BEYOND MODE-COVERING AND MODE-SEEKING: A CLOSER LOOK AT KL DIVERGENCE IN ON-POLICY DISTILLATION

Anonymous authors
Paper under double-blind review

ABSTRACT

On-policy distillation (OPD) is a de facto recipe for enhancing the reasoning capability of student models by matching the teacher's distribution over self-generated trajectories, where the choice of divergence is a critical design decision. Empirical guidance relies on geometric intuitions — framing forward KL as *mode-covering* and reverse KL as *mode-seeking* — offering little insight into how these objectives actually steer optimization. Moving beyond this dichotomy, we shift our focus to optimization dynamics. Specifically, we analyze the logit-level gradient fields that different divergences produce during training. And we show that no f-divergence guarantees that every logit update moves probability toward the teacher due to the Softmax coupling effect. Instead, effective optimization relies on coordinate-level sign alignment with the student-teacher probability gap. Under this framework, we prove that forward KL is the unique sign-aligned f-divergence, while reverse KL actively pushes over-allocated tokens in the wrong direction — a structural pathology that convex combinations of the two cannot resolve. To settle this, we introduce m-RKL, a minimal gradient-field correction that restores sign alignment while retaining RKL's beneficial tail damping. Extensive experiments on Qwen3 models demonstrate that m-RKL outperforms competitive KL-based objectives on a suite of math reasoning benchmarks without degrading the general capabilities.

1 INTRODUCTION

On-policy distillation (OPD) (Agarwal et al., 2024) has emerged as a critical component in the post-training pipeline of large language models (LLMs) (Zeng et al., 2026; Xu et al., 2026; Team et al., 2026). By training a student model to match a teacher's predictions along the reasoning trajectories sampled directly from the student's own policy, OPD mitigates the *exposure bias* (Pozzi et al., 2025) that plagues traditional teacher-forced distillation. At the heart of every OPD framework lies a vital design decision: the choice of divergence used to align the student's predictive distribution with the teacher's. Current studies primarily rely on two canonical choices — forward KL (FKL) and reverse KL (RKL). FKL encourages the student to assign probability mass across the teacher's full support, over-smoothing the distribution and potentially degrading the reasoning quality. RKL, by contrast, drives the student to concentrate mass on dominant teacher modes, producing confident reasoning paths at the risk of under-covering valid alternative solutions. This empirical trade-off has spawned a rich line of interpolating and re-weighting objectives (Ko et al., 2024; Peng et al., 2025; Jung et al., 2025; Wu et al., 2025; Jin et al., 2026), yet a fundamental theoretical principle explaining how these divergences steer optimization has remained elusive.

Existing literature predominantly analyzes divergences through the lens of continuous probability spaces — labeling FKL as *mode-covering* and RKL as *mode-seeking* (Song & Zheng, 2026). While these geometric descriptions accurately characterize the resulting distribution shape, they shed little light on how an objective steers the underlying optimization process. LLMs are optimized via gradient descent, which directly affects the logits rather than the probabilities, and the mapping between them is governed by the Softmax function (Kim et al., 2026). The essential question for OPD is therefore not "*What global shape does a divergence favor?*", but rather a mechanistic one: *at the logit level, where the optimizer directly makes a difference, how does each divergence's gradient field move the student, and what guarantees can we strictly prove about that movement?*

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.

Answering this question requires precise accounting of optimization mechanics. Given that Softmax couples all logits, the sign of an individual logit gradient does not dictate whether the corresponding token's probability moves toward or away from the teacher (Franke & Degen, 2023). In fact, that depends on the sign of a centered quantity $G_k - \mathbb{E}_{\boldsymbol{q}}[G]$, as we analyze in Section 3.1. Accordingly, it can be further deduced that *no f-divergence can guarantee monotonic movement of every token's probability toward the teacher*. To build a rigorous theoretical framework, we characterize objectives along two well-posed axes: *i)* whether the sign of each logit gradient aligns with the student-teacher probability gap, *i.e.*, $q_k - p_k$; *ii)* whether the induced gradient field provides an effective descent direction in an appropriate divergence landscape.

Within this framework, the KL family exhibits an unexpected structural asymmetry. First, we prove that FKL is the unique f-divergence whose raw logit gradient field is sign-aligned with the student-teacher probability gap — consistently pushing down the logits of over-allocated tokens ($q_k > p_k$) and pulling up those of under-allocated ones ($q_k < p_k$). Secondly, we discover that RKL is provably misaligned: for any over-allocated token that satisfies $\log(q_k/p_k) < D_{\mathrm{KL}}(\boldsymbol{q} \,\|\, \boldsymbol{p})$, RKL produces a negative gradient on its logit, causing the already high logit value to increase further, which possibly widens the student-teacher probability gap (Gao & Pavel, 2017). Crucially, we demonstrate that this pathology cannot be mixed away by convex combinations of FKL and RKL. Interpolating between divergences merely relocates the misalignment rather than removing it.

To resolve this structural flaw, we propose a gradient field correction scheme instead of a new loss function. Particularly, we introduce m-RKL, a minimally modified gradient field of RKL where the global divergence term is replaced by an adaptive, energy-balancing multiplicative factor. In a single optimization step, the resulting field: *i)* is sign-aligned with the student-teacher probability gap on every token; *ii)* preserves RKL's tail-damping property and optimization efficiency; *iii)* enforces the zero-sum conservation constraint, preventing the unbounded logit drift that a naive sign correction would introduce. Although m-RKL is non-conservative (it is not the gradient of any scalar loss), we prove that it constituents a strict descent direction for FKL at every configuration where the student differs from the teacher. In summary, our main contributions are presented as follows:

- We formalize divergence-based OPD via logit gradient fields on the Softmax manifold and disambiguate the coordinate-level and probability-level guarantees such a field can carry, showing that no f-divergence can guarantee strict token-level probability monotonicity.
- We prove that FKL is the unique f-divergence whose logit gradient is sign-aligned with the student-teacher probability gap, whereas RKL provably opposes the sign on over-allocated tokens — a basic pathology that no convex combination of FKL and RKL can eliminate.
- We propose m-RKL, a gradient-field correction method of standard RKL that restores sign alignment and zero-sum conservation while retaining RKL's tail-damping property, despite being non-conservative, we prove m-RKL represents a strict descent direction for FKL.
- Extensive evaluations on Qwen3 models of varying scales demonstrate that m-RKL delivers stronger performance than other KL-based objectives across a panel of mathematical reasoning benchmarks with no deterioration in base models' general capabilities.

2 PRELIMINARIES

2.1 ON-POLICY DISTILLATION

In traditional off-policy distillation (Hinton et al., 2015), the student is only exposed to target distributions during training but must generate from its own distribution at inference time. OPD resolves this train-test *distribution shift* problem by allowing the student to sample trajectories from its own policy and receiving the teacher's real-time supervision, effectively forcing the student to learn from its own mistakes. This philosophy can be traced back to the DAgger algorithm (Ross et al., 2011).

Formally, let $\mathcal{V} = \{1, 2, \ldots, V\}$ denote a finite vocabulary of size V. Let x denote an input prompt and $y = (y_1, \ldots, y_T)$ represent a trajectory generated autoregressively by the student model, where $y_t \in \mathcal{V}$. At time step t, the history of context available is denoted by $y_{<t} = (x, y_1, \ldots, y_{t-1})$. Then the output distributions of the teacher and the student conditioned on the same context can be defined as $p(\cdot|y_{<t}) \in \Delta^{V-1}$ and $q_\theta(\cdot|y_{<t}) \in \Delta^{V-1}$, respectively. Accordingly, the objective of OPD is to minimize the expected token-step divergence between the two conditional distributions

along the student-generated trajectory, mathematically expressed as

$$\mathcal{L}_{\text{OPD}}(\theta) = \mathbb{E}_{x\sim\mathcal{D}, y\sim q_\theta(\cdot|x)}\left[\sum_{t=1}^{T} D\left(p(\cdot|y_{<t}) \parallel q_\theta(\cdot|y_{<t})\right)\right] \tag{1}$$

where $D(\cdot\|\cdot)$ is an arbitrary divergence used to measure the distance between two distributions. For brevity, we drop the timestep index t and context $y_{<t}$ in subsequent sections to analyze single-step dynamics, representing teacher and student distributions simply as probability vectors $\boldsymbol{p}, \boldsymbol{q} \in \Delta^{V-1}$.

2.2 The Softmax Manifold and Its Tangent Space

In modern LLMs, the student's probability distribution $\boldsymbol{q}$ is parameterized by an unnormalized logit vector $\boldsymbol{z} = [z_1, \ldots, z_V]^\top$ via the Softmax map $\sigma : \mathbb{R}^V \to \Delta^{V-1}$:

$$q_k = \sigma(\boldsymbol{z})_k = \frac{e^{z_k}}{\sum_{j=1}^{V} e^{z_j}}, \quad \forall\, k \in \mathcal{V} \tag{2}$$

The Jacobian matrix $J_{\boldsymbol{z}}(\boldsymbol{q}) = \frac{\partial \boldsymbol{q}}{\partial \boldsymbol{z}} \in \mathbb{R}^{V\times V}$ of the Softmax transformation has elements given by

$$[J_{\boldsymbol{z}}(\boldsymbol{q})]_{kj} = \frac{\partial q_k}{\partial z_j} = q_k(\delta_{kj} - q_j) \tag{3}$$

where δ_{kj} is the Kronecker delta. Applying the multivariate chain rule, the gradient of any scalar loss $\mathcal{L}(\boldsymbol{q})$ with respect to the raw student logits $\boldsymbol{z}$ is

$$\nabla_{\boldsymbol{z}}\mathcal{L} = J_{\boldsymbol{z}}(\boldsymbol{q})^\top \nabla_{\boldsymbol{q}}\mathcal{L} = \left(\operatorname{diag}(\boldsymbol{q}) - \boldsymbol{q}\boldsymbol{q}^\top\right)\nabla_{\boldsymbol{q}}\mathcal{L} \tag{4}$$

One of the key properties of the Softmax map is translation invariance, *i.e.*, $\sigma(\boldsymbol{z} + c\mathbf{1}) = \sigma(\boldsymbol{z})$ for any scalar $c \in \mathbb{R}$, where $\mathbf{1} = [1, 1, \ldots, 1]^\top$. Consequently, the logit gradient of any mathematically consistent loss $\mathcal{L}$ must lie in the tangent space orthogonal to $\mathbf{1}$ (Li, 2025):

$$\mathbf{1}^\top \nabla_{\boldsymbol{z}}\mathcal{L} = \sum_{k=1}^{V} \frac{\partial \mathcal{L}}{\partial z_k} = 0 \tag{5}$$

We refer to Eq. (5) as the **zero-sum conservation constraint**. Satisfying this constraint ensures that parameter updates do not induce unbounded logit drift or numerical instability during training.

2.3 The f-Divergence Family

The similarity between the teacher's distribution $\boldsymbol{p}$ and the student's distribution $\boldsymbol{q}$ is broadly unified under the mathematical family of f-divergence. An f-divergence is defined by a convex generator function $f : \mathbb{R}_{>0} \to \mathbb{R}$ satisfying $f(1) = 0$, expressed as

$$D_f(\boldsymbol{p} \parallel \boldsymbol{q}) = \sum_{k=1}^{V} q_k f\left(\frac{p_k}{q_k}\right) \tag{6}$$

Different choices of the generator function $f(u)$ recover different objectives for distillation. The two most widely used are the forward KL and reverse KL divergences.

$$D_{\text{FKL}} = D_{\text{KL}}(\boldsymbol{p} \parallel \boldsymbol{q}) = \sum_{k=1}^{V} p_k \log\left(\frac{p_k}{q_k}\right); D_{\text{RKL}} = D_{\text{KL}}(\boldsymbol{q} \parallel \boldsymbol{p}) = \sum_{k=1}^{V} q_k \log\left(\frac{q_k}{p_k}\right) \tag{7}$$

Most of the existing literature characterizes their behaviors through the lens of global distribution geometry, positioning FKL as a *mode-covering* objective and RKL as a *mode-seeking* objective. Yet such macroscopic geometric taxonomies do not translate into computable principles governing how these objectives shape individual parameter updates. Therefore, we put our focus on the structure of gradients in logit space, where Lemma 1 serves as a unified starting point for our analysis.

Lemma 1 (General Logit Gradient of f-Divergence). *For any twice differentiable, strictly convex generator function $f : \mathbb{R}_{>0} \to \mathbb{R}$ with $f(1) = 0$, the partial derivative of $D_f(\boldsymbol{p}\|\boldsymbol{q})$ with respect to logit z_k is given by:*

$$\frac{\partial D_f(\boldsymbol{p} \parallel \boldsymbol{q})}{\partial z_k} = q_k\left[g\left(\frac{p_k}{q_k}\right) - \sum_{j=1}^{V} q_j g\left(\frac{p_j}{q_j}\right)\right] \tag{8}$$

where $g(u) = f(u) - uf'(u)$ is a strictly decreasing function on $(0, \infty)$. See Appendix B for proof.

3 METHODOLOGY

In this section, we conduct a rigorous logit-level gradient analysis of different objectives commonly used in OPD. Our central question is: *how does the gradient induced by a selected divergence shape the student's logits, and what can we precisely guarantee about the resulting optimization behavior?*

3.1 LOGIT GRADIENT, PROBABILITY FLOW, AND SIGN ALIGNMENT

In gradient-based policy optimization, a natural desideratum for each update to the logits entails that it should locally narrow the probability gap between the student and the teacher. Drawing inspiration from Gneiting & Raftery (2007), we formalize this idea at the logit coordinate level.

Definition 1 (**Strict Local Sign Monotonicity, SLSM**). A logit gradient field $G(\boldsymbol{z}) \in \mathbb{R}^V$ satisfies *strict local sign monotonicity* if and only if:

$$\operatorname{sign}(G_k) = \operatorname{sign}(q_k - p_k), \quad \forall\, k \in \mathcal{V},\ \forall\, \boldsymbol{p}, \boldsymbol{q} \in \Delta^{V-1} \tag{9}$$

Under gradient descent, *i.e.*, $\boldsymbol{z} \leftarrow \boldsymbol{z} - \eta G(\boldsymbol{z})$, Eq. (9) guarantees that over-allocated tokens ($q_k > p_k$) receive downward pressure on their logits, while under-allocated tokens ($q_k < p_k$) receive upward pressure. It is essential to be precise about what Eq. (9) does and does not guarantee. Since Softmax couples all logits, the sign of an individual logit gradient does not directly determine whether the corresponding token's probability moves toward or away from the teacher, as clarified in Lemma 2.

Lemma 2 (**Probability-Flow Decomposition**). *For a continuous gradient flow $\dot{\boldsymbol{z}} = -G(\boldsymbol{z})$ induced by a logit field G, the probability assigned to token k evolves as*

$$\frac{dq_k}{ds} = q_k \left(\mathbb{E}_{\boldsymbol{q}}[G] - G_k \right), \quad \mathbb{E}_{\boldsymbol{q}}[G] := \sum_{j=1}^{V} q_j G_j \tag{10}$$

Refer to Appendix B.2 for the proof. Based on Eq. (10), we can deduce that a sufficient condition for probability movement toward the teacher is the centered criterion $\operatorname{sign}(G_k - \mathbb{E}_{\boldsymbol{q}}[G]) = \operatorname{sign}(q_k - p_k)$ such that over-allocated tokens satisfy $dq_k/ds < 0$ and under-allocated tokens satisfy $dq_k/ds > 0$. However, given the zero-sum conservation constraint shown in Eq. (5), *i.e.*, $\sum_k G_k = 0$, the gradient expectation satisfies $\mathbb{E}_{\boldsymbol{q}}[G] = \sum_k q_k G_k \neq 0$ for any non-uniform distribution $\boldsymbol{q}$. As a result, SLSM is generally neither necessary nor sufficient for probability contraction. Actually, further derivation demonstrates that no gradient field of the single-token form in Eq. (8) guarantees probability-level monotonicity, implying no f-divergence ensures every token's probability moves toward the teacher at each single optimization step. The complete derivation process can be found in Appendix B.3.

Given the above analysis, one might wonder whether SLSM is therefore vacuous. We clarify that it is not the case: *i)* SLSM is uniquely achievable only by FKL among all f-divergences (Theorem 1), so it does isolate an objective; and *ii)* for fields satisfying SLSM, the logit dynamics provably moves each coordinate in the correct direction, which translates into a first-order contraction of the logits toward the teacher — a meaningful structural property even in the presence of Softmax coupling.

3.2 FORWARD KL: THE UNIQUE SIGN-ALIGNED f-DIVERGENCE

We present our first main theoretical result, establishing that forward KL occupies a unique position within the f-divergence family.

Theorem 1 (**Unique Sign Alignment of Forward KL**). *Forward KL divergence, $D_{\mathrm{KL}}(\boldsymbol{p} \| \boldsymbol{q})$, is the unique f-divergence whose un-modified raw gradient with respect to the logit $\boldsymbol{z}$ inherently satisfies strict local sign monotonicity (SLSM) across all distributions $\boldsymbol{p}, \boldsymbol{q} \in \Delta^{V-1}$.*

Proof. In terms of FKL, $f(u) = u \log u$, yielding $g(u) = -u$. Substituting it into Lemma 1 gives:

$$\frac{\partial D_{\mathrm{KL}}(\boldsymbol{p} \,\|\, \boldsymbol{q})}{\partial z_k} = q_k \left[-\frac{p_k}{q_k} + \sum_j p_j \right] = q_k - p_k \tag{11}$$

which trivially satisfies $\operatorname{sign}\left(\frac{\partial D_{\mathrm{KL}}(\boldsymbol{p} \| \boldsymbol{q})}{\partial z_k} \right) = \operatorname{sign}(q_k - p_k)$ unconditionally for all $k \in \mathcal{V}$. To prove the uniqueness, please refer to our detailed derivation in Appendix B.4. □

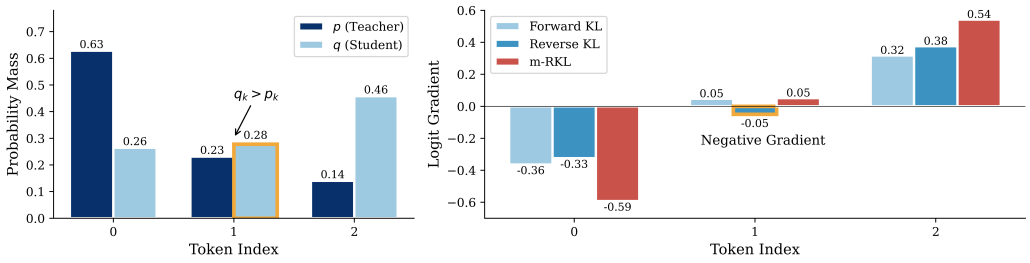

Figure 1: A schematic of the sign-reversal pathology described in Theorem 2.

3.3 Reverse KL: The Sign-Reversal Pathology

We now turn to RKL, where $f(u) = -\log u$ and $g(u) = \log u + 1$. In this case, Lemma 1 yields:

$$\frac{\partial D_{\mathrm{KL}}(\boldsymbol{q} \parallel \boldsymbol{p})}{\partial z_k} = q_k \left[\log \left(\frac{q_k}{p_k} \right) - D_{\mathrm{KL}}(\boldsymbol{q} \parallel \boldsymbol{p}) \right] \tag{12}$$

Obviously, the sign of this gradient depends on a *global comparison*. Only those tokens satisfying $\log(q_k/p_k) > D_{\mathrm{KL}}(\boldsymbol{q}\|\boldsymbol{p})$ receive a suppression force on their logits, while the rest are boosted, even if $q_k > p_k$. This means that the RKL's logit gradient can lead to the update direction opposite to the student-teacher probability gap on over-allocated tokens, as formalized in Theorem 2.

At this moment, we realize that the optimization behaviors of FKL and RKL diverge fundamentally due to the distinct structures of their gradients w.r.t. logit. For FKL, Eq. (11) represents an *absolute deviation* between the probabilities of student and teacher, which exerts non-zero corrective signals on all tokens where $q_k \neq p_k$. Conversely, for RKL, Eq. (12) stands for a *relative deviation* between the token-wise log-likelihood ratio and the global divergence term. Moreover, the scaling factor q_k acts as a gating function, damping the gradient contributions of tokens to which the student assigns negligible probability. Accordingly, RKL focuses on correcting a set of dominant tokens rather than striving to matching the teacher on every token. This is commonly considered to respect the OPD's sampling constraint and enable more efficient optimization, thus delivering better performance.

Theorem 2 (Sign-Reversal Pathology of Reverse KL). *The logit gradient of standard reverse KL is opposite in sign to the student-teacher probability gap (violating SLSM) on any token k satisfying*

$$0 < \log \left(\frac{q_k}{p_k} \right) < D_{\mathrm{KL}}(\boldsymbol{q} \parallel \boldsymbol{p}) \tag{13}$$

Under this condition, token k is over-allocated ($q_k > p_k$), yet its logit gradient $\frac{\partial D_{\mathrm{KL}}(\boldsymbol{q}\|\boldsymbol{p})}{\partial z_k}$ is strictly negative, forcing gradient descent to increase z_k further and likely exacerbate the over-allocation.

The proof of Theorem 2 is presented in Appendix B.5, and figure 1 shows a concrete example of the sign reversal phenomenon. We emphasize that while the elevation of logits on such tokens does not necessarily move them away from the teacher in probability space, the sign reversal does constitute a structural failure of alignment that, when it occurs on sufficiently many moderate-mass tokens, is empirically associated with degraded optimization.

A natural response to Theorem 2 one might think of is to combine FKL with RKL, aiming to retain FKL's sign monotonicity while benefiting from RKL's optimization efficiency. Consider the trivial convex combination $\lambda D_{\mathrm{KL}}(\boldsymbol{p}\|\boldsymbol{q}) + (1-\lambda) D_{\mathrm{KL}}(\boldsymbol{q}\|\boldsymbol{p})$, where $\lambda \in (0, 1)$, its gradient w.r.t. logit is

$$\frac{\partial \mathcal{L}_{\mathrm{mix}}}{\partial z_k} = \lambda \cdot (q_k - p_k) + (1-\lambda) \cdot q_k \left[\log \left(\frac{q_k}{p_k} \right) - D_{\mathrm{KL}}(\boldsymbol{q} \parallel \boldsymbol{p}) \right] \tag{14}$$

For any token satisfying Eq. (13), the two terms in Eq. (14) oppose each other. The former moves z_k the correct direction while the latter moves it the opposite direction. Particularly the latter term can dominate for any fixed λ, *e.g.*, at $q_k = p_k$ the mixture gradient is still strictly negative. Therefore, the convex combination merely delays the onset of pathology rather than eliminating it.

3.4 m-RKL: Correcting the Gradient Field on the Softmax Manifold

Since it is impossible to radically eliminate the sign-reversal pathology of RKL via a combination strategy, we turn to alternative methods that directly manipulate the gradient field. Our motivation

stems from the observation that the intrinsic factor giving rise to the non-monotonicity of RKL is the global divergence term $D_{\mathrm{KL}}(\boldsymbol{q}\|\boldsymbol{p})$. With it removed from Eq. (12), we are surprised to find that the resulting gradient form strictly satisfies SLSM, as shown in Eq. (15).

$$\operatorname{sign}\left(q_k \log\left(\frac{q_k}{p_k}\right)\right) = \operatorname{sign}(q_k - p_k), \quad \forall\, k \in \mathcal{V} \tag{15}$$

Meanwhile, this gradient also retains the tail-damping property of the original (still gated by q_k), thus maintaining the optimization efficacy. However, as previously discussed in Section 3.3, $D_{\mathrm{KL}}(\boldsymbol{q}\|\boldsymbol{p})$ is a necessary term to ensure the zero-sum conservation constraint, *i.e.*, Eq. (5). Without this term, the equivalence between the positive and negative gradient components is broken, thereby triggering gradient explosion during training.

To this end, we introduce a multiplicative scaling factor to rebalance the energies of the positive and negative gradient components. Specifically, we first partition the vocabulary by relative probability into two mutually exclusive sets: $\mathcal{V}^+ = \{k \in \mathcal{V} \mid q_k \geq p_k\}$ and $\mathcal{V}^- = \{k \in \mathcal{V} \mid q_k < p_k\}$. We then define the aggregate over-allocated and under-allocated energies as follows, respectively.

$$S^+ = \sum_{k \in \mathcal{V}^+} q_k \log\left(\frac{q_k}{p_k}\right) > 0, \quad S^- = \sum_{k \in \mathcal{V}^-} q_k \log\left(\frac{q_k}{p_k}\right) < 0 \tag{16}$$

Since $D_{\mathrm{KL}}(\boldsymbol{q}\|\boldsymbol{p}) = S^+ + S^- \geq 0$, we always have $|S^+| \geq |S^-|$ always. To allow the energies on both sides to offset each other, a straightforward idea is to proportionally scale up every negative gradient component by $\alpha = |S^+/S^-|$ so that the resulting gradient field satisfies both SLSM and the zero-sum conservation, which is defined as:

$$[G_{\text{m-RKL}}]_k := \begin{cases} q_k \log\left(\frac{q_k}{p_k}\right), & k \in \mathcal{V}^+ \\ \alpha \cdot q_k \log\left(\frac{q_k}{p_k}\right), & k \in \mathcal{V}^- \end{cases} \quad \text{where } \alpha = \left|\frac{S^+}{S^-}\right| \tag{17}$$

Regarding the corresponding loss function of $G_{\text{m-RKL}}$, we have the following proposition.

Proposition 1 (**Non-Integrability**). *For $V \geq 3$, the gradient field $G_{\text{m-RKL}}(\boldsymbol{z})$ is not conservative: there exists no differentiable scalar potential $\mathcal{L}(\boldsymbol{z})$ such that $G_{\text{m-RKL}} = \nabla_{\boldsymbol{z}}\mathcal{L}$. The non-integrability arises because α is a global (non-local) function of $\boldsymbol{q}$ — it depends on the configuration of all tokens — which breaks the cross-partial symmetry required by Clairaut's theorem. Therefore, m-RKL is not a modified f-divergence; it is a structurally corrected gradient field on the Softmax manifold.*

The non-conservative nature of m-RKL raises a legitimate question: without a scalar potential, what optimization guarantee does this corrected field carry? We answer it directly. Although $G_{\text{m-RKL}}$ is not the gradient of any loss, it represents a strict descent direction for forward KL in logit space at every non-degenerate configuration, as established by Proposition 2 (proof in Appendix B.7).

Proposition 2 (**m-RKL as a Descent Direction for FKL**). *For all $\boldsymbol{p}, \boldsymbol{q} \in \Delta^{V-1}$ with $\boldsymbol{q} \neq \boldsymbol{p}$, the m-RKL gradient field, i.e., Eq. (17) satisfies*

$$\langle G_{\text{m-RKL}}, \nabla_{\boldsymbol{z}} D_{\mathrm{KL}}(\boldsymbol{p} \parallel \boldsymbol{q})\rangle > 0 \tag{18}$$

where the inner product is taken over logit coordinates. Consequently, under gradient descent, *i.e.*, $\boldsymbol{z} \leftarrow \boldsymbol{z} - \eta G_{\text{m-RKL}}$, the forward KL divergence $D_{\mathrm{KL}}(\boldsymbol{p}\|\boldsymbol{q})$ strictly decreases on every step.

Implementation. $G_{\text{m-RKL}}$ is injected as an explicit gradient via a custom `autograd.Function` in PyTorch (see Appendix C for the implemented code). The forward pass computes the standard teacher-student logits; the backward pass overrides PyTorch's automatic gradient with Eq. (17).

4 Experiments

4.1 Experimental Setup

Models and Training Data. We conduct all experiments on the Qwen3 (Yang et al., 2025) family of non-thinking versions as both teacher and student policies. To investigate the effectiveness of the proposed approach under different student–teacher capability gaps, we select two models of varying scales as the students: the compact Qwen3-1.7B and the larger Qwen3-8B, which are distilled from the teachers Qwen3-4B-Instruct-2507 and Qwen3-30B-A3B-Instruct-2507, respectively. For training data, we use the DAPO-Math-17K dataset (Yu et al., 2026) following the practice of Li et al. (2026), with trajectories sampled from the student's active policy as prescribed in Section 2.1.

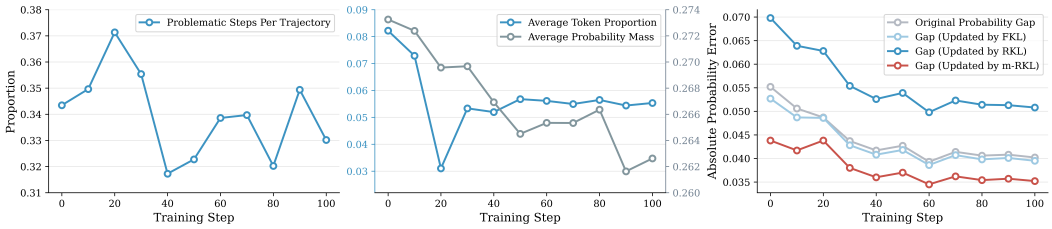

Figure 2: *Left)* The average proportion of generation steps with the sign-reversal pathology in each sampled on-policy trajectory. *Center)* The average proportion of tokens with sign-reversal gradients in the vocabulary, as well as their summed probability mass. *Right)* The mean absolute probability error between the student and the teacher on sign-reversal tokens before and after logit updates.

Evaluation. We primarily evaluate the distilled student models on six widely used math reasoning benchmarks, namely MATH-500 (Hendrycks et al., 2021), AIME'24 (Zhang & Math-AI, 2024), AIME'25 (Zhang & Math-AI, 2025), HMMT'25, HMMT'26 (Dekoninck et al., 2026), and Minerva-Math (Lewkowycz et al., 2022). Following Heo et al. (2026), we report **avg@4** accuracy on MATH-500 and Minerva-Math, and **avg@16** accuracy on the rest of four benchmarks. Also, we include four out-of-domain benchmarks — MMLU-Redux (Gema et al., 2024), IFEval (Zhou et al., 2023), HumanEval (Chen et al., 2021), and MBPP (Austin et al., 2021) — to investigate the impact of OPD on general capabilities of the base student. More evaluation details are deferred to Appendix A.

Baselines. We compare m-RKL against other KL-based distillation objectives, including FKL, RKL (Gu et al., 2024), symmetric KL (Sym-KL), which combines the two via 0.5 FKL + 0.5 RKL, Jensen–Shannon divergence (JS) (Agarwal et al., 2024), skewed FKL (SFKL) and skewed RKL (SRKL) (Ko et al., 2024) with the smoothing coefficient set as 0.1, Adaptive KL (AKL) (Wu et al., 2025). All methods are trained under the same setup, using the same dataset, identical hyperparameters, a shared teacher checkpoint, and the same student initialization to ensure a fair comparison.

Implementations. All our experiments are conducted on 2 compute nodes, each equipped with 8 NVIDIA A100-SXM4-80GB GPUs. We implement the OPD algorithm and various loss functions based on `TRL`'s (von Werra et al., 2020) `DistillationTrainer`. During training, we maintain two instances of the student: one undergoes standard parameter updates, while the other is hosted on a separate vLLM server to improve trajectory sampling efficiency. And the weights are synchronized from the training instance to the inference instance every training step. Besides, the training instance is optimized by the AdamW (Loshchilov & Hutter, 2017) optimizer, applying the cosine scheduler with a learning rate of 3×10^{-6}. Full implementation details are provided in Appendix A.

4.2 Main Results

Mechanism. We complement our theoretical analysis with an empirical study of *sign-reversal incidence* during training (Figure 2). As shown in the left panel, over 30% of generation steps on average consistently exhibit sign reversal for trajectories sampled from the student policy; further inspection (center panel) reveals that, within these affected steps, although only $\sim 5\%$ of vocabulary tokens are pathological, they collectively account for more than a quarter of the probability mass. The rightmost panel confirms the resulting degradation in probability flow behavior. Updating the logits with RKL gradient field enlarges the student-teacher probability gap on those sign-reversal tokens relative to the original gap before updates, whereas both SLSM-satisfying objectives (FKL and m-RKL) narrow the gap, with m-RKL achieving a greater degree of gap closure. These findings demonstrate that m-RKL indeed drives the student closer to the teacher at the probability-flow level.

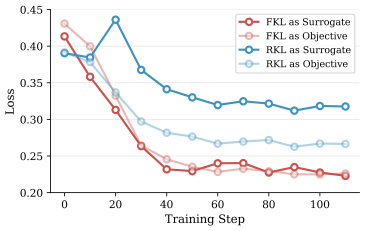

Figure 3: Convergence curve.

Convergence Inspection. Since the corrected gradient field $G_{\text{m-RKL}}$ we propose does not admit a scalar potential, we resort to conventional surrogate losses (*e.g.*, FKL or RKL) to monitor the convergence of the student during training. As shown in figure 3, training with $G_{\text{m-RKL}}$ leads to a steady decrease in both FKL and RKL. Moreover, compared with optimizing the model using their respective true gradients, our corrected gradient exhibits a loss curve highly consistent with that of FKL. This exactly provides the empirical evidence for proposition 2.

Table 1: Performance comparison of Qwen3-1.7B trained with different distillation objectives.

Method	MATH-500	Minerva	AIME'24	AIME'25	HMMT'25	HMMT'26	Average
Qwen3-1.7B (Student)	73.1	40.4	12.4	11.7	6.0	6.3	25.0
Qwen3-4B-IT (Teacher)	93.3	68.1	64.0	47.6	28.9	18.1	53.3
On-policy distillation							
FKL	72.2	**56.5**	41.3	36.7	21.7	12.2	40.1
RKL (Gu et al., 2024)	80.1	55.4	44.5	31.8	20.2	11.9	40.7
SFKL (Ko et al., 2024)	79.0	53.8	44.7	36.4	21.6	10.7	41.0
SRKL (Ko et al., 2024)	80.3	53.4	43.5	33.3	20.7	12.2	40.6
Sym-KL	78.5	55.1	45.9	36.9	22.3	12.3	41.8
JS (Agarwal et al., 2024)	81.1	53.8	45.5	35.3	21.3	11.2	41.4
AKL (Wu et al., 2025)	79.0	55.2	44.2	35.4	21.4	11.5	41.1
m-RKL (Ours)	**83.3**	56.1	**46.7**	**37.3**	**23.1**	**12.9**	**43.2**

Table 2: Performance comparison of Qwen3-8B trained with different distillation objectives.

Method	MATH-500	Minerva	AIME'24	AIME'25	HMMT'25	HMMT'26	Average
Qwen3-8B (Student)	84.8	62.4	28.0	23.7	11.3	16.8	37.8
Qwen3-30B-A3B (Teacher)	95.0	75.4	74.2	60.7	40.7	33.7	63.3
On-policy distillation							
FKL	85.8	69.1	71.0	65.0	33.1	20.3	57.4
RKL (Gu et al., 2024)	90.9	67.3	69.5	55.6	30.4	23.2	56.2
SFKL (Ko et al., 2024)	89.8	68.9	69.7	61.4	32.6	22.7	57.5
SRKL (Ko et al., 2024)	89.9	68.0	69.0	55.8	31.5	23.6	56.3
Sym-KL	89.3	67.0	71.2	62.8	32.7	20.3	57.2
JS (Agarwal et al., 2024)	90.1	68.0	70.4	61.2	30.7	23.3	57.3
AKL (Wu et al., 2025)	88.6	68.0	71.7	63.0	33.2	23.4	58.0
m-RKL (Ours)	**93.5**	**70.5**	**71.9**	**65.2**	**34.0**	**24.4**	**59.9**

Reasoning Performance on Math. Tables 1 and 2 summarize the performance of Qwen3-1.7B and Qwen3-8B students trained with different objectives on six mathematical reasoning benchmarks. Note that bold and underlined numbers denote the best and second-best results in each column, respectively. Overall, our m-RKL achieves the highest average accuracy on both model scales — 43.2% at 1.7B and 59.9% at 8B — outperforming the strongest baseline by 1.4 and 1.9 points and the standard RKL by 2.6 and 3.8 points, respectively. Our method attains the best result on five of six benchmarks at 1.7B and on all six at 8B; the only exception is Minerva at 1.7B, where m-RKL remains competitive with FKL (56.1% vs. 56.5%). Interestingly, FKL wins on Minerva but falls clearly behind on all four competition benchmarks, suggesting its mode-covering behavior is better suited to broad-coverage tasks than to hard reasoning. In contrast, the gains of m-RKL over RKL concentrate exactly where they matter most: on the most challenging benchmarks, *e.g.*, +5.5% on AIME'25 and +2.9% on HMMT'25 at 1.7B, and up to +9.6% on AIME'25 at 8B.

Two further observations are worth noting. First, the improvement over RKL grows with model capacity (from +2.6% at 1.7B to +3.8% at 8B), suggesting that correcting the logit gradient-field become increasingly impactful as the student grows stronger. Second, at 8B our method substantially narrows the gap to the Qwen3-30B-A3B teacher, raising the average from 37.8% (base student) to 59.9%, close to the teacher's 63.3%. These results corroborate our theoretical analysis: sign-reversal-prone objectives consistently rank at the bottom, while SLSM-satisfying objectives dominate the top, with m-RKL achieving the best overall trade-off between the two failure modes.

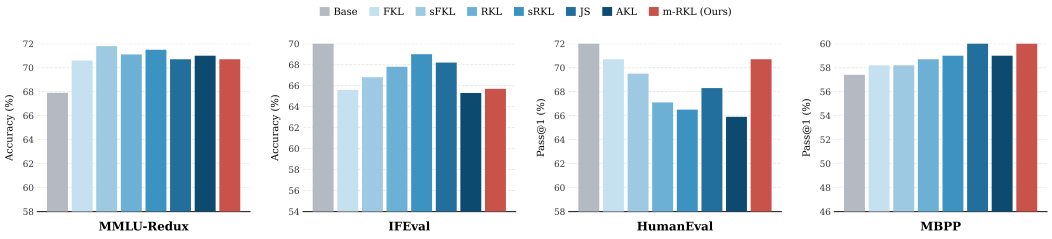

Figure 4: Performance on general benchmarks for Qwen3-1.7B training with different objectives.

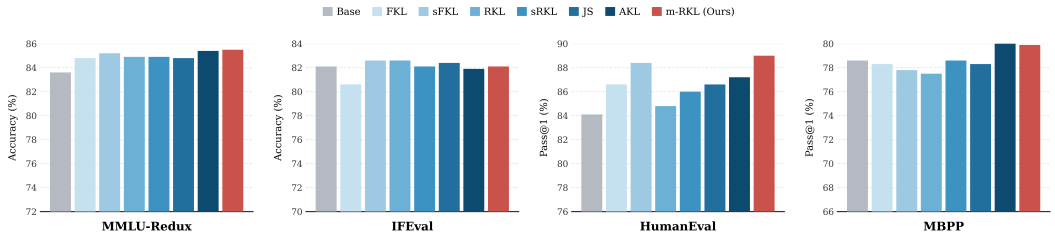

Figure 5: Performance on general benchmarks for Qwen3-8B training with different objectives.

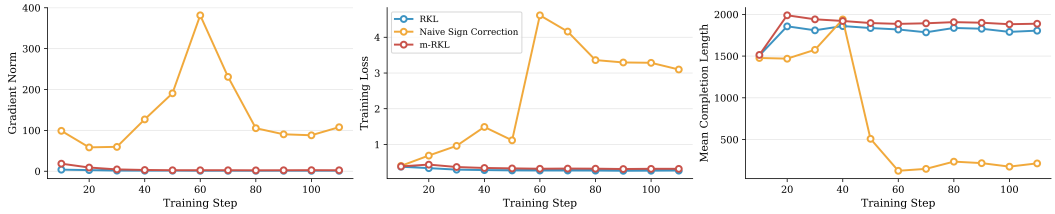

Figure 6: The curves of *gradient norm*, *training loss*, and *mean completion length* over training steps regarding the three objectives compared in Table 3.

Table 3: Ablation study.

Objective	Logit Gradient	Sign Alignment	Zero-Sum Conservation	Minerva	AIME'24	AIME'25	HMMT25
RKL	$q_k \left[\log\left(\frac{q_k}{p_k}\right) - D_{\mathrm{KL}}(\boldsymbol{q} \,\|\, \boldsymbol{p})\right]$	✗	✓	55.4	44.5	31.8	20.2
Naive Sign Correction	$q_k \log\left(\frac{q_k}{p_k}\right)$	✓	✗	39.3	7.5	9.8	1.3
m-RKL (Ours)	Eq. (17)	✓	✓	**56.1**	**46.7**	**37.3**	**23.1**

Results on General Benchmarks. Figures 4 and 5 further evaluate the distilled students on four general benchmarks to assess whether math-specific distillation compromises general capabilities. For Qwen3-1.7B, all objectives improve over the base model on MMLU-Redux, and m-RKL best preserves coding ability — remaining near the base on HumanEval (70.7%) where RKL degrades sharply (67.1%). IFEval is the only benchmark where all distilled models fall below the base, indicating a general sensitivity of instruction-following to domain-specific distillation. For Qwen3-8B, m-RKL attains the best results on MMLU-Redux and HumanEval, ranks second on MBPP, and stays on par with the base on IFEval. These results confirm that m-RKL preserves — and in several cases enhances — the general capabilities of the base model.

Ablation Study. Table 3 ablates the two defining properties of our corrected gradient field: sign alignment and zero-sum conservation. Dropping sign alignment recovers the standard RKL, which performs moderately but suffers from sign reversal. Enforcing sign alignment alone (*Naive Sign Correction*) breaks training: accuracy collapses to 7.5% and 9.8% on AIME'24 and AIME'25, and fails entirely (1.3%) on HMMT'25, as the broken zero-sum structure causes gradient explosion during training (Figure 6). Only when both properties hold does m-RKL attain its full performance, confirming that the two are complementary and jointly necessary.

5 Conclusion

This paper reframed divergence choice in on-policy distillation as a question about logit-level gradient fields rather than global distribution geometry. Softmax coupling separates two alignment notions the mode-covering/mode-seeking dichotomy conflates: no f-divergence guarantees every token's probability moves toward the teacher at each step (Proposition 3), yet forward KL is the unique f-divergence with sign-aligned logit gradients (Theorem 1), while reverse KL exerts a provably wrong-direction force on over-allocated tokens (Theorem 2) that convex mixtures cannot remove. We therefore proposed m-RKL, a non-conservative gradient field restoring sign alignment and zero-sum conservation while inheriting RKL's tail damping, proven to be a strict descent direction for FKL with a quantitative rate (Proposition 2). A pathology-targeted ablation confirms the causal role of sign reversal, and across Qwen3 students of two scales, m-RKL outperforms eight KL-family objectives on math reasoning without degrading general capabilities. Our evaluation covers one domain and model family; broader domains and preconditioned-optimizer-compatible guarantees remain future work.

AI USE STATEMENT

In this work, we used generative AI tools for **polishing and editing the written text**. We have not used generative AI tools for **generating primary research ideas, drafting sections of the paper, or retrieval and discovery (such as finding related work)**, and the remaining manuscript preparation steps are not applicable to this work. Additionally, we used generative AI tools for **assisting in the derivation and verification of mathematical proofs**. We have reviewed all AI-assisted work by thoroughly checking all AI-derived mathematical steps line-by-line for logical consistency and ensuring all text accurately reflects our original findings without altering the core scientific meaning. We take responsibility for the final content of this work, including text, claims or artifacts produced with the aid of generative AI.

ETHICS STATEMENT

This work studies optimization objectives for knowledge distillation between language models and uses only publicly available datasets and model checkpoints (DAPO-Math-17K, Qwen3, and standard evaluation benchmarks). It involves no human subjects, no personal data, and no deployment of trained models. We see no foreseeable dual-use risk specific to improved distillation objectives, beyond those generally applicable to stronger LLMs. The case study in Appendix D shows model-generated reasoning traces, which may contain errors; we annotate them accordingly.

REPRODUCIBILITY STATEMENT

Complete proofs of all theoretical claims are provided in Appendix B, and a self-contained PyTorch implementation of the m-RKL gradient field is provided in Appendix C. Training and evaluation hyperparameters are listed in Tables 4 and 5 of Appendix A, and implementation details (framework, optimizer, rollout synchronization) in Section 4.1. We will release all training code, custom autograd operators for all compared objectives, and evaluation scripts upon publication.

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.

Jacob Austin, Augustus Odena, Maxwell Nye, Maarten Bosma, Henryk Michalewski, David Dohan, Ellen Jiang, Carrie Cai, Michael Terry, Quoc Le, et al. Program synthesis with large language models. *arXiv preprint arXiv:2108.07732*, 2021.

Mark Chen, Jerry Tworek, Heewoo Jun, Qiming Yuan, Henrique Ponde de Oliveira Pinto, Jared Kaplan, Harri Edwards, Yuri Burda, Nicholas Joseph, Greg Brockman, Alex Ray, Raul Puri, Gretchen Krueger, Michael Petrov, Heidy Khlaaf, Girish Sastry, Pamela Mishkin, Brooke Chan, Scott Gray, Nick Ryder, Mikhail Pavlov, Alethea Power, Lukasz Kaiser, Mohammad Bavarian, Clemens Winter, Philippe Tillet, Felipe Petroski Such, Dave Cummings, Matthias Plappert, Fotios Chantzis, Elizabeth Barnes, Ariel Herbert-Voss, William Hebgen Guss, Alex Nichol, Alex Paino, Nikolas Tezak, Jie Tang, Igor Babuschkin, Suchir Balaji, Shantanu Jain, William Saunders, Christopher Hesse, Andrew N. Carr, Jan Leike, Josh Achiam, Vedant Misra, Evan Morikawa, Alec Radford, Matthew Knight, Miles Brundage, Mira Murati, Katie Mayer, Peter Welinder, Bob McGrew, Dario Amodei, Sam McCandlish, Ilya Sutskever, and Wojciech Zaremba. Evaluating large language models trained on code, 2021.

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. 2026. URL https://arxiv.org/abs/2605.00674.

Michael Franke and Judith Degen. The softmax function: Properties, motivation, and interpretation. 2023.

Bolin Gao and Lacra Pavel. On the properties of the softmax function with application in game theory and reinforcement learning. *arXiv preprint arXiv:1704.00805*, 2017.

Aryo Pradipta Gema, Joshua Ong Jun Leang, Giwon Hong, Alessio Devoto, Alberto Carlo Maria Mancino, Rohit Saxena, Xuanli He, Yu Zhao, Xiaotang Du, Mohammad Reza Ghasemi Madani, Claire Barale, Robert McHardy, Joshua Harris, Jean Kaddour, Emile van Krieken, and Pasquale Minervini. Are we done with mmlu?, 2024.

Tilmann Gneiting and Adrian E Raftery. Strictly proper scoring rules, prediction, and estimation. *Journal of the American statistical Association*, 102(477):359–378, 2007.

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.

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.

Byeongho Heo, Jaehui Hwang, Sangdoo Yun, and Dongyoon Han. On-policy delta distillation. *arXiv preprint arXiv:2607.15161*, 2026.

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

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

Seongryong Jung, Suwan Yoon, DongGeon Kim, and Hwanhee Lee. Todi: Token-wise distillation via fine-grained divergence control. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pp. 8089–8102, 2025.

Yeongmin Kim, Donghyeok Shin, Mina Kang, Byeonghu Na, and Il-Chul Moon. Distillation of large language models via concrete score matching. In *International Conference on Learning Representations*, volume 2026, pp. 16649–16682, 2026.

Jongwoo Ko, Sungnyun Kim, Tianyi Chen, and Se-Young Yun. Distillm: Towards streamlined distillation for large language models. *arXiv preprint arXiv:2402.03898*, 2024.

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.

Yingru Li. Logit dynamics in softmax policy gradient methods. *arXiv preprint arXiv:2506.12912*, 2025.

Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. *arXiv preprint arXiv:1711.05101*, 2017.

Jingyu Peng, Maolin Wang, Hengyi Cai, Yuchen Li, Kai Zhang, Shuaiqiang Wang, Dawei Yin, and Xiangyu Zhao. Adaswitch: Balancing exploration and guidance in knowledge distillation via adaptive switching. *arXiv preprint arXiv:2510.07842*, 2025.

Andrea Pozzi, Alessandro Incremona, Daniele Tessera, and Daniele Toti. Mitigating exposure bias in large language model distillation: An imitation learning approach. *Neural Computing and Applications*, 37(18):12013–12029, 2025.

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.

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

Kimi Team, Tongtong Bai, Yifan Bai, Yiping Bao, Jianfeng Cai, Xinyuan Cai, Peizhou Cao, Yuxuan Cao, Ziwei Chai, Y Charles, et al. Kimi k3: Open frontier intelligence. *arXiv preprint arXiv:2607.24653*, 2026.

Leandro von Werra, Younes Belkada, Lewis Tunstall, Edward Beeching, Tristan Thrush, Nathan Lambert, Shengyi Huang, Kashif Rasul, and Quentin Gallouédec. TRL: Transformers Reinforcement Learning, 2020. URL https://github.com/huggingface/trl.

Taiqiang Wu, Chaofan Tao, Jiahao Wang, Runming Yang, Zhe Zhao, and Ngai Wong. Rethinking kullback-leibler divergence in knowledge distillation for large language models. In *Proceedings of the 31st International Conference on Computational Linguistics*, pp. 5737–5755, 2025.

Anyi Xu, Bangcai Lin, Bing Xue, Bingxuan Wang, Bingzheng Xu, Bochao Wu, Bowei Zhang, Chaofan Lin, Chen Dong, Chenchen Ling, et al. Deepseek-v4: Towards highly efficient million-token context intelligence. *arXiv preprint arXiv:2606.19348*, 2026.

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

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. *Advances in Neural Information Processing Systems*, 38:113222–113244, 2026.

Aohan Zeng, Xin Lv, Zhenyu Hou, Zhengxiao Du, Qinkai Zheng, Bin Chen, Da Yin, Chendi Ge, Chenghua Huang, Chengxing Xie, et al. Glm-5: from vibe coding to agentic engineering. *arXiv preprint arXiv:2602.15763*, 2026.

Yifan Zhang and Team Math-AI. American invitational mathematics examination (aime) 2024, 2024. URL https://huggingface.co/datasets/math-ai/aime24.

Yifan Zhang and Team Math-AI. American invitational mathematics examination (aime) 2025, 2025. URL https://huggingface.co/datasets/math-ai/aime25.

Jeffrey Zhou, Tianjian Lu, Swaroop Mishra, Siddhartha Brahma, Sujoy Basu, Yi Luan, Denny Zhou, and Le Hou. Instruction-following evaluation for large language models, 2023. URL https://arxiv.org/abs/2311.07911.

A APPENDIX

Tables 4 and 5 list the training and evaluation hyperparameters.

Table 4: Training hyperparameters for OPD in our experiments unless otherwise specified.

Hyperparameter	**Value**
Training framework	`TRL`
Global batch size	128
Mini batch size	16
Precision	bf16
Training steps	110
Warmup steps	10
Rollout number	1
Maximum prompt length	1024
Maximum response length	2048
Temperature	1.0
Top-p	1.0
Learning rate	3e-6
Gradient clipping	1.0
vLLM sync frequency	every training step

Table 5: Evaluation protocol.

Benchmark	**Metric**	**Dataset size**	**Rollout number**	**Sample Paramters**			
				Temperature	**Top-p**	**Top-k**	**Max-tokens**
AIME'25	Avg@16	30	16	0.7	0.8	20	31744
AIME'24		30	16				
HMMT'25		30	16				
HMMT'26		33	16				
Minerva-MATH	Avg@4	272	4	0.7	0.8	20	31744
MATH-500		500	4				8192
MMLU-Redux	Overall accuracy	5700	1	0.7	0.8	20	31744
IFEval	Prompt level strict accuracy	541	1	0	-	-	4096
HumanEval	Pass@1	164	1	0	-	-	1024
MBPP		1401	1	0	-	-	1024

Overhead. m-RKL adds only two per-position reductions (S^+, S^-) over RKL — $O(V)$ extra work per position, no extra forward/backward passes, no teacher-side changes — so overhead should be within noise. Table 6 presents the specific statistics related to training efficiency.

Table 6: Statistics pertaining to training efficiency.

Objective	**Total Floating Point Operations**	**Total Training Time**	**Train Samples per Second**
FKL	2.175e+13	2003 s	7.05
RKL	2.175e+13	1973 s	7.15
JS	2.175e+13	2084 s	6.77
m-RKL (Ours)	2.175e+13	1997 s	7.08

B PROOFS OF STATEMENTS

This appendix collects the complete proofs of all theoretical claims in the main text.

B.1 PROOF OF LEMMA 1 (GENERAL LOGIT GRADIENT OF f-DIVERGENCE)

Proof. Recall the discrete f-divergence between two arbitrary distributions $\boldsymbol{p}$ and $\boldsymbol{q}$ is defined as

$$D_f(\boldsymbol{p} \parallel \boldsymbol{q}) = \sum_{j=1}^{V} q_j f\left(\frac{p_j}{q_j}\right) \tag{19}$$

Then the partial derivative of $D_f(\boldsymbol{p} \parallel \boldsymbol{q})$ with respect to each probability q_j is

$$\frac{\partial D_f}{\partial q_j} = \frac{\partial}{\partial q_j}\left[q_j f\left(\frac{p_j}{q_j}\right)\right] = f\left(\frac{p_j}{q_j}\right) + q_j \cdot f'\left(\frac{p_j}{q_j}\right) \cdot \left(-\frac{p_j}{q_j^2}\right) \tag{20}$$

Defining the likelihood ratio $u_j = \frac{p_j}{q_j} > 0$, this simplifies to

$$\frac{\partial D_f}{\partial q_j} = f(u_j) - u_j f'(u_j) \tag{21}$$

Let $g(u) = f(u) - uf'(u)$, applying the multivariate chain rule and the Softmax Jacobian yields

$$\frac{\partial D_f}{\partial z_k} = \sum_{j=1}^{V} \frac{\partial D_f}{\partial q_j}\frac{\partial q_j}{\partial z_k} = \sum_{j=1}^{V} g(u_j) q_j (\delta_{jk} - q_k) \tag{22}$$

Expanding the sum gives

$$\frac{\partial D_f}{\partial z_k} = \sum_{j=1}^{V} g(u_j) q_j \delta_{jk} - \sum_{j=1}^{V} g(u_j) q_j q_k = g(u_k) q_k - q_k \sum_{j=1}^{V} g(u_j) q_j \tag{23}$$

Factoring out q_k reveals an expectation under the distribution $\boldsymbol{q}$, *i.e.*

$$\frac{\partial D_f}{\partial z_k} = q_k \left(g(u_k) - \mathbb{E}_{\boldsymbol{q}}[g(u_j)]\right) = q_k \left[g\left(\frac{p_k}{q_k}\right) - \sum_{j=1}^{V} q_j g\left(\frac{p_j}{q_j}\right)\right] \tag{24}$$

Since f is strictly convex, $f''(u) > 0$. Differentiating $g(u)$, we obtain

$$g'(u) = f'(u) - (f'(u) + uf''(u)) = -uf''(u) < 0 \quad \forall\, u > 0 \tag{25}$$

Thus, $g(u)$ is strictly decreasing everywhere. This completes the proof. □

B.2 PROOF OF LEMMA 2 (PROBABILITY-FLOW DECOMPOSITION)

Recall that $q_k = e^{z_k}/Z$ with the normalizer $Z = \sum_j e^{z_j}$.

Differentiating gives the Softmax Jacobian *i.e.*, Eq. (3)

$$\frac{\partial q_k}{\partial z_j} = q_k(\delta_{kj} - q_j) \tag{26}$$

Under the continuous flow $\dot{\boldsymbol{z}} = -G(\boldsymbol{z})$ ($\dot{z}_k = -G_k$), the multivariate chain rule yields

$$\frac{dq_k}{ds} = \sum_{j=1}^{V} \frac{\partial q_k}{\partial z_j}\frac{dz_j}{ds} = \sum_{j=1}^{V} q_k(\delta_{kj} - q_j)\,(-G_j) = q_k\Big(-G_k + \sum_{j=1}^{V} q_j G_j\Big) = q_k\big(\mathbb{E}_{\boldsymbol{q}}[G] - G_k\big), \tag{27}$$

which is Eq. (10)

B.3 Proof of Impossibility of Probability-Level Monotonicity

Proposition 3 (Impossibility of Token-Level Probability Monotonicity for f-Divergences). *For $V \geq 3$, there is no twice-differentiable strictly convex generator f with $f(1) = 0$ whose logit gradient field in form of Eq. (8) satisfies the centered criterion:*

$$\operatorname{sign}\big(G_k - \mathbb{E}_{\boldsymbol{q}}[G]\big) = \operatorname{sign}(q_k - p_k), \quad \forall\, k \in \mathcal{V},\ \forall\, \boldsymbol{p}, \boldsymbol{q} \in \Delta^{V-1}. \tag{28}$$

Consequently, no f-divergence guarantees that every token's probability moves toward the teacher at each step of the induced flow (Lemma 2).

Proof. Write $u_j = p_j/q_j$, $\mu = \mathbb{E}_{\boldsymbol{q}}[g(u)]$, $m_2 = \sum_j q_j^2$, and $\nu = \sum_j q_j^2 g(u_j)$.

Since $\mathbb{E}_{\boldsymbol{q}}[G] = \sum_j q_j^2(g(u_j) - \mu) = \nu - \mu m_2$, for any token k with $u_k = 1$:

$$G_k - \mathbb{E}_{\boldsymbol{q}}[G] = q_k\big(g(1) - \mu\big) - \nu + \mu m_2. \tag{29}$$

Fix any $\boldsymbol{q}$ with two distinct coordinates $i \neq j$ such that $q_i \neq q_j$, and any token $k \notin i, j$ (exists since $V \geq 3$). Define $\boldsymbol{p}$ by transferring mass $\varepsilon > 0$ from p_i to p_j: $p_i = q_i - \varepsilon$, $p_j = q_j + \varepsilon$, $p_\ell = q_\ell$ otherwise; in particular $u_k = 1$. Eq. (28) at token k demands $G_k - \mathbb{E}_{\boldsymbol{q}}[G] = 0$ for every $\varepsilon > 0$.

At $\varepsilon = 0$ all ratios equal 1, so $\mu = g(1)$, $\nu = m_2 g(1)$, and Eq. (29) vanishes.

Differentiating Eq. (29) with respect to ε at $\varepsilon = 0$. Since $\frac{du_i}{d\varepsilon} = -1/q_i$ and $\frac{du_j}{d\varepsilon} = 1/q_j$, we have

$$\mu' = -g'(1) + g'(1) = 0, \tag{30}$$

$$\nu' = -q_i g'(1) + q_j g'(1) = g'(1)(q_j - q_i), \tag{31}$$

Hence

$$\frac{d}{d\varepsilon}\Big[G_k - \mathbb{E}_{\boldsymbol{q}}[G]\Big]_{\varepsilon=0} = -g'(1)(q_j - q_i). \tag{32}$$

By Eq. (25), $g'(1) = -f''(1) < 0$ strictly, and $q_j \neq q_i$ by construction, so the derivative is nonzero.

Therefore, for all sufficiently small $\varepsilon > 0$, $G_k - \mathbb{E}_{\boldsymbol{q}}[G] \neq 0$ while $\operatorname{sign}(q_k - p_k) = 0$, violating Eq. (28). Since f was arbitrary, the claim holds for the entire f-divergence family.

A concrete FKL instance. Let $\boldsymbol{p} = (0.2, 0.3, 0.5)$, $\boldsymbol{q} = (0.3, 0.3, 0.4)$.

FKL gives $G = \boldsymbol{q} - \boldsymbol{p} = (0.1, 0, -0.1)$ and $\mathbb{E}_{\boldsymbol{q}}[G] = -0.01 \neq 0$.

Token 2 has $q_2 = p_2$, yet

$$\frac{dq_2}{ds} = q_2(\mathbb{E}_{\boldsymbol{q}}[G] - G_2) = -0.003 < 0 \tag{33}$$

Although FKL satisfies SLSM, token 2's probability gap widens under the FKL flow. □

B.4 Proof of Theorem 1 (Unique Sign Alignment of Forward KL)

Proof. Based on the general gradient form of any f-divergence with respect to logit

$$G_k = \frac{\partial D_f(\boldsymbol{p} \,\|\, \boldsymbol{q})}{\partial z_k} = q_k\left(g(u_k) - \mathbb{E}_{\boldsymbol{q}}[g(u)]\right) \tag{34}$$

where $u_j = \frac{p_j}{q_j}$ and $g(u) = f(u) - uf'(u)$.

SLSM requires that for all valid probability distributions $\boldsymbol{p}, \boldsymbol{q}$:

$$\operatorname{sign}(G_k) = \operatorname{sign}(q_k - p_k) \quad \forall\, k \in \mathcal{V} \tag{35}$$

Since $q_k > 0$, we can divide by q_k without altering the sign. Moreover, noting that $\operatorname{sign}(q_k - p_k) = \operatorname{sign}\left(1 - \frac{p_k}{q_k}\right) = \operatorname{sign}(1 - u_k)$, the condition simplifies to

$$\operatorname{sign}\left(g(u_k) - \mathbb{E}_{\boldsymbol{q}}[g(u)]\right) = \operatorname{sign}(1 - u_k) \quad \forall\, k \in \mathcal{V} \tag{36}$$

Step 1: Establish the Affine Requirement on $g(u)$

Consider the boundary case where $p_k = q_k$, which means $u_k = 1$.

- Since $q_k - p_k = 0$, sign monotonicity requires $\text{sign}(G_k) = 0$, which implies $G_k = 0$.
- This forces:

$$g(1) = \mathbb{E}_{\boldsymbol{q}}[g(u)] = \sum_{j=1}^{V} q_j g\left(\frac{p_j}{q_j}\right) \tag{37}$$

For this equality to hold for any arbitrary distributions $\boldsymbol{p}$ and $\boldsymbol{q}$ over the $V-1$ free variables (where $\sum p_j = 1$ and $\sum q_j = 1$), the expected value $\mathbb{E}_{\boldsymbol{q}}[g(u)]$ must evaluate to a constant independent of the specific distribution over those variable states.

The only function $g(u)$ whose expectation $\mathbb{E}_{\boldsymbol{q}}[g(u)]$ is constant across all valid probability vectors $\boldsymbol{p}$ and $\boldsymbol{q}$ are the following affine functions:

$$g(u) = cu + d \quad \text{for constants } c, d \in \mathbb{R} \tag{38}$$

To verify this point, we substitute Eq. (38) to the expectation such that $\mathbb{E}_{\boldsymbol{q}}[g(u)] = \sum_{j=1}^{V} q_j g(u_j) = \sum_{j=1}^{V} q_j \left(c\frac{p_j}{q_j} + d\right) = c\sum_{j=1}^{V} p_j + d\sum_{j=1}^{V} q_j = c \cdot 1 + d \cdot 1 = c + d$.

Step 2: Solve the Ordinary Differential Equation for $f(u)$

Equating $g(u)$ to its definition yields a first-order differential equation for $f(u)$:

$$f(u) - uf'(u) = cu + d \tag{39}$$

Divide both sides by u^2 :

$$\frac{f(u) - uf'(u)}{u^2} = \frac{c}{u} + \frac{d}{u^2} \tag{40}$$

Recognizing the left-hand side as the negative derivative of $\frac{f(u)}{u}$:

$$-\frac{d}{du}\left(\frac{f(u)}{u}\right) = \frac{c}{u} + \frac{d}{u^2} \Longrightarrow \frac{d}{du}\left(\frac{f(u)}{u}\right) = -\frac{c}{u} - \frac{d}{u^2} \tag{41}$$

Integrating both sides with respect to u :

$$\frac{f(u)}{u} = -c\log u + \frac{d}{u} + C_0 \tag{42}$$

Multiplying by u :

$$f(u) = -cu\log u + d + C_0 u \tag{43}$$

Step 3: Apply f-Divergence Constraints

Every f-divergence generator function must satisfy $f(1) = 0$:

$$f(1) = -c\log(1) + d + C_0 = 0 \Longrightarrow C_0 = -d \tag{44}$$

Substituting $C_0 = -d$ back into the generator function:

$$f(u) = -cu\log u + d(1-u) \tag{45}$$

Next, evaluate the convexity condition ($f''(u) > 0$ for $u > 0$):

$$f'(u) = -c(\log u + 1) - d \Longrightarrow f''(u) = -\frac{c}{u} \tag{46}$$

For $f(u)$ to be strictly convex, we demand $c < 0$. Let $a = -c > 0$.

Step 4: Evaluate the Resulting Divergence

Substitute $f(u) = au\log u + d(1-u)$ back into the f-divergence formula:

$$\begin{aligned} D_f(\boldsymbol{p}\|\boldsymbol{q}) &= \sum_{j=1}^{V} q_j \left(a\frac{p_j}{q_j}\log\left(\frac{p_j}{q_j}\right) + d\left(1 - \frac{p_j}{q_j}\right)\right) \\ &= a\sum_{j=1}^{V} p_j \log\left(\frac{p_j}{q_j}\right) + d\sum_{j=1}^{V}(q_j - p_j) \end{aligned} \tag{47}$$

Since $\sum_j q_j = \sum_j p_j = 1$, the linear shift term $d\sum_j(q_j - p_j) = 0$ vanishes completely. Thus:

$$D_f(\boldsymbol{p}\|\boldsymbol{q}) = a\sum_{j=1}^{V} p_j \log\left(\frac{p_j}{q_j}\right) = a \cdot D_{\text{FKL}}(\boldsymbol{p}\|\boldsymbol{q}) \tag{48}$$

where $a > 0$ and setting $a = 1$ recovers the standard forward KL divergence.

In conclusion, FKL (up to a positive scale factor) is the unique f-divergence whose un-modified raw gradient with respect to the logit inherently satisfies *strict local sign monotonicity*. □

B.5 Proof of Theorem 2 (Sign-Reversal Pathology of Reverse KL)

Proof. Recall the sign-reversal condition, *i.e.*, Eq. (13)

$$0 < \log\left(\frac{q_k}{p_k}\right) < D_{\text{KL}}(\boldsymbol{q} \parallel \boldsymbol{p}) \tag{49}$$

If $\log(q_k/p_k) > 0$, which means $q_k > p_k$, so the student-teacher probability gap $q_k - p_k > 0$.

On the other hand, if $\log(q_k/p_k) < D_{\text{KL}}(\boldsymbol{q} \parallel \boldsymbol{p})$, according to the logit gradient of standard RKL, *i.e.*, Eq. (12), we obtain (given that $q_k > 0$ always)

$$\frac{\partial D_{\text{KL}}(\boldsymbol{q} \parallel \boldsymbol{p})}{\partial z_k} = q_k\left[\log\left(\frac{q_k}{p_k}\right) - D_{\text{KL}}(\boldsymbol{q} \parallel \boldsymbol{p})\right] < 0 \tag{50}$$

Therefore, the logit gradient of standard RKL is opposite in sign to the student-teacher probability gap (violating SLSM) on any token k satisfying Eq. (13). □

B.6 Proof of Proposition 1 (Non-Integrability)

Proof. We argue by contradiction. Recall that if a vector field on an open domain were the gradient of a C^2 (twice continuously differentiable) scalar potential, then Clairaut's theorem would force its Jacobian to be symmetric, *i.e.*, $\partial G_k/\partial z_l = \partial G_l/\partial z_k$ for all indices k, l, on every connected open neighborhood contained in the domain. We exhibit a configuration where this fails.

Since $V \geq 3$, as in the statement, choose a distribution pair with two distinct over-allocated tokens $k, l \in \mathcal{V}^+$ (so $q_k > p_k$ and $q_l > p_l$), and restrict to a sufficiently small neighborhood U on which the partition $\mathcal{V}^+/\mathcal{V}^-$ is constant (no token crosses its own boundary $q_j = p_j$).

Such a configuration always exists for $V \geq 3$; one example is $\boldsymbol{p} = (0.5, 0.4, 0.1)$ and $\boldsymbol{q} = (0.6, 0.25, 0.15)$, for which $\mathcal{V}^+ = \{1, 3\}$. On U the field is smooth, and for over-allocated tokens it does not involve the factor α:

$$G_k = q_k \log\left(\frac{q_k}{p_k}\right), \qquad G_l = q_l \log\left(\frac{q_l}{p_l}\right) \tag{51}$$

Differentiating G_k w.r.t. the other token's logit z_l ($k \neq l$), and applying the Softmax Jacobian,

$$\frac{\partial G_k}{\partial z_l} = \frac{\partial q_k}{\partial z_l}\left(\log\frac{q_k}{p_k} + 1\right) = q_k(\delta_{kl} - q_l)\left(1 + \log\frac{q_k}{p_k}\right) = -q_k q_l\left(1 + \log\frac{q_k}{p_k}\right) \tag{52}$$

where the last equality uses $\delta_{kl} = 0$. By the same computation,

$$\frac{\partial G_l}{\partial z_k} = q_l(\delta_{lk} - q_k)\left(1 + \log\frac{q_l}{p_l}\right) = -q_k q_l\left(1 + \log\frac{q_l}{p_l}\right) \tag{53}$$

The two mixed partials coincide if and only if

$$\log\left(\frac{q_k}{p_k}\right) = \log\left(\frac{q_l}{p_l}\right) \longrightarrow \frac{q_k}{p_k} = \frac{q_l}{p_l} \tag{54}$$

which is generically false (in the example, $q_1/p_1 = 1.2$ while $q_3/p_3 = 1.5$).

Thus $\partial G_k/\partial z_l \neq \partial G_l/\partial z_k$ on U, contradicting Clairaut's theorem. Consequently, no differentiable function $\mathcal{L}$ satisfies $G_{\text{m-RKL}} = \nabla_{\boldsymbol{z}}\mathcal{L}$, and in particular m-RKL is not the gradient of any scalar loss — hence not of any f-divergence. □

B.7 Proof of Proposition 2 (m-RKL as a Descent Direction for FKL)

Proof. Let $\mathcal{L}(\boldsymbol{z}) = D_{\mathrm{KL}}\left(\boldsymbol{p} \parallel \boldsymbol{q}(\boldsymbol{z})\right)$. We proceed in five steps.

Step 1: Obtain the Gradient of FKL w.r.t. the logits

According to Eq. (11), we get $\nabla_{z_k}\mathcal{L} = q_k - p_k$, thus $\nabla_{\boldsymbol{z}}\mathcal{L} = \boldsymbol{q} - \boldsymbol{p}$.

Step 2: Decompose the Inner Product

Substituting the m-RKL field, *i.e.*, Eq. (17), we obtain

$$\left\langle G_{\text{m-RKL}}, D_{\mathrm{KL}}\left(\boldsymbol{p} \parallel \boldsymbol{q}\right)\right\rangle = \sum_{k\in\mathcal{V}^+} q_k \log\left(\frac{q_k}{p_k}\right)(q_k - p_k) \;+\; \alpha \sum_{k\in\mathcal{V}^-} q_k \log\left(\frac{q_k}{p_k}\right)(q_k - p_k) \tag{55}$$

Step 3: Termwise Sign

For $k \in \mathcal{V}^+$ we have $q_k \geq p_k$, so $\log(q_k/p_k) \geq 0$ and $q_k - p_k \geq 0$; their product is non-negative.

For $k \in \mathcal{V}^-$ we have $q_k < p_k$, so $\log(q_k/p_k) < 0$ and $q_k - p_k < 0$; their product is strictly positive, and multiplying by $\alpha > 0$ keeps it strictly positive.

Hence every term is non-negative, and every term in the $\mathcal{V}^-$ sum is strictly positive.

Step 4: Strict Positivity

If $\boldsymbol{q} \neq \boldsymbol{p}$, and since $\sum_k q_k = \sum_k p_k = 1$, there must exist a token with $q_k < p_k$ (otherwise $q_k \geq p_k$ for every k would force $\boldsymbol{q} = \boldsymbol{p}$). That token lies in $\mathcal{V}^-$ and contributes a strictly positive term, so

$$\left\langle G_{\text{m-RKL}}, \boldsymbol{q} - \boldsymbol{p}\right\rangle > 0 \tag{56}$$

Step 5: Chain Rule

Along the flow $\dot{\boldsymbol{z}} = -G_{\text{m-RKL}}$,

$$\frac{d\mathcal{L}}{ds} = \left\langle \nabla_{\boldsymbol{z}}\mathcal{L}, \dot{\boldsymbol{z}}\right\rangle = \left\langle \boldsymbol{q} - \boldsymbol{p}, -G_{\text{m-RKL}}\right\rangle = -\left\langle G_{\text{m-RKL}}, \boldsymbol{q} - \boldsymbol{p}\right\rangle < 0 \tag{57}$$

Thus $D_{\mathrm{KL}}\left(\boldsymbol{p} \parallel \boldsymbol{q}\right)$ is strictly decreasing at every configuration with $\boldsymbol{q} \neq \boldsymbol{p}$, and stationary only in the limit $\boldsymbol{q} \to \boldsymbol{p}$.

The argument uses only the sign structure of Eq. (17): no differentiability of a potential, no convexity, and no small-step assumption. Because $G_{\text{m-RKL}}$ is a strict descent direction for $\mathcal{L}$ in the sense of Step 5, the discrete update $\boldsymbol{z} \leftarrow \boldsymbol{z} - \eta\, G_{\text{m-RKL}}$ strictly decreases $D_{\mathrm{KL}}\left(\boldsymbol{p} \parallel \boldsymbol{q}\right)$ for any sufficiently small step $\eta > 0$. □

C PyTorch Implementation

The custom `autograd` operator is shown below. It assumes `stu_logits` and `tea_logits` have been obtained from a forward pass and have shape `(batch_size, seq_len, vocab_size)`.

```python
import torch
import torch.nn.functional as F
from torch.autograd import Function

class CorrectedRKLLoss(Function):
    """
    Overrides the gradient by implementing a custom backward function.
    """

    @staticmethod
    def forward(ctx, stu_logits, tea_logits):
        # Stable log-probability computation
        stu_logprobs = F.log_softmax(stu_logits, dim=-1)
        tea_logprobs = F.log_softmax(tea_logits, dim=-1)

        # log(q / p)
        log_ratio = stu_logprobs - tea_logprobs

        # Standard reverse KL computation
        rkl = F.kl_div(tea_logprobs, stu_logprobs, reduction="none", log_target=True).sum(dim=-1, keepdim=True)

        # Standard gradient of RKL w.r.t. logit
        # grad_rkl = stu_logprobs.exp() * (log_ratio - rkl)
        # Monotonic gradient by removing global divergence term
        grad_new = stu_logprobs.exp() * log_ratio

        # --- Zero-sum projection ---
        P = F.relu(grad_new).sum(dim=-1, keepdim=True)
        N = F.relu(-grad_new).sum(dim=-1, keepdim=True)
        scale_coeff = P / (N + 1e-10)
        G = torch.where(grad_new > 0, grad_new, scale_coeff * grad_new)

        # Save for backward
        ctx.save_for_backward(G)

        # Return per-position RKL loss for logging
        # (Any tensor works here cause we override the gradient entirely)
        return rkl.squeeze(-1)

    @staticmethod
    def backward(ctx, grad_output):
        G, = ctx.saved_tensors
        while grad_output.dim() < G.dim():
            grad_output = grad_output.unsqueeze(-1)

        return grad_output * G, None
```

D CASE STUDY

Input Prompt

On a "prime date", both the month and the day are prime numbers. For example, Feb. 7 or 2/7 is a prime date. How many prime dates occurred in 2007?

Reasoning trace generated on-policy from Qwen3-1.7B trained with m-RKL (Ours)

We are asked to find how many prime dates occurred in **2007**.
A **prime date** is when **both the month and the day are prime numbers**.
Let's proceed step by step.
Step 1: Identify prime numbers $\leq$ 12. Months range from 1 to 12, and we want to see which are prime. Prime numbers ≤ 12: 1 not prime, 2 prime, 3 prime...
Step 2: List prime days (1 to 31) Prime numbers: $2, 3, 5, 7, 11, 13, 17, 19, 23, 29, 31$ ...
Prime Months: 2, 3, 5, 7, 11 ... We will go month by month.
...
Summary of valid prime dates:

Month	Prime days (valid in that month)
2 (Feb)	$2, 3, 5, 7, 11, 13, 17, 19, 23 \rightarrow 9$
3 (Mar)	$2, 3, 5, 7, 11, 13, 17, 19, 23, 29, 31 \rightarrow 11$
5 (May)	$2, 3, 5, 7, 11, 13, 17, 19, 23, 29, 31 \rightarrow 11$
7 (Jul)	$2, 3, 5, 7, 11, 13, 17, 19, 23, 29, 31 \rightarrow 11$
11 (Nov)	$2, 3, 5, 7, 11, 13, 17, 19, 23, 29 \rightarrow 10$

...
Let's count: Total $= 9 + 11 + 11 + 11 + 10 = \mathbf{52}$. So total prime dates in 2007 = **52**
Final Answer: $= \boxed{52}$

Reasoning trace generated on-policy from Qwen3-1.7B trained with RKL

To find how many **prime dates** occurred in **2007**, we need to identify **all days** where both the **month** and the **day** are **prime numbers**.
Step 1: List the prime months (1 to 12) Prime numbers between 1 and 12: - $2, 3, 5, 7, 11$ ...
Step 2: List the prime days (1 to 31) We will consider all days from 1 to 31, but **only those where the day is prime**. ...
Step 3: For each prime month, count the number of prime days in that month.
We check for each month (2, 3, 5, 7, 11) how many days between 1 and 31 are prime. Let's go through each:
...
Month = 3 (March). ... **So 2,3,5,7,11,13,17,19,23,29 $\rightarrow$ 10 days.**
10 prime dates in March.
...
Step 4: Add them up
- February: 9
- March: **10**
- May: 11
- July: 11
- November: 10

Total = 9+**10**+11+11+10=**51**
Final Answer:

$\boxed{51}$

Reasoning trace generated on-policy from Qwen3-1.7B trained with FKL

We are to find the number of **prime dates** in the year **2007**, where both the **day** and the **month** are **prime numbers**.
Step 1: Understand the format. A date like February 7, is a prime date only if both **2** and **7**...
Prime numbers:
2, 3, 5, 7, 11, 13, 17, 19, 23, 29
...
2. March Months with 31 days, all days are valid as $31 > 29$ ... **So: 10 days**
3. May Same logic as March. Valid days: all prime numbers ≤ 31 ... $\rightarrow$ **10 days**
4. July Same as March and May $\rightarrow$ 31 days $\rightarrow$ all primes ¡ 31: ... $\rightarrow$ **10 days**
5. November (30 days) Wait: November has 30 days. ... **29 ¿ 30** ... $\rightarrow$ **9 days**
Total Prime Dates in 2007:
- February: 9
- March: **10**
- May: **10**
- July: **10**
- November: **9**

Sum:

$$9 + \mathbf{10} + \mathbf{10} + \mathbf{10} + \mathbf{9} = \mathbf{48}$$

...
Final Answer:

$$\boxed{\mathbf{48}}$$

Reasoning trace generated on-policy from Qwen3-1.7B trained with JS

We are asked to count **how many prime dates occurred in 2007**, where a **prime date** means that both the month and the day are **prime numbers**.
Step 1: List of prime months (1 to 12) We use the months from **1 to 12**. ...
Step 2: For each prime month, list the prime days (1 to 31), but considering that days must be valid and not exceed 31.
...
Month = 2 (February). ...
Valid prime days in February: 2, 3, 5, 7, 11, 13, 17, 19, 23, 29.
That's 10 valid days.
Month = 3 (March). ...
Number of primes $\leq$ 31:
– 2, 3, 5, 7, 11, 13, 17, 19, 23,29 $\rightarrow$ 10 numbers ...
Month = 5 (May). Prime days: same list as before: $2, 3, 5, 7, 11, 13, 17, 19, 23, 29$... **10 primes.**
Month = 7 (July). Same primes: $2, 3, 5, 7, 11, 13, 17, 19, 23, 29$ ... $\rightarrow$ **10 days**
...
Total Prime Dates in 2007:
- February: **10**
- March: **10**
- May: **10**
- July: **10**
- November: 10

$$\text{Total} = \mathbf{10} \times 5 = \mathbf{50}$$

Final Answer:

$$\boxed{\mathbf{50}}$$