SAKI: MAXIMAL-COUPLING-ROUTED TEACHER SUPERVISION FOR ON-POLICY DISTILLATION

Anonymous authors
Paper under double-blind review

ABSTRACT

On-policy distillation (OPD) reduces train–test state mismatch by training a student on its own generated trajectories. However, a weak student may initially visit poor, teacher-misaligned prefixes, forcing the teacher to provide supervision on states that it would rarely generate under its own policy. Teacher-guided rollout policies improve the visited state distribution, but typically retain the same per-prefix reverse-KL objective. We introduce SAKI (Supervision Allocation with KL-constrained Interpolation), which **routes token-level supervision using realized accept/correction events from maximal coupling**. We construct a geometrically interpolated behavior policy inside a student-centered KL trust region and realize it through maximal coupling with the student. The coupling preserves a student proposal whenever possible and exposes a correction event precisely when realizing the guided policy requires an intervention. Accepted positions retain reverse-KL supervision, whereas correction positions switch to direct supervision on the teacher's highest-probability token. Because the **correction probability is exactly** $\mathrm{TV}(p_t, q_t)$, the same trust-region radius constrains rollout deviation and upper-bounds intervention and specialized-supervision frequency. An engine-resident speculative verifier preserves the exact-q trajectory distribution and coupling semantics while improving matched-workload rollout throughput by $4.22\times$. Across seven mathematical reasoning benchmarks, our method **improves the matched teacher-guided baseline in Mean@8 and Pass@8 for both 1.7B and 0.6B students**. Placement controls show that correction-triggered routing outperforms both equal-budget random and TV-weighted placement. Fixed-prefix analysis further shows persistent teacher alignment, with larger gains over random placement at higher initial student–teacher disagreement.

1 INTRODUCTION

Knowledge distillation (KD) transfers capabilities from a strong teacher model to a smaller student by matching its predictive behavior (Buciluǎ et al., 2006; Ba & Caruana, 2014; Hinton et al., 2015; Gou et al., 2021), with sequence-level distillation extending this idea to autoregressive generation (Kim & Rush, 2016). For autoregressive language models, however, conventional distillation suffers from a state-distribution mismatch: the student is often trained on fixed or teacher-generated prefixes, while at inference time it must condition on its own generations (Bengio et al., 2015; Ross et al., 2011). On-policy distillation (OPD) alleviates this mismatch by letting the student generate its own trajectories and querying the teacher on states actually visited by the student (Gu et al., 2024; Agarwal et al., 2024). This is particularly attractive for reasoning distillation, where an early generation decision can alter the entire subsequent trajectory and hence the states on which later supervision is provided.

Yet OPD's defining strength also creates an important limitation. *The effectiveness of distillation depends not only on what supervision the teacher provides, but also on the states at which that supervision is queried.* When the student is substantially weaker than the teacher, early mistakes can compound and drive its rollout toward prefixes that have low probability under the teacher's own behavior. Although the teacher distribution remains well-defined at such prefixes, the teacher is forced to continue from states that it would rarely generate under its own policy. The resulting conditional signal may therefore be less representative of the reasoning behavior that makes the teacher strong and less directly useful for capability transfer.

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.

This suggests that improving OPD requires considering both *where* teacher supervision is queried and *how* the student is updated at the resulting positions.

A natural approach is to guide the rollout distribution toward the teacher. Directly replacing student rollouts with teacher trajectories, however, introduces the opposite distribution shift: the resulting states may be too far from the student's current behavior to form an appropriate online learning distribution. Trust-Region Behavior Blending (TRB) (Plyusov et al., 2026) addresses this trade-off by constructing, at every prefix, an intermediate behavior distribution that moves toward the teacher while remaining inside a student-centered KL trust region. Given student distribution p_t and teacher distribution T_t, TRB constructs a teacher-guided behavior distribution q_t that moves toward T_t while satisfying a student-centered constraint $D_{\mathrm{KL}}(q_t \| p_t) \leq \epsilon$. TRB therefore addresses the *behavior-side* question of where supervision is queried while retaining the same per-prefix objective. This leaves a complementary *objective-side* question: should positions where the guided behavior retains a student proposal receive the same supervision as positions where it must override that proposal?

Reverse KL is student-weighted: tokens contribute in proportion to the student distribution. Consequently, teacher-preferred tokens with low student support receive weak gradient updates under RKL, highlighting the need for targeted direct supervision at high-conflict positions.

Sampling q_t through maximal coupling provides this distinction automatically. At each position, the student proposes from p_t; the coupling either retains the proposal or draws the residual correction required to realize q_t. We reuse this realized accept/correction event as an endogenous supervision router. Under maximal coupling, $\Pr(C_t = 1) = \mathrm{TV}(p_t, q_t)$, which is the minimum disagreement probability among couplings of p_t and q_t; the trust-region constraint further gives $\Pr(C_t = 1) \leq \sqrt{\epsilon/2}$. Section 3.3 formalizes these properties. We designate this framework SAKI (Supervision Allocation with KL-constrained Interpolation). Under this design, KL-constrained interpolation defines the target intermediate behavior distribution, while realized maximal-coupling events serve as an endogenous mechanism that adaptively routes token-level supervision.

This creates a natural distinction between two types of positions. At an *accepted position*, the student proposes behavior that remains compatible with the teacher-guided rollout. At a *correction position*, the student's proposed behavior cannot be retained and the intermediate policy must modify the trajectory. We hypothesize that these two regimes need not receive identical supervision.

We therefore retain the ordinary sampled-token reverse-KL (RKL) signal at accepted positions, while switching correction positions to direct supervision on the teacher's highest-probability token. Specifically, once maximal coupling identifies a correction, we query the teacher at the same prefix and optimize the negative log-likelihood of its Top-1 token. This exposes the student to the teacher-preferred mode precisely at positions where the guided rollout cannot retain the student proposal. Because corrections occur with probability $\mathrm{TV}(p_t, q_t)$, the resulting teacher supervision is sparse and follows a stochastic, conflict-adaptive routing pattern determined by the rollout coupling itself rather than by an external token-selection heuristic.

The residual correction token determines the subsequent rollout prefix, whereas the teacher Top-1 token determines the local parameter update. Thus, the same coupling process controls both trajectory construction and supervision routing.

Exact teacher-guided rollout requires online teacher distributions. We therefore use engine-resident speculative block verification to amortize teacher inference while preserving the exact-q trajectory distribution and the coupling semantics required by our objective.

Our contributions are summarized as follows:

- **Coupling-routed teacher-mode supervision.** We reuse realized correction events as an endogenous supervision router: accepted positions retain sampled-token RKL, while corrections receive direct supervision on the teacher's Top-1 token. Placement controls show gains over both count-matched random and TV-weighted conflict-aware placement, while fixed-prefix analysis shows persistent teacher support and larger gains over random placement at higher student–teacher conflict.

- **Minimal-intervention teacher-guided rollout.** We realize the TRB behavior policy through maximal coupling. The correction probability is exactly $\mathrm{TV}(p_t, q_t)$, the mini-

mum possible intervention probability among couplings with these marginals, and satisfies $\Pr(C_t = 1) \leq \sqrt{\epsilon/2}$ under the trust region.

- **Engine-resident exact-q rollout.** We implement maximal-coupling rollout with engine-resident speculative block verification, including exact residual correction and first-rejection commit/rollback. The system preserves the exact-q trajectory distribution and coupling semantics while providing a $4.22\times$ matched-workload speedup over the external-loop implementation.

2 Related Work

2.1 On-Policy Distillation and Teacher-Guided Rollouts

Knowledge distillation has been studied across both strong-to-weak and emerging weak-to-strong settings (Hinton et al., 2015; Chen et al., 2026), with sequence-level distillation extending the idea to autoregressive generation (Kim & Rush, 2016). For language models, fixed or teacher-generated trajectories create a mismatch between prefixes observed during training and those encountered when the student generates autonomously. MiniLLM (Gu et al., 2024) studies reverse-KL distillation for language generation, while Generalized Knowledge Distillation (GKD) (Agarwal et al., 2024) provides a framework for training students on their own generated outputs. These approaches motivate modern OPD, in which teacher supervision is delivered on states induced by the student's current policy.

On-policy training reduces train–test state mismatch, but it also makes the training state distribution depend on the current student's quality. TRB (Plyusov et al., 2026) addresses this issue by replacing pure student rollout with a teacher-guided behavior distribution constrained by a student-centered KL trust region, while retaining the same reverse-KL objective at visited prefixes. We adopt the same intermediate-policy construction, but additionally exploit the accept/correction information exposed when that policy is realized through maximal coupling. Recent work has also explored fine-grained distillation objectives and token-level supervision, including selective KL objectives and reflective credit assignment (Xing et al., 2026; Wei et al., 2026). Our focus is on whether the guided rollout process itself can provide the routing signal for such specialized supervision.

2.2 Speculative Decoding and Distillation

Speculative decoding accelerates autoregressive generation by allowing a draft model to propose multiple tokens that are verified in parallel by a stronger target model (Stern et al., 2018; Leviathan et al., 2023; Chen et al., 2023). Accepted prefixes can be committed in blocks, reducing the number of expensive serial target-model decoding steps while preserving the desired sampling distribution.

Draft-OPD (Lei et al., 2026) connects speculative verification with on-policy distillation and demonstrates that verification outcomes can provide useful training structure in addition to computational acceleration. Our use of speculative execution serves a different role: the student is the capability-distilled model itself, and speculative verification is used to efficiently realize a separate teacher-guided intermediate distribution q. The resulting maximal-coupling decisions are then reused to route token-level supervision. Thus, speculative execution in our framework simultaneously amortizes teacher inference and exposes the coupling events that connect trajectory construction with the distillation objective.

3 Method

Figure 1 summarizes one rollout-and-update step. At prefix h_t, the rollout student p_t and teacher T_t define the trust-region behavior distribution q_t. Maximal coupling then realizes q_t while exposing an accept/correction event that determines the rollout token and routes the local distillation objective.

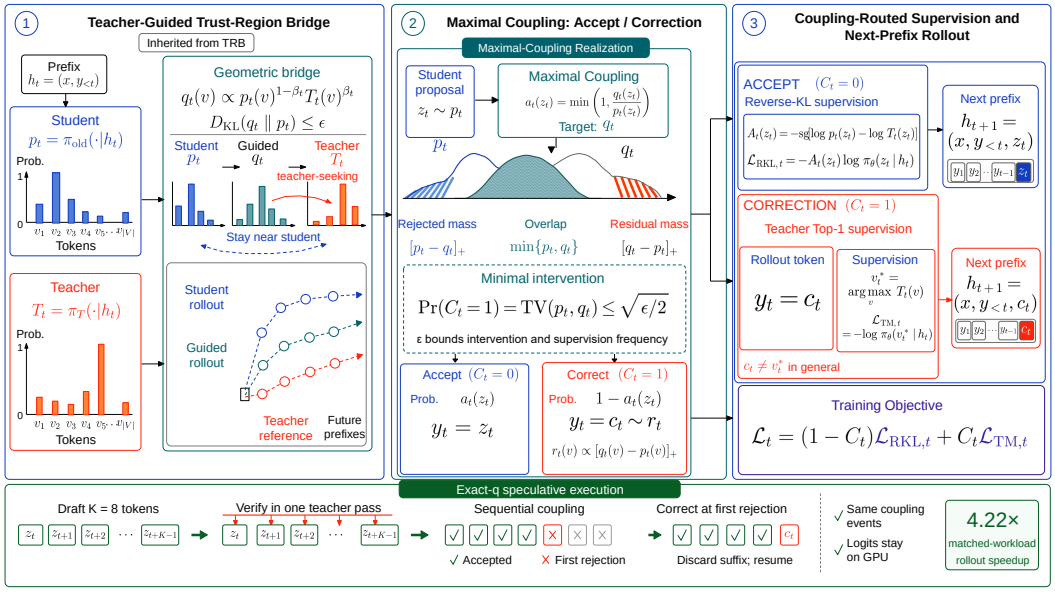


Figure 1: **Overview of SAKI and its exact-q execution.** The TRB bridge defines q_t; maximal coupling either retains a student proposal with RKL supervision or draws a residual rollout correction and routes teacher-Top-1 supervision. The residual token controls the next prefix, whereas the teacher mode controls the update. Bottom: an engine-resident K-token verifier commits proposals only through the first rejection, commits an exact residual correction there, and discards the invalid suffix.

3.1 Problem Setup

Let $h_t = (x, y_{<t})$ denote the prefix at decoding step t. We write $p_t(v) = \pi_{\text{old}}(v \mid h_t)$ for the frozen student behavior distribution and $T_t(v) = \pi_T(v \mid h_t)$ for the fixed teacher distribution. The trainable student is denoted by π_θ.

Standard student-rollout OPD samples $y_t \sim p_t$ and uses the detached log-ratio advantage

$$A_t = -\,\text{sg}[\log p_t(y_t) - \log T_t(y_t)]\,.$$

Under exact student on-policy sampling, this gives the standard one-sample score-function estimator associated with reverse KL (Williams, 1992); Appendix C provides the derivation. In our teacher-guided rollout, we reuse the same sampled log-ratio signal on student proposals retained by maximal coupling. As shown in Section 3.4, the resulting RKL supervision acts on the student–guided overlap mass $\min\{p_t, q_t\}$, while correction positions receive the teacher-mode objective introduced below.

3.2 Teacher-Guided Trust-Region Rollout

Following TRB (Plyusov et al., 2026), we construct the teacher-guided behavior policy by geometric interpolation between the student and teacher:

$$q_{\beta,t}(v) = \frac{p_t(v)^{1-\beta_t} T_t(v)^{\beta_t}}{Z_t(\beta_t)}, \qquad Z_t(\beta_t) = \sum_u p_t(u)^{1-\beta_t} T_t(u)^{\beta_t}. \tag{1}$$

At every position, we choose the largest feasible $\beta_t \in [0, 1]$ satisfying

$$D_{\text{KL}}(q_{\beta,t} \| p_t) \le \epsilon. \tag{2}$$

Thus, $\epsilon = 0$ recovers $q_t = p_t$, while a sufficiently large trust region permits $q_t = T_t$. For completeness, Appendix B outlines the derivation of this geometric bridge following TRB (Plyusov et al., 2026).

3.3 MAXIMAL COUPLING AND CORRECTION EVENTS

To sample exactly from q_t while exposing the relationship between student and guided behavior, we use maximal coupling (Lindvall, 2002).

The student first proposes

$$z_t \sim p_t. \tag{3}$$

The proposal is accepted with probability

$$a_t(z_t) = \min\left(1, \frac{q_t(z_t)}{p_t(z_t)}\right). \tag{4}$$

If accepted,

$$y_t = z_t, \qquad C_t = 0. \tag{5}$$

Otherwise, we draw a correction token from the positive residual distribution

$$r_t(v) = \frac{[q_t(v) - p_t(v)]_+}{\sum_u [q_t(u) - p_t(u)]_+}, \tag{6}$$

and set

$$y_t = c_t \sim r_t, \qquad C_t = 1. \tag{7}$$

The following proposition gives the key distributional semantics of the correction event.

Proposition 1 (Distributional semantics of coupling corrections)**.** *For maximal coupling between distributions p and q, the final output has marginal distribution q, and*

$$\Pr(C = 1) = \mathrm{TV}(p, q) = 1 - \sum_v \min\{p(v), q(v)\}. \tag{8}$$

The conditional rejected-proposal and residual-correction distributions are derived in Appendix G.

Proposition 2 (Minimal-intervention property)**.** *Let $\Gamma(p, q)$ denote the set of all couplings with marginals p and q. For any $\gamma \in \Gamma(p, q)$,*

$$\Pr_\gamma(Z \neq Y) \geq \mathrm{TV}(p, q), \tag{9}$$

with equality under maximal coupling. Thus, maximal coupling realizes the guided marginal with the minimum possible intervention probability.

Corollary 1 (Trust-region control of intervention rate)**.** *If the guided distribution satisfies*

$$D_{\mathrm{KL}}(q_t \| p_t) \leq \epsilon, \tag{10}$$

then the maximal-coupling correction probability satisfies

$$\Pr(C_t = 1) = \mathrm{TV}(p_t, q_t) \leq \sqrt{\frac{1}{2} D_{\mathrm{KL}}(q_t \| p_t)} \leq \sqrt{\frac{\epsilon}{2}}. \tag{11}$$

Thus, ϵ controls not only the distributional distance of the rollout from the student but also the maximum local probability that the rollout must override a student proposal.

Appendix J.2 reports the empirical correction trajectory under the annealed trust-region schedule: the observed correction probability remains below the Pinsker upper bound and vanishes when ϵ reaches zero.

The proof is given in Appendix G. Proposition 1 shows that a correction transfers the trajectory from probability mass overrepresented by the student relative to q toward mass underrepresented by the student relative to q. Hence, correction is not an arbitrary gating heuristic: it has an explicit distributional meaning.

Unlike scalar conflict measures such as TV or KL, C_t is a realized, proposal-dependent event: it identifies positions where the sampled student proposal cannot be retained under the minimum-intervention coupling that exactly realizes q_t.

For the geometric bridge, the positive residual is supported only on tokens whose teacher-to-student likelihood ratio exceeds a prefix-dependent threshold; Appendix H gives the derivation.

3.4 COUPLING-ROUTED SUPERVISION

SAKI allocates supervision using the realized coupling indicator C_t. Accepted positions retain the sampled-token RKL update, whereas correction positions receive teacher-mode supervision.

For a token v at prefix h_t, define the sampled-token RKL loss

$$L_{\mathrm{RKL},t}(v) = -A_t(v) \log \pi_\theta(v \mid h_t), \tag{12}$$

where

$$A_t(v) = -\operatorname{sg}\left[\log p_t(v) - \log T_t(v)\right]. \tag{13}$$

At an accepted position, we evaluate this loss on the retained proposal $v = z_t = y_t$. We retain the ordinary RKL signal at accepted positions and replace it with teacher-mode supervision when $C_t = 1$.

Teacher-mode supervision. At a correction position, let

$$v_t^* = \arg\max_v T_t(v) \tag{14}$$

denote the teacher's highest-probability token at the same prefix. We replace the RKL term with

$$L_{\mathrm{TM},t} = -\log \pi_\theta(v_t^* \mid h_t). \tag{15}$$

Thus, correction detection and correction supervision play decoupled roles: maximal coupling determines *where* specialized supervision is activated, while the teacher mode determines *what* the student learns. We adopt the deterministic teacher mode ($v_t^* = \arg\max_v T_t(v)$) rather than stochastic teacher sampling ($\tilde{v} \sim T_t$), as the mode provides a sharp, low-variance target precisely at high-conflict positions. Empirically, teacher-mode supervision outperforms stochastic teacher sampling by $+0.62$ Mean@8 and $+1.56$ Pass@8 on the 1.7B student (Appendix F, Table 8), confirming the benefit of mode-seeking updates under coupling-detected divergence.

This design naturally decouples exploration steering from parameter optimization: the residual token c_t steers the autoregressive prefix along the guided target distribution q_t, whereas the teacher mode v_t^* provides a low-variance, deterministic anchor for parameter updates at divergence points.

Let $M_t \in \{0, 1\}$ denote the valid response-token mask and let $N_{\mathrm{valid}} = \sum_t M_t$. Our coupling-aware objective is

$$L = \frac{1}{N_{\mathrm{valid}}} \left[\sum_t M_t (1 - C_t) L_{\mathrm{RKL},t} + \sum_t M_t C_t L_{\mathrm{TM},t} \right]. \tag{16}$$

At a fixed prefix, suppressing the time index for clarity, Proposition 1 gives

$$\mathbb{E}\left[(1 - C) L_{\mathrm{RKL}}(Z)\right] = \sum_v \min\{p(v), q(v)\} L_{\mathrm{RKL}}(v). \tag{17}$$

Thus, accepted-position RKL supervision acts exactly on the student–guided overlap mass $\min\{p, q\}$. Correction positions form the complementary branch of the same maximal-coupling realization: at these positions, the sampled student proposal cannot be retained while exactly realizing the guided rollout distribution q. We reuse this realized branch to switch from RKL to teacher-mode supervision, without introducing an additional token-selection rule or routing threshold.

In our main training schedule, this correction supervision is transient: we anneal ϵ to zero so that the method eventually returns exactly to student-rollout RKL training. Section 4.3 analyzes the distributional effect of this transition.

3.5 EFFICIENT ENGINE-RESIDENT EXACT-q ROLLOUT

Token-wise exact coupling requires both student and teacher distributions at every generated position and is therefore substantially more expensive than student-only OPD. We amortize this online computation with speculative block verification. At each wave, the frozen rollout student drafts up to K tokens from the current true prefix, and the teacher evaluates the corresponding proposal prefixes in one batched verification. Inside the inference engine, we construct q_t for all block positions,

solve the trust-region coefficients β_t in batch, and evaluate maximal-coupling acceptance from left to right.

The engine commits the longest consecutively accepted proposal prefix. If the first rejection occurs at position j, it samples $c_j \propto [q_j - p_j]_+$, commits that correction, discards all later speculative tokens and KV states, and resumes from the corrected prefix. Student and teacher full-vocabulary logits remain on GPU; the training loop receives only compact token-aligned metadata, including committed tokens, the correction mask, selected log-probabilities, and teacher-mode targets. Because proposal prefixes coincide with true prefixes up to the first rejection and no post-correction speculative state is reused, this execution has the same autoregressive law and coupling events as sequential exact-q sampling. Appendix I proves exactness, and Appendix K gives implementation and validation details.

4 Experiments

4.1 Experimental Setup

Models and shared protocol. We distill Qwen3-0.6B-Base and Qwen3-1.7B-Base students (Yang et al., 2025) from the same Qwen3-4B-Base-GRPO teacher on DAPO-Math-17K (Yu et al., 2025). Within each student scale, all methods use the same student and teacher checkpoints, training prompts, 200-step budget, rollout batch of 64 prompts with eight responses per prompt, and student-only evaluation protocol. Complete optimization, sampling, and model-compatibility details are in Appendix A.

Baselines. Where an RKL term is present, OPD, ExOPD, TRB, Random-TM, TV-Weighted-TM, and Ours use the same sampled-token log-ratio update. ExOPD (Yang et al., 2026) follows the official G-OPD code with $\lambda = 1.25$, `only_reverse_kl_advantages=True`, and the initial student as the fixed reference. SKD (Xu et al., 2025) is reproduced using its canonical Top-25 speculative rollout and full-distribution KL objective under our unified training and evaluation protocol.

Random-TM and TV-Weighted-TM are placement controls for the 1.7B student. Both match the number of teacher-mode updates used by Ours on each exact-q trajectory. Random-TM uses count-matched random placement, whereas TV-Weighted-TM samples positions without replacement according to the local $\mathrm{TV}(p_t, q_t)$ score. For TV-Weighted-TM, the realized correction mask determines only the per-trajectory supervision budget, not the selected locations.

TRB, Random-TM, TV-Weighted-TM, and Ours share an exact-q rollout with $K = 8$ and $\epsilon : 0.02 \to 0$ linearly over the first 50 steps, directly adopting the established schedule from Plyusov et al. (2026). To strictly isolate the algorithmic impact of supervision routing from confounding trajectory dynamics, all exact-q variants share the identical rollout schedule established in prior work (Plyusov et al., 2026). This strictly controlled protocol ensures that all empirical improvements are directly attributable to token-level supervision allocation rather than behavioral tuning artifacts.

Evaluation. We evaluate MATH-500 (Hendrycks et al., 2021; Lightman et al., 2024), HMMT-Feb26, AIME 2026, AIME 2025, AMC 2023, Minerva Math (Lewkowycz et al., 2022), and OlympiadBench (He et al., 2024). We sample eight responses per problem at temperature 1.0 with maximum response length 16,384. Mean@8 is the average correctness over the eight samples, Pass@8 is the fraction of problems with at least one correct sample, and Average is the unweighted mean over the seven displayed benchmarks. Full decoding settings are in Appendix A.

4.2 Main Results

Performance summary. Tables 1 and 2 show that Ours achieves the best macro Mean@8 and Pass@8 at both student scales. Relative to TRB, the 1.7B student improves from 27.9/44.6 to 29.0/47.5 ($+1.1/+2.9$ points), and the 0.6B student improves from 17.2/33.6 to 18.4/35.6 ($+1.2/+2.0$ points). Relative to SKD, the gains are $+4.5/+6.3$ points for 1.7B and $+6.0/+6.0$ points for 0.6B. Mean@8 improves over TRB in 13 of 14 student–benchmark pairs.

Table 1: **Mean@8 accuracy** (%). Average is the mean over the seven benchmarks; **bold** and **underline** denote the best and second-best trained method within each student size, respectively.

Method	MATH-500	HMMT	AIME26	AIME25	AMC23	Minerva	Olympiad	Average
Teacher (4B)	84.1	15.2	19.2	19.6	60.9	38.0	51.9	41.3
Student (1.7B)	10.2	0.0	0.4	0.4	3.8	3.1	3.2	3.0
OPD	70.7	2.3	6.3	8.3	42.8	27.2	33.7	27.3
ExOPD	69.5	2.7	5.4	7.9	44.4	27.5	32.1	27.1
SKD	64.3	3.4	5.4	4.6	40.9	23.9	29.1	24.5
TRB	70.7	4.9	6.3	8.8	43.1	26.5	34.7	<u>27.9</u>
SAKI (Ours)	70.9	5.7	7.9	8.3	47.2	28.1	34.9	**29.0**
Student (0.6B)	6.2	0.4	0.0	0.0	0.9	2.2	2.1	1.7
OPD	52.7	0.8	0.4	2.5	26.9	15.4	20.7	17.1
ExOPD	52.1	1.5	0.8	1.7	31.3	15.9	20.8	<u>17.7</u>
SKD	41.7	1.1	0.4	0.4	19.1	11.0	13.3	12.4
TRB	52.0	1.5	0.4	2.5	28.4	14.9	20.9	17.2
SAKI (Ours)	52.6	1.9	1.7	4.2	31.9	15.3	21.2	**18.4**

Table 2: **Pass@8 accuracy** (%). Average is the mean over the seven benchmarks; **bold** and **underline** denote the best and second-best trained method within each student size, respectively.

Method	MATH-500	HMMT	AIME26	AIME25	AMC23	Minerva	Olympiad	Average
Teacher (4B)	94.4	24.2	30.0	26.7	97.5	57.4	70.2	57.2
Student (1.7B)	49.6	0.0	3.3	3.3	20.0	19.9	18.4	16.4
OPD	90.0	9.1	20.0	20.0	72.5	46.3	57.2	<u>45.0</u>
ExOPD	87.6	6.1	20.0	20.0	75.0	45.6	56.7	44.4
SKD	86.6	6.1	16.7	13.3	67.5	44.5	53.6	41.2
TRB	90.4	9.1	16.7	23.3	70.0	44.9	57.9	44.6
SAKI (Ours)	90.0	18.2	20.0	23.3	75.0	47.4	58.5	**47.5**
Student (0.6B)	35.8	3.0	0.0	0.0	7.5	13.2	12.9	10.3
OPD	78.2	3.0	3.3	6.7	52.5	34.2	43.6	31.6
ExOPD	77.2	6.1	6.7	6.7	57.5	36.0	41.9	33.2
SKD	71.0	6.1	3.3	3.3	57.5	31.3	34.4	29.6
TRB	77.4	9.1	3.3	6.7	60.0	34.9	43.9	<u>33.6</u>
SAKI (Ours)	81.2	9.1	10.0	13.3	60.0	32.0	43.3	**35.6**

Placement controls. For the 1.7B student, Random-TM reaches 28.30 Mean@8 and 45.50 Pass@8. TV-Weighted-TM matches the same per-trajectory teacher-mode budget but weights placement by the local $\mathrm{TV}(p_t, q_t)$ score, reaching 28.24 Mean@8 and 46.77 Pass@8. Ours reaches 29.00/47.50, exceeding TV-Weighted-TM by $+0.76$ Mean@8 and $+0.73$ Pass@8. Thus, scalar conflict-aware placement is useful, while realized coupling corrections provide a stronger routing signal under the matched setup. Aggregate values are reported in Table 6; Figure 3(c) visualizes the Random-TM decomposition.

4.3 Mechanistic Analysis: Persistent Alignment and Conflict-Adaptive Teacher Support

Matched fixed-prefix probe. We compare matched TRB, Random-TM, and Ours runs on a frozen probe of 2,048 positions from 213 prompts. We track student probability on the teacher Top-1 token (C_1) and Top-16 set (C_{16}), and stratify positions by initial $D_{\mathrm{KL}}(T\|S)$. Differences are prompt-equal, and confidence intervals use prompt-clustered bootstrap. Correction-triggered supervision ends at step 51; full probe construction and statistics are in Appendix E.

Teacher-mode supervision establishes persistent alignment. Figure 2(a) shows that correction-triggered teacher-mode supervision rapidly increases teacher-supported probability mass during the correction-active phase. At step 50, Ours exceeds TRB by 4.63 percentage points in C_1 and 4.54 points in C_{16}. After teacher-mode supervision is disabled at step 51, these gains remain visible: at step 200, after 149 RKL-only steps, the corresponding advantages are still 4.26 and 2.94 points. Additional divergence and entropy statistics are reported in Appendix E.

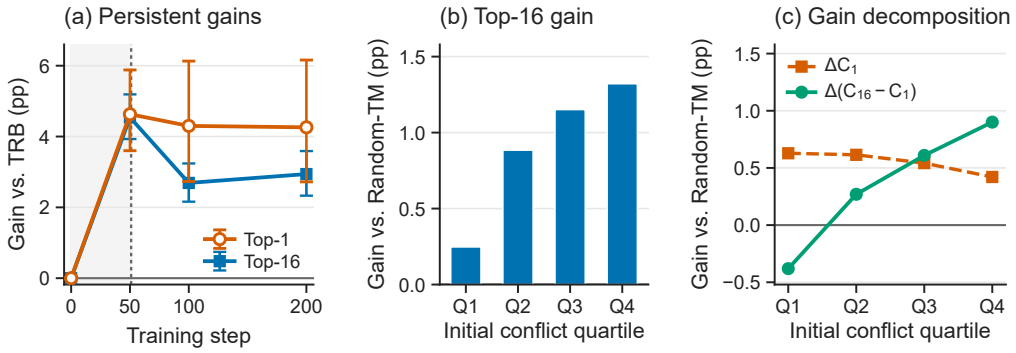


Figure 2: **Persistent and conflict-adaptive teacher support.** C_1 is student probability on the teacher Top-1 token, and C_{16} is student probability mass on the teacher Top-16 set. (a) Gains over TRB during training; shading marks the correction-active phase and the dotted line marks step 51. Error bars are 95% prompt-clustered bootstrap CIs. (b,c) Step-200 gains over Random-TM by initial $D_{\mathrm{KL}}(T\|S)$ quartile (Q1 lowest, Q4 highest). The Q4–Q1 contrasts are $+1.07$ pp for ΔC_{16} and $+1.28$ pp for $\Delta(C_{16} - C_1)$.

Table 3: **Exact-q rollout** throughput under the matched workload above.

Backend	Tok/s	Relative
Student-only reference	7,760	–
External-loop exact-q K8	776	1.00×
Engine-resident exact-q K8	3,276	**4.22**×

Coupling-based placement is conflict adaptive. The equal-budget Random-TM control tests whether correction-based placement provides benefits beyond teacher-mode supervision alone. Relative to Random-TM, correction-triggered placement increases C_1 by 0.648 points and C_{16} by 0.816 points at step 200. More importantly, Figure 2(b,c) shows that the C_{16} advantage grows from $+0.25$ points in the lowest-conflict quartile to $+1.32$ points in the highest-conflict quartile (Q4–Q1: $+1.07$ points; 95% CI: [0.87, 1.29]). This increase is primarily carried by teacher-supported non-argmax tokens (Q4–Q1: $+1.28$ points), while the corresponding C_1 interaction is not significant. These results show that maximal-coupling corrections provide an effective conflict-adaptive routing signal beyond the effect of supervision budget alone.

4.4 Exact-q Rollout Efficiency

We measure committed response tokens per generation wall-clock second under the same 64-prompt $\times$ 8-response, maximum-length 7,168, $K = 8$ workload. The engine-resident backend achieves 3,276 tokens/s compared to 776 tokens/s for the external-loop implementation—delivering a $4.22\times$ speedup under identical workloads. Remarkably, despite executing full online teacher verification and batched trust-region interpolation at every wave, the engine sustains over 42% of the theoretical throughput ceiling defined by unguided student-only generation (Table 3). Appendix K reports timing breakdowns, model/cache lifecycle, and correctness checks.

5 Conclusion

We introduce SAKI, which realizes a teacher-guided trust-region policy through maximal coupling and routes teacher-mode supervision at correction events. It achieves the best macro Mean@8 and Pass@8 at both student scales. Placement controls and fixed-prefix analysis support correction-based, conflict-adaptive routing. An engine-resident exact-q verifier preserves sampling semantics while providing a $4.22\times$ matched-workload speedup.

AI USE STATEMENT

In this work, generative AI tools were used to assist with checking and refining mathematical proofs, code development, and language editing of the manuscript. All mathematical arguments were independently verified by the authors. AI-assisted code was reviewed and tested by the authors for correctness, and all AI-assisted textual edits were reviewed before inclusion in the manuscript. The authors take full responsibility for the final content of this work, including text, claims, code, and other artifacts produced with the aid of generative AI.

REPRODUCIBILITY STATEMENT

We provide anonymous supplementary code containing a reference implementation of the core components of SAKI. Appendix A reports the training and evaluation configuration, Appendices G and I provide the coupling and exact-execution derivations, and Appendix K describes the systems implementation and validation details.

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 *International Conference on Learning Representations*, volume 2024, pp. 21246–21263, 2024.

Lei J Ba and Rich Caruana. Do deep nets really need to be deep? *Advances in neural information processing systems*, 27, 2014.

Samy Bengio, Oriol Vinyals, Navdeep Jaitly, and Noam Shazeer. Scheduled sampling for sequence prediction with recurrent neural networks. *Advances in neural information processing systems*, 28, 2015.

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, 2006.

Charlie Chen, Sebastian Borgeaud, Geoffrey Irving, Jean-Baptiste Lespiau, Laurent Sifre, and John Jumper. Accelerating large language model decoding with speculative sampling. *arXiv preprint arXiv:2302.01318*, 2023.

Zehao Chen, Gongxun Li, Tianxiang Ai, Zixuan Huang, Xiaodong Liu, Yifei Li, Wang Zhou, Fuzhen Zhuang, Xianglong Liu, Jianxin Li, et al. Weak-driven learning: How weak agents make strong agents stronger. *arXiv preprint arXiv:2602.08222*, 2026.

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

Bradley Efron, Robert J Tibshirani, et al. *An introduction to the bootstrap*. Boca Raton, Florida, 2000.

Jianping Gou, Baosheng Yu, Stephen J Maybank, and Dacheng Tao. Knowledge distillation: A survey. *International journal of computer vision*, 129(6):1789–1819, 2021.

Yuxian Gu, Li Dong, Furu Wei, and Minlie Huang. Minillm: Knowledge distillation of large language models. In *International Conference on Learning Representations*, volume 2024, pp. 32694–32717, 2024.

Chaoqun He, Renjie Luo, Yuzhuo Bai, Shengding Hu, Zhen Thai, Junhao Shen, Jinyi Hu, Xu Han, Yujie Huang, Yuxiang Zhang, et al. Olympiadbench: A challenging benchmark for promoting agi with olympiad-level bilingual multimodal scientific problems. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 3828–3850, 2024.

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

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

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

Haodi Lei, Yafu Li, Haoran Zhang, Shunkai Zhang, Qianjia Cheng, Xiaoye Qu, Ganqu Cui, Bowen Zhou, Ning Ding, Yun Luo, et al. Draft-opd: On-policy distillation for speculative draft models. *arXiv preprint arXiv:2605.29343*, 2026.

Yaniv Leviathan, Matan Kalman, and Yossi Matias. Fast inference from transformers via speculative decoding. In *International conference on machine learning*, pp. 19274–19286. PMLR, 2023.

Aitor Lewkowycz, Anders Andreassen, David Dohan, Ethan Dyer, Henryk Michalewski, Vinay Ramasesh, Ambrose Slone, Cem Anil, Imanol Schlag, Theo Gutman-Solo, et al. Solving quantitative reasoning problems with language models. *Advances in neural information processing systems*, 35:3843–3857, 2022.

Yaxuan Li, Yuxin Zuo, Bingxiang He, Jinqian Zhang, Chaojun Xiao, Cheng Qian, Tianyu Yu, Huanang Gao, Wenkai Yang, Zhiyuan Liu, et al. Rethinking on-policy distillation of large language models: Phenomenology, mechanism, and recipe. *arXiv preprint arXiv:2604.13016*, 2026.

Hunter Lightman, Vineet Kosaraju, Yuri 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*, volume 2024, pp. 39578–39601, 2024.

Torgny Lindvall. *Lectures on the coupling method*. Courier Corporation, 2002.

Daniil Plyusov, Alexey Gorbatovski, Alexey Malakhov, Nikita Balagansky, Boris Shaposhnikov, Daria Korotyshova, and Daniil Gavrilov. Trust-region behavior blending for on-policy distillation. *arXiv preprint arXiv:2605.31159*, 2026.

Stéphane 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*, pp. 627–635. JMLR Workshop and Conference Proceedings, 2011.

Mitchell Stern, Noam Shazeer, and Jakob Uszkoreit. Blockwise parallel decoding for deep autoregressive models. *Advances in Neural Information Processing Systems*, 31, 2018.

Zhenlin Wei, Pu Jian, Yingzhuo Deng, Xiaohan Wang, Jiajun Chai, Zhexin Hu, Wei Lin, Shanbin Zhang, and Guojun Yin. Amr-sd: Asymmetric meta-reflective self-distillation for token-level credit assignment. *arXiv preprint arXiv:2605.18529*, 2026.

Ronald J Williams. Simple statistical gradient-following algorithms for connectionist reinforcement learning. *Machine learning*, 8(3):229–256, 1992.

Xingrun Xing, Haoqing Wang, Boyan Gao, Ziheng Li, and Yehui Tang. Trust region on-policy distillation. *arXiv preprint arXiv:2606.01249*, 2026.

Wenda Xu, Rujun Han, Zifeng Wang, Long Le, Dhruv Madeka, Lei Li, William Wang, Rishabh Agarwal, Chen-Yu Lee, and Tomas Pfister. Speculative knowledge distillation: Bridging the teacher-student gap through interleaved sampling. In *International Conference on Learning Representations*, volume 2025, pp. 64616–64646, 2025.

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.

Table 4: **Training configuration** used to construct the **Qwen3-4B-Base-GRPO teacher**, following Li et al. (2026).

Setting	Value
Initialization	Qwen3-4B-Base
RL algorithm	GRPO
Training data	Processed DAPO-Math-17K
Training epochs	1
Prompt batch size	64
Reported micro-batch size	64
Responses per prompt	8
Maximum prompt length	1,024
Maximum response length	7,168
Learning rate	1×10^{-6}
Sampling temperature	1.0
Top-p	1.0
Repetition penalty	1.0
KL regularization	None
Additional KL coefficient	0
Loss aggregation	Token mean
Thinking template	Disabled

Wenkai Yang, Weijie Liu, Ruobing Xie, Kai Yang, Saiyong Yang, and Yankai Lin. Learning beyond teacher: Generalized on-policy distillation with reward extrapolation. *arXiv preprint arXiv:2602.12125*, 2026. URL https://arxiv.org/abs/2602.12125.

Qiying Yu, Zheng Zhang, Ruofei Zhu, Yufeng Yuan, Xiaochen Zuo, Yu Yue, Weinan Dai, Tiantian Fan, Gaohong Liu, Lingjun Liu, et al. DAPO: An open-source LLM reinforcement learning system at scale. *arXiv preprint arXiv:2503.14476*, 2025. URL https://arxiv.org/abs/2503.14476.

A Teacher Construction, Model Compatibility, and Training Details

Teacher construction. We initialize the teacher from Qwen3-4B-Base and reproduce the Qwen3-4B-Base-GRPO construction of Li et al. (2026). We train the teacher with GRPO on the processed DAPO-Math-17K dataset. Following their data format, each question is augmented with the instruction "Please reason step by step, and put your final answer within \boxed{}." We use exactly the reported GRPO hyperparameter recipe, summarized in Table 4.

The teacher uses the released rule-based mathematical reward function rather than a learned reward model. Format reward and reference-model KL regularization are disabled.

Model compatibility. Our rollout and maximal-coupling implementation operate at the exact token level. The publicly released Qwen3 Base and post-trained checkpoints use different default termination-token conventions. We therefore use Base-to-Base teacher–student pairs so that vocabulary, termination behavior, and response masking remain naturally aligned. All prompts are formatted with `enable_thinking=False`.

Matched student-training protocol. The 0.6B and 1.7B students use the same training protocol; the student parameter count is the primary experimental difference. Both models are trained on the same 17,917 DAPO-Math prompts for 200 optimization steps. Table 5 reports the shared configuration.

B Derivation of the Trust-Region Geometric Bridge

This section reproduces the geometric bridge underlying TRB (Plyusov et al., 2026) for completeness.

Table 5: **Shared training configuration** for the 0.6B and 1.7B students.

Setting	Value
Students	Qwen3-0.6B / 1.7B Base
Teacher	Qwen3-4B-Base-GRPO
Training	200 steps; AdamW; LR 1×10^{-6}
Weight decay / grad clip	0.01 / 1.0
Rollout batch	64 prompts $\times$ 8 responses $= 512$
Sampling	$T = 1.0$, top-$p = 1.0$, no top-k
Maximum lengths	7,168
Loss aggregation	Token mean over valid response tokens
Guided rollout(Ours)	$K = 8$, $\epsilon : 0.02 \to 0$ over 50 steps

At a fixed prefix, consider the problem

$$\begin{aligned} \min_{q} \quad & D_{\mathrm{KL}}(q\|T) \\ \text{s.t.} \quad & D_{\mathrm{KL}}(q\|p) \le \epsilon, \\ & \sum_{v} q(v) = 1. \end{aligned} \tag{18}$$

Its Lagrangian is

$$\mathcal{L}(q, \eta, \xi) = D_{\mathrm{KL}}(q\|T) + \eta \left(D_{\mathrm{KL}}(q\|p) - \epsilon\right) + \xi \left(\sum_{v} q(v) - 1\right), \tag{19}$$

where $\eta \ge 0$.

Expanding the KL terms gives

$$\begin{aligned} \mathcal{L} = & \sum_{v} q(v) \left[\log q(v) - \log T(v)\right] \\ & + \eta \sum_{v} q(v) \left[\log q(v) - \log p(v)\right] + \text{const}. \end{aligned} \tag{20}$$

Taking the derivative with respect to $q(v)$,

$$(1 + \eta) \log q(v) - \log T(v) - \eta \log p(v) + \text{const} = 0. \tag{21}$$

Hence,

$$q(v) \propto p(v)^{\frac{\eta}{1+\eta}} T(v)^{\frac{1}{1+\eta}}. \tag{22}$$

Defining

$$\beta = \frac{1}{1 + \eta} \in (0, 1], \tag{23}$$

we obtain

$$q_{\beta}(v) = \frac{p(v)^{1-\beta} T(v)^{\beta}}{\sum_{u} p(u)^{1-\beta} T(u)^{\beta}}. \tag{24}$$

If the teacher itself lies inside the trust region, the constraint is inactive, $\eta = 0$, and $\beta = 1$, giving $q = T$. As the trust-region radius approaches zero, $\eta \to \infty$, $\beta \to 0$, and $q \to p$.

C Single-Sample Estimation of Reverse KL

At a fixed prefix, the reverse KL from student p_θ to teacher T is

$$D_{\mathrm{KL}}(p_\theta\|T) = \sum_{v} p_\theta(v) \left[\log p_\theta(v) - \log T(v)\right]. \tag{25}$$

For a sample $Y \sim p_\theta$, define

$$k_1(Y) = \log p_\theta(Y) - \log T(Y). \tag{26}$$

Taking the expectation under the student distribution gives

$$\mathbb{E}_{Y\sim p_\theta}[k_1(Y)] = \sum_v p_\theta(v)\left[\log p_\theta(v) - \log T(v)\right] \tag{27}$$

$$= D_{\mathrm{KL}}(p_\theta \| T). \tag{28}$$

Thus, the sampled log-ratio is a one-sample Monte Carlo estimator of the full-vocabulary reverse-KL value under exact student sampling.

Since the teacher is fixed,

$$\nabla_\theta D_{\mathrm{KL}}(p_\theta\|T) = \sum_v \nabla_\theta p_\theta(v)\left[\log p_\theta(v) - \log T(v)\right] + \sum_v p_\theta(v)\nabla_\theta \log p_\theta(v). \tag{29}$$

Using

$$\nabla_\theta p_\theta(v) = p_\theta(v)\nabla_\theta \log p_\theta(v) \tag{30}$$

and the score-function identity

$$\mathbb{E}_{v\sim p_\theta}\left[\nabla_\theta \log p_\theta(v)\right] = 0, \tag{31}$$

we obtain

$$\nabla_\theta D_{\mathrm{KL}}(p_\theta\|T) = \mathbb{E}_{v\sim p_\theta}\left[\left(\log p_\theta(v) - \log T(v)\right)\nabla_\theta \log p_\theta(v)\right]. \tag{32}$$

Therefore, the same sampled log-ratio provides both a Monte Carlo estimator of the reverse-KL value in Eq. 28 and the coefficient of an unbiased score-function estimator of its gradient under exact student sampling.

Equivalently, defining a detached advantage

$$A(y) = -\operatorname{sg}\left[k_1(y)\right], \tag{33}$$

the policy-gradient loss

$$L_{\mathrm{PG}} = -A(y)\log p_\theta(y) \tag{34}$$

has gradient

$$\nabla_\theta L_{\mathrm{PG}} = k_1(y)\nabla_\theta \log p_\theta(y), \tag{35}$$

matching the one-sample gradient estimator in Eq. 32.

Remark 1 (Characterization on Guided Rollouts)**.** *Under teacher-guided rollout ($Y \sim q_t$), evaluating the log-ratio update exclusively on accepted proposals corresponds to a localized score-function estimator supported exactly on the student–guided intersection measure* $\min\{p_t, q_t\}$ *(Proposition 1). This formulation restricts policy-gradient updates strictly to the subspace where student generations remain compatible with the guided policy, while delegating diverging states to the direct teacher-mode target* L_{TM}.

D Additional Placement-Control Results

We compare two supervision-placement controls under the same exact-q rollout. Random-TM matches the number of teacher-mode updates per trajectory used by Ours while placing them at count-matched random locations.

For TV-Weighted-TM, let V_i denote the valid response positions of trajectory i and define

$$m_i = \sum_{t\in V_i} C_{i,t}, \qquad d_{i,t} = \mathrm{TV}(p_{i,t}, q_{i,t}).$$

We select m_i valid positions without replacement using $d_{i,t}$ as the sampling weight. Selected positions receive the same teacher-mode loss as Ours, while the remaining valid positions retain RKL supervision. The correction mask is used only to determine m_i and does not determine the selected locations. The exact-q rollout and maximal-coupling process are unchanged.

We implement the weighted sampling with exponential-race keys $-\log U_{i,t}/d_{i,t}$ and select the m_i smallest keys. If fewer than m_i valid positions have positive TV, the remaining positions are filled uniformly.

Table 6: **Matched supervision-placement controls** for the 1.7B student on the same seven-benchmark suite. Random-TM controls for the supervision budget, while TV-Weighted-TM additionally favors positions with larger local student–guided disagreement.

Method	Mean@8	Pass@8
TRB	27.90	44.60
Random-TM	28.30	45.50
TV-Weighted-TM	28.24	46.77
Ours	**29.00**	**47.50**

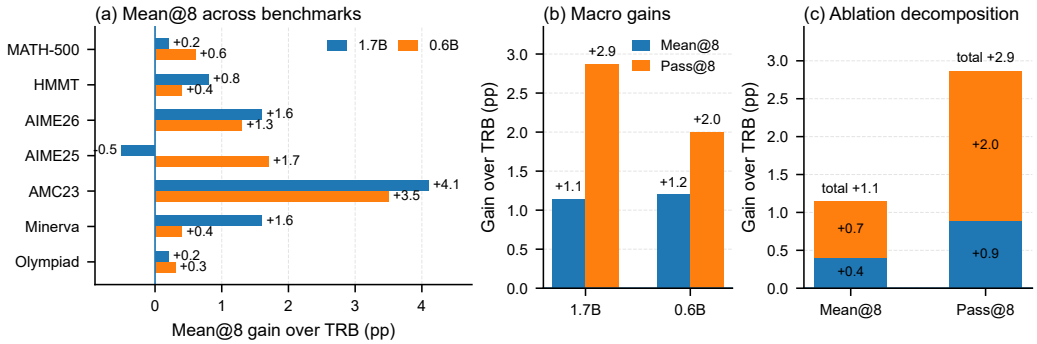


Figure 3: **Performance gains and supervision-placement ablation.** (a) **Mean@8 gains over the matched TRB baseline across seven benchmarks**. Our method improves six of seven benchmarks for the 1.7B student and all seven for the 0.6B student. (b) **Macro Mean@8 and Pass@8 improve at both student scales**. (c) The **lower segment** shows the gain obtained by applying the same number of teacher-mode updates at randomly selected positions (TRB → Random-TM), while the **upper segment** shows the additional gain from placing those updates at maximal-coupling corrections (Random-TM → Ours).TV-Weighted-TM is reported separately in Table 6.

E Additional Fixed-Prefix Results

Matched fixed-prefix probe. We compare three matched 1.7B runs: TRB, our correction-triggered teacher-mode objective, and a count-matched random-placement control. All three use the same Qwen3-1.7B-Base initialization, Qwen3-4B GRPO teacher, exact-q rollout, speculative block size $K = 8$, annealing schedule, training horizon, and seed. TRB retains RKL supervision at all valid positions. Our method replaces RKL with teacher-mode supervision at maximal-coupling corrections.

Using the initial student with $\epsilon = 0.02$, we construct a fixed probe containing 1,024 genuine maximal-coupling correction positions and 1,024 matched accepted positions. The fixed probe contains 2,048 positions spanning 213 prompts. Prefixes, response positions, teacher distributions, and initial conflict statistics are frozen throughout the analysis.

Let $v_T^*(h)$ denote the teacher's Top-1 token and let $\mathcal{S}_{16}^T(h)$ denote its Top-16 token set. For student distribution $S(\cdot \mid h)$, we measure

$$C_1(h) = S(v_T^*(h) \mid h), \qquad C_{16}(h) = \sum_{v \in \mathcal{S}_{16}^T(h)} S(v \mid h).$$

Thus, $C_{16} - C_1$ measures student probability on teacher-supported Top-16 alternatives excluding the teacher argmax.

We first average metrics within each prompt and then compute paired method differences, giving each prompt equal weight. Confidence intervals use 10,000 paired bootstrap resamples clustered by prompt (Efron et al., 2000).

To analyze heterogeneous placement effects, we rank fixed positions by their initial $D_{\mathrm{KL}}(T\|S)$ and partition them into four equal-sized conflict quartiles, from Q1 (lowest conflict) to Q4 (highest conflict).

The trust-region radius reaches exactly zero at step 51, after which $q_t = p_t$ and no correction-triggered teacher-mode updates occur.

Table 7 reports the complete fixed-prefix statistics underlying Figure 2(a). The reduction in $D_{\mathrm{KL}}(T\|S)$ persists after correction-triggered teacher-mode supervision is disabled at step 51, while the accompanying entropy reduction indicates that the student distribution becomes more concentrated around teacher-supported mass.

Table 7: **Prompt-equal paired differences** between correction-triggered teacher-mode supervision and the matched TRB control.

Step	ΔC_1	ΔC_{16}	$\Delta D_{\mathrm{KL}}(T\|S)$	$\Delta H(S)$
50	+4.63 pp	+4.54 pp	−0.133	−0.392
100	+4.30 pp	+2.69 pp	−0.154	−0.249
200	+4.26 pp	+2.94 pp	−0.196	−0.269

F Correction-Position Teacher Supervision

Teacher-mode supervision. At a correction prefix, let

$$v^* = \arg\max_v T(v).$$

Our teacher-mode loss is

$$L_{\mathrm{TM}} = -\log p_\theta(v^*).$$

Equivalently, because the teacher Top-1 distribution is the point mass δ_{v^*},

$$L_{\mathrm{TM}} = D_{\mathrm{KL}}(\delta_{v^*}\|p_\theta).$$

Thus, teacher-mode supervision can be viewed as forward KL from a degenerate teacher Top-1 target, while its implementation is simply hard-label negative log-likelihood on the teacher argmax token.

Teacher-sampled supervision. We additionally consider a stochastic correction objective that keeps the same maximal-coupling routing rule but samples the supervision target from the teacher distribution. At a correction prefix, we draw

$$\tilde{v} \sim T$$

and optimize

$$L_{\mathrm{TS}} = -\log p_\theta(\tilde{v}).$$

In expectation over $\tilde{v} \sim T$,

$$\mathbb{E}_{\tilde{v}\sim T}[L_{\mathrm{TS}}] = -\sum_v T(v)\log p_\theta(v),$$

so this stochastic objective has the same expected gradient as $D_{\mathrm{KL}}(T\|p_\theta)$.

Table 8: **Correction-supervision ablation** for the 1.7B student. The coupling-aware variants use the same maximal-coupling routing rule and trust-region schedule, differing only in the supervision target used at correction positions.

Correction supervision	Mean@8	Pass@8
TRB	27.90	44.60
Teacher-sampled	28.38	45.94
Teacher-mode (**SAKI**)	**29.00**	**47.50**

Teacher-sampled supervision improves over the matched TRB baseline, showing that coupling corrections are useful locations for direct teacher-directed supervision beyond the particular choice of a hard teacher-mode target. Teacher-mode supervision performs better in this setting, improving over teacher-sampled supervision by 0.62 Mean@8 and 1.56 Pass@8 points. We therefore use teacher-mode supervision in the main method.

G Proof of Proposition 1

Let

$$Z \sim p \tag{36}$$

be the student proposal. Given $Z = v$, it is accepted with probability

$$a(v) = \min\left(1, \frac{q(v)}{p(v)}\right). \tag{37}$$

Therefore,

$$\Pr(Z = v, C = 0) = p(v)a(v) = \min\{p(v), q(v)\}. \tag{38}$$

Summing over v,

$$\Pr(C = 0) = \sum_v \min\{p(v), q(v)\}. \tag{39}$$

Hence,

$$\Pr(C = 1) = 1 - \sum_v \min\{p(v), q(v)\} \tag{40}$$

$$= \frac{1}{2}\sum_v |p(v) - q(v)| \tag{41}$$

$$= \mathrm{TV}(p, q). \tag{42}$$

For the rejected proposal,

$$\Pr(Z = v, C = 1) = p(v)\left[1 - \min\left(1, \frac{q(v)}{p(v)}\right)\right] \tag{43}$$

$$= [p(v) - q(v)]_+. \tag{44}$$

Conditioning on $C = 1$ gives

$$\Pr(Z = v \mid C = 1) = \frac{[p(v) - q(v)]_+}{\mathrm{TV}(p, q)}. \tag{45}$$

On rejection, the correction token is sampled from

$$r(v) = \frac{[q(v) - p(v)]_+}{\sum_u [q(u) - p(u)]_+}. \tag{46}$$

Since

$$\sum_u [q(u) - p(u)]_+ = \mathrm{TV}(p, q), \tag{47}$$

we obtain

$$\Pr(Y = v \mid C = 1) = \frac{[q(v) - p(v)]_+}{\mathrm{TV}(p, q)}. \tag{48}$$

Finally, the unconditional probability of output token v is

$$\Pr(Y = v) = \Pr(Y = v, C = 0) + \Pr(Y = v, C = 1) \tag{49}$$

$$= \min\{p(v), q(v)\} + [q(v) - p(v)]_+ \tag{50}$$

$$= q(v). \tag{51}$$

Therefore, maximal coupling produces the exact q marginal.

G.1 MINIMAL-INTERVENTION PROPERTY

For any coupling $\gamma \in \Gamma(p, q)$,

$$\Pr_{\gamma}(Z = Y) = \sum_{v} \Pr_{\gamma}(Z = v, Y = v) \tag{52}$$

$$\leq \sum_{v} \min\{p(v), q(v)\}. \tag{53}$$

Therefore,

$$\Pr_{\gamma}(Z \neq Y) \geq 1 - \sum_{v} \min\{p(v), q(v)\} \tag{54}$$

$$= \mathrm{TV}(p, q). \tag{55}$$

Under the maximal-coupling construction in Proposition 1,

$$\Pr(Z = v, Y = v) = \min\{p(v), q(v)\}, \tag{56}$$

so equality is attained. This proves Proposition 2.

The intervention-rate bound in Corollary 1 follows immediately from Pinsker's inequality (Cover & Thomas, 2006),

$$\mathrm{TV}(p, q) \leq \sqrt{\frac{1}{2} D_{\mathrm{KL}}(q \| p)}, \tag{57}$$

together with the trust-region constraint.

H TEACHER–STUDENT INTERPRETATION OF THE RESIDUAL DISTRIBUTION

For the geometric bridge,

$$q_\beta(v) = \frac{p(v)^{1-\beta} T(v)^\beta}{Z_\beta}, \qquad Z_\beta = \sum_u p(u)^{1-\beta} T(u)^\beta. \tag{58}$$

Assuming $\beta > 0$ and positive softmax probabilities,

$$\frac{q_\beta(v)}{p(v)} = \frac{1}{Z_\beta} \left(\frac{T(v)}{p(v)} \right)^\beta. \tag{59}$$

Therefore,

$$q_\beta(v) > p(v) \iff \frac{q_\beta(v)}{p(v)} > 1 \tag{60}$$

$$\iff \left(\frac{T(v)}{p(v)} \right)^\beta > Z_\beta \tag{61}$$

$$\iff \log \frac{T(v)}{p(v)} > \frac{\log Z_\beta}{\beta}. \tag{62}$$

Hence, the positive residual

$$[q_\beta(v) - p(v)]_+ \tag{63}$$

is supported only on tokens whose teacher-to-student likelihood ratio exceeds a prefix-dependent threshold.

Symmetrically,

$$p(v) > q_\beta(v) \tag{64}$$

corresponds to teacher-to-student likelihood ratios below this threshold. Combined with Proposition 1, a correction can therefore be viewed as replacing probability mass that is excessive under the student relative to the bridge with probability mass favored by the bridge and associated with sufficiently high teacher-to-student likelihood ratios.

I EXACTNESS OF ENGINE-RESIDENT BLOCK VERIFICATION

Proposition 3 (Exactness of engine-resident block verification)**.** *Assume that the student and teacher distributions in a speculative block are evaluated on the same proposal prefixes as sequential maximal coupling. If proposals are committed only up to the first rejection, the correction is sampled from the exact residual distribution, and all subsequent speculative states are discarded, then the generated trajectory follows the same autoregressive distribution as sequential sampling from q:*

$$\Pr(y_{1:L} \mid x) = \prod_{t=1}^{L} q_t(y_t \mid x, y_{<t}). \tag{65}$$

Proof. Consider a speculative block beginning at a true prefix h_t. The student autoregressively proposes

$$z_t, z_{t+1}, \ldots, z_{t+K-1}. \tag{66}$$

The teacher and bridge distributions are evaluated on the corresponding proposal prefixes.

Suppose the first j proposals are accepted. Before any rejection occurs, the actual generated prefix is identical to the proposal prefix. Therefore, for each accepted position

$$t, t+1, \ldots, t+j-1, \tag{67}$$

the precomputed student and teacher distributions are exactly the same distributions that sequential maximal coupling would evaluate at the true prefix. Thus, committing these accepted tokens produces exactly the same conditional transition as sequential coupling.

If the next proposal is rejected, the algorithm samples a correction from

$$r_{t+j}(v) \propto [q_{t+j}(v) - p_{t+j}(v)]_+, \tag{68}$$

which is exactly the correction distribution used by sequential maximal coupling at that prefix. After this correction, the true prefix differs from the speculative proposal prefix. Consequently, all precomputed logits for later proposal positions are invalid and are discarded.

The next speculative block is constructed from the new true prefix. Therefore, after every committed token—whether accepted or corrected—the algorithm has the same true prefix and uses the same next-token transition kernel as sequential maximal coupling.

By induction over decoding steps,

$$Y_t \sim q_t(\cdot \mid x, Y_{<t}) \tag{69}$$

for every t, and hence

$$\Pr(y_{1:L} \mid x) = \prod_{t=1}^{L} q_t(y_t \mid x, y_{<t}). \tag{70}$$

Thus, speculative block verification changes only the execution schedule and does not alter the target autoregressive distribution. □

J LIMITING CASES AND SANITY CHECKS

J.1 $\epsilon = 0$: RECOVERY OF STANDARD STUDENT ROLLOUT

If

$$\epsilon = 0, \tag{71}$$

the only feasible bridge distribution is

$$q_t = p_t. \tag{72}$$

Therefore,

$$\mathrm{TV}(p_t, q_t) = 0, \tag{73}$$

and Proposition 1 gives

$$C_t = 0 \tag{74}$$

almost surely.

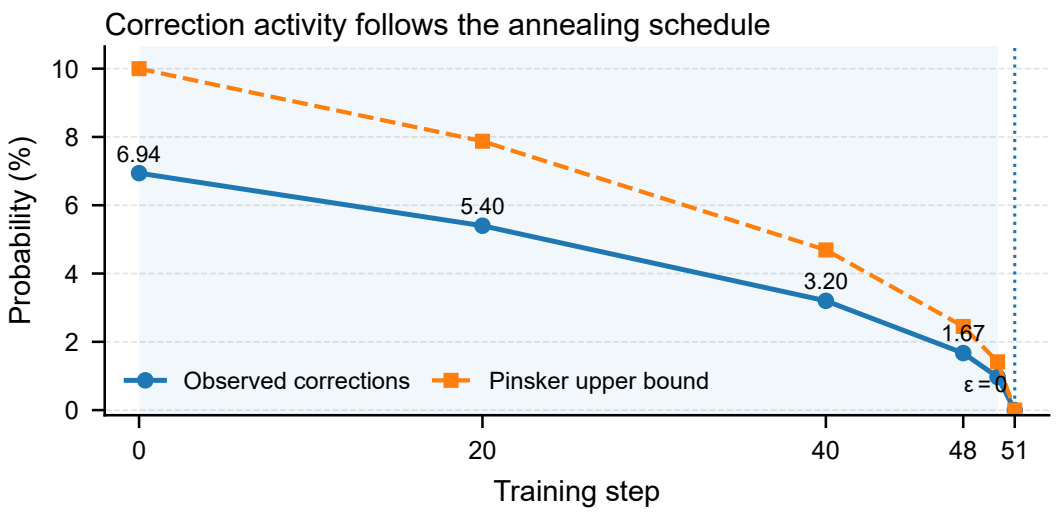


Figure 4: **Correction activity under annealing.** The observed correction probability decreases with the trust-region radius, remains below the Pinsker upper bound, and vanishes when ϵ reaches zero at step 51.

Consequently, the teacher-mode supervision term vanishes exactly:

$$\sum_t M_t C_t L_{\mathrm{TM},t} = 0. \tag{75}$$

The objective in Eq. 16 reduces to

$$L = \frac{1}{N_{\mathrm{valid}}} \sum_t M_t L_{\mathrm{RKL},t}, \tag{76}$$

which is the ordinary student-rollout RKL objective.

J.2 Correction Activity under Annealing

Figure 4 compares the observed maximal-coupling correction probability with the Pinsker upper bound induced by the current trust-region radius. As ϵ is annealed, correction activity decreases and vanishes at the student-rollout endpoint.

J.3 Teacher Endpoint

If the teacher distribution itself satisfies

$$D_{\mathrm{KL}}(T_t \| p_t) \le \epsilon, \tag{77}$$

then the optimal trust-region bridge takes

$$\beta_t = 1, \qquad q_t = T_t. \tag{78}$$

Thus, the bridge smoothly interpolates between student rollout and the teacher endpoint as the trust-region radius grows.

K Systems Implementation and Engineering Evolution

K.1 Why Exact-q Rollout Is Expensive

Vanilla OPD admits a computationally convenient execution schedule: the student can first generate a complete response, after which the fixed teacher scores the entire sequence in a batched forward

pass. Teacher-guided behavior blending instead requires the teacher distribution during generation in order to construct q_t at every visited prefix. TRB explicitly notes this online-inference overhead (Plyusov et al., 2026).

Ignoring transient activations, the generation-time resident memory can be approximated as

$$M_{\mathrm{OPD}}(t) \approx W_s + KV_s(n_t) \tag{79}$$

for student-only generation, whereas online teacher-guided generation requires

$$M_q(t) \approx W_s + W_T + KV_s(n_t) + KV_T(n_t). \tag{80}$$

The additional generation-time footprint is therefore approximately

$$\Delta M_{\mathrm{gen}}(t) \approx W_T + KV_T(n_t). \tag{81}$$

The arithmetic required by the geometric bridge itself is comparatively simple vector computation. In practice, the dominant difficulty in our original implementation was the execution structure surrounding this computation: token-level RPCs, transfer of full-vocabulary logits, host synchronization, repeated prefix handling, and Python control flow.

K.2 From Token-Wise to External Block Verification

Our first correctness-oriented exact-q implementation operated one token at a time. For every decoding position, it independently: (i) queried the student for a one-token proposal and full-vocabulary distribution, (ii) queried the teacher at the same prefix, (iii) transferred both distributions outside the inference servers, (iv) constructed the trust-region bridge in Python, (v) executed maximal coupling, and (vi) appended the resulting token before repeating the process.

For a response of length L, the execution cost was therefore approximately

$$T_{K=1} \approx L\left(T_{\text{student RPC}} + T_{\text{teacher RPC}} + T_{\text{dense transfer}} + T_{\text{bridge}} + T_{\text{coupling}} + T_{\text{host}}\right). \tag{82}$$

In an early unoptimized prototype ($K = 1$), external per-token RPC and logit serialization yielded only $\sim$5 tokens/s (compared to $\sim$3.02K tokens/s for student-only generation under the same pilot), illustrating the severe communication bottleneck of naive online bridge evaluation.

We next introduced external K-token proposals. For $K = 8$, one student request proposes a block and one teacher suffix-scoring request evaluates the corresponding proposal prefixes. This reduces model-service calls from approximately $O(L)$ to approximately $O(L/K)$ when blocks are largely accepted. However, bridge construction, dense-logit transfer, coupling, metadata assembly, and commit decisions still occur in the external Python coordinator.

K.3 Engine-Resident q Backend

The final backend moves the entire recurrent block loop inside the inference runtime. Each engine replica contains both a student draft model and the fixed teacher. For each speculative wave it performs:

1. student sampling of up to K proposal tokens;
2. teacher verification of the proposal path;
3. batched full-vocabulary construction of q_t;
4. fixed-wave batched solution of β_t;
5. vectorized maximal-coupling acceptance;
6. batched positive-residual correction sampling;
7. first-rejection KV commit/rollback; and
8. compact token-aligned metadata packing.

Table 9: **Engineering progression of the exact-q rollout implementation**. The first two rows reflect early exploratory setups, while the final two rows benchmark mature implementations under the matched workload.

Implementation	K	Tok/s	Execution
Token-wise prototype	1	$\sim$5	Per-token external RPC
Early external block path	8	165–299	External block verification
Final external loop	8	776	Mature Python/RPC pipeline
Engine-resident backend	8	3,276	GPU-resident bridge/coupling

Table 10: **Representative integrated training-step timing**.

Component	Time (s)
q generation	189.5
Actor update	24.9
Student weight sync	2.5
Whole step	232.6

Student and teacher logits remain inside the engine. For a batch of proposal blocks, they are represented as tensors of shape $[B, K, V]$ and reshaped to $[BK, V]$ for bridge construction. The trust-region solve uses 16 fixed bisection waves. Rows that have converged are masked on GPU rather than triggering host-synchronized early termination.

For a proposal token a, maximal-coupling acceptance is evaluated as

$$\alpha(a) = \exp\left(\min\left\{0, \log q(a) - \log p_s(a)\right\}\right). \tag{83}$$

A prefix cumulative product over the acceptance indicators identifies the longest consecutively accepted proposal prefix. If the first rejection occurs at position j, only proposals before j and the sampled correction are committed; every later speculative state is discarded because it was computed under an invalid prefix.

The external training loop receives only compact metadata required by the objective, rather than full-vocabulary logits.

K.4 Model and Cache Lifecycle

In our 8-GPU training layout, GPUs 0–3 execute the student actor update and GPUs 4–7 host q-engine replicas. Each q-engine keeps a fixed teacher and a draft replica of the evolving student resident in memory.

After every actor update, only the student draft weights are synchronized. The teacher remains fixed throughout training. Cached states associated with the previous student version are invalidated before new rollout requests resume. This avoids repeatedly redeploying or synchronizing the unchanged teacher.

The inference engines also use a separate cache lifecycle from the actor training path. Keeping the q-engine model and cache pools resident avoids repeated release/reinitialization overhead and ensures that speculative proposal and verification states remain under the engine's ownership.

Full-vocabulary distributions are consumed entirely inside the q-engine. The response returned to the training loop contains only token-aligned metadata, including proposal and committed tokens, the correction mask, selected student/q log-probabilities, β, bridge KL, residual mass, acceptance probability, and the effective trust-region radius.

Student synchronization accounts for only approximately 1.1% of the representative step wall-clock, indicating that online rollout rather than draft-weight synchronization remains the dominant systems cost.

K.5 Correctness Validation and Limitations

All system optimizations are required to preserve the same mathematical sampling semantics as the reference implementation. We validate the optimized backend along four dimensions.

Bridge correctness. GPU bridge construction is compared against a CPU reference on randomized small-vocabulary distributions. We verify normalization, the trust-region constraint, the exact $\epsilon = 0$ student endpoint, and the $\beta = 1$ teacher endpoint when feasible.

Coupling correctness. Monte Carlo samples from maximal coupling are compared against direct samples from q. We additionally test no-rejection blocks, first-rejection commit semantics, residual correction sampling, EOS handling in discarded speculative suffixes, and token-aligned metadata.

Training-state correctness. We verify that student weight synchronization changes the draft-model checksum while the teacher checksum remains fixed, and that cached states from old student versions are invalidated.

Routing correctness. Validation always generates from the current student alone. When the effective trust-region radius is exactly zero, the rollout follows the student-only fast path, teacher q-forward count is zero, and the correction mask is identically zero.

The optimized system does not eliminate the intrinsic cost of online teacher verification. Its throughput therefore remains below student-only generation and depends on response length, correction rate, and the number of committed tokens per verification wave.

L Additional Objective and Implementation Details

Correction masking. At correction positions, we set the RKL contribution exactly to zero. The correction loss does not stack with RKL.

Normalization. Both accepted-position RKL terms and correction terms are normalized by the total number of valid response tokens,

$$N_{\text{valid}} = \sum_t M_t. \tag{84}$$

Accepted-position RKL terms and correction-position teacher-mode terms receive unit weight and are jointly normalized by the total number of valid response tokens. No additional correction-loss coefficient is used.

Role of the residual correction token. The correction token sampled from the residual distribution becomes the actual response token at that position and therefore determines all subsequent prefixes. However, it is not used as the direct supervision target. At correction positions, the direct training target is the teacher's highest-probability token and therefore need not equal the sampled residual token.

Full-vocabulary rollout. The intermediate rollout distribution q_t and maximal-coupling computation operate over the full vocabulary. The teacher Top-1 target is used only for the correction supervision objective and does not truncate or otherwise modify the rollout distribution.

Validation. Evaluation should always generate from the current student alone, rather than from the teacher-guided rollout engine, so that reported benchmark performance measures the distilled student itself.