Lab, 2025); recent variants broaden support, adapt the divergence, or replace unreliable negatives with local distribution matching (Fu et al., 2026; Jin et al., 2026; Oh et al., 2026; Jia et al., 2026). LKR instead changes the recipient shares of an existing negative update while preserving its donor-coordinate gradient.

Selective and future-aware supervision. Teachability and prefix-drift methods select positions compatible with the student trajectory (Wang et al., 2026; Yang et al., 2026). LGR scores candidates by successor teacher confidence (Liu et al., 2026); TOPD uses continuations to identify reasoning forks (Jiang & Ferraro, 2026). LGR is the closest analogue: both evaluate candidate-induced successors, but LKR uses successor KL in a joint allocation instead of independent candidate rewards.

Probability routing. CR-OPD routes through normalized current-prefix pairwise preferences (Zhang, 2026); LKR adds successor KL under a bounded budget. Across these lines of work, LKR is distinguished by jointly using teacher support and successor KL to derive recipient shares under a bounded routing budget. Appendix A discusses broader background and the computational trade-off between current-prefix matching and lookahead.

6 Conclusion

LKR makes the recipient side of OPD explicit. Its dense routing target combines teacher support and successor agreement under a bounded budget adapted to event sampling; each sampled correction preserves the local donor gradient before optimizer processing. On the 1,000-problem Held-out set, three-seed results improve Mean@8/Cons@8 over OPD by 2.41/2.27 points; the corresponding public macro-average gains are 2.23/4.54 points. Single-run ablations and a 7B/32B comparison provide supporting evidence for the routing design and scale transfer.

Our evaluation focuses on mathematical reasoning, and routing is restricted to candidates ranked highly by both models. Successor queries also introduce additional training cost. Future work will explore broader candidate support, more efficient lookahead, and applications beyond mathematical reasoning.

AI use statement

Generative AI tools assisted with writing and language polishing, checking query-count aggregation, and revising figures. The authors take responsibility for the final content.

Ethics statement

This study uses publicly released model checkpoints and existing mathematical reasoning problems. No human participants were involved. The reported results concern the evaluated reasoning benchmarks.

Reproducibility statement

The appendix provides proofs, pseudocode, and experimental protocols. Section 4.1 links the public models and datasets.

References

Rishabh Agarwal, Nino Vieillard, Yongchao Zhou, Piotr Stanczyk, Sabela Ramos, Matthieu Geist, and Olivier Bachem. On-policy distillation of language models: Learning from self-generated mistakes. In *International Conference on Learning Representations*, 2024.

Chenxin An, Zhihui Xie, Xiaonan Li, Lei Li, Jun Zhang, Shansan Gong, Ming Zhong, Jingjing Xu, Xipeng Qiu, Mingxuan Wang, and Lingpeng Kong. POLARIS: A post-training recipe for scaling reinforcement learning on advanced reasoning models. Open-source project and dataset, 2025. URL https://huggingface.co/datasets/POLARIS-Project/Polaris-Dataset-53K.

Lei Jimmy Ba and Rich Caruana. Do deep nets really need to be deep? In *Advances in Neural Information Processing Systems*, volume 27, 2014. URL https://papers.nips.cc/paper_files/paper/2014/hash/b0c355a9dedccb50e5537e8f2e3f0810-Abstract.html.

Samy Bengio, Oriol Vinyals, Navdeep Jaitly, and Noam Shazeer. Scheduled sampling for sequence prediction with recurrent neural networks. In *Advances in Neural Information Processing Systems*, volume 28, 2015. URL https://papers.nips.cc/paper_files/paper/2015/hash/e995f98d56967d946471af29d7bf99f1-Abstract.html.

Lev M. Bregman. The relaxation method of finding the common point of convex sets and its application to the solution of problems in convex programming. *USSR Computational Mathematics and Mathematical Physics*, 7(3):200–217, 1967. doi: 10.1016/0041-5553(67)90040-7.

Cristian Buciluǎ, Rich Caruana, and Alexandru Niculescu-Mizil. Model compression. In *Proceedings of the 12th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pp. 535–541. ACM, 2006. doi: 10.1145/1150402.1150464. URL https://doi.org/10.1145/1150402.1150464.

Thomas M. Cover and Joy A. Thomas. *Elements of Information Theory*. Wiley, 2 edition, 2006. doi: 10.1002/047174882X.

Imre Csiszár. I-divergence geometry of probability distributions and minimization problems. *The Annals of Probability*, 3(1):146–158, 1975. doi: 10.1214/aop/1176996454.

DeepSeek-AI. DeepSeek-R1: Incentivizing reasoning capability in LLMs via reinforcement learning. *arXiv preprint arXiv:2501.12948*, 2025.

Jasper Dekoninck, Nikola Jovanović, Tim Gehrunger, Kári Rögnvaldsson, Ivo Petrov, Chenhao Sun, and Martin Vechev. Beyond benchmarks: MathArena as an evaluation platform for mathematics with LLMs. *arXiv preprint arXiv:2605.00674*, 2026. URL https://arxiv.org/abs/2605.00674.

Yuqian Fu, Haohuan Huang, Kaiwen Jiang, Yuanheng Zhu, and Dongbin Zhao. Revisiting on-policy distillation: Empirical failure modes and simple fixes. *arXiv preprint arXiv:2603.25562*, 2026.

Jianping Gou, Baosheng Yu, Stephen J. Maybank, and Dacheng Tao. Knowledge distillation: A survey. *International Journal of Computer Vision*, 129(6):1789–1819, 2021. doi: 10.1007/s11263-021-01453-z. URL https://doi.org/10.1007/s11263-021-01453-z.

Yuxian Gu, Li Dong, Furu Wei, and Minlie Huang. MiniLLM: Knowledge distillation of large language models. *International Conference on Learning Representations*, 2024.

Jujie He, Jiacai Liu, Chris Yuhao Liu, Rui Yan, Chaojie Wang, Peng Cheng, Xiaoyu Zhang, Fuxiang Zhang, Jiacheng Xu, Wei Shen, Siyuan Li, Liang Zeng, Tianwen Wei, Cheng Cheng, Bo An, Yang Liu, and Yahui Zhou. Skywork open reasoner 1 technical report. *arXiv preprint arXiv:2505.22312*, 2025. URL https://huggingface.co/Skywork/Skywork-OR1-Math-7B.

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. In *Advances in Neural Information Processing Systems*, volume 34, 2021. URL https://arxiv.org/abs/2103.03874.

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

Daniel G. Horvitz and Donovan J. Thompson. A generalization of sampling without replacement from a finite universe. *Journal of the American Statistical Association*, 47(260):663–685, 1952. doi: 10.1080/01621459.1952.10483446.

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*, pp. 8003–8017, 2023. doi: 10.18653/v1/2023.findings-acl.507. URL https://aclanthology.org/2023.findings-acl.507/.

Nan Jia, Haojin Yang, Xing Ma, Jiesong Lian, Shuailiang Zhang, Weipeng Zhang, Ke Zeng, Xunliang Cai, and Zequn Sun. Asymmetric on-policy distillation: Bridging exploitation and imitation at the token level. *arXiv preprint arXiv:2605.06387*, 2026.

Yuxuan Jiang and Francis Ferraro. Bridging reasoning trajectories in on-policy distillation via near-future guidance. *arXiv preprint arXiv:2606.00305*, 2026.

Woogyeol Jin, Taywon Min, Yongjin Yang, Swanand Ravindra Kadhe, Yi Zhou, Dennis Wei, Nathalie Baracaldo, and Kimin Lee. Entropy-aware on-policy distillation of language models. *arXiv preprint arXiv:2603.07079*, 2026.

Yoon Kim and Alexander M. Rush. Sequence-level knowledge distillation. In *Proceedings of EMNLP*, 2016.

Hunter Lightman, Vineet Kosaraju, Yura Burda, Harrison Edwards, Bowen Baker, Teddy Lee, Jan Leike, John Schulman, Ilya Sutskever, and Karl Cobbe. Let's verify step by step. In *International Conference on Learning Representations*, 2024. URL https://arxiv.org/abs/2305.20050.

Yanjiang Liu, Jie Lou, Xinyan Guan, Yuqiu Ji, Hongyu Lin, Ben He, Xianpei Han, Le Sun, Xing Yu, and Yaojie Lu. Your teacher can't help you here: Combating supervision fidelity decay in on-policy distillation. *arXiv preprint arXiv:2605.30833*, 2026.

Kevin Lu and Thinking Machines Lab. On-policy distillation. Thinking Machines Lab Blog, 2025. URL https://thinkingmachines.ai/blog/on-policy-distillation/.

Minjae Oh, Sangjun Song, Gyubin Choi, Yunho Choi, and Yohan Jo. KL for a KL: On-policy distillation with control variate baseline. *arXiv preprint arXiv:2605.07865*, 2026.

Stephane Ross, Geoffrey Gordon, and Drew Bagnell. A reduction of imitation learning and structured prediction to no-regret online learning. In *Proceedings of the Fourteenth International Conference on Artificial Intelligence and Statistics*, volume 15 of *Proceedings of Machine Learning Research*, pp. 627–635, 2011. URL https://proceedings.mlr.press/v15/ross11a.html.

Andrei A. Rusu, Sergio Gomez Colmenarejo, Çağlar Gülçehre, Guillaume Desjardins, James Kirkpatrick, Razvan Pascanu, Volodymyr Mnih, Koray Kavukcuoglu, and Raia Hadsell. Policy distillation. In *International Conference on Learning Representations*, 2016. URL https://arxiv.org/abs/1511.06295.

John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. Proximal policy optimization algorithms. *arXiv preprint arXiv:1707.06347*, 2017. URL https://arxiv.org/abs/1707.06347.

Mingyang Song and Mao Zheng. A survey of on-policy distillation for large language models. *arXiv preprint arXiv:2604.00626*, 2026.

Jonathan Uesato, Nate Kushman, Ramana Kumar, Francis Song, Noah Siegel, Lisa Wang, Antonia Creswell, Geoffrey Irving, and Irina Higgins. Solving math word problems with process- and outcome-based feedback. *arXiv preprint arXiv:2211.14275*, 2022. URL https://arxiv.org/abs/2211.14275.

Yuanyi Wang, Su Lu, Yanggan Gu, Pengkai Wang, Yifan Yang, Zhaoyi Yan, Congkai Xie, Jianmin Wu, and Hongxia Yang. Not all disagreement is learnable: Token teachability in on-policy distillation. *arXiv preprint arXiv:2605.26844*, 2026.

Liang Wen, Yunke Cai, Fenrui Xiao, Xin He, Qi An, Zhenyu Duan, Yimin Du, Junchen Liu, Lifu Tang, Xiaowei Lv, Haosheng Zou, Yongchao Deng, Shousheng Jia, and Xiangzheng Zhang. Light-R1: Curriculum SFT, DPO and RL for long COT from scratch and beyond. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 6: Industry Track)*, pp. 318–327, 2025. URL https://aclanthology.org/2025.acl-industry.24/.

Ronald J. Williams. Simple statistical gradient-following algorithms for connectionist reinforcement learning. *Machine Learning*, 8:229–256, 1992. doi: 10.1007/BF00992696.

Zhicheng Yang, Zhijiang Guo, Yifan Song, Minrui Xu, Yongxin Wang, Yiwei Wang, Xiaodan Liang, and Jing Tang. Prune-OPD: Efficient and reliable on-policy distillation for long-horizon reasoning. *arXiv preprint arXiv:2605.07804*, 2026.

Bowen Zhang. A formula-driven survey and research agenda for on-policy distillation. *arXiv preprint arXiv:2606.22793*, 2026.

A Expanded Related Work

A.1 Generative and on-policy distillation

Knowledge distillation originated in compressing the predictive function of a larger model or ensemble into a smaller network (Buciluǎ et al., 2006; Ba & Caruana, 2014), and temperature-softened logit matching became its standard neural formulation (Hinton et al., 2015). The literature now spans response, feature, and relation matching (Gou et al., 2021). For generation, sequence-level distillation transfers teacher-generated outputs (Kim & Rush, 2016), while rationale distillation transfers intermediate reasoning traces (Hsieh et al., 2023). Policy distillation provides a parallel lineage for sequential decision problems (Rusu et al., 2016).

For autoregressive generation, fixed or teacher-generated histories can differ from those induced by the student at inference time. This exposure mismatch is closely related to the motivation behind DAgger and scheduled sampling (Ross et al., 2011; Bengio et al., 2015). MiniLLM optimizes reverse KL on student samples, and GKD generalizes on-policy distillation to different divergences and mixtures of generated histories (Gu et al., 2024; Agarwal et al., 2024). A lightweight sampled-token realization queries the teacher for the realized token and turns the difference between teacher and student log probabilities into a policy-gradient-style signal (Lu & Thinking Machines Lab, 2025).

Recent OPD methods improve support, divergence selection, variance, or the treatment of negative signals. Top-K local matching broadens the support of the estimator (Fu et al., 2026); entropy-aware OPD adapts forward- and reverse-KL contributions according to teacher uncertainty (Jin et al., 2026); variance-reduced OPD introduces an analytic control variate (Oh et al., 2026); and AOPD replaces selected negative sampled-token updates with localized distribution matching (Jia et al., 2026). These methods change which signal is estimated or where it is trusted. LKR instead defines a dense routing target that changes a bounded portion of an existing negative update's non-donor destination, then estimates its correction through event sampling.

Full-vocabulary matching at observed prefixes and lookahead at alternative prefixes incur different costs. Once a student trajectory is available, teacher forcing can evaluate its observed prefixes together, and local distribution matching can use the vocabulary logits at those positions. LKR additionally queries candidate-induced successor prefixes absent from that trajectory. Event sampling limits the number of these branching queries; the sparse successor representation reduces the size of the retained distributions (Appendix D.2). These costs motivate reporting step-matched results separately from the approximate training-time comparison in Appendix E.2.

A.2 Selective and future-aware supervision

Teachability and prefix-drift methods retain positions at which teacher feedback remains compatible with the student trajectory (Wang et al., 2026; Yang et al., 2026). Other approaches use short-horizon information. LGR evaluates counterfactual candidates through the teacher confidence they induce at the next position (Liu et al., 2026), while TOPD uses continuations to distinguish meaningful reasoning forks from local token mismatches (Jiang & Ferraro, 2026). LGR is the closest mechanistic analogue to LKR because both evaluate candidate-induced successor states. LKR differs by using successor reverse KL from the student to the teacher inside one joint recipient allocation.

A.3 Vocabulary-level probability routing

The formula-driven survey of Zhang (2026) separates temporal credit assignment from *vocabulary-level probability routing* and proposes CR-OPD, a current-prefix pairwise operator for alternatives supported by both teacher and student. Its normalized candidate weights define relative pairwise preferences at the current prefix. LKR specifies a portion of the base donor-gradient budget and allocates it through a projection using current teacher support and candidate-induced successor KL. The routing residual preserves the base donor coordinate. These are distinct choices of objective and budget coupling within the shared probability-routing formulation.

B Proofs for Lookahead KL Routing

B.1 Reverse-KL gradients on underweighted alternatives

We derive equation 5. Suppress the prefix index and write

$$L_{\mathrm{RKL}} = D_{\mathrm{KL}}(p\|t) = \sum_j p(j) \log \frac{p(j)}{t(j)}, \qquad p = \mathrm{softmax}(z). \tag{17}$$

Using $\partial p(j)/\partial z_v = p(j)(\mathbb{I}\{j = v\} - p(v))$,

$$\frac{\partial L_{\mathrm{RKL}}}{\partial z_v} = \sum_j p(j)(\mathbb{I}\{j = v\} - p(v)) \left(\log \frac{p(j)}{t(j)} + 1 \right) \tag{18}$$

$$= p(v) \left(\log \frac{p(v)}{t(v)} - D_{\mathrm{KL}}(p\|t) \right). \tag{19}$$

For fixed $t(v) > 0$, both $p(v) \log p(v)$ and the remaining terms vanish as $p(v) \to 0$. This is a statement about the direct local logit gradient. Shared network parameters and updates to other logits can still change $p(v)$ indirectly, so the result does not imply that an underweighted token is mathematically unreachable. It instead explains why the student's current support is a poor sole criterion for choosing a specific replacement.

B.2 Two-token objective and the auxiliary distribution

Fix a routed event with $\rho > 0$ and suppress i. For $y(v) \geq 0$ on S with $\sum_{v \in S} y(v) = \rho$, define

$$\widetilde{p}(v) = \begin{cases} p(u) - \rho, & v = u, \\ p(v) + y(v), & v \in S, \\ p(v), & \text{otherwise.} \end{cases} \tag{20}$$

The constraint $\rho \leq \alpha[p(u) - t(u)]_+ < p(u)$ keeps the donor mass positive, and the added and removed masses cancel. Hence $\widetilde{p}$ is a probability distribution. It is an auxiliary allocation target used to define shares for the gradient construction.

Let $P_v(w) = \pi_\theta(w \mid c \oplus v)$ and $T_v(w) = \pi_T(w \mid c \oplus v)$ be fixed successor distributions. The two-token joint distributions $\widetilde{P}(v, w) = \widetilde{p}(v)P_v(w)$ and $T(v, w) = t(v)T_v(w)$ satisfy the KL chain rule

$$D_{\mathrm{KL}}(\widetilde{P}\|T) = D_{\mathrm{KL}}(\widetilde{p}\|t) + \sum_{v \in \mathcal{V}} \widetilde{p}(v) D_{\mathrm{KL}}(P_v\|T_v). \tag{21}$$

For fixed ρ, all non-candidate terms are constant in y. Writing $d(v) = D_{\mathrm{KL}}(P_v \| T_v)$ and weighting the successor term by β leaves the candidate objective

$$\sum_{v \in S} (p(v) + y(v)) \left[\log \frac{p(v) + y(v)}{t(v)} + \beta d(v) \right] = \sum_{v \in S} (p(v) + y(v)) \log \frac{p(v) + y(v)}{g(v)}, \tag{22}$$

where $g(v) = t(v) e^{-\beta d(v)}$. At $\beta = 1$, this is the variable part of the two-token joint KL; other values set the weight of successor agreement. The optimum determines $a = y^\star / \rho$. This optimality concerns the auxiliary probability target with fixed candidates, budget, and successor conditionals. Section B.6 connects this target to the logit correction through fixed-target distillation losses.

B.3 Water-filling solution

We prove Proposition 1. Fix an event with nonempty S, $g(v) > 0$, and $\rho > 0$, and suppress the subscript i. Let $x(v) = p(v) + y(v)$. The receiver-only projection becomes

$$\min_x \sum_{v \in S} x(v) \log \frac{x(v)}{g(v)} \quad \text{s.t.} \quad x(v) \geq p(v), \qquad \sum_{v \in S} x(v) = \sum_{v \in S} p(v) + \rho. \tag{23}$$

Because softmax probabilities are positive, $g(v) > 0$. The objective is strictly convex on the positive orthant and the feasible set is convex, so the solution is unique.

Introduce multiplier $\nu \in \mathbb{R}$ for the equality constraint and $\mu_v \geq 0$ for the lower bound. The KKT conditions are

$$\log \frac{x(v)}{g(v)} + 1 + \nu - \mu_v = 0, \tag{24}$$

$$\mu_v (x(v) - p(v)) = 0. \tag{25}$$

Set $\kappa = \exp(-1 - \nu)$. If $x(v) > p(v)$, complementary slackness gives $x(v) = \kappa g(v)$. If the lower bound is active, stationarity requires $p(v) \geq \kappa g(v)$. Hence

$$x(v) = \max\{p(v), \kappa g(v)\}, \qquad y(v) = [\kappa g(v) - p(v)]_+. \tag{26}$$

The excess-mass function

$$H(\kappa) = \sum_{v \in S} [\kappa g(v) - p(v)]_+ \tag{27}$$

is continuous, equals zero for sufficiently small κ, and is strictly increasing once positive. It diverges as $\kappa \to \infty$, so $H(\kappa) = \rho$ has a unique solution. For an unrouted event with $\rho = 0$, $y = 0$ is the unique feasible optimum, but routing shares need not be defined because its residual is zero. This proves the proposition.

If $\beta = 0$, then $g(v) = t(v)$ and allocation uses only current-prefix teacher support. If every $d(v)$ is equal, the common factor $e^{-\beta d(v)}$ is absorbed into κ, giving the same allocation.

Computing the threshold. Sort the ratios $r_j = p(v_j)/g(v_j)$ in nondecreasing order. For a proposed active prefix of length j, its threshold is

$$\kappa_j = \frac{\rho + \sum_{\ell=1}^{j} p(v_\ell)}{\sum_{\ell=1}^{j} g(v_\ell)}. \tag{28}$$

The valid prefix satisfies $r_j < \kappa_j \leq r_{j+1}$, with $r_{|S|+1} = +\infty$. Sorting and a cumulative scan give the solution in $O(|S| \log |S|)$ time. Coordinates at equality have zero increment.

B.4 BASE-UPDATE FACTORIZATION

Equation 4 follows directly from the definition of b. Suppressing event subscripts,

$$\gamma(e_u - p) = \gamma\left((1 - p(u))e_u - \sum_{v \neq u} p(v)e_v\right) \tag{29}$$

$$= \gamma(1 - p(u))\left(e_u - \sum_{v \neq u} \frac{p(v)}{1 - p(u)} e_v\right) \tag{30}$$

$$= B(e_u - b). \tag{31}$$

The vector b is a probability distribution supported only on non-donor coordinates.

B.5 DONOR PRESERVATION AND THE INTERPOLATED LOSS

The directed local loss and its interpolation with the base loss are

$$\ell_i^{\mathrm{dir}} = B_i\left[\log \pi_\theta(u_i \mid c_i) - \sum_{v \in S_i} a_i(v) \log \pi_\theta(v \mid c_i)\right], \tag{32}$$

$$\ell_i^{\mathrm{LKR}} = \ell_i^{\mathrm{base}} + s_i(\ell_i^{\mathrm{dir}} - \ell_i^{\mathrm{base}}). \tag{33}$$

Here B_i, s_i, and a_i are detached during differentiation.

We prove Proposition 2. Both a and b are probability distributions with zero donor coordinate. Therefore the residual $h = \rho(b - a)$ satisfies

$$h(u) = 0, \qquad \mathbf{1}^\top h = \rho(\mathbf{1}^\top b - \mathbf{1}^\top a) = 0. \tag{34}$$

Adding h to $B(e_u - b)$ cannot change the donor coordinate or the row sum. As $\rho = sB$,

$$B(e_u - b) + \rho(b - a) = B[e_u - (1 - s)b - sa]. \tag{35}$$

Since $s \in [0, 1]$, $(1 - s)b + sa$ is itself a non-donor probability distribution.

It remains to verify the loss realization. For softmax logits z,

$$\nabla_z \log \pi_\theta(j \mid c) = e_j - \pi_\theta(\cdot \mid c). \tag{36}$$

Using $\sum_v a(v) = 1$ and holding B and a fixed,

$$\nabla_z \ell^{\mathrm{dir}} = B\left[(e_u - \pi) - \sum_v a(v)(e_v - \pi)\right] \tag{37}$$

$$= B(e_u - a). \tag{38}$$

Combining this with $\nabla_z \ell^{\mathrm{base}} = B(e_u - b)$ yields

$$\nabla_z\left[(1 - s)\ell^{\mathrm{base}} + s\ell^{\mathrm{dir}}\right] = B[(1 - s)(e_u - b) + s(e_u - a)] \tag{39}$$

$$= B[e_u - (1 - s)b - sa], \tag{40}$$

which is equation 13. This proof is local in logit space and does not assume a particular neural parameterization.

B.6 FROM THE AUXILIARY PROJECTION TO THE LOGIT CORRECTION

Fix a routed event and let p^0 be a detached copy of its current student distribution. Hold a, b, ρ fixed and define

$$q^a = p^0 + \rho(a - e_u), \qquad q^b = p^0 + \rho(b - e_u). \tag{41}$$

Both targets sum to one. Their donor coordinate is $p^0(u) - \rho > 0$, and all other coordinates are nonnegative. The target q^a equals the optimal auxiliary distribution because $\rho a = y^\star$; q^b transfers the same mass using the implicit recipient shares.

For $p_\theta = \mathrm{softmax}(z)$, consider the fixed-target loss difference

$$\mathcal{R}(z) = D_{\mathrm{KL}}(q^a \| p_\theta) - D_{\mathrm{KL}}(q^b \| p_\theta). \tag{42}$$

Since $\nabla_z D_{\mathrm{KL}}(q\|p_\theta) = p_\theta - q$ for fixed q,

$$\nabla_z \mathcal{R} = (p_\theta - q^a) - (p_\theta - q^b) = \rho(b - a). \tag{43}$$

Consequently, $\ell^{\mathrm{base}} + \mathcal{R}$ reproduces the LKR local gradient at the event where the targets were constructed. This is an exact local gradient identity for the existing detached surrogate, requiring no change to the training rule. The reverse-KL projection selects the target; the forward-KL target losses specify how its difference from the implicit target enters the logit gradient. Their difference need not be nonnegative and is not itself a KL divergence.

This identity does not equate $y^\star$ with an actual probability update. For an isolated logit step Δz, the first-order probability change is

$$\Delta p = (\mathrm{diag}(p^0) - p^0(p^0)^\top)\Delta z + O(\|\Delta z\|^2). \tag{44}$$

Thus the threshold $\kappa g(v) > p^0(v)$ specifies positive auxiliary increments and routed gradient shares, not the support of the actual probability change. Routing changes the direct logit signal to recipients; it does not remove the softmax Jacobian or account for parameter sharing and optimizer processing. The unique optimum in Proposition 1 is global within the constrained auxiliary problem, not an optimal parameter update or a guarantee of descent on the original two-token reverse KL.

The outer budget rule chooses the largest dose satisfying both $\rho \le \delta$ and $\rho/q \le B$, hence $\rho = \min(\delta, qB)$. Using the same numerical dose for auxiliary mass transfer and gradient replacement couples two distinct spaces: the surplus cap ensures a valid probability target, and the replacement cap bounds the sampled gradient mixture. This coupling is chosen before projection; neither its scale nor α follows from projection optimality.

B.7 Coarse-grained successor KL

The implementation evaluates successor discrepancy on a vocabulary partition. At successor $c \oplus v$, let P_v and T_v be the student and teacher distributions. The retained singleton tokens are

$$R_v = \mathrm{Top20}(P_v) \cap \mathrm{Top20}(T_v),$$

where Top20 returns the 20 highest-probability token IDs. All other tokens form one tail bin. Set $P_v^{\mathrm{tail}} = 1 - \sum_{w \in R_v} P_v(w)$ and define T_v^{tail} analogously. The coarse discrepancy is

$$d^{\mathrm{coarse}}(v) = \sum_{w \in R_v} P_v(w) \log \frac{P_v(w)}{T_v(w)} + P_v^{\mathrm{tail}} \log \frac{P_v^{\mathrm{tail}}}{T_v^{\mathrm{tail}}}. \tag{45}$$

The same partition is used for student and teacher at each successor. The following result relates this objective to full-vocabulary KL.

Lemma 4 (Coarse-grained KL lower bound)**.** *Let P and Q be distributions on a finite vocabulary and let $\{A_1, \ldots, A_J\}$ be a partition. Define $\bar{P}_j = P(A_j)$ and $\bar{Q}_j = Q(A_j)$. Then*

$$D_{\mathrm{KL}}(P\|Q) \ge D_{\mathrm{KL}}(\bar{P}\|\bar{Q}). \tag{46}$$

Proof. For each bin, the log-sum inequality gives

$$\sum_{x \in A_j} P(x) \log \frac{P(x)}{Q(x)} \ge P(A_j) \log \frac{P(A_j)}{Q(A_j)}. \tag{47}$$

Summing over bins proves the claim. Equivalently, the result follows from the data-processing inequality for KL under the deterministic map from tokens to partition indices.

Replacing d with d^{coarse} gives the allocation objective for these partitioned successor distributions. The lower bound alone does not ensure identical candidate rankings under coarse and full KL. Both the sampled correction and its budget-dependent projected target use this discrepancy.

C Budgeted Sampling Analysis

C.1 Conditional unbiasedness

For each eligible event, use the clipped budget $\rho_i = \min\{q_i B_i, \delta_i\}$ and solve the water-filling projection at that budget to obtain a_i. Define the resulting residual

$$h_i = \rho_i(b_i - a_i). \tag{48}$$

Condition on the rollout and on the potential annotation (ρ_i, b_i, a_i) at every eligible event. If event i is included with probability $q_i > 0$, then

$$\mathbb{E}_I\left[\frac{I_i}{q_i} h_i\right] = h_i. \tag{49}$$

Let $\mathcal{I}$ contain all rollout positions and $\mathcal{E} \subseteq \mathcal{I}$ the eligible events. Linearity of expectation gives

$$\mathbb{E}_I\left[\sum_{i\in\mathcal{I}} G_i^{\text{base}} + \sum_{i\in\mathcal{E}} \frac{I_i}{q_i} h_i\right] = \sum_{i\in\mathcal{I}} G_i^{\text{base}} + \sum_{i\in\mathcal{E}} h_i. \tag{50}$$

No independence assumption is required. The same proof applies to the loss estimator

$$\widehat{L}_{\text{SQ}} = L_{\text{base}} + \sum_{i\in\mathcal{E}} \frac{I_i}{q_i} s_i(\ell_i^{\text{dir}} - \ell_i^{\text{base}}), \tag{51}$$

because its local residual has gradient h_i.

Donor neutrality and the zero-row-sum property hold for every sample because scalar inclusion weighting preserves both invariants. The sample-wise budget bound is proved in Section C.3.

C.2 Uniform sampling without replacement

Let $N = |\mathcal{E}|$ and sample $m < N$ events uniformly without replacement. Then $q_i = m/N$ and every queried residual receives weight N/m. If $m \geq N$, every event is included with weight one. For vector residuals, let

$$\bar{h} = \frac{1}{N}\sum_i h_i, \qquad \Sigma_h = \frac{1}{N-1}\sum_i (h_i - \bar{h})(h_i - \bar{h})^\top. \tag{52}$$

The covariance of the inverse-probability residual estimator is

$$\text{Cov}\left(\frac{N}{m}\sum_{i:I_i=1} h_i\right) = \frac{N^2}{m}\left(1 - \frac{m}{N}\right)\Sigma_h. \tag{53}$$

This identity holds for each fixed query budget. Changing m changes both q_i and the projected residuals h_i, as well as the sampling covariance and query cost. The target therefore depends on the query budget.

C.3 Clipped coefficients and projection optimality

For $B_i > 0$, the coefficient at a queried event is

$$\eta_i = \frac{\rho_i}{q_i B_i} = \min\left\{\frac{\delta_i}{q_i B_i}, 1\right\}, \qquad \tau_i = I_i\eta_i \in [0,1]. \tag{54}$$

Events with $B_i = 0$ contribute no residual. Combining the base gradient with the sampled correction gives

$$\widehat{G}_i = B_i[e_{u_i} - (1-\tau_i)b_i - \tau_i a_i]. \tag{55}$$

Since a_i and b_i are probability distributions with zero donor mass, this is a convex recipient mixture for every sample, with donor coordinate B_i and zero row sum. Moreover, $\|I_i h_i/q_i\|_1 \leq 2I_i B_i$. Together with the expectation calculation above, this proves Proposition 3.

The clipped budget is the largest dose satisfying both the surplus cap $\rho_i \leq \delta_i$ and the sample-wise constraint $\rho_i/(q_i B_i) \leq 1$. The water-filling level is solved using this dose, so Proposition 1 gives its unique optimal auxiliary allocation. The fixed-target identity in Section B.6 then yields the residual $\rho_i(b_i - a_i)$. All these statements use the same clipped budget and recomputed shares. Projection optimality concerns the auxiliary distribution; it does not assert optimality of a parameter update after optimizer processing.

D TRAINING REALIZATION

The main text defines LKR at the level of a local student distribution. The training realization follows the same two-stage structure as on-policy policy optimization. A behavior-policy snapshot generates the rollout and determines the event inclusion probabilities, candidate support, clipped routing budget, and water-filling shares computed at that budget. The subsequent policy update treats these quantities as stopped-gradient targets. In the reported configuration, each rollout batch is consumed by one policy update, so the annotation and loss anchors coincide at the start of the update. All donor-preservation and conservation claims are row-wise statements about the loss gradient with respect to the local logits, before parameter sharing, optimizer preconditioning, and momentum are applied.

OPD and LKR use the same clipped token-level signal to reconstruct the base coefficient and donor budget (Appendix E.6).

Batch aggregation and global gradient clipping. Global gradient clipping is applied to the parameter gradient after batch aggregation with a norm threshold of 1.0. In the logged 250-step OPD and LKR runs, this rule triggers at 8 of 250 steps (3.2%) for each method. This is an aggregate-gradient diagnostic, distinct from the per-event routing coefficient clipping in equation 54 and the response clipping rate in Figure 5. Conditional unbiasedness applies to the gradient estimator before global gradient clipping and optimizer processing; this subsequent nonlinear operation need not preserve its expectation.

D.1 END-TO-END SQ-LKR ALGORITHM

Algorithm 1 SQ-LKR training realization

Require: Student π_θ, teacher π_T, query budget m, candidate size k

for each training iteration **do**
  Generate an on-policy rollout and compute the base OPD loss
  Form $\mathcal{E} = \{i : p_i(u_i) > t_i(u_i)\}$
  if $\mathcal{E} = \varnothing$ or $m = 0$ **then**
    Update θ using L_{base} and continue
  end if
  Sample $\mathcal{Q} \subseteq \mathcal{E}$ uniformly without replacement, with $|\mathcal{Q}| = \min(m, |\mathcal{E}|)$
  Set $q_i = \min(1, m/|\mathcal{E}|)$ and $L \leftarrow L_{\text{base}}$
  for $i \in \mathcal{Q}$ **do**
    Obtain the base coefficient γ_i and compute $B_i = \gamma_i(1 - p_i(u_i))$
    Compute $\delta_i = \alpha[p_i(u_i) - t_i(u_i)]_+$ and $\rho_i = \min(q_i B_i, \delta_i)$
    Construct S_i; if $S_i = \varnothing$ or $\rho_i = 0$, continue
    Set $s_i = \rho_i / B_i$ and $\eta_i = s_i / q_i \le 1$
    for $v \in S_i$ **do**
      Query the successor distributions at $c_i \oplus v$ and compute $g_i(v)$
    end for
    Solve $y_i^\star(v) = [\kappa_i g_i(v) - p_i(v)]_+$ with $\sum_v y_i^\star(v) = \rho_i$
    Set $a_i(v) = y_i^\star(v)/\rho_i$
    $L \leftarrow L + \eta_i(\ell_i^{\text{dir}} - \ell_i^{\text{base}})$
  end for
  Update θ using L
end for

The base loss remains active at every token. Sampling is applied only to the donor-neutral residual, which is the distinction required by Equation 15. All teacher quantities, routing budgets, shares, inclusion weights, and local scalar coefficients are detached; gradients flow through the student log-probabilities in the base and directed losses.

D.2 Sparse support and successor discrepancy

The default candidate set contains up to $k = 4$ tokens, in teacher rank order, from the intersection of teacher and student top-20 lists, excluding the donor. An empty candidate set contributes no routing residual. Successor KL is computed on a separate vocabulary partition at each successor prefix $c_i \oplus v$. Here S_i indexes candidate tokens to append, whereas R_v indexes the successor tokens retained as singleton bins after appending v. Each model returns its successor top-20 token IDs and probabilities; R_v is their intersection, without a further top-k restriction. Both models use this same set, with all remaining tokens merged into a tail bin. The retained probabilities are taken from the full distributions without renormalization over R_v; the tail probability is their remaining mass, as in Equation 45. If R_v is empty, both tail probabilities equal one and the coarse KL is zero, so this candidate's score reduces to $g_i(v) = t_i(v)$. By Lemma 4, this discrepancy is a lower bound on full-vocabulary KL. The water-filling solution applies to the partitioned objective, but the lower bound does not guarantee identical recipient shares.

The partition represents each successor distribution by $|R_v| + 1 \leq 21$ probabilities rather than $|\mathcal{V}|$. It therefore reduces the size of the retained distribution representation; it does not remove the model forward pass needed to evaluate that successor. SQ-LKR separately reduces the number of successor prefixes evaluated through event sampling.

The realized teacher probability for the sampled token is retained as a separate quantity rather than inferred from whether that token belongs to the teacher top list. This separation prevents support truncation from silently changing the sign of the negative-credit gate.

D.3 Reported training configuration

All reported LKR experiments use SQ-LKR with a per-event coefficient cap of one. Annotation computes $\rho_i = \min(q_i B_i, \delta_i)$ and solves the water-filling allocation at this clipped dose before constructing the loss. The base loss remains active at all positions; the residual coefficient is $\eta_i = \rho_i/(q_i B_i)$, with inclusion weighting applied exactly once. The conditional expectation in Equation 16 uses these same budgets, candidate supports, and coarse successor discrepancies. Inclusion probabilities are computed from the eligible set before sampling; skipping an empty candidate set does not change them.

E Additional Experimental Details

E.1 Training-data overlap check

POLARIS-53K draws from DeepScaleR-40K and AReaL-boba-106K (An et al., 2025). An independent audit reported 196 prompt matches between DeepScaleR-Preview-Dataset and MATH-500 under number-insensitive exact and 32-gram matching (Wen et al., 2025). We therefore checked the final 47,909 training prompts used in this work against all six public benchmarks: AIME 2024, AIME 2025, AIME 2026, MATH-500, AMC 2023, and HMMT February 2025. No retained matches were found. The 1,000 Held-out problems are also excluded from this training split. The reported 196 matches concern an upstream dataset and are not a count of examples removed from our training split.

E.2 Reported experimental settings

Table 4 collects the on-policy training and evaluation settings for the 1.5B/7B comparison. Appendix E.6 specifies baseline objectives, and Appendix E.5 gives the separate SFT configuration. The primary configuration is described in Section 4.1; the larger model pair is specified in Section 4.4.

The main comparison (Table 1) reports three training seeds; ablations, scale transfer, and diagnostic plots report individual runs. Sample standard deviations use Bessel's correction (denominator $3 - 1 = 2$). For Public Avg., we report the mean and sample standard deviation of the six-benchmark macro-average across seeds. Checkpoint trajectories and pointwise paired 95% confidence intervals are reported for both Held-out Mean@8 and Cons@8 in Figure 2. The Cons@8 intervals use a paired percentile bootstrap over problems. The on-policy comparison uses the same prompt order

and prespecified step-250 checkpoint; SFT uses its final step-240 checkpoint after three epochs. Section 4.2 additionally reports OPD continued to step 450. Default LKR hyperparameters were fixed before the sensitivity sweeps; validation was not used to select them. Native thinking mode remains enabled during rollout. Checkpoints within a table are evaluated in the same round, and the teacher uses the same decoding protocol. Final benchmark evaluation uses generation seed 20260817, distinct from the training seeds. For scoring, we remove a completed `<think>...</think>` span and extract the boxed answer; an unclosed thinking span is counted as incorrect.

Consensus accuracy (Cons@8). For each problem, we sample eight independent responses with the same decoding configuration. We extract and lightly normalize each final answer by removing formatting-only differences, including whitespace, braces, and common LaTeX commands. We select the most frequent normalized answer and break ties deterministically by first occurrence. A problem is correct when the selected answer matches the reference under our rule-based mathematical equivalence checker. Cons@8 is the fraction of problems whose consensus answer is correct. All dataset-level results are aggregated over individual problems rather than by averaging shard-level accuracies.

Longer OPD training. To assess whether LKR's gains can be explained by additional computation, we extend OPD to 450 steps. This approximately matches the estimated training time of 250-step LKR, based on the $1.77\times$ per-step overhead reported in Appendix E.3. OPD reaches $24.64 \pm 0.25\%$ Mean@8 and $34.97 \pm 0.15\%$ Cons@8 on Held-out, while LKR reaches $25.87 \pm 0.56\%$ and $36.40 \pm 0.17\%$ at 250 steps. Thus LKR retains 1.23/1.43 percentage-point gains under this approximate training-time budget.

Table 4: **Shared 1.5B/7B on-policy training, routing, and evaluation settings.** SFT training uses the separate configuration in Appendix E.5.

Stage	Setting	Value
Training	Optimizer steps (on-policy comparison)	250
	Global batch	128 prompts
	Responses per rollout prompt	1
	Optimizer / weight decay	AdamW / 0.01
	Learning rate	10^{-5}
	Warmup steps	0
	Gradient clipping norm threshold	1.0
	Arithmetic	BF16
	Training mode	full-parameter FSDP2
	GPU count	8
	Student / teacher tensor parallelism	1 / 4
	Maximum prompt / response length	1,024 / 9,216 tokens
	Rollout temperature / top-p	1.0 / 1.0
	Rollout top-k	disabled (-1)
	Token-level $k1$ signal clamp	$[-10, 10]$
	Task reward / reference KL penalty	both disabled
LKR	Queried events per rollout m	20
	Maximum candidates per event k	4
	Candidate pool	intersection of teacher and student top-20 lists
	Successor KL support	intersection of successor top-20 lists
	Successor tail bin	all tokens outside the intersection
	Surplus scale α / lookahead scale β	0.25 / 1.0
	Event sampling	uniform without replacement
	Sampled routing coefficient	capped at 1; projection uses clipped budget
Evaluation	Responses per problem	8
	Temperature / top-p	0.8 / 0.95
	Top-k	disabled
	Maximum generation length	65,536 tokens
	Final evaluation seed	20260817
	Unclosed thinking span	counted as incorrect

E.3 Computational cost and routing behavior

We report the resource measurements complementing Figure 4. Timings are retained separately for each sweep; nominal defaults from different sweeps are not pooled. Query totals below use the complete 250-step default-run log.

Query budget and candidate count. Increasing m from 10 to 20 raises step time from $1.64\times$ to $1.77\times$ OPD; $m = 40$ costs $2.16\times$ without further Mean@8 improvement. In the complete default LKR run, successor-prefix teacher queries total 2,429,984, averaging 9.72×10^3 per optimizer step over all 250 steps. In the separate k sweep, step time is $1.68\times$, $1.63\times$, and $2.12\times$ OPD at $k = 2, 4, 8$, respectively. The m sweep changes both the query cost and the projected target through $\rho_i = \min(q_i B_i, \delta_i)$; its performance differences therefore combine changes in routing allocation and sampling variability. The k sweep also changes candidate support, with the reported increase improving Cons@8 while leaving Mean@8 unchanged.

Routing strength and lookahead cost. As α rises from 0.125 to 0.25 and 0.5, the recorded redirected fraction increases from 3.45% to 7.18% and 14.16%. The performance tradeoff appears in Figure 3. The β sweep records higher step time when successor evidence is enabled. These measurements describe overhead under the reported settings, not an equal-compute comparison.

E.4 Full training trajectories

Figure 5 complements the checkpoint comparison in Figure 2 with complete per-step logs from the same OPD and LKR runs. LKR retains higher token entropy through most of training, while its responses remain long. Both methods exhibit fluctuations near the end of training, which are retained in the figure. The routing panels show that LKR redirects a small fraction of the donor budget while applying routing to most of the logged donor events. This event rate is computed within the logged, budgeted event set, not over every negative-credit position in a rollout.

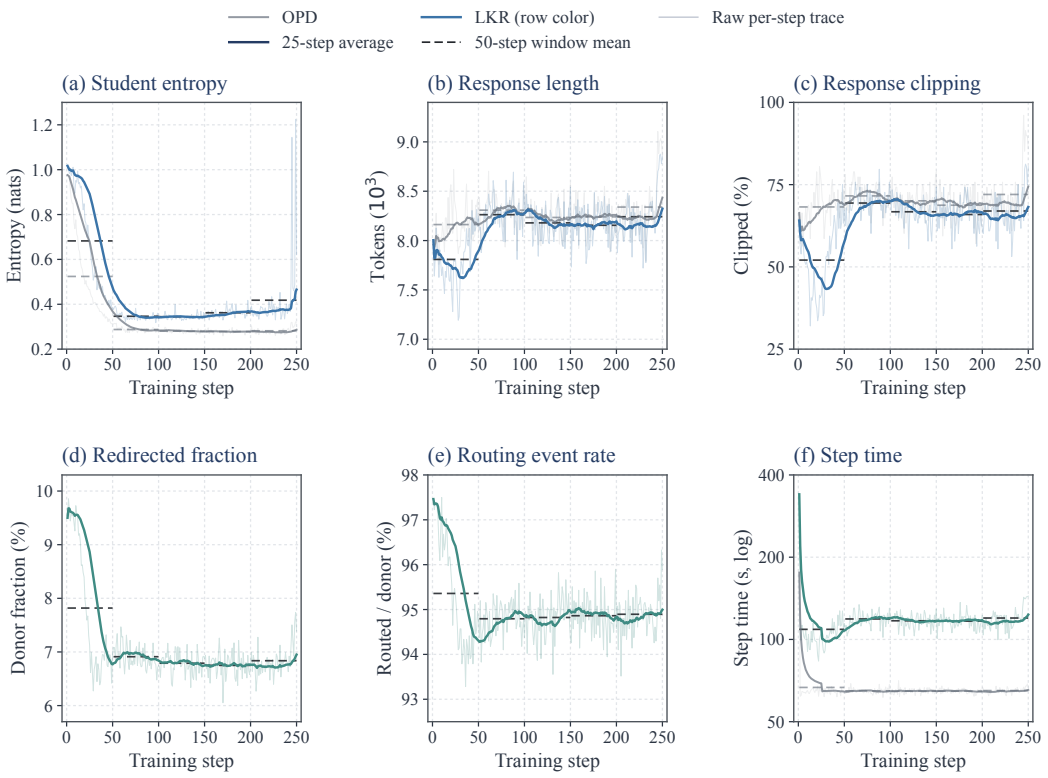

Figure 5: **Full training trajectories.** (a–c) Logged token entropy, mean response length, and response clipping rate. (d–e) LKR's mean redirected donor fraction and routed-to-donor event ratio. (f) Step time excluding checkpoint saving, with a logarithmic vertical axis. Gray solid lines denote OPD. For LKR, blue groups the training-state metrics in the upper row; teal groups routing behavior and step time below. Each panel uses a closed axis box and light dashed horizontal and vertical grid lines. Faint thin traces show all 250 per-step observations, bold lines show trailing 25-step averages, and short black (LKR) or gray (OPD) dashed segments show descriptive 50-step window means. The first 24 trailing averages use the available observations. No terminal observations are excluded. Panels (d–e) are LKR-specific. These are single-run training traces, not confidence intervals.

E.5 SFT ON TEACHER TRAJECTORIES (SEQKD)

Generation and filtering. We generate one Skywork-OR1-Math-7B response for each of the 47,909 clean training prompts, using the teacher's chat template, a maximum generation length of 9,216 tokens, and eight generation shards. The SFT baseline in Table 1 retains responses containing the closing `</think>` marker, without filtering by final-answer correctness. This leaves 20,612 traces (43.0%): 15,314 scored correct and 5,298 scored incorrect. A correctness-filtered subset contains only the former 15,314 traces; it is not the SFT baseline reported here. Each prompt has only one generated response, so there is no best-of-N selection.

Serialization and reproducibility. Each retained example pairs the original user question with the raw teacher response. The generation template opens the thinking span before generation; the SFT conversion restores the leading `<think>` tag and newline in the assistant message. Training uses a tokenizer/chat-template patch for DeepSeek-R1-Distill-Qwen-1.5B to handle this serialization. The patch is not currently included in the repository and is needed for exact reproduction. The original generation command was not retained, so its temperature, top-p, sampling seed, and tensor-parallel configuration could not be verified.

Training and checkpoint. We perform full-parameter SFT on the 20,612 retained traces for three epochs with eight-GPU FSDP. Table 5 lists the training settings. With global batch 256 and `drop_last=True`, each epoch contains $\lfloor 20{,}612/256 \rfloor = 80$ optimizer steps and omits 132 examples from its final partial batch. The reported checkpoint is the final step 240. This is a separate offline training budget from the 250-step on-policy comparison; Table 1 does not impose a shared step or compute budget on SFT.

Table 5: **SFT (SeqKD) training configuration.**

Setting	Value
Training traces	20,612; closing thinking marker required
Epochs / final checkpoint	3 / step 240
Global batch / microbatch per GPU	256 / 1
Maximum sequence length	10,240 tokens
Maximum tokens per GPU	20,480
Optimizer / weight decay	AdamW / 0.01
Learning rate / schedule	10^{-5} / constant
Gradient clipping norm threshold	1.0
Training mode	Full-parameter SFT; eight-GPU FSDP
Final partial batch	Dropped each epoch

E.6 IMPLEMENTED BASELINE OBJECTIVES AND SETTINGS

The on-policy baselines use the shared 1.5B student, 7B teacher, 47,909-prompt training split, and training configuration in Table 4. The settings below specify each implemented objective and gradient path. SeqKD is the separate offline SFT baseline in Appendix E.5.

OPD. For OPD and LKR, the realized-token log-ratio $\log[p_i(u_i)/t_i(u_i)]$ is clipped to $[-10, 10]$ and its negative is used as the detached policy-gradient advantage. LKR uses the same clipped donor coefficient γ_i to construct $B_i = \gamma_i(1 - p_i(u_i))$.

Top-K OPD. The teacher request returns its top-64 tokens, but the training objective uses only its top-32 support. Student and teacher probabilities are each renormalized on this same teacher-selected support before computing reverse KL. The gradient differentiates this distribution-matching loss directly, rather than using it as a detached policy-gradient advantage.

AOPD. We set the branching threshold to $\tau = 0$ and use the exact realized-token probabilities to evaluate $p_T(u) - p_S(u)$, independently of top-list membership. The negative branch uses supervised top-32 forward-KL matching; the other branch uses the sampled-token policy-gradient update.

EOPD. The entropy-gating parameters are $\tau = 0.8$ and $\alpha = 1$. Teacher entropy is computed exactly over the full vocabulary. The forward-KL component uses the normalized teacher top-16 distribution. Training combines the OPD policy-gradient term with entropy-gated supervised forward KL. These parameters are specific to EOPD, not the LKR surplus and lookahead settings.

LGR. The configuration samples 20 positions per sequence and applies a top-20 entropy approximation with threshold $\tau = 0.2$. The candidate pool is the union of student top-10 and teacher top-10 tokens, with at most eight candidates retained. The LGR coefficient is $\gamma = 1$, and the LGR component is enabled every other training step. Its signal enters through a policy-gradient advantage; no supervised distribution-matching gradient is added.