DFPO: ENHANCING ROBUSTNESS AND GENERALIZATION FOR ROBUST RL THROUGH DISTRIBUTIONAL VALUE FLOW MODELING IN LLM POST-TRAINING

Anonymous authors
Paper under double-blind review

ABSTRACT

Training reinforcement learning (RL) systems in real-world environments remains challenging due to noisy supervision and poor out-of-domain (OOD) generalization, especially in LLM post-training. Recent distributional RL methods improve robustness by modeling values with multiple quantile points, but they still learn each quantile independently as a scalar. This results in rough-grained value representations that lack fine-grained conditioning on state information, struggling under complex and OOD conditions. We propose **DFPO** (Distributional Value Flow Policy Optimization), a robust distributional RL framework that models values as continuous flows across time steps. By scaling value modeling through learning of a value flow field instead of isolated quantile predictions, DFPO captures richer state information for more accurate advantage estimation. To stabilize training under noisy feedback, DFPO further integrates conditional risk control and consistency constraints along value flow trajectories. Experiments on dialogue, math reasoning, and scientific tasks show that DFPO outperforms PPO, FlowRL, and other robust baselines under noisy supervision, achieving improved training stability and generalization.

1 INTRODUCTION

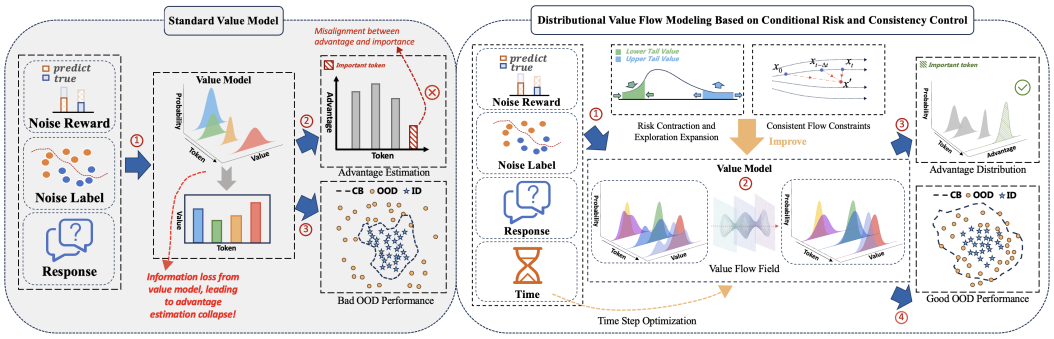

Figure 1: Comparison between the standard scalar value model and DFPO with distributional value flow modeling. The standard value model is sensitive to noisy and biased reward signals, which often leads to unstable value estimation and unreliable advantage learning. In contrast, DFPO models a token-level distributional value flow field across time steps, thereby capturing state information more effectively under noisy supervision. By integrating conditional risk control and consistency constraints along value flow trajectories, DFPO suppresses spurious value fluctuations while preserving high-value exploration. CB denotes the capacity boundary of the model.

Reinforcement Learning has achieved remarkable success across diverse application domains Xi et al. (2025b; 2024b;a); Ding et al. (2025); Caregnato-Neto et al. (2024). However, real-world RL training remains challenging due to severe instability and poor cross-domain generalization, primarily

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.

caused by pervasive noise and incomplete supervision Geng et al. (2024); Parać et al. (2024); Zhu et al. (2025a). This issue is particularly pronounced in LLM post-training and agent-based RL, where learning relies on noisy, high-variance, semantically complex feedback for human value alignment or self-improvement.

To address these challenges, prior work has explored advantage control-based RL methods Wang et al. (2025). Clipping-based approaches Xi et al. (2025a); Liu et al. (2025) stabilize policy updates by reshaping or truncating noisy advantage estimates, while methods like KTAE Sun et al. (2025) reweight signals via semantic key words to suppress spurious noise. Despite effective variance reduction, these methods focus on stabilizing advantage fluctuations rather than explicitly guiding OOD learning, limiting their generalization in real-world OOD scenarios.

Recent studies adopt distributional value models Zhu et al. (2025b) and some distributionally robust Bellman methods Ma et al. (2021); Hu et al. (2025) to enhance robustness under noise and OOD conditions. By modeling value supervision across multiple dimensions, these methods capture more comprehensive state and uncertainty information, improving advantage estimation in complex environments. However, they typically rely on discrete scalar quantile learning for each value dimension, leading to rough-grained state information conditioning and impeding the generalization of value predictions in complex, dynamic OOD scenarios.

In this work, we introduce flow-based modeling into robust distributional RL. Instead of relying on isolated quantile points, we further scale value modeling by representing the value distribution for each state as a continuous generative flow process defined over a virtual time horizon. Learning value distributions as a continuous flow field enables more expressive semantic representations and finer-grained uncertainty modeling. We impose principled constraints inspired by conditional risk and consistency theories on this generative flow field, which encourages near-optimal transport behavior under noisy and OOD conditions and enables accurate advantage estimation.

We propose **DFPO** (Distributional Value Flow Policy Optimization with Conditional Risk and Consistency Control), a robust RL framework integrating continuous value flow modeling with risk and consistency control. DFPO further scales value modeling by representing token-level values as continuous distributions via flow dynamics. This approach encourages high-value exploration while suppressing lower-tail risk, and enforces consistency constraints along the virtual flow paths to stabilize the generative process and guide coherent predictions.

Evaluations on multi-turn dialogue, math reasoning, and scientific QA tasks show that DFPO consistently outperforms standard PPO, GRPO, distributionally robust Bellman methods, and FlowRL under noisy and OOD settings. By combining continuous flow supervision with risk and consistency control, DFPO achieves superior training stability and generalization, providing a scalable solution for real-world robust RL. Our main contributions are summarized as follows:

- We propose a distributional value flow framework that scales value modeling by replacing discrete quantile learning with continuous generative flow modeling over a virtual horizon.
- We introduce a conditional risk and consistency flow control mechanism, jointly optimizing bounded risk distributions and trajectory coherence to balance exploration and constraint and improve training stability.
- Extensive experiments demonstrate DFPO's superiority under noisy and OOD conditions, offering a scalable solution for real-world robust RL.

2 Related Work

Distributional Reinforcement Learning Distributional Reinforcement Learning (DRL) extends traditional reinforcement learning by modeling the full probability distribution of future returns rather than just their expectation. It provides a richer signal for capturing higher-order statistics, including variance, skewness, and multimodality, to enhance decision stability and robustness. Early works like C51, QR-DQN, and IQN Bellemare et al. (2017); Dabney et al. (2017; 2018) attempted to model such full distributions. Recent RL from Human Feedback (RLHF) advances adopt distributional ideas. Q# Zhou et al. (2025) applies DRL to LLM post-training, computing a provably optimal KL-regularized Q-function via the reference policy's cumulative return distribution. Current distribution constraint methods use distribution distances and robust Bellman-based pessimistic optimization. Kan et al.

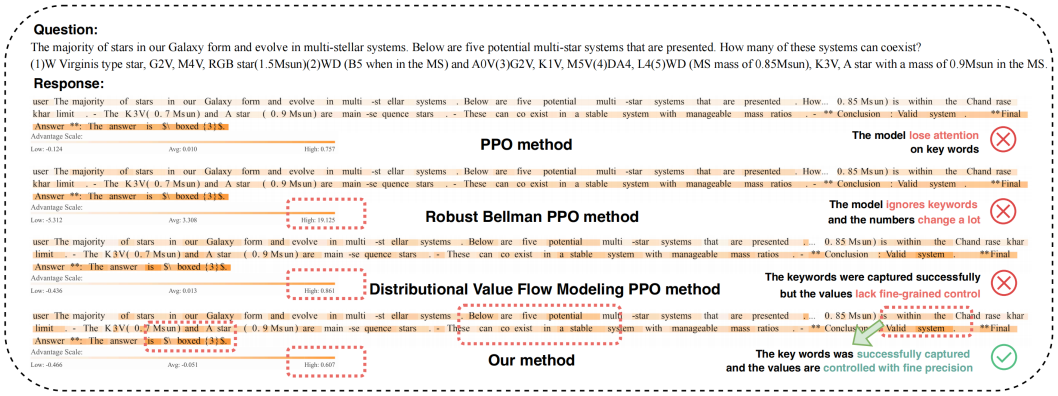

Figure 2: Token-level advantage estimation for the same response across different methods. Our method demonstrates better alignment between high advantage scores and key words.

controls multi-distribution divergence via Sinkhorn Divergence and gradientsSun et al. (2024); VDRL Hu et al. (2025) mitigates Q-value overestimation by discarding lowest multi-Q estimates; CODAC Ma et al. (2021) penalizes quantiles to learn point-wise lower bounds for conservatism. Building on these, this work introduces distributional supervision and conditional value risk constraints into value modeling to improve stability and generalization under noisy supervision.

Flow Reinforcement Learning Flow Reinforcement Learning extends traditional continuous control and value estimation by leveraging flow-based generative models to model complex, multi-modal distributions beyond Gaussian approximations. It offers superior expressivity for capturing high-dimensional action policies and return distributions while mitigating the sampling overhead of diffusion models Ghugare & Eysenbach (2025). In policy optimization, FPO McAllister et al. (2025) integrates conditional flow matching (CFM) into gradient frameworks, while ReinFlow Zhang et al. (2026) and Riemannian Flow Matching Braun et al. (2024) add noise injection and geometric constraints for robotic exploration. Unlike these CFM-based policies, FlowRL Zhu et al. (2025d) matches reward distributions at the trajectory level via GFlowNet trajectory balance with a learnable partition function. For value estimation, recent works focus on distributional fidelity and scalability. Value Flows Dong et al. (2025) models return distributions via density paths and flow derivative ODEs for uncertainty quantification; floq Agrawalla et al. (2025) parameterizes Q-functions as velocity fields for test-time scaling. To ensure robustness in offline and embodied settings, Akimov et al. Akimov et al. (2023) enforces latent-space conservatism; FM-IRL Wan et al. (2025) unifies reward modeling and policy regularization. Building on these, DFPO integrates flow learning into value distribution modeling, enabling fine-grained generalization and robustness control via state guidance.

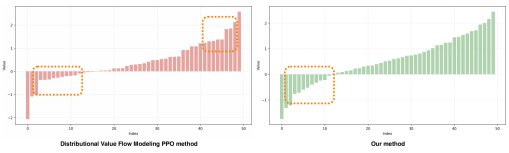

Figure 3: **Comparison of output value distributions.** (Left) Standard distributional PPO shows sharp, unstable lower-tail expansion. (Right) DFPO constrains lower-tail risk while promoting upper-tail exploration.

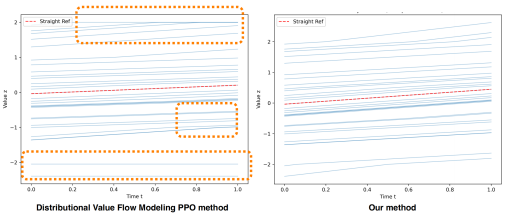

Figure 4: **Comparison of value flow trajectories.** (Left) Standard PPO generates intertwined, chaotic trajectories. (Right) DFPO produces smooth, coherent paths for accurate advantage estimation.

3 METHOD

3.1 MOTIVATION

To enhance the generalization and robustness of RL value models in complex real-world scenarios, prior work Ma et al. (2021); Hu et al. (2025) models values via dense distributions with multiple quantile points. However, existing distributional methods typically learn each quantile independently as a scalar. This results in rough-grained representations that hinder the generalization of value predictions in dynamic environments. To address this limitation, we further scale value modeling. Inspired by VRPO Zhu et al. (2025a), which filters value noise through a normal-signal information bottleneck to enhance robustness and generalization, we aim to transform the raw scalar value into a quasi-normal distributional signal. Flow matching excels at mapping such distributions while providing fine-grained distribution control. We therefore introduce flow-based value modeling, which models continuous value flows across time steps for each state; mapping the signal to a quasi-normal distribution balances the signal profile and guides generalized learning beyond discrete quantiles.

To further scale value modeling and model the value flow field across time, standard flow matching methods Liu et al. (2022) typically sample virtual time steps at random during training to enable learning over entire trajectories. However, this random sampling introduces significant noise into the learned flows Zhang et al. (2026); Braun et al. (2024). As a result, flow directions often become unstable, leading to large variations in the generative paths and major instability in the resulting value distributions. As shown in Figure 4, a PPO-based baseline with value distribution modeling and unconstrained flow learning produces heavily overlapping and intertwined trajectories during generation. Furthermore, Figure 3 reveals sharp, unstable lower-tail growth in the value distributions, reflecting excessive risk accumulation and unreliable advantage estimates.

Inspired by consistency-based learning Lu & Song (2025), we stabilize generation via flow anchoring with directional consistency constraints and explicit risk-aware distribution modeling while retaining high-value exploration. Our DFPO sculpts an asymmetric quasi-normal value signal balancing robustness and generalization. In particular, we explore what kind of signal best balances training stability and generalization, and find that flow-guided signals achieve this balance, enabling reliable advantage estimation under noisy settings.

3.2 AUTO-REGRESSIVE FLOW-BASED VALUE MODELING

To capture the complex, multi-modal distribution of future returns, we construct a distributional value model V_θ on top of a pre-trained auto-regressive transformer backbone. Let $\mathbf{s} = (w_1, \ldots, w_L)$ denote an input sequence of length L, and $\mathbf{h_s} \in \mathbb{R}^d$ denote the corresponding hidden state representation generated by the transformer. Unlike traditional approaches that predict a scalar value or a static categorical distribution, we introduce a Value Flow Head, parameterized as a continuous-time neural Ordinary Differential Equation (ODE) vector field v_θ. This vector field defines the evolution of the value distribution's path over a virtual time horizon $t \in [0, 1]$.

The architecture processes inputs sequentially. First, the time step t is mapped to a high-dimensional embedding using sinusoidal positional encodings, followed by a Multi-Layer Perceptron (MLP) with Mish activations, denoted as $\phi(t)$. This embedding is then conditioned on the hidden state h_s and the noisy value input z via concatenation. Finally, to ensure the Lipschitz continuity of the learned vector field and stabilize the flow dynamics, we apply spectral normalization to the linear layers within the value head. Formally, the output of the value head approximates the velocity of the probability path: $v_\theta(z_t, t, \mathbf{h_s}) \approx \frac{dz_t}{dt}$ During inference, the return distribution is approximated by sampling an initial noise batch $\{z_0^{(k)}\}_{k=1}^K \sim \mathcal{N}(0, 1)$ and solving the initial value problem (IVP) from $t = 0$ to $t = 1$ using an Euler solver Liu et al. (2022). The terminal particles $\{z_1^{(k)}\}_{k=1}^K$ represent the empirical quantiles of the predicted return distribution.

3.3 DISTRIBUTIONAL GENERALIZED ADVANTAGE ESTIMATION

To effectively train the distributional critic, we extend the Generalized Advantage Estimation (GAE) to the distributional setting. Traditional GAE operates on scalar expectations, which loses information

regarding the aleatoric uncertainty of the environment. We introduce a Distributional GAE Processor that operates directly in the quantile space.

Let $Z(s)$ denote the random variable representing the return at state s. We represent distributions using a fixed set of quantile positions $\tau = \{\tau_1, \ldots, \tau_K\}$. The distributional temporal difference (TD) error at time step t, denoted as $\delta_{D,t}$, is defined as a distribution $\delta_{D,t} := R_t + \gamma Z(s_{t+1}) - Z(s_t)$, where R_t is the scalar reward (broadcasted to the distribution dimension), $\gamma \in [0, 1]$ is the discount factor, and arithmetic operations are performed element-wise on the sorted quantile supports.

The distributional advantage $A_D(s_t)$ is computed recursively as $A_D(s_t) = \delta_{D,t} + (\gamma\lambda)A_D(s_{t+1})$, where $\lambda \in [0, 1]$ serves as a distributional variance smoothing parameter, and $\gamma\lambda$ scales the distribution spread around its mean. This mechanism propagates the full distributional shape of value estimates along trajectories, providing fine-grained supervision for risk-sensitive optimization. All operations run per-quantile on sorted supports, with quantiles coupled across timesteps by rank order. With K quantile particles, this recursion defines an empirical surrogate target for the distributional Bellman equation; its contraction behavior under the 1-Wasserstein metric is analyzed in Appendix B.

3.4 Robust Training via Risk-Sensitive Flow Matching

We train the value model using a composite objective function that combines Uncertainty-Weighted Distributional Conditional Flow Matching (UDCFM) for generative modeling with geometric and risk-based constraints for robustness.

3.4.1 Robust Distributional Flow Learning

To effectively model complex return distributions in high-dimensional environments, we propose a robust training paradigm that anchors the flow field to stable estimates and enforces geometric consistency.

Uncertainty-Weighted Distributional CFM (UDCFM) The primary objective regresses the neural vector field v_θ to a target probability path. Given a source distribution $p_0 = \mathcal{N}(0, 1)$ and a target return sample x_1 derived from the GAE buffer, we define the optimal transport path as $z_t = t \cdot x_1 + (1 - t) \cdot x_0$. The target velocity is $u_t(z_t|x_1, x_0) = x_1 - x_0$. To mitigate the impact of noisy supervision, we introduce a state-dependent sensitivity weight $w_{\text{sens}}(\mathbf{s})$ that focuses learning capacity on complex states, where the flow mapping is highly sensitive to input perturbations. This complexity is estimated from the Jacobian evolution $\|J(1)\|^2$ via the sensitivity ODE. The weight is derived as $w_{\text{sens}}(\mathbf{s}) = \sigma(\|J(1)\|^2/\tau_{\text{temp}}) + 0.5$, where σ is the sigmoid function and τ_{temp} is a temperature scaling factor (detailed proof in Appendix B.5). The weighted UDCFM objective is:

$$\mathcal{L}_{\text{UDCFM}}(\theta) = \mathbb{E}_{t\sim\mathcal{U}[0,1],\, x_0,\, x_1,\, \mathbf{s}}[w_{\text{sens}}(\mathbf{s}) \cdot \left\|v_\theta(z_t, t, \mathbf{h_s}) - (x_1 - x_0)\right\|^2] \tag{1}$$

Bootstrapped Anchor Regularization (BCFM) In the early stages of training, the high variance of the stochastic target x_1 can destabilize the learning of the vector field. To mitigate this, we implement a Bootstrapped CFM term using an "anchor" target $x_1^{\text{anc}} = \text{StopGrad}(\mathbb{E}_\theta[x_1|\mathbf{s}])$, which represents the current stable estimate of the expected return. Here, the operator StopGrad$(\cdot)$ denotes the stop-gradient operation (or detach), which blocks gradient backpropagation through the target term. This ensures that x_1^{anc} serves as a fixed regression target during the update step, preventing the optimization from collapsing into trivial solutions where the target shifts to match the prediction. A secondary flow loss minimizes the deviation towards this anchor:

$$\mathcal{L}_{\text{BCFM}}(\theta) = \mathbb{E}_{t\sim\mathcal{U}[0,1],x_0,\mathbf{s}}[w_{\text{sens}}(\mathbf{s}) \cdot \|v_\theta(z_t^{\text{anc}}, t, \mathbf{h_s}) - (x_1^{\text{anc}} - x_0)\|^2] \tag{2}$$

where z_t^{anc} is the interpolated state along the anchor path.

Geometric Consistency Regularization To ensure the learned vector field constitutes a mathematically consistent ODE trajectory, we enforce a geometric consistency constraint. By minimizing the consistency loss, we implicitly encourage the flow trajectories to approximate Optimal Transport paths. This geometric rectification is crucial for enabling accurate single-step inference during the policy optimization phase. Under the linear flow assumption, the terminal state x_1 can be projected from any intermediate state z_t via $\hat{x}_1(z_t, v_t) = z_t + (1 - t) \cdot v_t$. We employ Symmetric Time

Sampling: for $t \sim \mathcal{U}[0,1]$ and its conjugate $t' = 1 - t$, we enforce that projections yield an identical terminal value:

$$\mathcal{L}_{\text{cons}}(\theta) = \|\hat{x}_1(z_t, v_\theta(z_t)) - \hat{x}_1(z_{t'}, v_\theta(z_{t'}))\|^2 \tag{3}$$

3.4.2 Risk-Controlled Distributional Constraints

To enhance policy robustness against OOD states, we impose explicit constraints on the shape of the learned distribution. Let $\{\hat{z}_{(k)}\}_{k=1}^K$ be the sorted predicted quantiles and $\{z_{(k)}^{\text{tgt}}\}_{k=1}^K$ be the sorted target quantiles (target returns).

Conditional Value Risk Optimization We enforce a risk-averse constraint on the left tail and a gain constraint on the right tail. For a risk level $\alpha \in (0,1)$ and gain level $\beta \in (0,1)$, we define the index cutoffs $k_\alpha = \lfloor \alpha K \rfloor$ and $k_\beta = \lfloor (1-\beta)K \rfloor$. The risk loss $\mathcal{L}_{\text{Risk}}$ is defined as:

$$\mathcal{L}_{\text{Risk}} = \left\| \frac{1}{k_\alpha} \sum_{k=1}^{k_\alpha} (\hat{z}_{(k)} - z_{(k)}^{\text{tgt}}) \right\|^2 + \left\| \frac{1}{K - k_\beta + 1} \sum_{k=k_\beta}^{K} (\hat{z}_{(k)} - z_{(k)}^{\text{tgt}}) \right\|^2 \tag{4}$$

Tail Curvature and Shape Regularization To balance robustness and generalization, we penalize the second-order finite differences $\nabla^2 \hat{z}_{(k)} = \hat{z}_{(k+2)} - 2\hat{z}_{(k+1)} + \hat{z}_{(k)}$ in the tail regions $\mathcal{I}_L$ (left) and $\mathcal{I}_R$ (right). We define the tail index sets as $\mathcal{I}_L = \{1, \ldots, k_\alpha - 2\}$ for the left tail and $\mathcal{I}_R = \{k_\beta, \ldots, K-2\}$ for the right tail. We enforce concavity in the left tail and convexity in the right tail using the ReLU operator $[\cdot]_+$:

$$\mathcal{L}_{\text{Shape}} = \frac{1}{|\mathcal{I}_L|} \sum_{k \in \mathcal{I}_L} \left[\nabla^2 \hat{z}_{(k)}\right]_+ + \frac{1}{|\mathcal{I}_R|} \sum_{k \in \mathcal{I}_R} \left[-\nabla^2 \hat{z}_{(k)}\right]_+ \tag{5}$$

3.4.3 Total Objective

The final objective function for the value model combines the generative flow modeling with distributional robustness constraints:

$$\mathcal{L}_{\text{Total}} = \mathcal{L}_{\text{UDCFM}} + \lambda_{\text{reg}} \mathcal{L}_{\text{BCFM}} + \lambda_{\text{cons}} \mathcal{L}_{\text{cons}} + \lambda_{\text{risk}} \mathcal{L}_{\text{Risk}} + \lambda_{\text{shape}} \mathcal{L}_{\text{Shape}} \tag{6}$$

where $\lambda_{\text{reg}}, \lambda_{\text{cons}}, \lambda_{\text{risk}}, \lambda_{\text{shape}}$ are hyperparameters balancing the regularization terms.

4 Experiments

4.1 Experimental Setup

We evaluate DFPO on dialogue, math reasoning and scientific QA under both model-based and rule-based reward supervision. All experiments adopt a unified training pipeline for fair comparison.

Tasks and Datasets To evaluate DFPO under realistic model-based reward supervision, we conduct multi-turn dialogue experiments on the **Honor-Dialogue Dataset**, a task-oriented, multi-domain corpus. The model is trained to act as an assistant producing helpful and task-completing responses under noisy reward feedback. To assess robustness under rule-based supervision, we adopt the Light-R1 dataset Wen et al. (2025) for math training and SuperGPQA Team et al. (2025) for scientific training. We evaluate generalization across a diverse set of benchmarks, including math datasets (MATH500 Hendrycks et al. (2021), AIME24, Minerva-Math Lewkowycz et al. (2022), AMC23) and science datasets (SimpleQA Wei et al. (2024), GPQA, and HLE). Across all benchmarks, we convert all questions from multiple-choice format to open-ended form with no answer options provided, and correctness is judged by generated-text matching; this intentionally removes the guessing advantage of the original multiple-choice settings, so the absolute accuracies in Tables 2, 3, and 4 are naturally lower, and only relative comparisons across methods should be drawn.

Baselines and Initialization DFPO and all baselines are initialized from Qwen3-8B Yang et al. (2025). For dialogue tasks, both the policy and reward models are further fine-tuned on the Honor-Dialogue dataset. We compare DFPO against standard policy optimization methods (PPO, GRPO, Reinforce++), robust RL baselines (KTAE Sun et al. (2025), BAPO Xi et al. (2025a), λ-GRPO Wang et al. (2025)), flow-based RL (FlowRL Zhu et al. (2025d)), and an adapted distributional robust Bellman PPO baseline.

Table 1: Performance comparison (%) of RL methods on real-world dialogue domains at 1000 training steps. DFPO achieves the highest overall accuracy 87.31%, effectively **preventing the catastrophic forgetting** observed in baselines. Specifically, GRPO and Dr.GRPO suffer severe performance collapse (<50%) late in training, while PPO exhibits significant instability. Furthermore, under the influence of real-world noise, the Robust Bellman PPO, BAPO and FlowRL methods generated outputs with **formatting errors**, making performance evaluation infeasible.

Domain	In-Domain	Out-of-Domain				ALL
Method	Life Services	Transportation & Travel	Healthcare & Wellness	Social & Entertainment	Financial Services	AVG
Baseline	**86.73%**	84.50%	**90.23%**	**87.13%**	82.70%	86.26%
GRPO	29.23%	30.00%	24.90%	25.60%	29.37%	27.82%
PPO	59.20%	64.70%	62.67%	59.70%	61.47%	61.55%
Reinforce++	85.17%	76.00%	75.63%	85.40%	83.53%	81.15%
Dr.GRPO	44.67%	49.87%	50.17%	50.80%	46.60%	48.42%
DFPO (Ours)	85.67%	**90.50%**	89.70%	86.40%	**84.30%**	**87.31%**

Table 2: Accuracy (%) of different methods trained on the **Science Domain**. DFPO achieves the best performance on most benchmarks and consistently leads in ID, OOD, and overall averages, demonstrating strong robustness to noisy supervision and superior generalization under OOD conditions.

Domain	In-Domain (Science & QA)			Out-of-Domain (Math)				ID	OOD	ALL
Method	SimpleQA	GPQA(ALL)	HLE	MATH500	AIME24	Minerva-Math	AMC23	AVG	AVG	AVG
Base	2.89%	3.10%	2.89%	87.40%	41.67%	28.68%	75.83%	2.96%	58.40%	34.64%
GRPO	3.03%	2.98%	3.24%	86.80%	45.00%	26.84%	82.50%	3.08%	60.29%	35.77%
PPO	2.82%	2.17%	3.29%	87.00%	51.67%	29.41%	85.00%	2.76%	63.27%	37.34%
Reinforce++	3.19%	4.35%	3.29%	89.40%	50.00%	30.51%	83.33%	3.61%	63.31%	37.72%
Dr. GRPO	2.50%	3.99%	3.10%	88.60%	48.33%	27.21%	80.00%	3.20%	61.04%	36.25%
KTAE	3.17%	3.99%	2.64%	87.20%	46.67%	30.15%	82.50%	3.27%	61.63%	36.62%
λ-GRPO	2.77%	4.35%	3.57%	89.60%	43.33%	29.04%	82.50%	3.56%	61.12%	36.45%
BAPO	3.10%	3.99%	3.15%	84.40%	**60.00%**	26.84%	82.50%	3.41%	63.44%	37.71%
Robust Bellman	3.14%	4.17%	3.38%	87.40%	45.00%	27.21%	83.33%	3.56%	60.74%	36.23%
FlowRL	**3.26%**	4.17%	3.24%	89.60%	50.00%	31.62%	86.67%	3.56%	64.47%	38.37%
DFPO (Ours)	3.17%	**5.43%**	**4.03%**	**91.80%**	56.67%	**32.35%**	**88.33%**	**4.21%**	**67.29%**	**40.25%**

Training Noise Noise is pervasive in both model-based and rule-based reward supervision. As shown in Figure 5, rule-based rewards derived from pseudo-labels contain frequent mislabels, causing erroneous supervision under noisy scenarios in math and science tasks.

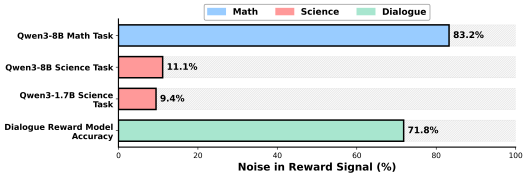

Figure 5: Noise statistics in the various tasks. A significant portion of rewards contains inaccuracies.

Extended details regarding dataset preprocessing, pseudo-label generation, baseline architecture adaptations, and specific training hyperparameters are provided in Appendix C.

4.2 Evaluation Metrics

For math reasoning and scientific QA, we use accuracy as the primary metric and perform iterative generation at 4096-token context length under GPU memory constraints; this setup intentionally skips 5-shot Chain-of-Thought prompting to assess purely intrinsic reasoning capabilities, yielding absolute accuracy lower than official leaderboard values. For multi-turn dialogue, GPT-4o evaluates responses via three metrics measuring task effectiveness and human alignment (TCR, ACR, GCR), whose detailed definitions and human-validation are given in Appendix C.

4.3 Experiment results

DFPO demonstrates strong stability under noisy real-world feedback Tables 1, 2, and 3 summarize the performance of different methods under noisy and imperfect reward supervision

Table 3: Accuracy (%) of different methods trained on the **Math Domain**. DFPO achieves the best or tied-best performance across most benchmarks, and attains the highest ID, OOD, and overall averages, demonstrating superior robustness to noisy supervision and improved cross-domain generalization.

Domain	In-Domain (Math)				Out-of-Domain (Science & QA)			ID	OOD	ALL
Method	MATH500	AIME24	Minerva-Math	AMC23	SimpleQA	GPQA(ALL)	HLE	AVG	AVG	AVG
Base	87.40%	41.67%	28.68%	75.83%	2.89%	3.10%	2.89%	58.40%	2.96%	34.64%
GRPO	89.20%	35.00%	28.68%	84.17%	2.91%	4.17%	3.57%	59.26%	3.55%	35.39%
PPO	86.40%	36.67%	30.51%	80.00%	2.70%	3.99%	3.34%	58.40%	3.34%	34.80%
Reinforce++	90.20%	50.00%	30.88%	84.17%	**3.19%**	**4.53%**	3.71%	63.81%	3.81%	38.10%
Dr. GRPO	90.80%	46.67%	30.88%	**88.33%**	3.12%	4.35%	3.48%	64.17%	3.65%	38.23%
Robust Bellman	89.40%	36.67%	**31.99%**	84.17%	2.61%	3.62%	3.43%	60.56%	3.22%	35.98%
KTAE	86.00%	38.33%	29.41%	78.33%	2.91%	3.99%	3.38%	58.02%	3.43%	34.62%
λ-GRPO	90.00%	50.00%	31.62%	80.00%	2.91%	4.35%	3.71%	62.91%	3.66%	37.51%
BAPO	87.00%	38.33%	30.51%	75.00%	3.12%	4.35%	3.06%	57.71%	3.51%	34.48%
FlowRL	90.20%	**55.00%**	29.41%	86.67%	2.84%	3.80%	3.56%	65.32%	3.40%	38.78%
DFPO (Ours)	**91.00%**	**55.00%**	**31.99%**	85.83%	2.84%	**4.53%**	**4.22%**	**65.96%**	**3.86%**	**39.34%**

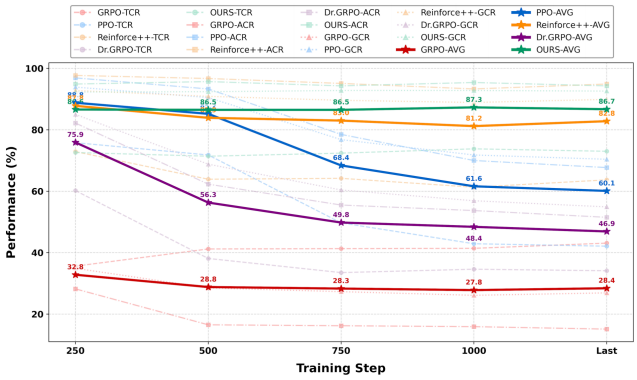

Figure 6: Training performance on dialogue tasks. DFPO demonstrates exceptional stability, maintaining a high average performance of 86.7% throughout the training process.

across multiple domains. On real-world dialogue benchmarks, as shown in Figure 6, DFPO continues to improve model performance throughout training and reaches an average accuracy of 86.65%, without experiencing catastrophic performance collapse. By contrast, GRPO and Dr. GRPO are highly sensitive to reward noise and show substantial performance collapse, with GRPO exhibiting the most severe instability. Moreover, DFPO preserves strong performance on reasoning tasks. After training on Math and Science benchmarks, DFPO achieves overall average accuracies of 39.34% and 40.25%, respectively, outperforming GRPO, Reinforce++, and FlowRL, which highlights its robustness under multi-task and multi-domain noisy training.

DFPO maintains consistent effectiveness across in-domain and cross-domain evaluations. As reported in Tables 1, 2, and 3, DFPO consistently delivers strong in-domain and cross-domain performance. In dialogue settings, DFPO sustains high task completion rates on previously OOD domains and shows stable performance improvements over the baseline, exceeding the results of Reinforce++ and FlowRL. Similarly, when trained on either Math or Science tasks, DFPO consistently achieves the strongest overall performance among all compared methods. In contrast, robust Bellman PPO exhibit stable convergence, but their generalization to OOD domains remains limited, often producing overly long and incoherent responses. FlowRL, while improving exploration through reward distribution matching, provides limited gains in advantage estimation, which constrains its overall improvement.

4.4 Ablation Experiment and Discussion

Effectiveness of loss components for robust and generalizable value flow learning Table 4 presents an ablation study of DFPO's loss components on both ID (Science & QA) and OOD (Math) benchmarks. We observe that adding distributional and flow matching losses yields clear gains, raising overall accuracy from 34.64% to 38.32%, which confirms that modeling values as flow fields provides

Table 4: Ablation of DFPO loss components on scientific tasks. Distributional value modeling with flow matching improves average accuracy from 34.64% to 38.32%, while tail and consistency constraints boost OOD accuracy from 64.16% to 67.12%. The full method achieves the highest overall accuracy 40.25%. DCFM denotes Distributional Conditional Flow Matching loss. Consistency Loss denotes Bootstrapped Anchor & Geometric Consistency Regularization Loss. **The Standard Distributional Value Flow Modeling PPO** results are under the Above+DCFM Loss methods.

Domain	**In-Domain (Science & QA)**			**Out-of-Domain (Math)**				**ID**	**OOD**	**ALL**
Loss Component	**SimpleQA**	**GPQA(ALL)**	**HLE**	**MATH500**	**AIME24**	**Minerva-Math**	**AMC23**	**AVG**	**AVG**	**AVG**
Base	2.89%	3.10%	2.89%	87.40%	41.67%	28.68%	75.83%	2.96%	58.40%	34.64%
Above+Distributional Loss	2.94%	4.17%	3.85%	90.80%	45.00%	30.51%	79.17%	3.65%	61.37%	36.63%
Above+DCFM Loss	3.19%	3.80%	3.80%	89.60%	48.33%	31.99%	87.50%	3.60%	64.36%	38.32%
Above+Conditional Risk Loss	3.10%	4.89%	3.80%	90.40%	51.67%	31.25%	83.33%	3.93%	64.16%	38.35%
Above+Tail Curvature and Shape Loss	2.70%	4.89%	3.66%	89.60%	51.67%	30.88%	**88.33%**	3.75%	65.12%	38.82%
Above+Uncertainty DCFM Loss	3.10%	4.35%	**4.08%**	89.80%	53.33%	31.25%	86.67%	3.84%	65.26%	38.94%
Above+Consistency Loss	3.01%	4.53%	3.24%	90.40%	**58.33%**	**33.09%**	86.67%	3.59%	67.12%	39.90%
Above+Lipschitz Optimization(OURS)	**3.17%**	**5.43%**	4.03%	**91.80%**	56.67%	32.35%	**88.33%**	**4.21%**	**67.29%**	**40.25%**

a stronger signal under noisy supervision. The most significant improvements occur with tail-related and flow consistency constraints. For instance, OOD accuracy increases from 64.16% to 65.12% with tail-related constraints, and further to 67.12% with consistency loss. This indicates that constraining value flow trajectories effectively reduces unnecessary exploration and improves robustness in OOD domains. Our complete model, which integrates these constraints with Lipschitz-based optimization, achieves the highest average accuracy of 40.25%. This suggests that Lipschitz optimization helps consolidate structural integrity.

What Constitutes an Effective Distributional Signal for Robustness and Generalization? Beyond scaling value model capacity via continuous flow fields, a core open question in distributional RL remains: *what geometric properties of a learned return distribution optimally balance noise robustness and cross-domain generalization?*

Conventional distributional RL treats the critic as a passive density estimator of raw empirical returns; under noisy supervision this agnostic fitting either explodes the lower tail from reward noise (Figure 3, Left) or collapses the variance under naive penalties, both symptoms of a signal that fails to converge to a well-posed target. Our analysis recasts these operations as signal-convergence attempts: shaping the two tails is not an ad-hoc regularizer but a mechanism for steering the empirical return signal toward a convergent, structured geometry. An effective distributional signal keeps the central probability mass smooth and unimodal around the expected return, anchoring the expectation and eliminating chaotic gradient fluctuations. Its tail dynamics are asymmetric yet complementary: the lower tail is contracted via strict concavity and conditional risk bounds (Eq. 4, 5), suppressing aleatoric noise and preventing spurious negative spikes from eroding pre-trained capabilities; the upper tail is expanded via convexity regularization, preserving rare high-reward reasoning chains that drive cross-domain generalization rather than penalizing them as variance. Both operations converge the signal from opposite directions, stabilizing noisy mass on the left while preventing informative mass collapse on the right, and as the ablation in Appendix C.6 shows, the gain materializes better when the two are combined asymmetrically.

5 Conclusion

We present DFPO, a robust distributional RL framework that models value as continuous temporal flows. Replacing discrete scalar quantiles with learned value flow fields, it improves robustness and generalization under noisy supervision and scales value modeling for real-world LLM post-training. DFPO integrates conditional risk control and trajectory consistency constraints to balance stable optimization and exploratory efficiency, and shapes the value distribution into a regularized quasi-normal profile with asymmetric tail dynamics, offering a principled paradigm for optimal distributional guidance signals. Experiments on dialogue, math, and scientific tasks show DFPO consistently outperforms PPO, FlowRL and other baselines, providing a scalable, practical solution for real-world RL and LLM post-training.

AI USE STATEMENT

In the preparation of this paper, large language models (LLMs) were used solely as an assistive tool for writing refinement, grammatical checking, and language polishing. The core research ideas, mathematical formulations and theoretical derivations (including the contraction proof of the Distributional GAE operator and consistency regularization), algorithm design (DFPO), experimental execution, and result analysis across multi-turn dialogue, math, and scientific reasoning tasks were entirely conceived and conducted by the human authors. The LLM was employed after the intellectual substance and technical methodologies of the work were fully established, specifically to improve readability, fluency, and overall clarity of expression in certain parts of the manuscript. It did not contribute to the scientific ideation, algorithmic development, or research conclusions. All final content was thoroughly reviewed, verified, and approved by the authors.

ETHICS STATEMENT

This work introduces DFPO, an algorithmic reinforcement learning framework designed to improve the robustness and generalization of LLM post-training under noisy supervision. For mathematical and scientific reasoning, all experiments are conducted on publicly accessible datasets (Light-R1, SuperGPQA, MATH500, AIME24, Minerva-Math, AMC23, SimpleQA, GPQA, and HLE) under standard academic research protocols. For the multi-turn dialogue benchmark, interactions are evaluated using the Honor-Dialogue corpus. All personal identifying details and sensitive user attributes in this dataset have been rigorously masked and anonymized (e.g., masking user identifiers), and the dataset usage protocol has been reviewed and approved by the relevant institutional ethics committee, involving no human subjects or private personal data exposure. Beyond academic evaluation, we recognize that RL post-training for large language models carries broader societal implications: potential safety and hallucination risks if models are over-relied upon in high-stakes domains (such as healthcare and financial services examined in our dialogue tasks), the computational energy footprint of distributed model training across large GPU clusters, and alignment biases under imperfect or corrupted reward feedback. We encourage responsible deployment through rigorous out-of-domain safety verification, verifiable reward engineering, and transparent model evaluation. This work adheres to the ICLR Code of Ethics and aims to foster safer, more reliable, and resilient AI systems through principled distributional value modeling.

REPRODUCIBILITY STATEMENT

To ensure the reproducibility of our theoretical and empirical findings, we provide comprehensive algorithmic specifications, implementation configurations, formal proofs, as well as code and data resources:

1. **Source Code and Data Samples in Supplementary Material:** We provide a Source Code containing the complete implementation of DFPO (including model architectures, training routines, and evaluation pipelines) along with representative anonymized data samples from the benchmark suites in the Supplementary Material. Full code and dataset resources will also be made publicly available upon paper acceptance.
2. **Algorithm and Mathematical Proofs:** The full training procedure of DFPO is detailed in pseudocode format in Appendix A (Algorithm 1). Complete mathematical derivations and convergence proofs—including the contraction mapping of the Distributional GAE operator under the 1-Wasserstein metric, sensitivity ODE analysis, the straight-line regularization induced by geometric consistency, and the sufficiency of the 1-step Euler solver under the straight-trajectory regime—are rigorously provided in Appendix B.
3. **Experimental Setup and Hyperparameters:** The model architecture (e.g., value flow head parameterized by continuous-time neural ODEs), training protocols, and exact hyperparameters (e.g., regularization weights $\lambda_{\text{reg}} = 0.1$, $\lambda_{\text{cons}} = 0.01$, $\lambda_{\text{risk}} = 0.5$, $\lambda_{\text{shape}} = 0.5$, risk quantiles $\alpha = \beta = 0.1$, flow sampling steps $K = 50$, and GPU cluster specifications) are explicitly detailed in Section 3.4 and Appendix C.
4. **Datasets and Evaluation Rubrics:** Preprocessing steps and pseudo-label filtering thresholds (via 5-round majority voting) for the math and science tasks, along with the composition of

the Honor-Dialogue benchmark across 11 service scenarios, are elaborated in Section 3.2 and Appendix C. Additionally, the evaluation rubrics, 1–5 scoring guidelines for GPT-4o, and human-model agreement validations (Table 6) are fully documented in Appendix C. 3 and Figure 9.

REFERENCES

Bhavya Agrawalla, Michal Nauman, Khush Agrawal, and Aviral Kumar. floq: Training critics via flow-matching for scaling compute in value-based rl, 2025. URL https://arxiv.org/abs/2509.06863.

Dmitriy Akimov, Vladislav Kurenkov, Alexander Nikulin, Denis Tarasov, and Sergey Kolesnikov. Let offline rl flow: Training conservative agents in the latent space of normalizing flows, 2023. URL https://arxiv.org/abs/2211.11096.

Marc G. Bellemare, Will Dabney, and Rémi Munos. A distributional perspective on reinforcement learning, 2017. URL https://arxiv.org/abs/1707.06887.

Max Braun, Noémie Jaquier, Leonel Rozo, and Tamim Asfour. Riemannian flow matching policy for robot motion learning, 2024. URL https://arxiv.org/abs/2403.10672.

Angelo Caregnato-Neto, Luciano Cavalcante Siebert, Arkady Zgonnikov, Marcos Ricardo Omena de Albuquerque Maximo, and Rubens Junqueira Magalhães Afonso. ARMCHAIR: integrated inverse reinforcement learning and model predictive control for human-robot collaboration. *arXiv e-prints*, art. arXiv:2402.19128, February 2024. doi: 10.48550/arXiv.2402.19128.

Ricky T. Q. Chen, Yulia Rubanova, Jesse Bettencourt, and David Duvenaud. Neural ordinary differential equations, 2019. URL https://arxiv.org/abs/1806.07366.

Will Dabney, Mark Rowland, Marc G. Bellemare, and Rémi Munos. Distributional reinforcement learning with quantile regression, 2017. URL https://arxiv.org/abs/1710.10044.

Will Dabney, Georg Ostrovski, David Silver, and Rémi Munos. Implicit quantile networks for distributional reinforcement learning, 2018. URL https://arxiv.org/abs/1806.06923.

Yiwen Ding, Zhiheng Xi, Wei He, Zhuoyuan Li, Yitao Zhai, Xiaowei Shi, Xunliang Cai, Tao Gui, Qi Zhang, and Xuanjing Huang. Mitigating tail narrowing in llm self-improvement via socratic-guided sampling, 2025. URL https://arxiv.org/abs/2411.00750.

Perry Dong, Chongyi Zheng, Chelsea Finn, Dorsa Sadigh, and Benjamin Eysenbach. Value flows, 2025. URL https://arxiv.org/abs/2510.07650.

Emilien Dupont, Arnaud Doucet, and Yee Whye Teh. Augmented neural odes, 2019. URL https://arxiv.org/abs/1904.01681.

Wei Geng, Baidi Xiao, Rongpeng Li, Ning Wei, Dong Wang, and Zhifeng Zhao. Noise distribution decomposition based multi-agent distributional reinforcement learning, 2024. URL https://arxiv.org/abs/2312.07025.

Raj Ghugare and Benjamin Eysenbach. Normalizing flows are capable models for rl, 2025. URL https://arxiv.org/abs/2505.23527.

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, 2021. URL https://arxiv.org/abs/2103.03874.

Xiaoliang Hu, Fuyun Wang, Tong Zhang, and Zhen Cui. Value diffusion reinforcement learning. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems*, 2025. URL https://openreview.net/forum?id=thJ6aFoKrh.

Aitor Lewkowycz, Anders Andreassen, David Dohan, Ethan Dyer, Henryk Michalewski, Vinay Ramasesh, Ambrose Slone, Cem Anil, Imanol Schlag, Theo Gutman-Solo, Yuhuai Wu, Behnam Neyshabur, Guy Gur-Ari, and Vedant Misra. Solving quantitative reasoning problems with language models, 2022. URL https://arxiv.org/abs/2206.14858.

Yunfei Li, Xiao Ma, Jiafeng Xu, Yu Cui, Zhongren Cui, Zhigang Han, Liqun Huang, Tao Kong, Yuxiao Liu, Hao Niu, Wanli Peng, Jingchao Qiao, Zeyu Ren, Haixin Shi, Zhi Su, Jiawen Tian, Yuyang Xiao, Shenyu Zhang, Liwei Zheng, Hang Li, and Yonghui Wu. Gr-rl: Going dexterous and precise for long-horizon robotic manipulation, 2025. URL https://arxiv.org/abs/2512.01801.

Yaron Lipman, Ricky T. Q. Chen, Heli Ben-Hamu, Maximilian Nickel, and Matt Le. Flow matching for generative modeling, 2023. URL https://arxiv.org/abs/2210.02747.

Jijia Liu, Feng Gao, Bingwen Wei, Xinlei Chen, Qingmin Liao, Yi Wu, Chao Yu, and Yu Wang. What can rl bring to vla generalization? an empirical study, 2026. URL https://arxiv.org/abs/2505.19789.

Xingchao Liu, Chengyue Gong, and Qiang Liu. Flow straight and fast: Learning to generate and transfer data with rectified flow, 2022. URL https://arxiv.org/abs/2209.03003.

Zongkai Liu, Fanqing Meng, Lingxiao Du, Zhixiang Zhou, Chao Yu, Wenqi Shao, and Qiaosheng Zhang. Cpgd: Toward stable rule-based reinforcement learning for language models, 2025. URL https://arxiv.org/abs/2505.12504.

Cheng Lu and Yang Song. Simplifying, stabilizing and scaling continuous-time consistency models, 2025. URL https://arxiv.org/abs/2410.11081.

Yecheng Jason Ma, Dinesh Jayaraman, and Osbert Bastani. Conservative offline distributional reinforcement learning, 2021. URL https://arxiv.org/abs/2107.06106.

David McAllister, Songwei Ge, Brent Yi, Chung Min Kim, Ethan Weber, Hongsuk Choi, Haiwen Feng, and Angjoo Kanazawa. Flow matching policy gradients, 2025. URL https://arxiv.org/abs/2507.21053.

Roko Parać, Lorenzo Nodari, Leo Ardon, Daniel Furelos-Blanco, Federico Cerutti, and Alessandra Russo. Learning robust reward machines from noisy labels. In *Proceedings of the TwentyFirst International Conference on Principles of Knowledge Representation and Reasoning*, KR-2024, pp. 909–919. International Joint Conferences on Artificial Intelligence Organization, November 2024. doi: 10.24963/kr.2024/85. URL http://dx.doi.org/10.24963/kr.2024/85.

R Tyrrell Rockafellar, Stanislav Uryasev, et al. Optimization of conditional value-at-risk.

Ke Sun, Yingnan Zhao, Wulong Liu, Bei Jiang, and Linglong Kong. Distributional reinforcement learning with regularized wasserstein loss, 2024. URL https://arxiv.org/abs/2202.00769.

Wei Sun, Wen Yang, Pu Jian, Qianlong Du, Fuwei Cui, Shuo Ren, and Jiajun Zhang. Ktae: A model-free algorithm to key-tokens advantage estimation in mathematical reasoning, 2025. URL https://arxiv.org/abs/2505.16826.

M-A-P Team, Xinrun Du, Yifan Yao, Kaijing Ma, Bingli Wang, Tianyu Zheng, Kang Zhu, Minghao Liu, Yiming Liang, Xiaolong Jin, Zhenlin Wei, Chujie Zheng, Kaixing Deng, Shuyue Guo, Shian Jia, Sichao Jiang, Yiyan Liao, Rui Li, Qinrui Li, Sirun Li, Yizhi Li, Yunwen Li, Dehua Ma, Yuansheng Ni, Haoran Que, Qiyao Wang, Zhoufutu Wen, Siwei Wu, Tianshun Xing, Ming Xu, Zhenzhu Yang, Zekun Moore Wang, Junting Zhou, Yuelin Bai, Xingyuan Bu, Chenglin Cai, Liang Chen, Yifan Chen, Chengtuo Cheng, Tianhao Cheng, Keyi Ding, Siming Huang, Yun Huang, Yaoru Li, Yizhe Li, Zhaoqun Li, Tianhao Liang, Chengdong Lin, Hongquan Lin, Yinghao Ma, Zhongyuan Peng, Zifan Peng, Qige Qi, Shi Qiu, Xingwei Qu, Yizhou Tan, Zili Wang, Chenqing Wang, Hao Wang, Yiya Wang, Yubo Wang, Jiajun Xu, Kexin Yang, Ruibin Yuan, Yuanhao Yue, Tianyang Zhan, Chun Zhang, Jingyang Zhang, Xiyue Zhang, Xingjian Zhang, Yue Zhang, Yongchi Zhao, Xiangyu Zheng, Chenghua Zhong, Yang Gao, Zhoujun Li, Dayiheng Liu, Qian Liu, Tianyu Liu, Shiwen Ni, Junran Peng, Yujia Qin, Wenbo Su, Guoyin Wang, Shi Wang, Jian Yang, Min Yang, Meng Cao, Xiang Yue, Zhaoxiang Zhang, Wangchunshu Zhou, Jiaheng Liu, Qunshu Lin, Wenhao Huang, and Ge Zhang. Supergpqa: Scaling llm evaluation across 285 graduate disciplines, 2025. URL https://arxiv.org/abs/2502.14739.

Zhenglin Wan, Jingxuan Wu, Xingrui Yu, Chubin Zhang, Mingcong Lei, Bo An, and Ivor Tsang. Fm-irl: Flow-matching for reward modeling and policy regularization in reinforcement learning, 2025. URL https://arxiv.org/abs/2510.09222.

Yining Wang, Jinman Zhao, Chuangxin Zhao, Shuhao Guan, Gerald Penn, and Shinan Liu. λ-grpo: Unifying the grpo frameworks with learnable token preferences, 2025. URL https://arxiv.org/abs/2510.06870.

Jason Wei, Nguyen Karina, Hyung Won Chung, Yunxin Joy Jiao, Spencer Papay, Amelia Glaese, John Schulman, and William Fedus. Measuring short-form factuality in large language models, 2024. URL https://arxiv.org/abs/2411.04368.

Liang Wen, Yunke Cai, Fenrui Xiao, Xin He, Qi An, Zhenyu Duan, Yimin Du, Junchen Liu, Lifu Tang, Xiaowei Lv, Haosheng Zou, Yongchao Deng, Shousheng Jia, and Xiangzheng Zhang. Light-r1: Curriculum sft, dpo and rl for long cot from scratch and beyond, 2025. URL https://arxiv.org/abs/2503.10460.

Zhiheng Xi, Wenxiang Chen, Boyang Hong, Senjie Jin, Rui Zheng, Wei He, Yiwen Ding, Shichun Liu, Xin Guo, Junzhe Wang, Honglin Guo, Wei Shen, Xiaoran Fan, Yuhao Zhou, Shihan Dou, Xiao Wang, Xinbo Zhang, Peng Sun, Tao Gui, Qi Zhang, and Xuanjing Huang. Training large language models for reasoning through reverse curriculum reinforcement learning, 2024a. URL https://arxiv.org/abs/2402.05808.

Zhiheng Xi, Yiwen Ding, Wenxiang Chen, Boyang Hong, Honglin Guo, Junzhe Wang, Dingwen Yang, Chenyang Liao, Xin Guo, Wei He, Songyang Gao, Lu Chen, Rui Zheng, Yicheng Zou, Tao Gui, Qi Zhang, Xipeng Qiu, Xuanjing Huang, Zuxuan Wu, and Yu-Gang Jiang. Agentgym: Evolving large language model-based agents across diverse environments, 2024b. URL https://arxiv.org/abs/2406.04151.

Zhiheng Xi, Xin Guo, Yang Nan, Enyu Zhou, Junrui Shen, Wenxiang Chen, Jiaqi Liu, Jixuan Huang, Zhihao Zhang, Honglin Guo, Xun Deng, Zhikai Lei, Miao Zheng, Guoteng Wang, Shuo Zhang, Peng Sun, Rui Zheng, Hang Yan, Tao Gui, Qi Zhang, and Xuanjing Huang. Bapo: Stabilizing off-policy reinforcement learning for llms via balanced policy optimization with adaptive clipping, 2025a. URL https://arxiv.org/abs/2510.18927.

Zhiheng Xi, Jixuan Huang, Chenyang Liao, Baodai Huang, Honglin Guo, Jiaqi Liu, Rui Zheng, Junjie Ye, Jiazheng Zhang, Wenxiang Chen, Wei He, Yiwen Ding, Guanyu Li, Zehui Chen, Zhengyin Du, Xuesong Yao, Yufei Xu, Jiecao Chen, Tao Gui, Zuxuan Wu, Qi Zhang, Xuanjing Huang, and Yu-Gang Jiang. Agentgym-rl: Training llm agents for long-horizon decision making through multi-turn reinforcement learning, 2025b. URL https://arxiv.org/abs/2509.08755.

An Yang, Anfeng Li, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Gao, Chengen Huang, Chenxu Lv, Chujie Zheng, Dayiheng Liu, Fan Zhou, Fei Huang, Feng Hu, Hao Ge, Haoran Wei, Huan Lin, Jialong Tang, Jian Yang, Jianhong Tu, Jianwei Zhang, Jianxin Yang, Jiaxi Yang, Jing Zhou, Jingren Zhou, Junyang Lin, Kai Dang, Keqin Bao, Kexin Yang, Le Yu, Lianghao Deng, Mei Li, Mingfeng Xue, Mingze Li, Pei Zhang, Peng Wang, Qin Zhu, Rui Men, Ruize Gao, Shixuan Liu, Shuang Luo, Tianhao Li, Tianyi Tang, Wenbiao Yin, Xingzhang Ren, Xinyu Wang, Xinyu Zhang, Xuancheng Ren, Yang Fan, Yang Su, Yichang Zhang, Yinger Zhang, Yu Wan, Yuqiong Liu, Zekun Wang, Zeyu Cui, Zhenru Zhang, Zhipeng Zhou, and Zihan Qiu. Qwen3 technical report, 2025. URL https://arxiv.org/abs/2505.09388.

Tonghe Zhang, Chao Yu, Sichang Su, and Yu Wang. Reinflow: Fine-tuning flow matching policy with online reinforcement learning, 2026. URL https://arxiv.org/abs/2505.22094.

Jin Peng Zhou, Kaiwen Wang, Jonathan Chang, Zhaolin Gao, Nathan Kallus, Kilian Q. Weinberger, Kianté Brantley, and Wen Sun. $q\sharp$: Provably optimal distributional rl for llm post-training, 2025. URL https://arxiv.org/abs/2502.20548.

Dingwei Zhu, Shihan Dou, Zhiheng Xi, Senjie Jin, Guoqiang Zhang, Jiazheng Zhang, Junjie Ye, Mingxu Chai, Enyu Zhou, Ming Zhang, Caishuang Huang, Yunke Zhang, Yuran Wang, and Tao Gui. Vrpo: Rethinking value modeling for robust rl training under noisy supervision, 2025a. URL https://arxiv.org/abs/2508.03058.

Dingwei Zhu, Zhiheng Xi, Shihan Dou, Yuhui Wang, Sixian Li, Junjie Ye, Honglin Guo, Shichun Liu, Chenhao Huang, Yajie Yang, Junlin Shang, Senjie Jin, Ming Zhang, Jiazheng Zhang, Caishuang Huang, Yunke Zhang, Demei Yan, Yuran Wang, and Tao Gui. Dvpo: Distributional value modeling-based policy optimization for llm post-training, 2025b. URL https://arxiv.org/abs/2512.03847.

Fengqi Zhu, Rongzhen Wang, Shen Nie, Xiaolu Zhang, Chunwei Wu, Jun Hu, Jun Zhou, Jianfei Chen, Yankai Lin, Ji-Rong Wen, and Chongxuan Li. Llada 1.5: Variance-reduced preference optimization for large language diffusion models, 2025c. URL https://arxiv.org/abs/2505.19223.

Xuekai Zhu, Daixuan Cheng, Dinghuai Zhang, Hengli Li, Kaiyan Zhang, Che Jiang, Youbang Sun, Ermo Hua, Yuxin Zuo, Xingtai Lv, Qizheng Zhang, Lin Chen, Fanghao Shao, Bo Xue, Yunchong Song, Zhenjie Yang, Ganqu Cui, Ning Ding, Jianfeng Gao, Xiaodong Liu, Bowen Zhou, Hongyuan Mei, and Zhouhan Lin. Flowrl: Matching reward distributions for llm reasoning, 2025d. URL https://arxiv.org/abs/2509.15207.

A Additional Details for DFPO

A.1 Pseudocode

The full algorithm of DFPO is detailed in Algorithm 1.

Algorithm 1 DFPO Training Algorithm

Require: Dataset $\mathcal{D}$, Policy π_θ, Value Model V_ϕ with Flow Head v_ϕ.
Require: Hyperparameters: $\lambda_{\text{reg}}, \lambda_{\text{cons}}, \lambda_{\text{risk}}, \lambda_{\text{shape}}$, quantile levels α, β.
Ensure: Optimized Policy π_θ and Value Model V_ϕ.

for each iteration **do**
  Sample Trajectories: Generate trajectories $\tau = \{(s_t, a_t, r_t, s_{t+1})\}$ using current policy π_θ.
  Distributional Value Estimation (Inference):
  for each state s_t in trajectories **do**
    Extract hidden state: $\mathbf{h_s} = \text{Transformer}(s_t)$
    Sample initial noise batch: $\{z_0^{(k)}\}_{k=1}^K \sim \mathcal{N}(0,1)$
    Solve ODE IVP (Euler): $\{z_1^{(k)}\} = \text{ODESolve}(v_\phi, z_0^{(k)}, t=0 \to 1, \mathbf{h_s})$
    Obtain predicted value distribution quantiles: $\hat{Z}(s_t) = \text{Sort}(\{z_1^{(k)}\})$
  end for
  Compute Distributional GAE:
  for each timestep t reversed **do**
    Calculate distributional TD error (element-wise on sorted supports):
    $\delta_{D,t} = R_t + \gamma \hat{Z}(s_{t+1}) - \hat{Z}(s_t)$
    Recursive Distributional Advantage:
    $A_D(s_t) = \delta_{D,t} + (\gamma\lambda) A_D(s_{t+1})$
    Compute Target Returns for Value Model: $Z^{\text{tgt}}(s_t) = \hat{Z}(s_t) + A_D(s_t)$
  end for
  Value Model Update:
  for each mini-batch B sampled from buffer **do**
    1. Flow Matching Objectives:
    Sample time $t \sim \mathcal{U}[0,1]$, noise $x_0 \sim \mathcal{N}(0,1)$, target sample $x_1 \sim Z^{\text{tgt}}(s)$
    Compute Sensitivity Weight: $w_{\text{sens}}(\mathbf{s}) = \sigma(\|J(1)\|^2/\tau_{\text{temp}}) + 0.5$
    Construct optimal path: $z_t = tx_1 + (1-t)x_0$
    Anchor target: $x_1^{\text{anc}} = \text{StopGrad}(\mathbb{E}[x_1|\mathbf{s}])$, path $z_t^{\text{anc}} = tx_1^{\text{anc}} + (1-t)x_0$
    $\mathcal{L}_{\text{UDCFM}} = \mathbb{E}_{t,x_0,x_1}[w_{\text{sens}}(\mathbf{s}) \| v_\phi(z_t, t, \mathbf{h_s}) - (x_1 - x_0)\|^2]$
    $\mathcal{L}_{\text{BCFM}} = \mathbb{E}_{t,x_0}[w_{\text{sens}}(\mathbf{s}) \| v_\phi(z_t^{\text{anc}}, t, \mathbf{h_s}) - (x_1^{\text{anc}} - x_0)\|^2]$
    2. Geometric Consistency Regularization:
    Sample $t \sim \mathcal{U}[0,1]$, set conjugate $t' = 1 - t$
    Project terminals: $\hat{x}_1(z_t) = z_t + (1-t)v_\phi(z_t), \quad \hat{x}_1(z_{t'}) = z_{t'} + (1-t')v_\phi(z_{t'})$
    $\mathcal{L}_{\text{cons}} = \|\hat{x}_1(z_t) - \hat{x}_1(z_{t'})\|^2$
    3. Risk-Controlled Distributional Constraints:
    Sort current predictions $\hat{z}$ and targets z^{tgt}
    Calculate Risk Loss (Left/Right Tails):
    $\mathcal{L}_{\text{Risk}} = \|\text{Tail}_L(\hat{z}) - \text{Tail}_L(z^{\text{tgt}})\|^2 + \|\text{Tail}_R(\hat{z}) - \text{Tail}_R(z^{\text{tgt}})\|^2$
    Calculate Shape Loss (Curvature Penalty):
    $\mathcal{L}_{\text{Shape}} = \frac{1}{|\mathcal{I}_L|}\sum_{k\in\mathcal{I}_L}[\nabla^2 \hat{z}_{(k)}]_+ + \frac{1}{|\mathcal{I}_R|}\sum_{k\in\mathcal{I}_R}[-\nabla^2 \hat{z}_{(k)}]_+$
    Total Critic Loss:
    $\mathcal{L}_{\text{Total}} = (\mathcal{L}_{\text{UDCFM}} + \lambda_{\text{reg}}\mathcal{L}_{\text{BCFM}} + \lambda_{\text{cons}}\mathcal{L}_{\text{cons}}) + (\lambda_{\text{risk}}\mathcal{L}_{\text{Risk}} + \lambda_{\text{shape}}\mathcal{L}_{\text{Shape}})$
    Update ϕ using $\nabla_\phi \mathcal{L}_{\text{Total}}$
  end for
  Policy Update:
  Scalarize Advantage for PPO: $\hat{A}_t = \text{Mean}(A_D(s_t))$
  $\mathcal{L}_{\text{PPO}} = \mathbb{E}[\min(\rho_t \hat{A}_t, \text{clip}(\rho_t, 1-\epsilon, 1+\epsilon)\hat{A}_t)]$
  Update θ using $\nabla_\theta \mathcal{L}_{\text{PPO}}$
end for

A.2 Efficient Jacobian Estimation via Sensitivity ODE

To quantify flow uncertainty efficiently, we approximate the sensitivity of the terminal state z_1 with respect to the initial noise z_0 without relying on computationally expensive Monte Carlo sampling. Dong et al. (2025) We employ the adjoint sensitivity method to propagate the local

Jacobian $J(t) := \frac{\partial z_t}{\partial z_0}$ along the flow trajectory. Differentiating the flow equation $\frac{dz_t}{dt} = v_\theta(z_t, t, \mathbf{h_s})$, the evolution of the Jacobian follows the linear ODE:

$$\frac{dJ(t)}{dt} = \frac{\partial v_\theta(z_t, t, \mathbf{h_s})}{\partial z_t} J(t), \quad \text{with } J(0) = 1. \tag{7}$$

In our implementation, we solve this coupled system (z_t, J_t) in a single forward pass using a fixed-step Euler solver. Specifically, we set the number of ODE solver steps to $N = 10$. It is crucial to note that $N = 10$ represents the temporal discretization resolution of the numerical integration, not the number of random samples. At each step t_k, we compute the scalar gradient $\nabla_z v_\theta$ via automatic differentiation and update the Jacobian: $J_{t_{k+1}} \approx J_{t_k} + (\nabla_z v_\theta \cdot J_{t_k})\Delta t$. Since we utilize a 1D value distribution, this process incurs only negligible computational overhead (computing a single scalar gradient per step) compared to the Transformer backbone, while providing a stable deterministic estimate of flow sensitivity.

B Theoretical Analysis

In this section, we provide the theoretical justification for the convergence of DFPO.

B.1 Contraction of the Distributional GAE Operator

Proposition 1. *Let $\mathcal{T}^\pi$ denote the distributional Bellman operator. The Distributional GAE operator $\mathcal{T}^\pi_{\mathrm{GAE}}$, defined as a geometric mixture of k-step distributional Bellman operators, is a contraction mapping under the supremum 1-Wasserstein metric for any $\gamma < 1$. Consequently, the iterative application of $\mathcal{T}^\pi_{\mathrm{GAE}}$ converges to a unique fixed-point return distribution.*

Proof. Let $\mathcal{Z}$ denote the space of return distributions with finite first moments. We equip this space with the supremum 1-Wasserstein metric (also known as the Earth Mover's Distance):

$$\bar{d}_1(Z_1, Z_2) := \sup_{s,a} W_1(Z_1(s,a), Z_2(s,a)). \tag{8}$$

First, consider the standard one-step distributional Bellman operator $\mathcal{T}^\pi$, defined as:

$$\mathcal{T}^\pi Z(s,a) \overset{D}{=} R(s,a) + \gamma Z(s', a'), \qquad a' \sim \pi(\cdot \mid s'). \tag{9}$$

Leveraging the translation invariance ($W_1(X + c, Y + c) = W_1(X, Y)$) and absolute homogeneity ($W_1(cX, cY) = |c| W_1(X, Y)$) of the Wasserstein metric, and assuming a synchronous coupling of transitions and actions, we have:

$$\begin{aligned} W_1(\mathcal{T}^\pi Z_1(s,a), \mathcal{T}^\pi Z_2(s,a)) &= \gamma\, W_1\big(Z_1(s', a'), Z_2(s', a')\big) \\ &\le \gamma\, \bar{d}_1(Z_1, Z_2). \end{aligned} \tag{10}$$

Taking the supremum over (s, a) yields $\bar{d}_1(\mathcal{T}^\pi Z_1, \mathcal{T}^\pi Z_2) \le \gamma\, \bar{d}_1(Z_1, Z_2)$, establishing that $\mathcal{T}^\pi$ is a γ-contraction. By induction, the k-step operator $(\mathcal{T}^\pi)^k$ is a γ^k-contraction.

Next, consider the Distributional GAE operator $\mathcal{T}^\pi_{\mathrm{GAE}}$. It is constructed as a geometric mixture of k-step returns, parameterized by $\lambda \in [0, 1]$:

$$\mathcal{T}^\pi_{\mathrm{GAE}} Z \overset{D}{=} \sum_{k=1}^{\infty} (1-\lambda)\lambda^{k-1} (\mathcal{T}^\pi)^k Z. \tag{11}$$

A key property of the 1-Wasserstein metric is its convexity with respect to mixture distributions. For any set of distributions $\{\mu_k, \nu_k\}$ and weights w_k summing to 1, $W_1(\sum w_k \mu_k, \sum w_k \nu_k) \le$

$\sum w_k W_1(\mu_k, \nu_k)$. Applying this property to the GAE operator:

$$\begin{aligned}
&\bar{d}_1(\mathcal{T}^\pi_{\mathrm{GAE}} Z_1, \mathcal{T}^\pi_{\mathrm{GAE}} Z_2) \\
&\leq \sum_{k=1}^{\infty} (1-\lambda)\lambda^{k-1}\, \bar{d}_1\Big((\mathcal{T}^\pi)^k Z_1, (\mathcal{T}^\pi)^k Z_2\Big) \\
&\leq (1-\lambda) \sum_{k=1}^{\infty} \lambda^{k-1}\gamma^k\, \bar{d}_1(Z_1, Z_2) \\
&= \gamma(1-\lambda) \left(\sum_{j=0}^{\infty} (\lambda\gamma)^j\right) \bar{d}_1(Z_1, Z_2) \\
&= \frac{\gamma(1-\lambda)}{1-\lambda\gamma}\, \bar{d}_1(Z_1, Z_2).
\end{aligned} \tag{12}$$

Let $\Gamma := \frac{\gamma(1-\lambda)}{1-\lambda\gamma}$. Since $\gamma < 1$ and $\lambda \in [0, 1]$, it follows that $\Gamma < 1$. Thus, $\mathcal{T}^\pi_{\mathrm{GAE}}$ is a contraction mapping with factor Γ. By the Banach Fixed-Point Theorem, repeated application of $\mathcal{T}^\pi_{\mathrm{GAE}}$ converges to a unique fixed-point return distribution. □

Remark. The contraction argument above analyzes the geometric-mixture (n-step mixing) operator at the level of return *distributions*. The implementation in Sec. 3.3 instead accumulates a rank-wise weighted sum of per-quantile TD errors, which couples quantiles by their sorted index rather than by mixing full distributions; these are not the same object (an equal-weight mixture of δ_0 and δ_2 differs from the pointwise average of the two constants, δ_1). We therefore use the contraction result as an idealized convergence analysis, and treat the rank-wise recursion as an empirical surrogate target whose practical value is the improved mean-advantage estimate it provides for the actor, rather than a claim of exact recovery of the true return distribution.

B.2 Sensitivity-Aware Flow Optimization

Proposition 2. *The Uncertainty-Weighted Distributional CFM (UDCFM) objective enhances the learning of complex flow dynamics by prioritizing regions with high flow sensitivity, thereby preventing over-smoothing in state-dependent transport paths.*

Proof. Consider the regression target in Flow Matching, which aims to fit the vector field v_θ to the target velocity $u_t = x_1 - x_0$. In real-world RL, the supervision is often multi-modal or sparse, leading to regions where the optimal transport path exhibits high complexity.

Let $\mathcal{R}_{\text{complex}}$ denote the subset of the state space where the flow dynamics exhibit high sensitivity to initial conditions, characterized by a large Jacobian norm $\|J(1)\|^2$. Standard Flow Matching optimization minimizes the expected MSE: $\mathcal{L}_{\text{FM}} = \mathbb{E}[\|v_\theta - u_t\|^2]$. Under limited capacity or noisy supervision, the minimizer of this unweighted objective tends to converge to the conditional mean $\mathbb{E}[x_1|x_0]$, which often results in *over-smoothed* vector fields that fail to capture the sharp transitions in $\mathcal{R}_{\text{complex}}$.

Our UDCFM objective introduces a sensitivity-dependent weight:

$$w_{\text{sens}}(\mathbf{s}) \propto \sigma(\|J(1)\|^2/\tau) \tag{13}$$

Since the sigmoid function is monotonically increasing, this weighting scheme assigns higher importance to samples with large Jacobian norms (i.e., high-sensitivity regions). Instead of treating high sensitivity purely as aleatoric noise to be discarded, we interpret it as a signal of topological complexity or hard examples that require stronger gradient updates to be learned correctly.

By assigning $w_{\text{sens}}(\mathbf{s}) > 1$ to samples in $\mathcal{R}_{\text{complex}}$, the gradient descent process focuses more capacity on minimizing the approximation error in these difficult regions:

$$\nabla\mathcal{L}_{\text{UDCFM}} \approx \mathbb{E}\left[(1 + \alpha\|J(1)\|^2)\nabla\|v_\theta - u_t\|^2\right] \tag{14}$$

This acts as a form of curriculum learning or focal loss, ensuring that the learned policy does not collapse to a trivial mean solution in high-dynamic scenarios, thus improving the robustness of the value distribution's structural integrity. □

B.3 Geometric Consistency Regularizes Trajectories toward Straight Lines

Proposition 3. *If the geometric consistency loss $\mathcal{L}_{cons}$ is minimized to zero for all $t \in [0, 1)$, then the learned vector field v_θ along the flow trajectory is constant in time, implying that the generated flow trajectories are straight lines.*

Proof. Let z_t be the state at time t governed by the ODE $dz_t/dt = v_\theta(z_t, t)$. The geometric consistency constraint enforces that the projected terminal state $\hat{x}_1(z_t)$ is invariant with respect to time t. The projection operator is defined based on a linear flow assumption:

$$\hat{x}_1(z_t, t) = z_t + (1 - t)v_\theta(z_t, t) \tag{15}$$

Minimizing $\mathcal{L}_{\text{cons}}$ to zero implies that $\hat{x}_1(z_t, t)$ is constant for all t along the trajectory. Taking the total time derivative of $\hat{x}_1$ and setting it to zero:

$$\frac{d}{dt}\hat{x}_1(z_t, t) = \frac{dz_t}{dt} + \frac{d}{dt}\left[(1 - t)v_\theta(z_t, t)\right] = 0 \tag{16}$$

Substituting $dz_t/dt = v_\theta(z_t, t)$ and applying the product rule:

$$\begin{aligned} v_\theta(z_t, t) + \left[(-1) \cdot v_\theta(z_t, t) + (1 - t)\frac{d}{dt}v_\theta(z_t, t)\right] &= 0 \\ v_\theta(z_t, t) - v_\theta(z_t, t) + (1 - t)\frac{dv_\theta}{dt} &= 0 \\ (1 - t)\frac{dv_\theta}{dt} &= 0 \end{aligned} \tag{17}$$

For any $t < 1$, this implies that the total derivative of the velocity field along the trajectory is zero:

$$\frac{dv_\theta(z_t, t)}{dt} = 0 \implies v_\theta(z_t, t) = \mathbf{c} \quad \text{(constant vector)} \tag{18}$$

Consequently, the trajectory z_t is the integral of a constant velocity, which is a straight line:

$$z_t = z_0 + \int_0^t \mathbf{c}\, d\tau = z_0 + t \cdot \mathbf{c} \tag{19}$$

This proves that enforcing geometric consistency is theoretically equivalent to regularizing the flow towards a straight-line Optimal Transport path. □

Remark. In practice the consistency loss is minimized to a small but nonzero value with finite K quantile particles and finite samples, so the straight-line property above should be read as an *empirical* regularizer that drives trajectories toward, rather than exactly onto, straight lines; our single-step inference relies on the induced straightness being good enough for the 1-step Euler solver (Sec. B.4), not on exact equality.

B.4 Sufficiency of 1-Step Euler Solver

Proposition 4. *Under the assumption that the flow matching objective and consistency regularization are optimized, the learned vector field approximates a straight-line trajectory. In this regime, the numerical error of a 1-Step Euler solver vanishes, rendering it sufficient for high-precision inference.*

Proof. Our training objective combines Conditional Flow Matching with Geometric Consistency. As shown in Proposition 3, the consistency term encourages the vector field v_θ to be approximately constant in time, i.e., $v_\theta(z_t, t) \approx v_0$ for all $t \in [0, 1]$.

Consider the numerical integration of the ODE $dz_t = v_\theta(z_t, t)dt$ from $t = 0$ to $t = 1$. The exact solution is:

$$z_1 = z_0 + \int_0^1 v_\theta(z_t, t)dt \tag{20}$$

A 1-Step Euler solver approximates this integral as:

$$\hat{z}_1^{\text{Euler}} = z_0 + (1 - 0) \cdot v_\theta(z_0, 0) \tag{21}$$

The local truncation error of the Euler method is bounded by $\mathcal{O}(\Delta t^2 \max_t \|\ddot{z}_t\|)$. Since $\ddot{z}_t = \frac{d}{dt} v_\theta(z_t, t)$, and Proposition 3 implies $\frac{dv_\theta}{dt} \to 0$, the acceleration term vanishes ($\ddot{z}_t \approx 0$).

Therefore, as the flow trajectory becomes a straight line (Straight Flow), the Euler approximation becomes exact:

$$\int_0^1 v_\theta(z_t, t)dt = \int_0^1 v_0 dt = v_0 = v_\theta(z_0, 0) \tag{22}$$

This theoretical result aligns with the findings of Rectified Flow Liu et al. (2022), which demonstrates that minimizing the transport cost yields straight displacement paths. Our method effectively performs a "One-Step Rectification" during training by enforcing the target $x_1 - x_0$ (straight path) and the consistency constraint, allowing the model to generate high-fidelity samples with a single function evaluation (NFE=1) during inference, avoiding the computational cost of iterative solvers used in diffusion models. □

Remark. With independently sampled $x_0 \sim \mathcal{N}(0, 1)$ and $x_1 \sim Z^{\text{tgt}}(\mathbf{s})$, the standard (unregularized) CFM optimum at $t = 0$ is $v^*(z_0, 0, \mathbf{s}) = \mathbb{E}[x_1 \mid \mathbf{s}] - z_0$, so a 1-step Euler projection would output $\hat{z}_1 = \mathbb{E}[x_1 \mid \mathbf{s}]$ for every particle, collapsing to the conditional mean rather than reproducing a multi-modal target. This is a property of unregularized CFM; in DFPO the flow is additionally straightened by $\mathcal{L}_{\text{cons}}$ and its tail structure is preserved by $\mathcal{L}_{\text{Risk}}$ and $\mathcal{L}_{\text{Shape}}$, so the 1-step projection no longer coincides with the conditional mean. We therefore treat single-step fidelity as an empirical property of the regularized objective, supported by the sampling-step ablation (Table 8), rather than as a guarantee inherited from unregularized CFM.

B.5 Stationary Point Invariance of UDCFM Objective

Proposition 5. (Target Distribution Matching under State-Dependent Weighting) *Let $w_{sens}(\mathbf{s}) > 0$ be a strictly positive, state-dependent sensitivity weight. The optimal neural vector field v_θ^* that minimizes the weighted UDCFM objective in Eq. (1) coincides point-wise with the optimal solution of the standard unweighted Conditional Flow Matching objective. Consequently, UDCFM strictly preserves the target distribution matching property.*

Proof. In conditional flow matching, the objective is formulated over the joint probability path of states $\mathbf{s}$, latent virtual time $t \sim \mathcal{U}[0, 1]$, initial noise $x_0 \sim \mathcal{N}(0, 1)$, and target return samples $x_1 \sim Z^{\text{tgt}}(\mathbf{s})$. Given intermediate state $z_t = tx_1 + (1 - t)x_0$ and transformer hidden state $\mathbf{h_s}$, the optimal vector field $v^*(z_t, t, \mathbf{h_s})$ solves:

$$v^*(z_t, t, \mathbf{h_s}) = \arg\min_v \mathbb{E}_{\mathbf{s},t,x_0,x_1} \left[w_{\text{sens}}(\mathbf{s}) \cdot \|v(z_t, t, \mathbf{h_s}) - (x_1 - x_0)\|^2 \right]. \tag{23}$$

By applying the law of total expectation, we condition the expectation on the realized flow state z_t, virtual time t, and state conditioning $\mathbf{s}$ (represented by $\mathbf{h_s}$):

$$v^*(z_t, t, \mathbf{h_s}) = \arg\min_v \mathbb{E}_{\mathbf{s},t,z_t} \left[w_{\text{sens}}(\mathbf{s}) \cdot \mathbb{E}_{x_0,x_1|z_t,t,\mathbf{s}} \left[\|v(z_t, t, \mathbf{h_s}) - (x_1 - x_0)\|^2 \right] \right]. \tag{24}$$

Since the minimizer can be determined pointwise for each fixed tuple $(z_t, t, \mathbf{s})$, and crucially because the sensitivity weight $w_{\text{sens}}(\mathbf{s}) = \sigma(\|J(1)\|^2/\tau_{\text{temp}}) + 0.5$ is strictly positive ($w_{\text{sens}}(\mathbf{s}) \geq 0.5 > 0$) and independent of the random sampling of (x_0, x_1) along the conditional path:

$$v^*(z_t, t, \mathbf{h_s}) = \arg\min_v \mathbb{E}_{x_0,x_1|z_t,t,\mathbf{s}} \left[\|v(z_t, t, \mathbf{h_s}) - (x_1 - x_0)\|^2 \right]. \tag{25}$$

By the standard property of L_2 regression, the unique minimizer is the conditional expectation of the target vector field:

$$v^*(z_t, t, \mathbf{h_s}) = \mathbb{E}_{x_0,x_1|z_t,t,\mathbf{s}}[x_1 - x_0 \mid z_t, t, \mathbf{s}]. \tag{26}$$

This establishes that multiplying by the state-dependent scalar $w_{\text{sens}}(\mathbf{s})$ does not shift the global stationary point. By the continuity equation governing probability density paths (Lipman et al., 2023), the learned vector field generates trajectories whose push-forward distribution at $t = 1$ strictly recovers the true empirical target return distribution $Z^{\text{tgt}}(\mathbf{s})$. □

Remark. The invariance argument treats $w_{\text{sens}}(\mathbf{s})$ as fixed when optimizing v_θ. In our implementation the weight is computed from a detached sensitivity estimate (stop-gradient on $\|J(1)\|^2$) so that it acts

as a constant per state rather than as an additional function of x_1; this is the sense in which the target-distribution matching property holds. As a consequence the weighting only rescales the per-sample contribution to the regression objective and does not alter which empirical return distribution the flow matches.

C Additional Experimental Details

C.1 Additional Experimental Setup

We evaluate DFPO on dialogue, mathematical reasoning, and scientific QA tasks under both model-based and rule-based reward supervision. All experiments follow a unified training pipeline to ensure fair comparison across methods, while the learning objectives and value modeling differ across approaches.

Evaluation Method We employ a strict, unguided evaluation pipeline: all models are evaluated using zero-shot greedy decoding (temperature $= 0$) with a multi-turn generation horizon of 4096 tokens. This setup intentionally omits 5-shot Chain-of-Thought prompting, which artificially guides output formatting, in order to test purely intrinsic reasoning capabilities. The full, unsplit test sets are used. **While the resulting absolute accuracy scores are lower than official leaderboard numbers, our internal comparisons are strictly fair, all RL methods and baselines are evaluated under this exact same pipeline.**

Dialogue Task To evaluate DFPO under realistic model-based reward supervision, we conduct multi-turn dialogue experiments on the **Honor-Dialogue Dataset**. This dataset consists of task-oriented, multi-domain conversations collected from real-world usage scenarios.Reinforcement learning is performed using noisy feedback from a dialogue reward model on real multi-turn trajectories. Evaluation is carried out through interactions with GPT-4o, following the dialogue evaluation methods described in Section C.3. All data have been anonymized, and the dataset usage has been reviewed and approved by the ethics committee.

Math and Science Tasks To assess robustness under rule-based supervision, we further evaluate DFPO on mathematical and scientific reasoning tasks. We adopt the Light-R1 dataset Wen et al. (2025) for math training and SuperGPQA Team et al. (2025) for scientific training. For each dataset, we first generate multiple candidate solutions using the initial model and apply majority voting to obtain pseudo-labels for reinforcement learning. Evaluation is conducted on a diverse set of benchmarks, including math datasets MATH500 Hendrycks et al. (2021), AIME24, Minerva-Math Lewkowycz et al. (2022), AMC23 and science datasets SimpleQA Wei et al. (2024), GPQA, and HLE.All benchmarks are reformulated as open-ended QA without multiple-choice comparison, so the reported accuracies reflect pure generation performance and are lower than the original multiple-choice settings. Due to memory constraints, generation is performed over multiple rounds, with a maximum length of 4096 tokens per round, and the final round output is used for evaluation.

Model Initialization and Baselines For math and science tasks, DFPO and all baselines are initialized from Qwen3-8B Yang et al. (2025). For dialogue tasks, both the policy model and reward model are initialized from Qwen3-8B and further fine-tuned on the Honor-Dialogue dataset. In addition to robust RL baselines like KTAE, BAPO, and λ-GRPO Sun et al. (2025); Wang et al. (2025); Xi et al. (2025a), we include flow-based RL methods like FlowRL Zhu et al. (2025d) and robust distributional value RL methods that construct multiple value estimates per token and aggregate them to guide policy optimization under noisy rewards. Since no relevant work exists in the LLM domain, we implemented a distributional PPO algorithm based on the robust Bellman operator inspired by recent studies Ma et al. (2021); Hu et al. (2025). This algorithm maintains a 200-dimensional distributional return estimate per token and then collapses it to a scalar pessimistic lower bound by taking the minimum, simulating a worst-case learning scenario; because the policy update consumes only this scalar minimum, the baseline retains per-token distributional statistics internally but lacks the token-level distributional resolution that DFPO's value flow provides.

Training Noise Noise is pervasive in both model-based and rule-based reward supervision. Figure 5 shows the error rates of different reward signals. Specifically, rule-based rewards from majority-voted

Table 5: Statistics of the validation dataset across 11 real-world service scenarios. The dataset features diverse interaction types with varying dialogue lengths, providing a robust testbed for evaluating generalization.

Scenario	Wealth	Rental	Insurance	Food	Express	Promotion	Loan	Housing	Service	Product	General	Avg
Dialogue Count	87	99	138	120	215	66	70	87	67	69	92	100.91
Avg Turns	5.40	4.22	5.22	3.37	3.76	4.55	4.33	4.68	5.46	4.44	3.46	4.44

generations contain mislabels, causing erroneous supervision under noisy scenarios in math and science tasks.

Training Configuration During the exploration phase, we employ a sampling temperature of 1.0 with a top-p setting of 1.0. For the optimization objective, the hyperparameters balancing the loss components are configured as follows: the regularization weights are set to $\lambda_{\text{reg}} = 0.1$ for Bootstrapped CFM and $\lambda_{\text{cons}} = 0.01$ for geometric consistency; the distributional constraint weights are set to $\lambda_{\text{risk}} = 0.5$, and $\lambda_{\text{Shape}} = 0.5$. In the distributional value model, the risk level α and gain level β are both set to 0.1 (corresponding to the 10% and 90% quantiles). The number of flow sampling steps (quantiles) is set to $K = 50$, and the Jacobian estimation for sensitivity weighting utilizes 10 ODE steps. The model generates sequences of length 4096 for reasoning tasks and 1024 for dialogue tasks. The reinforcement learning training is performed for 1 epoch per iteration on a cluster of 8 $\times$ NVIDIA A100 80GB GPUs.

C.2 Data Collection and Preprocessing

Math and Scientific Reasoning For mathematical reasoning tasks, we utilize Qwen3-8B as the base model to generate pseudo-labels from 39,000 samples in the Light-R1 dataset Wen et al. (2025). A plurality-consistency filtering process is applied over 5 candidate responses per sample: a sample is retained as a pseudo-label only when at least 3 out of 5 responses are identical (an unambiguous plurality; responses are not aggregated by strict majority vote, and samples with no consistent answer are discarded). This results in a final dataset of 31,209 instances suitable for RL training. For scientific QA tasks, pseudo-labels are derived from 26,529 samples in the SuperGPQA dataset. Applying the same 5-response consistency filtering but with a lower threshold (at least 2 identical responses), we preserve 10,075 samples for training. To prevent data leakage and ensure a fair assessment of generalization, the evaluation benchmarks are strictly disjoint from the training data.

Real-World Dialogue We utilize Honor-Dialogue, a dataset derived from authentic real-world interactions. The construction of Honor-Dialogue follows a goal-oriented, scenario-centric paradigm. We initially defined the agent's role as a pragmatic, context-aware assistant, imposing rigorous constraints regarding accuracy, naturalness, and adherence to service norms. We then curated many representative domains, one general and lots of professional domains, based on high-frequency real-life interactions. Each domain is defined by explicit triggers, such as user intent and contextual features, alongside standardized protocols for response logic and information prioritization. For every domain, we synthesized realistic user inputs comprising current messages and dialogue history, paired with standardized outputs aligned with specific goals. To ensure high-quality supervision, we explicitly annotated dialogue states, response content, and intended targets. A representative sample is shown in Figure 8, where sensitive details (e.g., user IDs) are masked with "xxx" to strictly adhere to data privacy regulations.

For the reward model, training involves 36,000 labeled instances, with a subset of 3,000 held out for validation. The policy model undergoes fine-tuning and subsequent reinforcement learning on a separate corpus of 50,952 conversations; notably, this training phase excludes data from the financial, social entertainment, transportation, and healthcare sectors. To evaluate performance, we test across five distinct scenarios: Daily Services (1,004 samples), Financial Services (700), Social Entertainment (534), Transportation (463), and Healthcare (380). By encompassing a wide array of practical tasks, this dataset mirrors real-world complexities. Crucially, this level of multi-category, real-scenario diversity is absent in contemporary datasets.

C.3 Dialogue Evaluation Methods

For multi-turn dialogue tasks, we adopt a three-part evaluation methods: (1) Task Completion Rate (TCR), measuring whether the intended task is successfully completed; (2) Ask Completion Rate (ACR), evaluating whether key user requests are fully addressed; (3) Goal Completion Rate (GCR), assessing like overall fluency, coherence, logicality and response quality and so on. This evaluation framework allows us to measure both task effectiveness and alignment with human communication standards.

Our evaluation employs GPT-4o strictly as a rubric-based evaluator rather than a subjective judge. To ensure reproducibility, the model is constrained by a prompt containing precise metrics, a 1–5 grading scale, scoring examples, and formatting requirements. We confirmed reliability via **1)** stability testing through multiple runs and **2)** validation against human scores. Specifically, we conducted a comparative study using 1,110 samples across 11 scenarios, the statistics of which appear in Table 5. Four professional annotators scored this data using identical guidelines. As presented in Table 6,

Table 6: Agreement analysis between human annotators and GPT-4o. The average scores are extremely close, indicating high reliability of the automated evaluation.

Category	Human Annotators				GPT Models		Human AVG	Model AVG
Score	4.595	4.689	4.490	4.330	4.510	4.501	4.526	4.505

the results indicate near-perfect alignment with a marginal mean score difference of 0.021 between human annotators and GPT-4o. These findings confirm a strong consistency between human and automated scores, thereby validating the effectiveness of our rubric. While we acknowledge that establishing full external validity and reproducibility requires further independent annotation, we are in the process of releasing our rubric, evaluation prompts, and the real-world dialogue dataset to facilitate community replication. To illustrate our assessment logic, Figure 9 displays the core prompt section used for evaluating response content relevance.

C.4 Additional experimental details in the Dialogue Task

DFPO establishes superior final convergence stability Table 7 reports the overall dialogue performance at step 1000 across all five domains. While standard RL baselines suffer from severe instability, DFPO achieves the highest step-1000 D-AVG of 87.3%, outperforming Reinforce++ (81.1%) and PPO (61.5%). This advantage persists at the final checkpoint (the **Last** rows of Table 16): DFPO 86.7% vs. Reinforce++ 82.8% and PPO 60.1%. Notably, PPO and GRPO-based methods exhibit a drastic performance collapse at the end of training, with PPO dropping to near 60% averages in most domains and GRPO falling below 30%. This validates our core motivation: without the continuous, risk-controlled flow field, scalar value models are prone to learning unstable policy updates that eventually diverge. In contrast, DFPO maintains robust convergence, matching or exceeding the strongest baselines even in the final training stages.

DFPO prevents catastrophic forgetting and oscillation Figure 6 shows the training dynamics, and integrating the trajectory analysis from Table 16 with the final results in Table 7 reveals a critical advantage in learning dynamics. Baselines like PPO and Reinforce++ exhibit a "peak-and-degrade" pattern: they reach competitive performance at intermediate steps, such as Step 250, but suffer from catastrophic forgetting as training proceeds. For instance, PPO degrades from an early peak of 88.8% to a final 60.1%. DFPO, however, exhibits a flat and stable performance curve, sustaining an average of $\sim$86.6% throughout the entire training process. This empirically proves that the continuous, risk-controlled flow field learned by DFPO effectively regularizes the learning path, preventing the policy from collapsing into erroneous attractors induced by noisy supervision.

Robust generalization across diverse domains At the 1000 training steps, DFPO demonstrates remarkable resilience in OOD settings where other methods falter. In the Healthcare & Wellness domain, DFPO achieves an average of 89.7%, significantly surpassing Reinforce++ (75.6%) and PPO (62.7%). Similarly, in the Transportation & Travel domain, DFPO maintains 90.5%, whereas PPO collapses to 64.7%. Even in domains with highly complex semantics, such as the Social & Entertainment domain, DFPO retains high precision (86.4%), largely outperforming Dr.GRPO

Table 7: Comprehensive results on the Real-world Dialogue Benchmark at 1000 training steps under noisy reward supervision. We report Task Completion (TC), Ask Completion (AC), and Goal Completion (GC) rates for each domain. DomainAVG (D-AVG) represents the average performance across these three metrics within a domain, while AVG denotes the overall average across all five domains. DFPO achieves the highest overall average, consistently outperforming all prior RL baselines in both in-domain and out-of-domain settings.

Method	Life Services (In-Domain)				Transportation & Travel				Healthcare & Wellness				Social & Entertainment				Financial Services				AVG
	TC	AC	GC	D-AVG	TC	AC	GC	D-AVG	TC	AC	GC	D-AVG	TC	AC	GC	D-AVG	TC	AC	GC	D-AVG	
Baseline	**72.5**	**95.0**	**92.7**	**86.7**	66.7	**96.5**	90.3	84.5	**77.2**	97.8	**95.7**	**90.2**	**75.3**	92.0	**94.1**	**87.1**	65.7	91.4	**91.0**	82.7	86.3
GRPO	47.4	14.6	25.7	29.2	41.0	20.5	28.5	30.0	24.6	21.6	28.5	24.9	43.3	10.0	23.5	25.6	50.9	12.7	24.5	29.4	27.8
PPO	43.8	63.4	70.4	59.2	46.2	77.1	70.8	64.7	31.6	77.4	79.0	62.7	47.4	61.5	70.2	59.7	45.3	70.4	68.7	61.5	61.5
Reinforce++	70.9	92.5	92.1	85.2	54.1	87.9	86.0	76.0	45.8	94.3	86.8	75.6	66.1	**99.2**	90.9	85.4	**69.6**	92.4	88.6	83.5	81.1
Dr.GRPO	34.5	45.3	54.2	44.7	37.0	53.0	60.0	49.9	23.0	66.0	62.0	50.2	44.0	50.0	58.0	50.8	35.0	54.0	51.0	46.6	48.4
DFPO (Ours)	70.8	94.4	91.8	85.7	**82.1**	94.5	**94.9**	**90.5**	75.4	**98.8**	94.9	89.7	71.1	96.8	91.3	86.4	69.4	**92.6**	90.9	**84.3**	**87.3**

Table 8: Performance comparison on the number of sampling steps used to integrate the probability-flow ODE during training. Single-step sampling achieves the best overall performance, reaching 40.25% average accuracy, with strong results on MATH500 (91.80%) and AIME24 (56.67%). Increasing the sampling steps to 5 leads to a clear performance drop, while larger steps (10 and 20) partially recover performance on some benchmarks but remain lower overall.

Domain	In-Domain (Science & QA)			Out-of-Domain (Math)				ID	OOD	ALL
Sampling Steps	SimpleQA	GPQA(ALL)	HLE	MATH500	AIME24	Minerva-Math	AMC23	AVG	AVG	AVG
Base	2.89%	3.10%	2.89%	87.40%	41.67%	28.68%	75.83%	2.96%	58.40%	34.64%
1 (Ours)	**3.17%**	**5.43%**	4.03%	**91.80%**	**56.67%**	**32.35%**	88.33%	**4.21%**	**67.29%**	**40.25%**
5	2.87%	4.17%	3.10%	90.20%	50.00%	31.62%	85.83%	3.38%	64.41%	38.26%
10	3.05%	4.17%	3.34%	89.40%	50.00%	31.99%	86.67%	3.52%	64.52%	38.37%
20	2.98%	4.89%	**4.22%**	90.60%	55.00%	29.04%	**90.00%**	4.03%	66.16%	39.53%

(50.8%). These results confirm that the continuous, risk-controlled flow field learned by DFPO captures stable semantic features that generalize well, ensuring reliable performance even when baseline methods fail to converge.

C.5 Theoretical Grounding and Literature Motivation for Regularization Objectives

To address the concern regarding the mathematical grounding of the composite objective in Section 3.4, we elucidate the theoretical origins and design motivations for each regularization term:

Uncertainty-Weighted DCFM ($\mathcal{L}_{\text{UDCFM}}$, Eq. 1): While standard flow matching treats all training samples uniformly, return distributions in RL exhibit non-uniform semantic complexity. Weighting flow matching objectives was conceptually pioneered in Dong et al. (2025) to prevent probability mass collapse under complex topologies. Our design innovates by grounding this weighting in the sensitivity ODE's Jacobian norm $\|J(1)\|^2$. Rather than filtering out sensitivity as aleatoric noise, high Jacobian values indicate critical decision points (e.g., pivot tokens in reasoning or dialogue state transitions) where minute latent variations dramatically change expected returns. Weighting ensures the neural ODE allocates sufficient model capacity to resolve these high-frequency semantic bifurcations without altering the stationary point (Proposition B.5).

Bootstrapped Anchor Regularization ($\mathcal{L}_{\text{BCFM}}$, Eq. 2): In actor-critic RL, early-stage bootstrap targets $x_1 \sim Z^{\text{tgt}}(\mathbf{s})$ suffer from severe stochastic variance. Unconstrained regression on such volatile targets causes erratic flow vector oscillations. Grounded in continuous-time consistency models (Lu & Song, 2025) and flow-anchoring principles (Agrawalla et al., 2025), $\mathcal{L}_{\text{BCFM}}$ regularizes intermediate velocity toward a detached expectation anchor $x_1^{\text{anc}} = \text{StopGrad}(\mathbb{E}[x_1 \mid \mathbf{s}])$. By bounding estimation variance through target detach, this prevents compounding shortcut errors and eliminates early-phase policy divergence. Treating the anchor as a fixed target, the combined objective $\mathcal{L}_{\text{UDCFM}} + \lambda_{\text{reg}}\mathcal{L}_{\text{BCFM}}$ corresponds to flow matching toward the mixture distribution $(p_{\text{tgt}} + \lambda_{\text{reg}}\,\delta_a)/(1+\lambda_{\text{reg}})$; this deliberately introduces a small, interpretable bias (shrinking the modeled distribution toward the anchor mean) that detach does not remove.

Table 9: Performance comparison on the effect of the risk level α (tail probability, Eq. equation 4) in DFPO. A moderate risk level of $\alpha = 0.1$ leads to the best balance between stability and generalization across in-domain (Science & QA) and out-of-domain (Math) benchmarks. $\alpha = 0$ disables the risk constraint entirely; the risk loss weight $\lambda_{\text{risk}} = 0.5$ is fixed across all settings.

Domain	**In-Domain (Science & QA)**			**Out-of-Domain (Math)**				**ID**	**OOD**	**ALL**
Risk Interval	**SimpleQA**	**GPQA(ALL)**	**HLE**	**MATH500**	**AIME24**	**Minerva-Math**	**AMC23**	**AVG**	**AVG**	**AVG**
Base	2.89%	3.10%	2.89%	87.40%	41.67%	28.68%	75.83%	2.96%	58.40%	34.64%
0	3.01%	4.17%	3.52%	89.20%	50.00%	30.51%	83.33%	3.57%	63.26%	37.68%
0.05	3.21%	5.07%	**4.08%**	90.60%	55.00%	31.99%	84.17%	4.12%	65.44%	39.16%
0.1 (Ours)	3.17%	**5.43%**	4.03%	**91.80%**	**56.67%**	**32.35%**	**88.33%**	**4.21%**	**67.29%**	**40.25%**
0.2	**3.31%**	4.71%	3.38%	89.20%	46.67%	29.04%	84.17%	3.80%	62.27%	37.21%

Table 10: Performance comparison on a Qwen3-1.7B model under noisy supervision. DFPO consistently achieves the best overall performance 33.75%, demonstrating strong scalability and robustness across domains.

Domain	**In-Domain (Science & QA)**			**Out-of-Domain (Math)**				**ID**	**OOD**	**ALL**
Method	**SimpleQA**	**GPQA(ALL)**	**HLE**	**MATH500**	**AIME24**	**Minerva-Math**	**AMC23**	**AVG**	**AVG**	**AVG**
Base	1.78%	2.36%	2.87%	84.80%	35.00%	19.49%	69.17%	2.34%	52.12%	30.78%
GRPO	1.60%	1.99%	3.94%	82.20%	33.33%	21.32%	77.50%	2.51%	53.59%	31.70%
PPO	**1.91%**	2.54%	3.94%	83.80%	33.33%	22.43%	**79.17%**	2.80%	54.68%	32.45%
Reinforce++	1.48%	2.36%	3.52%	84.60%	38.33%	23.16%	75.83%	2.45%	55.48%	32.75%
Dr. GRPO	1.50%	2.17%	3.10%	84.00%	35.00%	22.79%	76.67%	2.26%	54.62%	32.18%
KTAE	1.64%	**2.90%**	**4.36%**	82.20%	26.67%	23.90%	64.17%	**2.97%**	49.24%	29.41%
λ-GRPO	1.62%	**2.90%**	3.89%	**85.00%**	**40.00%**	22.06%	70.83%	2.80%	54.47%	32.33%
BAPO	1.55%	2.36%	3.48%	82.40%	30.00%	19.85%	64.17%	2.46%	49.11%	29.12%
Robust Bellman	1.64%	2.17%	4.17%	84.60%	31.67%	22.79%	75.00%	2.66%	53.52%	31.72%
FlowRL	1.76%	1.81%	3.99%	84.00%	36.67%	22.06%	75.83%	2.52%	54.64%	32.30%
DFPO (Ours)	1.76%	2.54%	3.99%	**85.00%**	**40.00%**	**24.63%**	78.33%	2.76%	**56.99%**	**33.75%**

Geometric Consistency Regularization ($\mathcal{L}_{\text{cons}}$, Eq. 3): Flow trajectory crossing and entanglement are well-known failure modes in generative ODEs (Dupont et al., 2019). Grounded in the semigroup properties of continuous flows and Rectified Flow theory (Liu et al., 2022), enforcing identical projections from symmetric time points (t and $1 - t$) encourages the acceleration $\frac{dv_\theta}{dt}$ toward zero along each path (Proposition B.3 and its Remark; the property holds exactly only under the stronger time-invariance idealization). This straightens the generative trajectories toward Optimal Transport displacement interpolations, serving as the practical regularizer that makes single-step Euler solver inference ($NFE = 1$, Proposition B.4) accurate in the straightened regime.

Conditional Value Risk Optimization ($\mathcal{L}_{\text{Risk}}$, Eq. 4): Traditional variance penalties constrain the entire distribution symmetrically, which inadvertently penalizes positive exploratory upside. Grounded in Conditional Value-at-Risk (CVaR) optimization (Rockafellar et al.), $\mathcal{L}_{\text{Risk}}$ establishes an asymmetric risk objective. By explicitly bounding the lower-α tail distance to the target, the agent suppresses disastrous lower-tail catastrophic outcomes (e.g., formatting collapse, toxic outputs) while leaving the upper-β tail free to pursue high-return, out-of-domain exploratory reasoning paths. This is a calibration of the critic's distributional estimate rather than a direct risk constraint on the policy objective.

Tail Curvature and Shape Regularization ($\mathcal{L}_{\text{Shape}}$, Eq. 5): In distributional RL, Dabney et al. (2018) established that applying non-linear distortion risk measures to cumulative return functions directly shapes risk-averse or risk-seeking policies. Grounded in this framework, $\mathcal{L}_{\text{Shape}}$ regularizes the second-order finite differences of quantile particles. By enforcing concavity ($\nabla^2 \leq 0$) on the left tail $\mathcal{I}_L$, the value model exhibits strict local risk aversion that contracts spurious negative reward noise. Conversely, enforcing convexity ($\nabla^2 \geq 0$) on the right tail $\mathcal{I}_R$ encourages risk-seeking behavior that preserves sparse positive rewards critical for OOD generalization.

C.6 Additional Ablation Experiment

Single-step trajectory sampling is sufficient for stable flow learning Table 8 reports the results with different numbers of trajectory sampling steps during training.

Table 11: Performance comparison on the weight of the flow consistency loss. A moderate setting (0.01) achieves the best overall performance, reaching the highest average accuracy of 40.25%. Stronger or weaker consistency constraints lead to lower performance, especially on OOD benchmarks.

Domain	In-Domain (Science & QA)			Out-of-Domain (Math)				ID	OOD	ALL
Consistency Weight	SimpleQA	GPQA(ALL)	HLE	MATH500	AIME24	Minerva-Math	AMC23	AVG	AVG	AVG
Base	2.89%	3.10%	2.89%	87.40%	41.67%	28.68%	75.83%	2.96%	58.40%	34.64%
0.001	**3.17%**	3.99%	3.75%	90.00%	48.33%	31.25%	80.83%	3.64%	62.60%	37.33%
0.005	2.96%	4.35%	3.71%	90.60%	**60.00%**	29.78%	**88.33%**	3.67%	67.18%	39.96%
0.01 (Ours)	**3.17%**	**5.43%**	**4.03%**	**91.80%**	56.67%	**32.35%**	**88.33%**	**4.21%**	**67.29%**	**40.25%**
0.05	3.01%	3.99%	3.61%	89.60%	51.67%	31.25%	83.33%	3.54%	63.96%	38.07%

Table 12: Performance comparison on the effect of tail contraction and expansion in DFPO. Each row is labeled as a "lower-tail / upper-tail" treatment: “Contract” imposes a risk-averse constraint that shrinks the corresponding tail, while “Expand” applies a gain-seeking constraint. The asymmetric setting “Contract / Expand” (lower tail contracted, upper tail expanded) achieves the best balance between stability and generalization across in-domain (Science & QA) and out-of-domain (Math) benchmarks.

Domain	In-Domain (Science & QA)			Out-of-Domain (Math)				ID	OOD	ALL
Tail Treatment	SimpleQA	GPQA(ALL)	HLE	MATH500	AIME24	Minerva-Math	AMC23	AVG	AVG	AVG
Base	2.89%	3.10%	2.89%	87.40%	41.67%	28.68%	75.83%	2.96%	58.40%	34.64%
Expand / Expand	**3.19%**	5.07%	3.66%	90.00%	41.67%	29.04%	84.17%	3.97%	61.22%	36.69%
Expand / Contract	3.14%	4.53%	3.99%	89.60%	48.33%	31.25%	86.67%	3.89%	63.96%	38.22%
Contract / Expand (Ours)	**3.17%**	**5.43%**	**4.03%**	**91.80%**	**56.67%**	**32.35%**	**88.33%**	**4.21%**	**67.29%**	**40.25%**
Contract / Contract	2.80%	4.53%	3.15%	89.00%	51.67%	31.25%	86.67%	3.49%	64.65%	38.44%

We observe that sampling a single trajectory achieves the best overall performance, indicating that one-step sampling is sufficient to learn the global flow structure of trajectories. When the sampling number is increased to 5, performance drops noticeably, likely due to sampling bias that increases the impact of noisy or wrong paths. As the number of sampling steps further increases, performance gradually recovers on some benchmarks, but still does not surpass the single-step setting. This suggests that while more sampling steps provide richer local flow information, they also make the value model more sensitive to erroneous trajectory regions. In contrast, single-step sampling captures the overall trajectory distribution while reducing the impact of noisy fluctuations, leading to more stable learning and better generalization.

Proper risk interval control improves robustness and generalization. Table 9 illustrates the effect of different risk interval settings in DFPO. When risk interval control is removed and the setting is 0, the value model becomes more sensitive to noisy supervision and shows weaker and less stable performance across tasks. When the interval constraint is further increased to a strong level, overall performance drops on most benchmarks, suggesting that excessive restriction limits effective exploration. Overall, the moderate setting 0.1 adopted in DFPO achieves the best balance between stability and exploration, reaching the highest average accuracy of 40.25%.

Asymmetric tail contraction and expansion improves robustness and generalization. Table 12 reports the effect of contraction and expansion applied to the two tails of the value distribution, where contraction imposes a risk-averse constraint and expansion a gain-seeking one; each row is labeled by the treatment on the lower tail followed by that on the upper tail. When both tails are expanded, the value model becomes overly sensitive to noisy supervision and shows the weakest gains across tasks, confirming that naive tail expansion amplifies reward noise rather than capturing it. Introducing contraction on the lower tail consistently improves performance regardless of the upper-tail setting: the overall average rises from 36.69% to 40.25% with an upper-tail expansion, and from 38.22% to 38.44% with an upper-tail contraction, indicating that risk aversion toward the lower tail is the primary source of training stability. On top of a contracted lower tail, expanding the upper tail further improves out-of-domain and overall accuracy, from 64.65% to 67.29% and from 38.44% to 40.25% respectively; the same upper-tail expansion is ineffective when the lower tail remains expanded, an asymmetry no uniform tail policy can capture. Overall, the asymmetric setting that contracts the lower tail and expands the upper tail, adopted in DFPO, achieves the best balance between stability and generalization, reaching the highest average accuracy of 40.25%.

Table 13: Performance comparison of DFPO on the reasoning tasks across multiple random seeds. The results demonstrate that DFPO maintains stable and superior performance regardless of seed variance, with only minor fluctuations across both ID and OOD benchmarks.

Domain	In-Domain (Science & QA)			Out-of-Domain (Math)				ID	OOD	ALL
Seed	SimpleQA	GPQA(ALL)	HLE	MATH500	AIME24	Minerva-Math	AMC23	AVG	AVG	AVG
22	3.05%	4.83%	**4.26%**	90.80%	55.00%	**32.72%**	87.50%	4.05%	66.51%	39.74%
32	3.10%	5.25%	4.03%	91.40%	55.00%	31.99%	87.50%	4.13%	66.47%	39.75%
42 (Ours)	3.17%	**5.43%**	4.03%	**91.80%**	**56.67%**	32.35%	88.33%	**4.21%**	67.29%	**40.25%**
52	**3.21%**	4.53%	3.66%	90.60%	**56.67%**	31.99%	**90.00%**	3.80%	**67.32%**	40.09%

Table 14: Performance comparison with the DVPO baseline on the reasoning tasks. DFPO consistently outperforms DVPO across both in-domain (Science & QA) and out-of-domain (Math) benchmarks.

Domain	In-Domain (Science & QA)			Out-of-Domain (Math)				ALL
Method	SimpleQA	GPQA(ALL)	HLE	MATH500	AIME24	Minerva-Math	AMC23	AVG
DVPO	**3.21%**	4.71%	3.57%	90.60%	**56.67%**	31.99%	86.67%	39.63%
DFPO (Ours)	3.17%	**5.43%**	**4.03%**	**91.80%**	**56.67%**	**32.35%**	**88.33%**	**40.25%**

DFPO scales effectively across different model sizes Table 10 reports results on a smaller-scale model under noisy supervision. Despite the reduced model capacity, DFPO consistently outperforms PPO, Reinforce++, and FlowRL. In particular, DFPO achieves the highest out-of-domain average accuracy of 56.99% on math tasks and maintains competitive performance on science benchmarks. These results show that DFPO remains stable and effective even on smaller models, indicating good scalability and robust generalization under noisy training conditions.

Proper Consistency Constraints Balance Stability and Flexibility in Flow Learning Table 11 shows the effect of different flow consistency loss weights during training. A proper consistency constraint helps the model reach a good balance between constrained and divergent flow learning. With a moderate weight of 0.01, DFPO achieves the best overall performance, reaching an average accuracy of 40.25%, which is clearly higher than the stronger setting of 0.05 (38.07%). When the consistency weight is too large, the flow becomes overly constrained, making value guidance conservative and limiting generalization to out-of-domain states. In contrast, when the weight is too small, the flow tends to diverge, and noisy trajectories introduce instability during training. These results show that an appropriate consistency constraint is important for stable optimization and robust generalization.

DFPO Demonstrates Strong Statistical Stability Across Multiple Training Seeds To evaluate statistical significance and training stability, we conducted additional experiments using multiple training random seeds on the scientific reasoning tasks. As shown in Table 13, DFPO exhibits strong robustness against initialization and sampling variance. The overall average accuracy shows only minor fluctuations, consistently remaining within a tight and high-performing range of 39.74% to 40.25%. Notably, on complex out-of-domain mathematical benchmarks such as MATH500 and AIME24, the model's performance remains highly stable, maintaining scores strictly above 90.60% and 55.00%, respectively, regardless of the seed setting. These results confirm that the superior performance of DFPO is statistically robust and deeply rooted in the method's algorithmic design rather than random chance.

Comparison with DVPO Baseline We have now implemented and evaluated the DVPO Zhu et al. (2025b) baseline across our reasoning and dialogue benchmarks, where DFPO consistently outperforms DVPO. Due to space constraints, we present the comparative results on the Science tasks and the 1000 training steps for the Dialogue tasks in Tables 14 and 15. DFPO's core advantage is its fine-grained flow control, which captures richer state information for accurate advantage estimation and better generalization. Conversely, DVPO only optimizes the overall distribution without fine-grained trajectory constraints. Consequently, DVPO suffers a performance collapse around 1000 steps in the dialogue task, whereas DFPO explicitly prevents this catastrophic forgetting.

Table 15: Performance comparison with the DVPO baseline on the dialogue tasks at 1000 training steps (Task Completion Rate). DVPO suffers a performance collapse, whereas DFPO maintains high task completion rates.

Method	Life Services	Transportation & Travel	Healthcare & Wellness	Social & Entertainment	Financial Services	AVG
DVPO	74.63%	74.30%	81.43%	73.17%	71.33%	74.97%
DFPO (Ours)	**85.67%**	**90.50%**	**89.70%**	**86.40%**	**84.30%**	**87.31%**

C.7 Additional Discussion

Training Flow-based Value Models is Simpler than Policy Models Compared with policy models, training flow-based value models is simpler and more stable. Unlike policy models, value models do not generate language tokens directly; instead, they focus on learning training dynamics and guiding future returns, avoiding sequence generation challenges and reducing sensitivity to linguistic variability. Additionally, value model learning does not rely on importance sampling ratios or explicit action probability estimation, eliminating common approximation errors in policy optimization Zhu et al. (2025c) and resulting in smaller training variance and more reliable gradient signals. As shown in Figure 4, even a simple flow formulation for value models can produce smooth trajectories that closely follow optimal transport paths, highlighting the practical advantages of applying flow modeling at the value model.

Flow constraints enable fine-grained robustness and generalization Flow field modeling enables fine-grained control over training robustness and generalization. As shown in Figure 2, 4, DFPO and unconstrained flow methods exhibit similar relative advantage patterns across response tokens. However, unconstrained flow fields tend to have overlapping and collapsing trajectories, reducing exploration and increasing instability. By introducing consistency-based flow constraints, DFPO guides value flow toward smoother, more coherent transport paths, keeping the learned flow closer to optimal transport under noisy supervision. This allows small, controlled trajectory adjustments to enhance value and advantage estimation, enabling precise cross-domain guidance and improved robustness in noisy environments.

Mathematical superiority of time-step flow over scalar regression Fundamentally, traditional scalar learning treats value estimation as a regression problem, minimizing prediction error to approximate the conditional expectation $\mathbb{E}[Y|X]$. In complex environments with high variance or multi-modality, this averaging effect leads to *distribution collapse*, where the model outputs a safe mean value that discards critical information regarding risk and diversity Zhu et al. (2025b). In contrast, time-step flow modeling learns a continuous evolution process rather than a static target. By modeling a time-dependent vector field $v(z, t)$, the flow defines a diffeomorphism that continuously deforms a simple noise distribution into the complex target distribution via ODE integration Dong et al. (2025). This topological flexibility allows the model to preserve high-order statistical moments (e.g., variance and skewness) and faithfully represent disjoint, multi-modal return landscapes that scalar models inherently fail to capture.

Single-step flow inference as a robust global approximation While our empirical results favor single-step inference, this approach differs fundamentally from standard single-step direct sampling (e.g., direct diffusion generation or regression). Standard direct sampling often attempts a point-to-point fitting of noisy targets, leading to overfitting of high-frequency reward noise and poor generalization. Conversely, the single-step inference in DFPO represents a global expectation approximation of the entire flow trajectory. By imposing a geometric consistency constraint, we encourage the vector field to align along straight paths across all time steps t (Prop. B.3 and the accompanying remark). Mathematically, this regularizes the initial velocity $v(z, 0)$ toward the global average velocity pointing to the target, so that single-step projection acts as a low-pass filter over the trajectory. Consequently, DFPO's single-step projection $z_1 = z_0 + v(z_0, 0)$ functions as a low-pass filter: it effectively captures the global trend of value evolution while filtering out the transient stochastic gradient noise accumulated during multi-step integration. This mechanism renders DFPO significantly more robust in noisy RL post-training regimes compared to standard multi-step solvers.

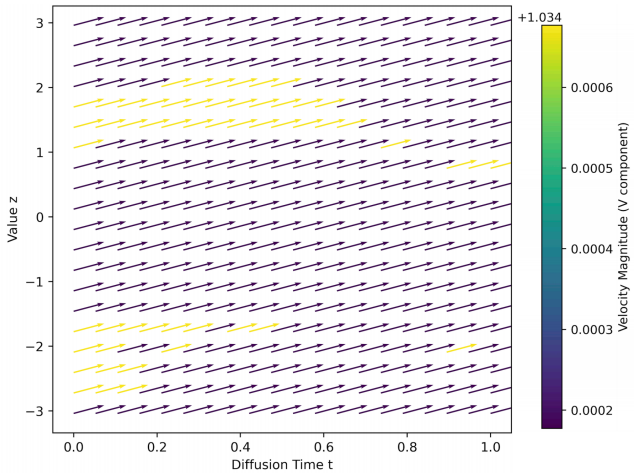

Figure 7: Visualization of the learned velocity vector field of one token for the answer part. The model encodes complex distributional structure through the fine-grained modulation of velocity magnitude (indicated by the color scale). As discussed, the vector field dynamically allocates speed based on spatial position, instructing particles in specific regions to accelerate (yellow/green) while neighboring particles decelerate (purple). Through this spatial divergence, the flow effectively compresses, stretches, and splits probability densities to reconstruct complex multi-peak return landscapes at the terminal state.

Modeling complex multi-peak distributions via velocity modulation and Optimal Transport A critical question is how straight-line trajectories can theoretically model complex multi-peak return distributions. The mechanism lies in the fine-grained modulation of velocity magnitude within the vector field. As shown in Figure 7, while trajectories are geometrically rectified to be straight, the vector field $v(z, t)$ dynamically allocates speed based on spatial position, for instance, instructing particles in the center to accelerate leftward while neighboring particles decelerate. Through this spatial divergence, the flow effectively compresses, stretches, and splits probability densities to reconstruct complex multi-peak landscapes. Lipman et al. (2023); Chen et al. (2019) Furthermore, we enforce Optimal Transport (OT) paths to ensure non-crossing and parallel dynamics. In ODE theory, crossing trajectories imply conflicting velocity directions at a single point, leading to numerical collapse; OT naturally selects a "non-entangled" transport plan (e.g., mapping left-side noise to left-side targets) Dupont et al. (2019), yielding a smooth, continuous vector field that is inherently robust. Mathematically, straight paths minimize the Wasserstein transport cost (energy) Liu et al. (2022), representing the simplest constant-velocity function for neural networks to approximate. This imposes a strong regularization that prevents the model from wasting capacity on chaotic, winding paths, allowing it to focus entirely on capturing the distribution's shape. Ultimately, this balance between constraint and expressivity serves as the bedrock for exploration: by ensuring that the predicted multi-peak structure and variance reflect true environmental uncertainty rather than approximation artifacts, the model provides high-fidelity signals for risk-aware decision-making.

DFPO maintains small computational overhead during training and inference DFPO introduces a small computational footprint during both training and inference. During training, because the underlying flow dynamics operate on a 1-dimensional value space, the Jacobian matrix required for flow sensitivity collapses into a single scalar derivative. This streamlined deterministic approach explicitly avoids the computationally expensive Monte Carlo sampling or Hutchinson trace estimators typically required for high-dimensional flow matching. This operation adds limited overhead relative to the standard forward and backward passes of the Transformer backbone. Measured under identical batch size, sequence length, and GPU count, DFPO increases peak GPU memory by approximately 0.7 GB per card and the average per-step training time by 1.7 seconds relative to the standard PPO baseline. During inference, DFPO employs a 1-Step Euler solver (NFE=1) for value projection. As validated in our ablation studies, this single-step evaluation generates reliable value samples in a single forward pass, bypassing the computational cost typically associated with iterative flow solvers.

Token-level value modeling ensures robust adherence to formatting constraints The formatting errors observed in Robust Bellman PPO, BAPO, and FlowRL during real-world dialogue evaluation can be attributed to their reliance on coarse-grained value supervision. Robust Bellman PPO optimizes the lower bound of scalar global returns, which provides a trajectory-level signal but lacks the token-level resolution necessary to enforce strict structural formatting. Similarly, BAPO applies global policy clipping without fine-grained regularization across the token sequence, making it susceptible to overlooking structural tokens under high noise. FlowRL models action-space reward distributions globally rather than tracking token-level value trajectories, resulting in supervision that struggles to stably guide the generation of specific formatting markers. In contrast, DFPO models continuous distributional value flows at the token level. This token-level supervision, combined with consistency constraints, provides the fine-grained guidance required to maintain adherence to structural and formatting constraints under noisy and out-of-domain conditions.

Mechanistic Analysis of Baseline Collapse and Formatting Degradation A striking empirical phenomenon observed in Table 1 and Table 7 is the catastrophic collapse of GRPO (accuracies dropping below 30%) and Dr.GRPO late in training, as well as evaluation failures in Robust Bellman PPO, BAPO, and FlowRL due to severe formatting errors. We provide a mechanistic explanation for these vulnerabilities:

1. **Why GRPO Suffers Severe Training Collapse:** GRPO entirely eliminates the independent critic model and computes advantages solely via sample normalization within small generation groups: $A_i = (R_i - \text{Mean}(R))/\text{Std}(R)$. Under severe, non-Gaussian real-world noise (e.g., noisy dialogue reward models or noisy majority-voting pseudo-labels), the intra-group mean and variance estimates become highly corrupted and stochastic Liu et al. (2026); Li et al. (2025). Lacking an independent, slowly-updating Value Model to serve as an expectation anchor, these randomized group advantages inject high-variance, contradictory gradient updates directly into the policy. This rapidly triggers reward hacking, the policy exploits perceived length shortcuts, over-generates unstructured tokens, and loses its pre-trained structural priors (e.g., misclassifying dialogue states or dropping critical conversation tags). PPO retains modest stability because its scalar critic absorbs a fraction of this variance, whereas DFPO stabilizes training by providing a continuous, risk-bounded value flow that acts as a robust gradient anchor.
2. **Why Coarse-Grained Baselines Suffer Formatting Failures:** Robust Bellman PPO optimizes pessimistic lower bounds on scalar trajectory returns, which provides only coarse, global supervision without token-level resolution. Similarly, BAPO applies global policy clipping without token-sequence regularizers, and FlowRL models distributions globally over action spaces rather than tracking token-level value trajectories. In complex multi-turn dialogue, adherence to strict task guidelines (such as generating domain tags, status indicators, and balanced turns) depends on high-precision advantage attribution on specific structural tokens. Under noisy supervision, coarse-grained baselines fail to recognize these critical structural tokens, treating them with uniform weights (as visualized in Figure 10 and Figure 11) or predicting extreme advantage spikes (up to 19.125). In contrast, DFPO models continuous value flows at the token level, precisely highlighting key words (Figure 13) while strictly preserving syntactic and formatting markers across out-of-domain transfers.

D Additional Visualization

Figures 10, 11, 12, and 13 illustrate a comparative analysis of advantage estimations among PPO, Robust Bellman PPO, Standard Distributional Value Flow Modeling PPO, and our proposed DFPO method for the same output sequence. Notably, significant discrepancies in advantage estimation patterns and semantic sensitivity are observed across the four methods. For PPO (Figure 10), the method fails to capture key semantic information, treating most tokens with uniform importance, which reflects its limited capability in distinguishing critical decision points. Similarly, the Robust Bellman PPO method (Figure 11) also fails to recognize key semantic tokens. Furthermore, it demonstrates severe instability in value estimation, characterized by excessive fluctuations and significant overestimation, with advantage values peaking at an abnormal high of 19.125. The Distributional Value Flow Modeling PPO method (Figure 12) shows improvement by successfully identifying and highlighting key words. However, it falls short in the precise regulation of advantage values, lacking fine-grained control over the numerical details, which may lead to suboptimal policy

updates. In contrast, our DFPO method (Figure 13) demonstrates superior performance. It not only accurately captures key semantic terms (e.g., "maximum" and "mass") by assigning distinct and interpretable advantage weights but also achieves fine-grained control over the advantage values. This balance ensures that the advantage signals are both semantically meaningful and numerically stable, validating the effectiveness of our proposed risk-controlled flow modeling.

E LIMITATIONS

While DFPO demonstrates strong robustness and generalization under noisy supervision and OOD scenarios, several limitations remain. First, the optimal configuration of the composite objective function, specifically the balance between flow matching terms ($\lambda_{\text{reg}}, \lambda_{\text{cons}}$) and distributional risk constraints ($\lambda_{\text{risk}}, \lambda_{\text{shape}}$) may vary across different domains, requiring task-specific tuning to achieve the best trade-off between exploration and stability. Second, although the 1-step Euler solver significantly accelerates inference, the assumption of straight-line trajectories relies heavily on the convergence of the geometric consistency loss; in scenarios with extremely highly non-linear value dynamics, the approximation error might increase. Finally, while DFPO improves stability under moderate noise via sensitivity weighting and anchor regularization, extreme reward corruption or completely noisy supervision can still degrade performance, as the flow field may eventually collapse to erroneous attractors.

Table 16: Trajectory-level generalization performance in the dialogue task across different methods and training steps. We report Task Completion Rates (TCR), Ask Completion Rates (ACR), and Goal Completion Rates (GCR) for each method. Our method maintains high performance even in last stages, beyond GRPO, PPO, Reinforce++, and Dr.GRPO.

Method	**Step**	**TCR**	**ACR**	**GCR**	**AVG**
Cold Start	0	71.5%	94.5%	92.8%	86.3%
GRPO	250	35.6%	**28.2%**	**34.9%**	**32.8%**
	500	41.2%	16.5%	28.5%	28.8%
	1000	41.4%	15.9%	26.1%	27.8%
	Last	**43.1%**	15.1%	26.9%	28.4%
PPO	250	**75.8%**	**96.9%**	**93.9%**	**88.8%**
	500	71.8%	93.3%	90.5%	85.2%
	750	49.7%	78.5%	76.8%	68.4%
	1000	42.9%	70.0%	71.8%	61.5%
	Last	42.1%	67.7%	70.4%	60.1%
Reinforce++	250	**72.9%**	**97.7%**	**92.8%**	**87.8%**
	500	63.9%	96.7%	90.9%	83.9%
	750	64.2%	95.1%	89.6%	83.0%
	1000	61.3%	93.3%	88.9%	81.1%
	Last	63.7%	94.9%	89.9%	82.8%
Dr.GRPO	250	**60.2%**	**82.2%**	**85.0%**	**75.9%**
	500	38.1%	62.3%	68.8%	56.3%
	750	33.5%	55.5%	60.4%	49.8%
	1000	34.6%	53.7%	56.9%	48.4%
	Last	34.1%	51.5%	54.9%	46.9%
OURS	250	72.6%	94.9%	92.3%	86.6%
	500	71.4%	**95.7%**	92.4%	86.5%
	750	72.4%	94.3%	**92.8%**	86.5%
	1000	**73.8%**	95.4%	**92.8%**	**87.3%**
	Last	73.0%	94.3%	92.6%	86.7%

System Prompt

Role: You are the AI call assistant of xxx Inc., capable of answering phone calls clearly and politely on behalf of the user. Based on predefined dialogue goals, you should first identify the appropriate goal according to the caller's intent and then respond accordingly. You do not have access to any personal information about the user beyond the task description, and you must inform the caller that any provided information will be conveyed to the user.
Dialogue Goals (Customized):

- **0: General Domain** – Used when the call does not belong to a specific domain. Maintain politeness and professionalism, clarify the caller's intent.
- **1: Medical & Health (Treatment Adjustment)** – Express concern about medication changes, inquire about alternatives and scientific evidence, and ask about reports or prescription updates.
- **2: Life Services** – If service completion is reported, express dissatisfaction, request compensation, and ask about service details and response time.
- **3: Medical & Health (Insurance Policy Update)** – Express satisfaction with policy changes, notify the user, raise concerns about reimbursement of specific medications, and inquire about the conditions and follow-up services of newly covered dental examinations.

Caller History and New Input

Conversation History:
Caller: Hello, this is the Municipal Medical Insurance Service Center. We would like to inform you that there have been recent adjustments to the medical insurance policy, including the removal of a certain application process and the addition of dental examinations to the reimbursement list. These changes will officially take effect at the end of this month. Please also note that some specific surgical procedures will no longer be covered.
Assistant: Hello, this call is currently handled by the AI assistant on behalf of the user. May I ask whether there are any changes to the reimbursement policy for specific medications?

Caller New Input: Don't worry, there are no changes to the reimbursement policy for specific medications. The adjustments mainly concern surgical procedures. Medication-related updates were already covered in last month's notice.

Model Response

Target Domain: *3: Medical & Health (Insurance Policy Update)*
Dialogue State: *Ongoing*
Response: Thank you for the clarification. Could you please provide more details about the specific conditions and follow-up services for the newly covered dental examination reimbursement?

Figure 8: A multi-turn example from the Honor-Dialogue dataset under the medical insurance policy update scenario. The model correctly identifies the dialogue goal and maintains an ongoing conversation state while proactively requesting detailed conditions for newly introduced reimbursement items.

Core Prompt for Response Content Relevance

This task evaluates the content relevance of the callee's latest response in an incoming call scenario. The callee is expected to communicate clearly and naturally with the caller while actively collecting useful information.
When scoring the callee's latest response, please follow the steps below:
Step 1. Read and understand the full dialogue history and the callee's latest response.
Step 2. Score from the dimension of content relevance. Content relevance is defined as whether the callee's latest response is directly related to the caller's statements or questions, and whether it provides a reasonable and context-aware reply based on the caller's input. Use a 1–5 scoring scale with the following criteria:

- **1 point:** The response is completely irrelevant to the caller's input and dialogue context. It fails to address the caller's statement or question and provides unrelated or meaningless information.
- **2 points:** The response is weakly related. Although it stays within the same general scenario, it contains noticeable irrelevant content or fails to properly respond to the caller's main intent.
- **3 points:** The response is partially relevant. It aligns with the general scenario, but the callee misunderstands the caller's key intent, shifts focus away from the main issue, or guides the caller toward an action that is reasonable but beyond the scope of the current conversation.
- **4 points:** The response is mostly relevant. It addresses the caller's input and provides additional useful information, though the suggested action or guidance may not fully comply with business requirements.
- **5 points:** The response is fully relevant. It directly addresses the caller's content, strictly follows business requirements, and all information provided is appropriate and consistent with the current dialogue context.

Step 3. Based on the above criteria and the full dialogue context, assign a reasonable score to the callee's latest response only. Only output the final score in the required format without any additional explanation.

Figure 9: A core prompt for response content relevance evaluation in intelligent call assistant scenarios. The rubric provides a clear definition of content relevance, explicit 1–5 point scoring criteria, and step-by-step evaluation instructions, enabling consistent and fine-grained assessment of the callee's latest response.

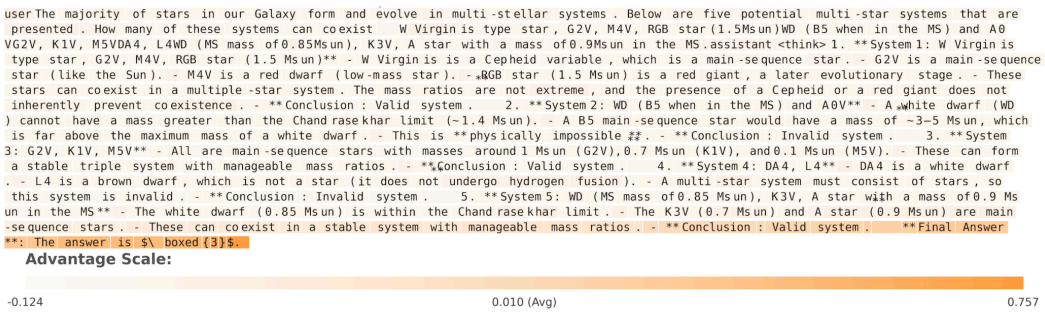

Figure 10: **Advantage estimation visualization of the PPO method.** The method fails to effectively highlight or capture key semantic tokens within the generated response.

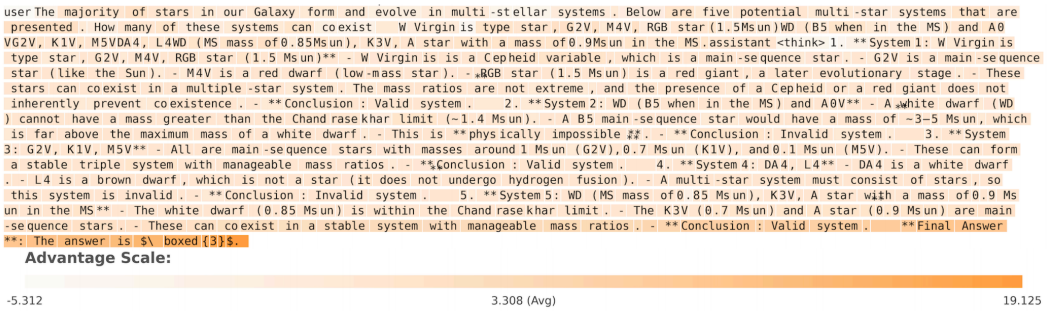

Figure 11: **Advantage estimation visualization of the Robust Bellman PPO method.** The method demonstrates significant instability and fails to localize key semantic terms. It exhibits extreme fluctuations in advantage values, with peak overestimation reaching as high as 19.125.

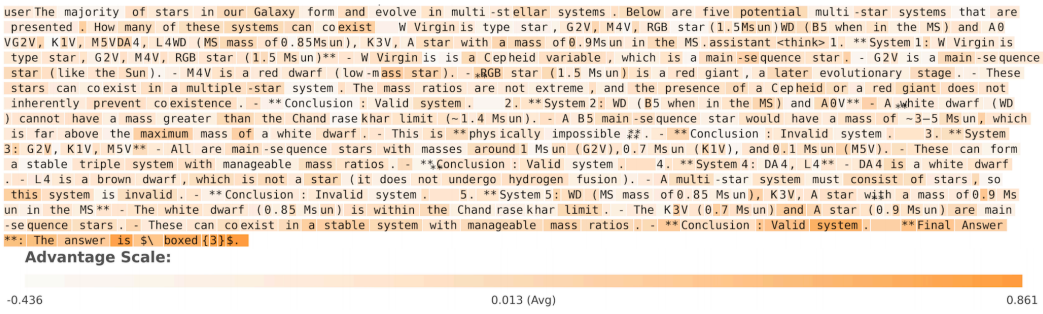

Figure 12: **Advantage estimation visualization of the Standard Distributional Value Flow Modeling PPO method.** While this method successfully identifies key words, it lacks fine-grained control over the numerical magnitude of advantage estimates, resulting in suboptimal value precision.

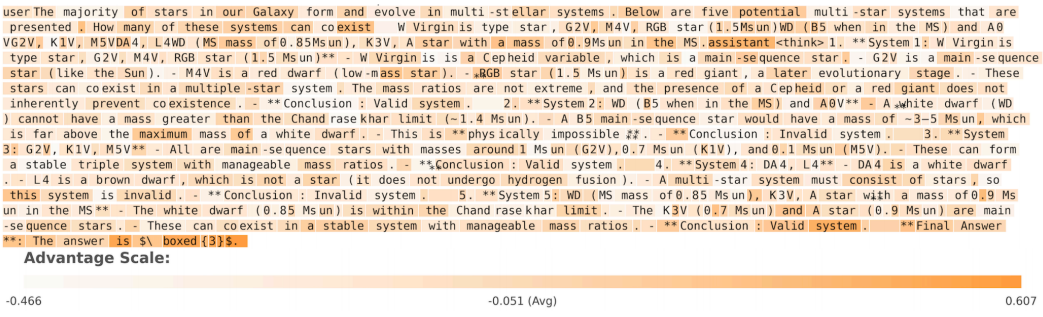

Figure 13: **Advantage estimation visualization of our proposed DFPO method.** The method accurately captures key semantic terms (e.g., "maximum" and "mass") via distinct advantage weights while simultaneously achieving fine-grained control over the value details, demonstrating superior stability under noisy supervision.