USING REWARD UNCERTAINTY TO INDUCE DIVERSE BEHAVIOUR IN REINFORCEMENT LEARNING

Anonymous authors
Paper under double-blind review

ABSTRACT

Classical reinforcement learning (RL) typically seeks a deterministic policy that maximizes the expected sum of a scalar reward. Yet, modern applications such as language model fine-tuning or scientific discovery demand diversity. Existing remedies such as entropy regularization or diversity bonuses often require fragile trade-offs that sacrifice performance for stochasticity or rely on heuristic metrics that can misalign policy rankings. We argue that diversity is more naturally understood as the rational response to *uncertainty* in the reward. When the reward function is not perfectly known—as is the case with ambiguous preferences or imperfect reward models—committing to a single action can be sub-optimal. Building on this, we propose a fundamental reformulation of the RL objective by replacing the scalar reward with a distribution over reward functions, and applying a non-linear objective over *sets* of actions. The result is a framework in which *calibrated* behavioural diversity emerges naturally, remains controllable through the reward function distribution, and is obtained without sacrificing expected reward. Focusing on the contextual bandit setting as commonly used in large language model (LLM) post-training, we derive a principled gradient estimator for this objective and prove that our formulation naturally generalizes both vanilla policy gradient and more recently developed action-set approaches. We provide didactic experiments which complement our theoretical results, and our large-scale empirical results in LLM reasoning further demonstrate that this framework offers a robust and theoretically grounded alternative for complex RL tasks where the traditional formulation of the problem fails to induce the desired breadth of agent behaviour.

1 INTRODUCTION

Classical reinforcement learning (RL) theory rests on the foundational assumption that preferences can be captured by the expected sum of a scalar reward function. A well-known consequence of this assumption is the existence of an optimal deterministic policy (Puterman, 1994), and, accordingly, most standard RL methods are designed to converge to such a solution. However, as RL is increasingly applied to open-ended domains, convergence to a single behaviour has become a significant limitation. In settings such as the fine-tuning of large language models (LLMs), it is often essential to maintain the diversity of the pre-trained policy in order to preserve its usefulness for creative tasks and chain-of-thought reasoning (Kirk et al., 2024; Chung et al., 2025; Ismayilzada et al., 2025; Cui et al., 2025; West & Potts, 2025; Zhao et al., 2025; Shypula et al., 2025; Yang & Holtzman, 2025; Yun et al., 2025). Beyond language, the capacity for inference-time exploration is vital for scientific discovery, where agents must navigate vast search spaces to identify rare, high-value solutions (Romera-Paredes et al., 2024; Hubert et al., 2026; Aygün et al., 2025). Furthermore, convergence to a deterministic optimal policy is detrimental when the reward model itself is only an imperfect proxy for some true underlying preference, and over-optimization results in the deterioration of generation quality (Gao et al., 2022; Coste et al., 2024; Moskovitz et al., 2024; Rafailov et al., 2024; Lambert, 2026).

Existing approaches for inducing diversity include using entropy regularization or diversity bonuses. Both families of approaches can be thought of as modifications to the original reward function, resulting in policies that are highly sensitive to the explicit choice of diversity measures, and must be carefully tuned. A variety of approaches make use of policy entropy penalties (Williams & Peng, 1991; Mnih et al., 2016; Todorov, 2006; Ziebart et al., 2008; Haarnoja et al., 2017), although this forces a trade-off, in which a more stochastic policy can be obtained at the cost of reduced expected

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.

rewards (Jhaveri et al., 2025). Diversity bonuses that get added to the rewards similarly introduce a tension between optimizing the original reward function and focusing on the bonus (Li et al., 2025; Hamid et al., 2026; Orney et al., 2026), and may result in undesirable ranking of sub-optimal policies, as we discuss further below.

In this work, we propose a shift in perspective: diversity should be understood as the rational response to *reward uncertainty*—whether arising from ambiguity in under-specified user preferences, or from the epistemic gap between an imperfect reward model and the true underlying preference—so that the learning objective is more holistically described by a *distribution* over reward functions, rather than by augmenting a single reward function. However, standard policy gradient updates over such a distribution collapse the policy to average, non-specialized behaviour. In response, we propose policy gradients that depend non-linearly on *sets of actions* (Koyamada et al., 2022; Tang et al., 2025; Hamid et al., 2026) and sampled reward functions; while set-based updates have previously been explored to promote exploration in Markov decision processes (Koyamada et al., 2022), we provide an in-depth analysis as applied to generative modelling. Concretely, focusing on the contextual bandit setting, we derive a principled gradient estimator for this objective, prove that our formulation naturally generalizes both vanilla policy gradient and recently developed action-set approaches, and demonstrate empirically that the resulting policies flexibly learn diverse behaviours across the reward function distribution, rather than collapsing onto optimal actions for the most frequently encountered reward function.

Our core contributions are as follows:

- We introduce a new family of RL objectives that provides calibrated diversity control through a distribution over reward functions, which we refer to as *Randomized Objectives, Set Actions* (ROSA) (Sec. 3).
- We provide an analysis of this new family of objectives, both in terms of the global optimizing policies and the landscape of the objectives themselves. This simultaneously provides theoretical guarantees for the objectives and informs useful hyperparameter settings in practice (Sec. 4).
- We complement this analysis with a variety of empirical studies, both in toy domains, to develop a fine-grained understanding of the method's performance, and in larger-scale domains, to demonstrate the broad applicability of the proposed objectives in practical scenarios (Sec. 5).

2 Background

We consider a contextual bandit scenario (Langford & Zhang, 2007; Sutton & Barto, 2018) defined over action space $\mathcal{Y}$ and state space $\mathcal{X}$. This encompasses both abstract bandit problems, as well as large-scale generative modelling such as single-turn fine-tuning of language models. A typical RL problem is specified via a reward function $R : \mathcal{X} \times \mathcal{Y} \to \mathbb{R}$, and the goal is to optimize a policy $\pi : \mathcal{X} \to \mathscr{P}(\mathcal{Y})$ according to the **mean reward criterion** for some state distribution μ,

$$\mathcal{J}(\pi) = \mathbb{E}_{X\sim\mu, Y\sim\pi(\cdot|X)}\big[R(X,Y)\big]\,. \tag{1}$$

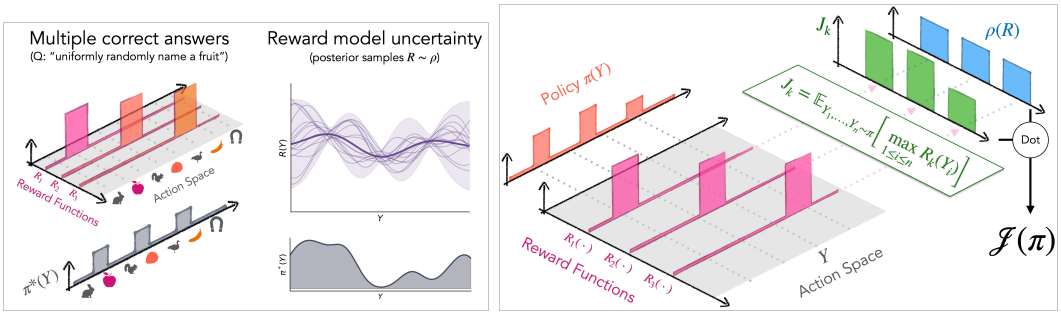
Policy gradient (PG) methods are common means of optimizing the policy π with respect to the objective in Eq. (1). The prototypical PG method is REINFORCE (Williams, 1992); given a parametrized policy π_θ and a state–action sample (X, Y) drawn according to π_θ, the algorithm updates the parameters θ in the direction

$$\nabla_\theta \mathcal{J}_{\text{PG}}(\pi_\theta) = \mathbb{E}_{X\sim\mu, Y\sim\pi_\theta(\cdot|X)}\big[R(X,Y)\nabla_\theta \log \pi_\theta(Y|X)\big]\,. \tag{2}$$

3 Optimizing behaviour for distributions of reward functions

Our goal is to develop a policy gradient method that learns diverse behaviours capable of reflecting three key criteria: (a) uncertainty in the objective function; (b) variance in human judgment regarding what constitutes a high-quality action (e.g., scenarios where a single query yields multiple valid answers); and (c) a strategic desire to maintain policy flexibility for future fine-tuning, such as adapting to individual user preferences. The existing families of approaches for inducing diversity fall short of these criteria in complementary ways: *entropy regularization* (Williams & Peng, 1991; Ziebart et al., 2008; Haarnoja et al., 2017) prevents convergence to a deterministic policy, but its optimum places non-zero mass on *all* actions—including very poor ones; *diversity bonuses* (Li et al., 2025;

Hamid et al., 2026) similarly retain poor actions in some variants, and both can prefer a lower-reward (but higher-diversity) policy over one with higher average reward; and *multi-objective RL* (Roijers et al., 2013; Hayes et al., 2022) optimizes scalarized objectives whose optima are either deterministic (expected scalarized returns—like vanilla policy gradient and pass-at-k/best-of-n gradients) or lack unbiased sample-based gradient estimators (scalarized expected returns, $s(\mathbb{E}[\cdot])$). We validate these failure modes numerically in Section 3.2 (Figure 2), summarize them in Table 1, and defer detailed discussion of related work to Appendix A.



(a) Examples of distributions of reward functions and the desired policy. **(Left)** A binary reward setting with a variety of correct actions; we prefer a policy that places probability mass on all correct actions. **(Right)** We only have access to approximate reward functions ($\hat{R}$'s, shaded lines). There is epistemic uncertainty about the value of the true underlying reward (bold line), and we wish to avoid overfitting to the approximate $\hat{R}$.

(b) ROSA+Max optimizing a policy for a distribution of reward functions). Given some policy (orange) and reward functions R_k (pink), we first compute the policy's max-of-n performance for *each* reward function, J_k (green). We then take the dot product with the reward function probabilities ($\rho(R_k)$, blue) to get the ROSA+Max score. Optimizing this score calibrates the policy to the reward function uncertainties. Appendix A contains more details and comparisons with regular PG.

Figure 1: Many complex tasks are naturally expressed as distributions over reward functions.

3.1 The Randomized Objectives, Set Actions (ROSA) criterion

We now make our core algorithmic proposal, aiming to close the drawbacks described in the previous section. We build on the approach of working with multiple reward functions that express our desired diverse behaviours, and develop an approach that (i) naturally allows for arbitrary distributions over reward functions, (ii) avoids inducing learnt policies to take bad actions solely for the sake of increased diversity, (iii) has straightforwardly-implementable, unbiased policy gradient estimates. The key idea of our proposal is to generalize Eq. (1) by considering (i) *multiple* actions sampled i.i.d. from the policy π (Tang et al., 2025; Hamid et al., 2026); and (ii) a *distribution* ρ over reward functions R. This gives rise to the family of ***R**andomized **O**bjectives, **S**et **A**ctions* (**ROSA**) criteria.

Formally, let $\rho \in \mathscr{P}(\mathbb{R}^{\mathcal{X}\times\mathcal{Y}})$ be a distribution over reward functions, and let $\mathbf{Y} = (Y_i)_{i=1}^n$ denote a *multiset* of actions induced by $Y_1, \ldots, Y_n$. We use tuple notation to signify multiplicities are retained, but any orders are ignored. We use the shorthand $\mathbf{Y} \sim \pi_\theta$ to denote the n actions are sampled i.i.d. from policy π_θ. Let $R \sim \rho$ be a sample from the distribution over reward functions. For brevity, we denote the reward multiset as $R(\mathbf{Y}) = (R(Y_i))_{i=1}^n$. While our general objective makes use of a set function $f : \mathbb{R}^n \to \mathbb{R}$ (general case in Section 4.1), we begin by studying the $f = \max$ set function,

$$\mathcal{J}_{\text{ROSA+Max}}(\pi_\theta) = \mathbb{E}_{X\sim\mu,\, \mathbf{Y}\overset{\text{i.i.d.}}{\sim}\pi_\theta(\cdot|X)}\left[\mathbb{E}_{R\sim\rho}\left[\max_{1\le i\le n} R(X, Y_i)\right]\right], \tag{3}$$

where the $\max$ function aggregates over the reward of the n actions in a permutation-invariant way for each reward function. Figure 1b contains an example in the case of a discrete set of reward functions.

Intuitively, the ROSA+Max criterion provides an *asymmetric* summary of performance on each sampled reward function, encouraging *at least one* action in the multiset to perform well in each case. This is related to the expected utilities used in multi-objective RL, described above, but differs in several important ways: (i) the form of optimism used is strictly more expressive than that typically used in MORL (see Appendix A) leading to more desirable optimal policies, (ii) the objective has a

straightforwardly derivable unbiased policy gradient estimator (see below), and (iii) the objective naturally supports arbitrary distributions over reward functions (rather than an explicitly declared finite number of reward functions). Additionally, Koyamada et al. (2022) and Nishimori et al. (2026) propose the ReMax objective which coincides with ROSA+Max in the bandit setting and is similarly motivated by an observation about the induced optimal policy becoming stochastic. They focus on exploration algorithms for MDPs without explicit reward/value distributions, rather than policy diversity for generative modelling (see Appendix A.1 for further discussion). As we will demonstrate in Section 3.2, a key consequence of the ROSA+Max objective is that its optimal policy assigns action probabilities over the set of actions that are optimal under *each* $R \sim \rho$. Finally, note that the distribution over *reward functions* in ROSA differs from the distribution over *rewards / returns* modelled in distributional RL (Chung & Sobel, 1987; Morimura et al., 2010; Bellemare et al., 2023); we discuss their relationship in Appendix A.2.

We now derive an unbiased, variance-reduced policy gradient estimator to optimize our proposed objective. Given m sampled reward functions $(R_k)_{k=1}^m$ from ρ, and n sampled actions $(Y_i)_{i=1}^n$ from $\pi_\theta(\cdot|X)$, an unbiased gradient estimator of Equation (3) is given by:

$$\hat{g}(\theta) = \frac{1}{m}\sum_{k=1}^{m}\left[\sum_{i=1}^{n}\Big(\max_{1\le j\le n} R_k(X,Y_j) \;-\; \max_{\substack{1\le j\le n,\\ j\neq i}} R_k(X,Y_j)\Big)\nabla_\theta \log \pi_\theta(Y_i|X)\right], \tag{4}$$

which has the effect of only the maximal reward action contributing a non-zero term to the estimator. Note the term $\max_{1\le j\le n,\, j\neq i} R_k(Y_j)$ is an optional control variate for variance reduction, which is analogous (in the single reward function setting) to the estimator used in Tang et al. (2025). We provide derivations for a generalization of the above gradient estimator (for general set functions and control variates) and all other theoretical results in the main paper in Appendix B.

Remark 3.1. While set-function objectives based on best-of-n have been investigated in the single-reward setting as a means of inducing generative diversity (see, e.g., Tang et al. 2025; Walder & Karkhanis 2026), we emphasize such formulations do not induce a stochastic optimal policy—a deterministic optimal policy always exists for the single reward function. By contrast, in our setting, the combination of randomized reward functions *and* set function objectives induces an optimal stochastic policy, which balances the incentives of the randomized reward functions.[1]

3.2 Theoretical analysis of ROSA+Max

Having introduced the ROSA+Max update in Eq. (4), we now examine it theoretically. Without loss of generality, we consider the single-state case, and drop state-dependence from our notation. Our theoretical analysis will assume each reward function $R \sim \rho$ is binary and "one-hot", assigning reward 1 to a distinct optimal action $y^* \in \mathcal{Y}$ and 0 otherwise. We note that the ROSA objective and gradient estimator support arbitrary reward functions in practice.

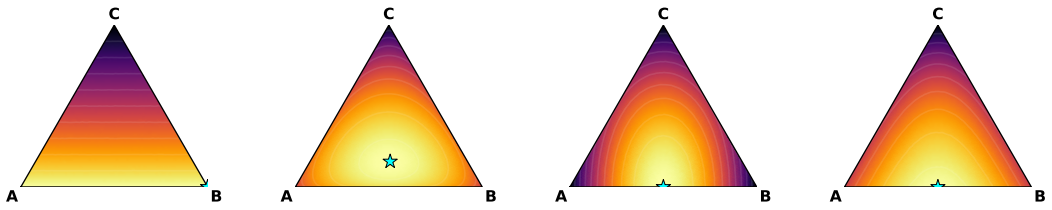
Proposition 3.2. ***Optimal policy of ROSA+Max with uniform reward function distributions.*** *Consider m binary reward functions $(r_i)_{i=1}^m$, each with a single distinct optimal action $y_i^* \in \mathcal{Y}$, and set ρ uniform over $(r_i)_{i=1}^m$. Then, writing δ_y for the Dirac delta distribution at y, the ROSA+Max objective with any action set size $n \ge 2$ has unique optimal policy*

$$\pi^* = \sum_{i=1}^{m}\frac{1}{m}\delta_{y_i^*}\,.$$

This result verifies that the ROSA+Max objective achieves our intended goal of having the maximally diverse policy over correct responses at its unique optimizer. To develop intuition further, we numerically simulate the global objective landscape over all policies in an example with two reward functions (Figure 2). We compare ROSA with other diversity-inducing RL objectives, as well as the standard PG objective (Figure 2a), that does not induce any kind of diversity among optimal actions.

Firstly, adding entropy regularization to a standard RL objective (Todorov, 2006; Ziebart et al., 2008; Haarnoja et al., 2017) changes the ordering of preferred policies to favour non reward-maximizing

[1]This observation, in the case of the max set function, was also made by Koyamada et al. (2022); Nishimori et al. (2026), along with an unbiased max-at-k gradient estimator. However, these works do not develop it in the generative modelling setting. See Appendix A.1 for further discussions.



(a) Vanilla PG (b) Entropy Reg (c) Mult. Diversity (d) ROSA+Max

Figure 2: Simplices of global objective landscapes over 3-action categorical policies for four RL objectives. Actions `A` and `B` receive reward +1, and `C` receives reward 0. Details in Appendix D.1.

policies with higher entropy (Figure 2b), and disproportionally favour actions with minor improvements in rewards (GX-Chen et al., 2026). We also consider diversity bonuses (Li et al., 2025; Hamid et al., 2026; Orney et al., 2026), such as in the form of a multiplicative diversity bonus, e.g., $\bar{r}(X, Y_i) = r(X, Y_i)\text{Div}(X, \mathbf{Y})$ where $\text{Div}(X, \mathbf{Y})$ is the average distance to all other samples in the action multiset. This family of objectives can induce a global optimum near/at a reward maximizing policy, but re-orders sub-optimal policies where a high-diversity, low-reward policy can be favoured over a high-reward policy (Figure 2c). Finally, ROSA+Max does not have either of these issues: its gradient points solely in the direction of both higher reward and higher action diversity (Figure 2d). It should be noted that both "diversity bonus" methods and ROSA require more information than the original reward function, while entropy regularization does not; diversity bonus requires a diversity function, while ROSA requires a *distribution* of reward functions.

Finally, it is interesting to note that the action sampling parameter n in Prop. 3.2 does *not* affect the optimal policy: *any* $n \geq 2$ induces a maximally diverse policy over all reward functions, so the global optimum can be obtained with small parallel sampling budgets (e.g., just $n = 2$ i.i.d. action samples from π_θ). In practice, however, the choice of n matters for algorithmic performance: analyzing the curvature of the objective at the optimal policy suggests $2 \leq n < 2m$ as a sensible hyperparameter range (when m is known); see Appendix C.1 for the analysis and visualization.

4 Beyond the Max: ROSA with General Distributions and Set Functions

Having established a result on the optimal solution in the case of uniform distribution over rewards and the $\max$ set function in Proposition 3.2, we now generalize it in two ways: to a general family of set functions, and to non-uniform reward function distributions.

4.1 A Family of Set Functions with Maximally Diverse Optimal ROSA Policies

So far, we have shown that the ROSA+Max criterion has the desirable property of having an optimal policy that places mass equally over all optimal answers. The reader may wonder if set functions other than "max" can be used to aggregate over rewards $(R(Y_i))_{i=1}^n$. Here, we present an additional theoretical result which complements Proposition 3.2 and establishes this optimal policy guarantee for a more general family of set functions.

We focus our analysis on the binary reward setting. Under binary rewards, any general multi-set function $f(R(X, Y_1), \ldots, R(X, Y_n))$ can be simplified into a *success-count reward function* $\tilde{f}(\sum_{i=1}^n R(X, Y_i))$ depending only on the *success count* $\sum_{i=1}^n R(X, Y_i)$; because the individual rewards are binary, the function's value is determined entirely by the number of correct actions.

Theorem 4.1. ***Optimal policy for general*** f. *Consider m uniformly distributed (binary, distinct) reward functions $(r_i)_{i=1}^m$, each with a single distinct optimal action $y_i^* \in \mathcal{Y}$, and a set function f, with corresponding success-count reward function $\tilde{f}$ (defined above) which is strictly increasing as well as strictly concave in the sum of rewards. Then the ROSA+f objective has a global optimum which samples correct actions from all reward functions uniformly,*

$$\pi^* = \frac{1}{m} \sum_{i=1}^{m} \delta_{y_i^*} .$$

Theorem 4.1 allows the construction of many different kinds of set functions which all have the same optimum at the maximally diverse policy, yet can have vastly different optimization properties. One function from this family is the Softmax, which is a natural extension of the max that varies smoothly between f_{max} and f_{mean},

$$f_{\text{Softmax}}\left(\left(R(X,Y_i)\right)_{i=1}^n\right) = \sum_{i=1}^n \frac{\exp\left(R(X,Y_i)\right)}{\sum_{j=1}^n \exp\left(R(X,Y_j)\right)} R(X,Y_i)\,. \quad (5)$$

Numerical simulations in Figure 3 illustrate that, like the Max, the Softmax also has a maximally diverse global optimum, though with different optimization landscape. Notably, the Softmax function also removes the sparsity of the variance-reduced gradient in Eq. (4) in the case of the max set function. We leave exploration of other set functions for future work and focus primarily on f_{Max} and f_{Softmax} in the next sections. See also Verdun et al. (2025) for the use of softmax distributions as an alternative to best-of-n sampling in the single-reward setting.

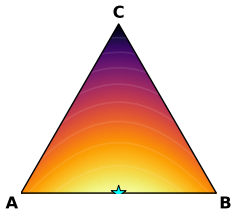


Figure 3: ROSA + Softmax simplex in same setting as Figure 2.

4.2 Non-uniform reward function distribution induces controllable diversity

We now consider non-uniform weightings in the reward function distribution ρ. In this setting, the relationship between ρ and optimal policy action probabilities is more complex. We derive a specific result for the ROSA+Max objective and leave the extension to other set functions for future work.

Proposition 4.2. *Consider m distinct, binary reward functions $(r_i)_{i=1}^m$ (each with a single distinct optimal action $y_i^* \in \mathcal{Y}$) with probabilities $(\alpha_i)_{i=1}^m$ under ρ, and assume without loss of generality that $\alpha_1 \geq \cdots \geq \alpha_m$. The ROSA+Max objective with $n \geq 2$ action samples has an optimal policy which samples optimal actions from the top-k reward functions (ordered by α's), each with probability,*

$$p_i^* = 1 - \frac{(k-1)\,\alpha_i^{-1/(n-1)}}{\sum_{j=1}^k \alpha_j^{-1/(n-1)}}\,, \quad (6)$$

where k is the largest integer between 1 and m where p_k^* is positive.

Proposition 4.2 allows us to *predict*, given reward function probabilities $(\alpha_1, ..., \alpha_m)$, how frequently the optimal policy under ROSA+Max will sample from *each* of the distinct correct answers (corresponding to each reward function). This gives us the ability to precisely predict and/or control the diversity in the optimal policy through the reward function distribution. Figure 4 numerically illustrates this point: the ROSA+Max optimum smoothly interpolates between uniformly sampling optimal actions, to preferentially sampling the optimal action under the more probable reward function.

Algorithm 1 ROSA+Max single-state policy update

Require: Policy π_θ, state X, reward functions $\{R_k\}_{k=1}^m$ each with probabilities $\{\alpha_k\}_{k=1}^m$.

1: Sample actions $Y_1, \ldots, Y_n \overset{\text{i.i.d.}}{\sim} \pi_\theta(\cdot \mid X)$.
2: **for** each action Y_i and reward function R_k **do**
3: Calculate the effective advantage:

$$A_i^{(k)} = \max_{1\leq j\leq n} R_k(X,Y_j) - \max_{\substack{1\leq j\leq n \\ j\neq i}} R_k(X,Y_j)$$

4: **end for**
5: Compute gradient estimator:

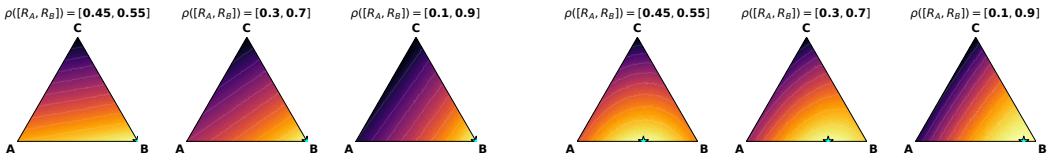
$$\hat{g} = \sum_{i=1}^n \Big(\sum_{k=1}^m \alpha_k \cdot A_i^{(k)}\Big) \nabla_\theta \log \pi_\theta(Y_i \mid X)$$

6: Update weights $\theta' \leftarrow \texttt{Optimizer}(\theta, \hat{g})$

5 Experiments

We now evaluate empirically how our proposed objective function, **ROSA+Max**, and its variant **ROSA+Softmax** compare with the standard policy gradient objective and other variants in experiments meant to illustrate a variety of real-world applications.

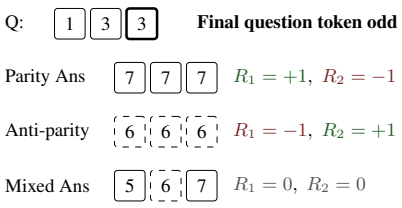
A practical ROSA+Max gradient update is provided in Alg. 1. Although we write it iteratively for clarity, it can be implemented efficiently via matrix operations (over n actions and m sampled reward functions). Note that the same algorithm can be applied to any (multi)set functions by replacing the $\max$ with the desired set function.



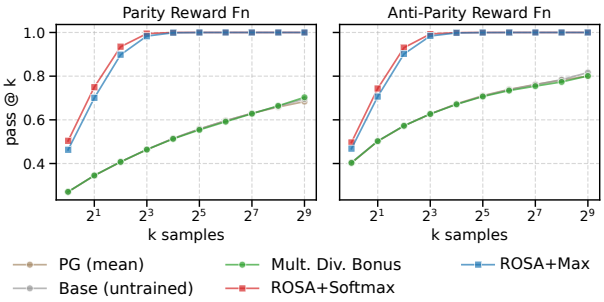
(a) Vanilla PG

(b) ROSA+Max

Figure 4: Global objective landscape over simplex of 3-category policies, with non-uniformly weighted reward functions R_A and R_B. Vanilla PG can only induce global optima at deterministic policies, while ROSA+Max naturally induces global optima at optimal stochastic policies. The probability of sampling each unique optimal action is characterized in Proposition 4.2.



(a) Sequence generation task with competing reward functions. R_1 prefers answers with the same parity as the last question token, R_2 prefers opposite parity.



(b) Evaluation result of different gradient estimators.

Figure 5: Comparison of vanilla PG, **ROSA+Max** and **ROSA+Softmax** in scenario of competing preferences. PG (even if augmented with a multiplicative diversity reward, `Mult. Div. Bonus`) suffers from reward cancellation, resulting in *no learning*—its pass@k curve is identical to the `Base` (untrained) model. However, ROSA+Max and ROSA+Softmax cleanly balance both preferences: starting at ∼50% accuracy and generating both answer types as k increases. Details in Appendix D.2.

5.1 Diverse, competing preferences

We first investigate the regime of **conflicting preferences**, where $R \sim \rho$ represents a distribution over diverse, contradictory reward functions. In this setting, vanilla policy gradient is fundamentally incapable of optimization: by aggregating the return as $\bar{r}(Y) = \mathbb{E}_{R\sim\rho}[R(Y)]$, opposing reward signals mutually cancel out, resulting in no reward.

We construct a didactic experiment with a small transformer outputting only integers. There are two competing reward functions: R_1 is $+1$ if all generated integers have the same parity (i.e. same odd/even-ness) as the last integer in the prompt, -1 if all generated integers have the opposite parity as the last integer in the prompt, and 0 if the generated integers have mixed parities. R_2 is the negative of R_1 (see Figure 5a for example and explanation). As Figure 5 demonstrates, ROSA is the only method capable of making any progress at all in this setting.

5.2 Multiple correct answers

Many tasks in the real world can naturally contain multiple correct answers (or correct actions). For instance, a coding task asking the user to "reverse a Python list" can be implemented in a number of syntactically distinct, yet functionally equivalent ways, or a task of discovering a drug having a high reward (defined by some properties) may have multiple drug molecules achieving the highest reward. In these scenarios, the user is not interested in finding a deterministic policy which achieves the highest reward on one solution but finding a stochastic policy which discovers many correct solutions.

We can model this scenario by defining a reward function distribution over multiple reward functions, each corresponding to a unique, correct answer. Note that optimizing the sum/average i.e. "joint" reward function $\bar{R}(Y_i) = \sum_{a\in\mathcal{A}^*} R_a(Y_i)$, reduces to a regular reward function that gives "1" to all correct answers $Y_i \in \mathcal{A}^*$, and 0 otherwise. We construct a setting in mathematical reasoning where we prefer answers that are *correct*, but we want diversity in the answer lengths: short answers are concise and to the point, while long answers can be more didactic and detailed. Specifically, we train

with four different reward functions with different length preferences. Figure 6 shows the output pass@k from sampling the resulting policies, where ROSA achieves the same overall correctness as regular policy gradient training, but is able to sample from all length preferences specified. We show short and long example responses from a ROSA trained policy in Figure 17.

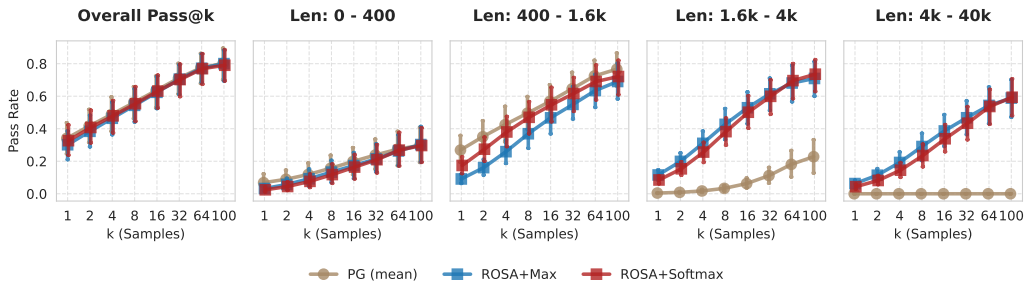


Figure 6: Comparison of ROSA+Max/Softmax to PG in a setting where we train on `MATH` and with multiple reward functions preferring different lengths. Each reward function gives 1 if the answer is both correct *and* within its preferred character length. Training with vanilla PG results in generations mostly in the 400-1.6k length range, while the ROSA-trained policy samples from all four preferred length ranges, without sacrificing any accuracy (overall pass@k rate).

5.3 Learning with unknown ground-truth rewards

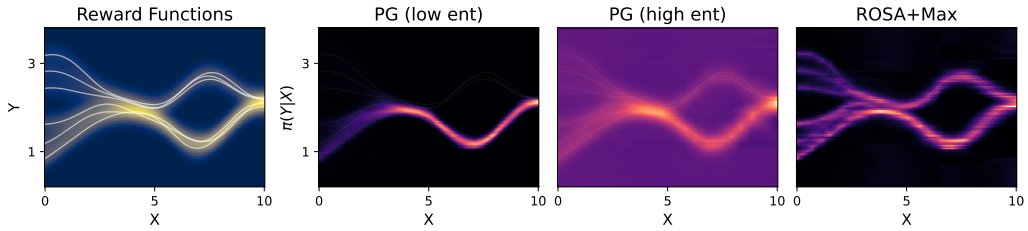
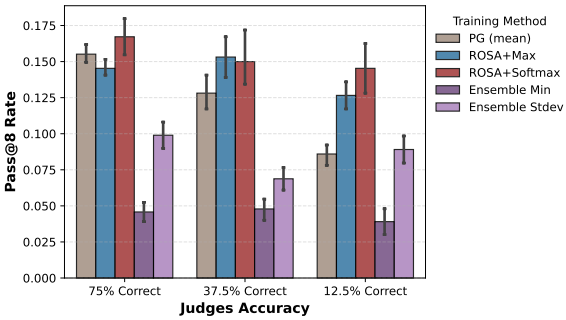


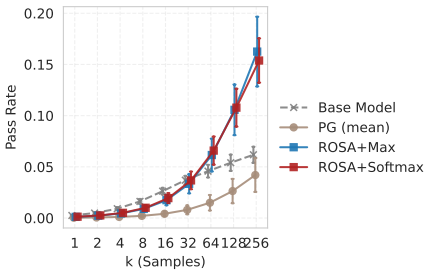
Figure 7: Left: Reward functions; white lines denote maximum reward y for each x, with rewards decaying exponentially away from the maximum. Right: Trained policy $\pi(Y|X)$.

We now consider the case where we are *uncertain* about the *true* reward function. This naturally occurs in reward modelling, or if LLMs are used as the reward function. Here, due to finite data or stochasticity in the judge, we only have an approximately correct reward function. This epistemic uncertainty is naturally expressed as a reward function distribution $R \sim \rho$, and we will subsequently show that ROSA+Max/Softmax naturally captures the uncertainty in the reward function in the resulting policy's action distribution. We first set up a didactic task where there are multiple reward functions that do not always agree. Specifically, we consider the case of one-dimensional state X and action Y; see Figure 7(left), illustrating multiple diverse, non-equally weighted reward functions. Figure 7(right) displays the results of training ROSA+Max as well as PG with low/high entropy regularization. PG training with low-entropy regularization converges to the highest weighted reward functions, while high-entropy regularization results in wider coverage but also action probabilities over many bad actions that are not optimal under *any* reward functions. ROSA+Max covers all reward modes in proportion to their reward function's weight. See Appendix D.4 for more experimental details and additional figures comparing specific slices of $\pi(Y|X)$ between methods.

Finally, we extend our evaluation to an "ensemble of judges" setting for LLM post-training. We construct a mathematical reasoning task in which each problem is paired with a panel of eight judges. A response receives reward of 1 from a judge if it matches the judge's target, though only some judges are correct, and the learner is not told which ones. We train under varying levels of judge noise, with 6, 3, or 1 correct judges out of 8 (corresponding to 75%, 37.5%, and 12.5% judges' accuracy in Figure 8a). Across these settings, ROSA is more robust to noisy rewards than both standard PG and ensemble-regularized pessimistic baselines (Figure 8a). To test out-of-distribution generalization, we further evaluate models trained with noisy MATH judges on AIME2025. ROSA-trained models achieve higher out-of-distribution pass@k (Figure 8b).



(a) In domain (MATH) evaluation with different judge mixtures



(b) Out-of-distribution pass@k performance on unseen AIME 2025 problem set

Figure 8: Gemma 2B performance when trained on MATH where the reward is given by an ensemble of noisy judges. (8a) In distribution evaluation performance. (8b) Out-of-distribution pass@k evaluation on unseen AIME 2025 set. Ensemble min and stddev refer to taking $\min\left((r_i)_{i=1}^m\right)$ or $\text{avg}((r_i)_{i=1}^m) - \text{stddev}((r_i)_{i=1}^m)$ of the m rewards. All details are in Appendix D.5

6 Conclusion

We presented a novel, generalized reinforcement learning framework that unifies optimization for *sets of actions* over *distributions of reward functions*. By shifting the objective away from maximizing expected scalar returns, we obtain policy gradient objectives whose reward-maximizing solutions are by design *distributions*, thereby resolving the tension between reward maximization and policy entropy. This addresses a variety of limitations with standard policy gradients, such as mode collapse and brittleness under reward function errors. We provide rigorous theoretical analysis for the behaviour, optimization efficiency, and optimal solution of this objective, and also provide empirical evidence at small and large scales for how ROSA can be applied to a wide variety of example settings. Ultimately, this work provides a principled foundation for training stochastic policies that are as diverse and adaptable as the environments they inhabit.

AI use statement

In this work, we used generative AI tools (large language model assistants) for the following tasks requiring disclosure. *Mathematical claims and proofs:* all propositions and proofs were formulated and written by the authors; LLMs were used to double-check the correctness of completed proofs, in addition to the authors' own verification. *Implementing methods:* the ROSA objectives were implemented on existing training infrastructure; LLM coding assistants helped edit code, set up smaller-scale experiments, and supporting code such as plotting. All estimator implementations were checked for correctness by the authors. We did not use generative AI tools to develop the theoretical framework, propose or refine hypotheses, design the methodology or experiments, or interpret results. Generating synthetic datasets, translation, dataset cleaning, and qualitative or thematic analysis are not applicable to this work. Additionally, we used generative AI tools to produce and adjust figures, to identify relevant literature, and to edit the manuscript only for readability through iterative revision with the authors. We have reviewed all AI-assisted work and take responsibility for the final content of this paper, including all text, claims, and artifacts produced with the aid of generative AI.

Reproducibility statement

To support reproducibility, we provide complete statements and proofs of all theoretical results in Appendix B. Appendix D.1 describes how the objective landscapes in the main text were computed. Full experimental details, including task construction, model architectures, and hyperparameter tables for the competing-preferences transformer experiment (Appendix D.2), the multiple-correct-answers MATH experiment (Appendix D.3), the unknown ground-truth reward experiment (Appendix D.4), and the noisy MATH judges experiment (Appendix D.5), are given in the appendix. The ROSA+Max update is stated in full in Algorithm 1.

REFERENCES

Abbas Abdolmaleki, Sandy Huang, Leonard Hasenclever, Michael Neunert, Francis Song, Martina Zambelli, Murilo Martins, Nicolas Heess, Raia Hadsell, and Martin Riedmiller. A distributional view on multi-objective policy optimization. In *Proceedings of the International Conference on Machine Learning*, 2020.

Mridul Agarwal, Vaneet Aggarwal, and Tian Lan. Multi-objective reinforcement learning with non-linear scalarization. In *Proceedings of the International Conference on Autonomous Agents and Multiagent Systems*, 2022.

Eitan Altman. *Constrained Markov decision processes*. Routledge, 1999.

Eser Aygün, Anastasiya Belyaeva, Gheorghe Comanici, Marc Coram, Hao Cui, Jake Garrison, Renee Johnston, Anton Kast, Cory Y. McLean, Peter Norgaard, Zahra Shamsi, David Smalling, James Thompson, Subhashini Venugopalan, Brian P. Williams, Chujun He, Sarah Martinson, Martyna Plomecka, Lai Wei, Yuchen Zhou, Qian-Ze Zhu, Matthew Abraham, Erica Brand, Anna Bulanova, Jeffrey A. Cardille, Chris Co, Scott Ellsworth, Grace Joseph, Malcolm Kane, Ryan Krueger, Johan Kartiwa, Dan Liebling, Jan-Matthis Lueckmann, Paul Raccuglia, Xuefei Wang, Katherine Chou, James Manyika, Yossi Matias, John C. Platt, Lizzie Dorfman, Shibl Mourad, and Michael P. Brenner. An AI system to help scientists write expert-level empirical software. *arXiv preprint arXiv:2509.06503*, 2025.

Ryan Bahlous-Boldi, Isha Puri, Idan Shenfeld, Akarsh Kumar, Mehul Damani, Sebastian Risi, Omar Khattab, Zhang-Wei Hong, and Pulkit Agrawal. Vector policy optimization: Training for diversity improves test-time search. *arXiv preprint arXiv:2605.22817*, 2026.

Qinbo Bai, Mridul Agarwal, and Vaneet Aggarwal. Joint optimization of concave scalarized multi-objective reinforcement learning with policy gradient based algorithm. *Journal of Artificial Intelligence Research*, 74:1565–1597, 2022.

Ananth Balashankar, Ziteng Sun, Jonathan Berant, Jacob Eisenstein, Michael Collins, Adrian Hutter, Jong Lee, Chirag Nagpal, Flavien Prost, Aradhana Sinha, Ananda Theertha Suresh, and Ahmad Beirami. InfAlign: Inference-aware language model alignment. In *Proceedings of the International Conference on Machine Learning*, 2024.

Ahmad Beirami, Alekh Agarwal, Jonathan Berant, Alexander D'Amour, Jacob Eisenstein, Chirag Nagpal, and Ananda Theertha Suresh. Theoretical guarantees on the best-of-n alignment policy. In *Proceedings of the International Conference on Machine Learning*, 2025.

Marc G Bellemare, Will Dabney, and Rémi Munos. A distributional perspective on reinforcement learning. In *International conference on machine learning*, pp. 449–458. Pmlr, 2017.

Marc G Bellemare, Will Dabney, and Mark Rowland. *Distributional reinforcement learning*. MIT Press, 2023.

V Joseph Bowman Jr. On the relationship of the Tchebycheff norm and the efficient frontier of multiple-criteria objectives. In *Multiple Criteria Decision Making: Proceedings of a Conference Jouy-en-Josas, France May 21–23, 1975*, pp. 76–86. Springer, 1976.

Yilei Chen, Souradip Chakraborty, Lorenz Wolf, Yannis Paschalidis, and Aldo Pacchiano. Post-training large language models for diverse high-quality responses. In *Proceedings of the International Conference on Learning Representations*, 2026.

Taehyun Cho, Seungyub Han, Heesoo Lee, Kyungjae Lee, and Jungwoo Lee. Pitfall of optimism: Distributional reinforcement learning by randomizing risk criterion. In *Advances in Neural Information Processing Systems*, 2023.

Yinlam Chow and Mohammad Ghavamzadeh. Algorithms for CVaR optimization in MDPs. In *Advances in Neural Information Processing Systems*, 2014.

Yinlam Chow, Guy Tennenholtz, Izzeddin Gur, Vincent Zhuang, Bo Dai, Sridhar Thiagarajan, Craig Boutilier, Rishabh Agarwal, Aviral Kumar, and Aleksandra Faust. Inference-aware fine-tuning for best-of-m sampling in large language models. In *Proceedings of the International Conference on Learning Representations*, 2024.

John Joon Young Chung, Vishakh Padmakumar, Melissa Roemmele, Yuqian Sun, and Max Kreminski. Modifying large language model post-training for diverse creative writing. In *Proceedings of the Conference on Language Modeling*, 2025.

Kun-Jen Chung and Matthew J Sobel. Discounted MDP's: Distribution functions and exponential utility maximization. *SIAM Journal on Control and Optimization*, 25(1):49–62, 1987.

Anthony Coache and Sebastian Jaimungal. Robust reinforcement learning with dynamic distortion risk measures. *SIAM Journal on Mathematics of Data Science*, 8(1):1–22, 2026.

Thomas Coste, Usman Anwar, Robert Kirk, and David Krueger. Reward model ensembles help mitigate overoptimization. In *Proceedings of the International Conference on Learning Representations*, 2024.

Ganqu Cui, Yuchen Zhang, Jiacheng Chen, Lifan Yuan, Zhi Wang, Yuxin Zuo, Haozhan Li, Yuchen Fan, Huayu Chen, Weize Chen, Zhiyuan Liu, Hao Peng, Lei Bai, Wanli Ouyang, Yu Cheng, Bowen Zhou, and Ning Ding. The entropy mechanism of reinforcement learning for reasoning language models. *arXiv preprint arXiv:2505.22617*, 2025.

Will Dabney, Georg Ostrovski, David Silver, and Rémi Munos. Implicit quantile networks for distributional reinforcement learning. In *Proceedings of the International Conference on Machine Learning*, 2018.

Maria Dimakopoulou, Nikos Vlassis, and Tony Jebara. Marginal posterior sampling for slate bandits. In *Proceedings of the International Joint Conference on Artificial Intelligence*, 2019.

Matthias Ehrgott. *Multicriteria optimization*. Springer, 2005.

M Milani Fard and Joelle Pineau. Non-deterministic policies in Markovian decision processes. *Journal of Artificial Intelligence Research*, 40:1–24, 2011.

Zoltán Gábor, Zsolt Kalmár, and Csaba Szepesvári. Multi-criteria reinforcement learning. In *Proceedings of the International Conference on Machine Learning*, 1998.

Swetha Ganesh and Vaneet Aggarwal. Breaking the bias barrier in concave multi-objective reinforcement learning. *arXiv preprint arXiv:2603.08518*, 2026.

Leo Gao, John Schulman, and Jacob Hilton. Scaling laws for reward model overoptimization. *arXiv preprint arXiv:2210.10760*, 2022.

Xiangming Gu, Soham De, Larisa Markeeva, Petar Veličković, and Razvan Pascanu. Understanding performance gap between parallel and sequential sampling in large reasoning models. *arXiv preprint arXiv:2604.05868*, 2026.

Lin Gui, Cristina Gârbacea, and Victor Veitch. Bonbon alignment for large language models and the sweetness of best-of-n sampling. In *Advances in Neural Information Processing Systems*, 2024.

Anthony GX-Chen, Jatin Prakash, Jeff Guo, Rob Fergus, and Rajesh Ranganath. KL-regularized reinforcement learning is designed to mode collapse. In *Proceedings of the International Conference on Learning Representations*, 2026.

Tuomas Haarnoja, Haoran Tang, Pieter Abbeel, and Sergey Levine. Reinforcement learning with deep energy-based policies. In *Proceedings of the International Conference on Machine Learning*, 2017.

Jubayer Ibn Hamid, Ifdita Hasan Orney, Ellen Xu, Chelsea Finn, and Dorsa Sadigh. Polychromic objectives for reinforcement learning. In *Proceedings of the International Conference on Learning Representations*, 2026.

Conor F Hayes, Roxana Rădulescu, Eugenio Bargiacchi, Johan Källström, Matthew Macfarlane, Mathieu Reymond, Timothy Verstraeten, Luisa M Zintgraf, Richard Dazeley, Fredrik Heintz, Enda Howley, Athirai A. Irissappane, Patrick Mannion, Ann Nowé, Gabriel Ramos, Marcello Restelli, Peter Vamplew, and Diederik M. Roijers. A practical guide to multi-objective reinforcement learning and planning. *Autonomous Agents and Multi-Agent Systems*, 36(1):26, 2022.

Audrey Huang, Adam Block, Dylan J Foster, Dhruv Rohatgi, Cyril Zhang, Max Simchowitz, Jordan T Ash, and Akshay Krishnamurthy. Self-improvement in language models: The sharpening mechanism. In *Proceedings of the International Conference on Learning Representations*, 2025.

Thomas Hubert, Rishi Mehta, Laurent Sartran, Miklós Z. Horváth, Goran Žužić, Eric Wieser, Aja Huang, Julian Schrittwieser, Yannick Schroecker, Hussain Masoom, Ottavia Bertolli, Tom Zahavy, Amol Mandhane, Jessica Yung, Iuliya Beloshapka, Borja Ibarz, Vivek Veeriah, Lei Yu, Oliver Nash, Paul Lezeau, Salvatore Mercuri, Calle Sönne, Bhavik Mehta, Alex Davies, Daniel Zheng, Fabian Pedregosa, Yin Li, Ingrid von Glehn, Mark Rowland, Samuel Albanie, Ameya Velingker, Simon Schmitt, Edward Lockhart, Edward Hughes, Henryk Michalewski, Nicolas Sonnerat, Demis Hassabis, Pushmeet Kohli, and David Silver. Olympiad-level formal mathematical reasoning with reinforcement learning. *Nature*, 651:607–613, 2026.

Mete Ismayilzada, Antonio Laverghetta Jr, Simone A Luchini, Reet Patel, Antoine Bosselut, Lonneke van der Plas, and Roger Beaty. Creative preference optimization. In *Findings of the Association for Computational Linguistics*, 2025.

Yash Jhaveri, Harley Wiltzer, Patrick Shafto, Marc G Bellemare, and David Meger. Convergence theorems for entropy-regularized and distributional reinforcement learning. In *Advances in Neural Information Processing Systems*, 2025.

Satyen Kale, Lev Reyzin, and Robert E Schapire. Non-stochastic bandit slate problems. In *Advances in Neural Information Processing Systems*, 2010.

Robert Kirk, Ishita Mediratta, Christoforos Nalmpantis, Jelena Luketina, Eric Hambro, Edward Grefenstette, and Roberta Raileanu. Understanding the effects of RLHF on LLM generalisation and diversity. In *Proceedings of the International Conference on Learning Representations*, 2024.

Sotetsu Koyamada, Paavo Parmas, Tadashi Kozuno, and Shin Ishii. Emergence of exploration in policy gradient reinforcement learning via resetting. ICLR Submission, 2022. URL https://openreview.net/forum?id=GKsNIC_mQRG.

Nathan Lambert. Over-optimization. In *Reinforcement Learning from Human Feedback*. Online, 2026.

John Langford and Tong Zhang. The epoch-greedy algorithm for multi-armed bandits with side information. In *Advances in Neural Information Processing Systems*, 2007.

Haim Levy. *Stochastic dominance: Investment decision making under uncertainty*. Springer, third edition, 2016.

Tianjian Li, Yiming Zhang, Ping Yu, Swarnadeep Saha, Daniel Khashabi, Jason Weston, Jack Lanchantin, and Tianlu Wang. Jointly reinforcing diversity and quality in language model generations. *arXiv preprint arXiv:2509.02534*, 2025.

Xi Lin, Xiaoyuan Zhang, Zhiyuan Yang, Fei Liu, Zhenkun Wang, and Qingfu Zhang. Smooth Tchebycheff scalarization for multi-objective optimization. In *Proceedings of the International Conference on Machine Learning*, 2024.

Shie Mannor and Nahum Shimkin. The steering approach for multi-criteria reinforcement learning. In *Advances in Neural Information Processing Systems*, 2001.

Kaisa Miettinen. *Nonlinear multiobjective optimization*, volume 12. Springer Science & Business Media, 1999.

Volodymyr Mnih, Adria Puigdomenech Badia, Mehdi Mirza, Alex Graves, Timothy Lillicrap, Tim Harley, David Silver, and Koray Kavukcuoglu. Asynchronous methods for deep reinforcement learning. In *Proceedings of the International Conference on Machine Learning*, 2016.

Tetsuro Morimura, Masashi Sugiyama, Hisashi Kashima, Hirotaka Hachiya, and Toshiyuki Tanaka. Nonparametric return distribution approximation for reinforcement learning. In *Proceedings of the International Conference on Machine Learning*, 2010.

Ted Moskovitz, Aaditya K Singh, DJ Strouse, Tuomas Sandholm, Ruslan Salakhutdinov, Anca D Dragan, and Stephen McAleer. Confronting reward model overoptimization with constrained RLHF. In *Proceedings of the International Conference on Learning Representations*, 2024.

Soichiro Nishimori, Paavo Parmas, Sotetsu Koyamada, Tadashi Kozuno, Toshinori Kitamura, Shin Ishii, and Yutaka Matsuo. Emergence of exploration in policy gradient reinforcement learning via retrying. *arXiv preprint arXiv:2606.00151*, 2026.

Ifdita Hasan Orney, Jubayer Ibn Hamid, Shreya S Ramanujam, Shirley Wu, Hengyuan Hu, Noah Goodman, Dorsa Sadigh, and Chelsea Finn. Poly-EPO: Training exploratory reasoning models. *arXiv preprint arXiv:2604.17654*, 2026.

Ian Osband, Daniel Russo, and Benjamin Van Roy. (more) efficient reinforcement learning via posterior sampling. *Advances in Neural Information Processing Systems*, 26, 2013.

Vishakh Padmakumar and He He. Does writing with language models reduce content diversity? In *Proceedings of the International Conference on Learning Representations*, 2023.

Martin L. Puterman. *Markov Decision Processes: Discrete Stochastic Dynamic Programming*. Wiley, 1994.

Filip Radlinski, Robert Kleinberg, and Thorsten Joachims. Learning diverse rankings with multi-armed bandits. In *Proceedings of the International Conference on Machine Learning*, 2008.

Rafael Rafailov, Yaswanth Chittepu, Ryan Park, Harshit Sushil Sikchi, Joey Hejna, Brad Knox, Chelsea Finn, and Scott Niekum. Scaling laws for reward model overoptimization in direct alignment algorithms. In *Advances in Neural Information Processing Systems*, 2024.

Diederik M Roijers, Peter Vamplew, Shimon Whiteson, and Richard Dazeley. A survey of multi-objective sequential decision-making. *Journal of Artificial Intelligence Research*, 48:67–113, 2013.

Bernardino Romera-Paredes, Mohammadamin Barekatain, Alexander Novikov, Matej Balog, M. Pawan Kumar, Emilien Dupont, Francisco J. R. Ruiz, Jordan S. Ellenberg, Pengming Wang, Omar Fawzi, Pushmeet Kohli, and Alhussein Fawzi. Mathematical discoveries from program search with large language models. *Nature*, 625(7995):468–475, 2024.

Pier Giuseppe Sessa, Robert Dadashi, Léonard Hussenot, Johan Ferret, Nino Vieillard, Alexandre Ramé, Bobak Shariari, Sarah Perrin, Abe Friesen, Geoffrey Cideron, Sertan Girgin, Piotr Stanczyk, Andrea Michi, Danila Sinopalnikov, Sabela Ramos, Amélie Héliou, Aliaksei Severyn, Matt Hoffman, Nikola Momchev, and Olivier Bachem. BOND: Aligning LLMs with best-of-n distillation. In *Proceedings of the International Conference on Learning Representations*, 2025.

Alexander Shypula, Shuo Li, Botong Zhang, Vishakh Padmakumar, Kayo Yin, and Osbert Bastani. Evaluating the diversity and quality of LLM generated content. In *Proceedings of the Conference on Language Modeling*, 2025.

Ralph E Steuer and Eng-Ung Choo. An interactive weighted Tchebycheff procedure for multiple objective programming. *Mathematical programming*, 26(3):326–344, 1983.

Nisan Stiennon, Long Ouyang, Jeffrey Wu, Daniel Ziegler, Ryan Lowe, Chelsea Voss, Alec Radford, Dario Amodei, and Paul F. Christiano. Learning to summarize with human feedback. In *Advances in Neural Information Processing Systems*, 2020.

Peter Sunehag, Richard Evans, Gabriel Dulac-Arnold, Yori Zwols, Daniel Visentin, and Ben Coppin. Deep reinforcement learning with attention for slate Markov decision processes with high-dimensional states and actions. *arXiv preprint arXiv:1512.01124*, 2015.

Richard S. Sutton and Andrew G. Barto. *Reinforcement Learning: An Introduction*. The MIT Press, second edition, 2018.

Yunhao Tang, Kunhao Zheng, Gabriel Synnaeve, and Rémi Munos. Optimizing language models for inference time objectives using reinforcement learning. In *Proceedings of the International Conference on Machine Learning*, 2025.

Emanuel Todorov. Linearly-solvable Markov decision problems. In *Advances in Neural Information Processing Systems*, 2006.

Amos Tversky and Daniel Kahneman. Advances in prospect theory: Cumulative representation of uncertainty. *Journal of Risk and Uncertainty*, 5(4):297–323, 1992.

Peter Vamplew, Richard Dazeley, Adam Berry, Rustam Issabekov, and Evan Dekker. Empirical evaluation methods for multiobjective reinforcement learning algorithms. *Machine learning*, 84 (1):51–80, 2011.

Claudio Mayrink Verdun, Alex Oesterling, Himabindu Lakkaraju, and Flavio P Calmon. Soft best-of-n sampling for model alignment. In *IEEE International Symposium on Information Theory (ISIT)*. IEEE, 2025.

Christian Walder and Deep Tejas Karkhanis. Pass@k policy optimization: Solving harder reinforcement learning problems. In *Advances in Neural Information Processing Systems*, 2026.

Shaun Wang. Premium calculation by transforming the layer premium density. *ASTIN Bulletin: The Journal of the IAA*, 26(1):71–92, 1996.

Peter West and Christopher Potts. Base models beat aligned models at randomness and creativity. In *Proceedings of the Conference on Language Modeling*, 2025.

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

Ronald J Williams and Jing Peng. Function optimization using connectionist reinforcement learning algorithms. *Connection Science*, 3(3):241–268, 1991.

Menahem E Yaari. The dual theory of choice under risk. *Econometrica: Journal of the Econometric Society*, pp. 95–115, 1987.

Chenghao Yang and Ari Holtzman. LLM probability concentration: How alignment shrinks the generative horizon. *arXiv preprint arXiv:2506.17871*, 2025.

Yisong Yue and Carlos Guestrin. Linear submodular bandits and their application to diversified retrieval. In *Advances in Neural Information Processing Systems*, 2011.

Longfei Yun, Chenyang An, Zilong Wang, Letian Peng, and Jingbo Shang. The price of format: Diversity collapse in LLMs. *arXiv preprint arXiv:2505.18949*, 2025.

Yiming Zhang, Harshita Diddee, Susan Holm, Hanchen Liu, Xinyue Liu, Vinay Samuel, Barry Wang, and Daphne Ippolito. NoveltyBench: Evaluating language models for humanlike diversity. In *Proceedings of the Conference on Language Modeling*, 2025.

Rosie Zhao, Alexandru Meterez, Sham Kakade, Cengiz Pehlevan, Samy Jelassi, and Eran Malach. Echo chamber: RL post-training amplifies behaviors learned in pretraining. In *Proceedings of the Conference on Language Modeling*, 2025.

Brian D Ziebart, Andrew L Maas, J Andrew Bagnell, and Anind K Dey. Maximum entropy inverse reinforcement learning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, 2008.

APPENDICES

A Related work

In this section, we provide more technical discussion with related subfields of RL.

A.1 Overview

ROSA generalizes several existing RL objectives. Specifically, in the *single reward function* case, ROSA with the set mean function $f_{\text{mean}}\big((U_i)_{i=1}^n\big) = \frac{1}{n}\sum_{i=1}^n U_i$ recovers the vanilla PG criterion (Eq. (1)), while the set max $f_{\max}\big((U_i)_{i=1}^n\big) = \max_{1\le i\le n} U_i$ coincides with the "best-of-n" training objective (Stiennon et al., 2020). With multiple reward functions, the set mean function f_{mean} recovers linear scalarization from multi-objective RL (Vamplew et al., 2011).

Improving diversity in post-training. Many recent works (Huang et al., 2025; Padmakumar & He, 2023; GX-Chen et al., 2026) have pointed to the problem of mode collapse in RL post-trained models compared to supervised fine-tuned models. Li et al. (2025) proposes to address this by jointly training the model on a reward function which is a product of diversity term and task reward. The diversity term is computed as a function of pairwise distances over a batch of samples for each prompt where the more unique samples are assigned a higher reward. While this approach preserves the diversity during post-training and shows significant gains in performance, we observe that this diversity bonus undesirably distorts the ordering of sub-optimal policies (Figure 2c). Similarly, Hamid et al. (2026); Orney et al. (2026) train on product of reward and diversity defined at a set level, at the cost of having a non reward-maximizing policy as the optimum of this objective (Figure 11). We discuss in detail in Appendix A.4.

Optimization over a set of responses. The best-of-n approach to inference-time policy improvement is a popular approach to improving LLM performance (Stiennon et al., 2020). Several recent works have studied the modification of RL objectives to take this inference-time technique into account (Koyamada et al., 2022; Beirami et al., 2025; Balashankar et al., 2024; Chow et al., 2024; Tang et al., 2025; Walder & Karkhanis, 2026; Chen et al., 2026), and other works use the implicit resulting improved policy in training objectives (Gui et al., 2024; Sessa et al., 2025). Distinct from our contributions of dealing with distribution of reward functions, these methods are concerned with optimization of a single reward function, and thus have deterministic policies as objective optima. Earlier related work in this vein is motivated by recommender systems (Radlinski et al., 2008), including online learning methods (Kale et al., 2010; Yue & Guestrin, 2011; Dimakopoulou et al., 2019) and deep reinforcement learning (Sunehag et al., 2015). Fard & Pineau (2011) study the problem of identifying all near-optimal actions in MDPs via mixed-integer programming methods.

Another closely related approach is the ReMax objective (Koyamada et al., 2022; Nishimori et al., 2026), which coincides with ROSA+Max (Equation (3)) in the bandit setting. Like ROSA+Max, ReMax observes that the optimal bandit policy under this objective is stochastic, unlike in standard policy-gradient methods. However, while the initial motivations are similar, ReMax primarily targets improved exploration in MDPs. Algorithmically, Nishimori et al. (2026) propose an expected-improvement-based estimator for best-of-n gradients, but dispense with explicit reward or value distributions, instead interpreting the non-stationary Q-function being learned as samples from the action-value distribution. In contrast, ROSA is designed specifically for generative model post-training: it uses different gradient estimators and, crucially, maintains explicit reward uncertainty as a natural consequence of real-world, multiply satisfiable task constraints. Whereas ReMax uses stochasticity instrumentally to discover better deterministic MDP policies, ROSA treats policy stochasticity and reward uncertainty as central objects of the optimization problem.

Multi-objective reinforcement learning. Multi-objective reinforcement learning (Roijers et al., 2013; Hayes et al., 2022) focuses on modelling problems that naturally have several, potentially conflicting, notions of reward. Key questions include optimization of reward subject to cost constraints (Altman, 1999), and discovery of policies on the Pareto frontier (Gábor et al., 1998; Mannor & Shimkin, 2001; Abdolmaleki et al., 2020; Bahlous-Boldi et al., 2026). Most closely connected to the problem we study is *scalarization*, in which a single scalar objective is constructed from several reward signals. However, these approaches are typically limited to expected utilities, which we show in Appendix A.3.1 are either insufficiently expressive to capture our desired diverse policies as

solutions, or otherwise suffer from bias issues, unlike the ROSA objective proposed here. Concurrent with this paper, Bahlous-Boldi et al. (2026) propose a method for Pareto optimization that combines multiple reward functions and sets of actions; we discuss comparison with this work in more detail in Appendix A.3.2.

A.2 Risk-sensitive reinforcement learning

The set max function used in ROSA+Max can be interpreted as optimizing a function of the reward distribution that is *not* expressible as an expected utility, but rather as a *distortion risk measure* (Yaari, 1987; Wang, 1996). The particular distributional property is summarized below (Wang, 1996).

Lemma A.1. *The expected max-of-n reward can be expressed through a distorted expectation,*

$$\mathbb{E}_{Y_1,\dots,Y_n \overset{i.i.d.}{\sim} \pi}\left[\max(R(Y_1),\dots,R(Y_n))\right] = \int_0^1 F^{-1}(u^{1/n})\,\mathrm{d}u\,, \tag{7}$$

where F is the CDF of $R(Y)$, $Y \sim \pi$.

Proof. Recall that $R(Y)$ is equal in distribution to $F^{-1}(U)$, where $U \sim \text{Uniform}([0,1])$, and F^{-1} is the quantile function of $R(Y)$. We then have that $(R(Y_1)\ \dots, R(Y_n))$ is equal in distribution to $(F^{-1}(U_1),\dots,F^{-1}(U_n))$, where $(U_i)_{i=1}^n \overset{\text{i.i.d.}}{\sim} \text{Uniform}([0,1])$. It follows that,

$$\max(F^{-1}(U_1),\dots,F^{-1}(U_n)) = F^{-1}(\max(U_1,\dots,U_n))\,,$$

due to monotonicity of F^{-1}. Now, $\max(U_1,\dots,U_n)$ is distributed equally to $U_1^{1/n}$, since

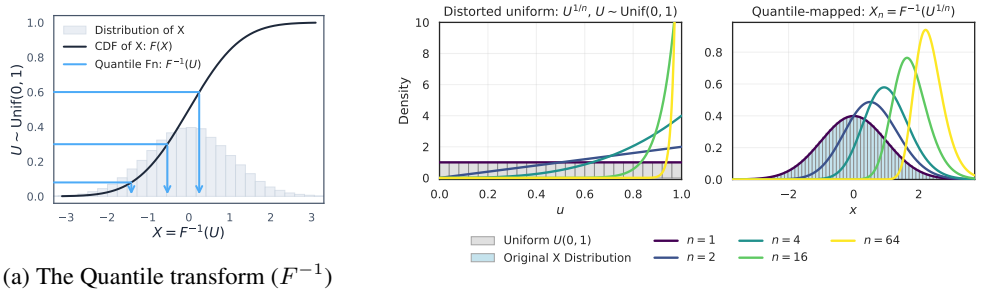
$$\mathbb{P}(\max(U_1,\dots,U_n) \le z) = \mathbb{P}(\forall i\,, U_i \le z) = \mathbb{P}(U_1 \le z)^n = z^n = \mathbb{P}(U_1 \le z^n) = \mathbb{P}(U_1^{1/n} \le z)\,.$$

Hence, we have

$$\mathbb{E}_{Y_{1:n} \overset{i.i.d.}{\sim} \pi}[\max(R(Y_1),\dots,R(Y_n))] = \mathbb{E}_{U\sim\text{Uniform}([0,1])}[F^{-1}(U^{1/n})]\,,$$

as required. □

Note that without the $1/n$ exponent, the expression on the right-hand side of Eq. (7) would evaluate to the mean of the distribution, $\mathbb{E}_{Y\sim\pi}[R(Y)]$. The integral in Lemma A.1 therefore reports an asymmetric summary of the distribution by *distorting* the variable of integration.



(a) The Quantile transform (F^{-1}) commutes with set max operation

(b) Max-of-n (right) as transforms of distorted uniform (left)

Figure 9: Max-of-n as an estimator of a distortion risk measure. (**a**) The quantile transform allows for sampling of any 1D random variable X by mapping a uniform random variable through its quantile function (i.e., the inverse CDF). Because this transformation commutes with order-preserving operations like the maximum, a max-of-n distribution can be identically modelled by taking the max-of-n of uniform variables before applying the quantile transform. (**b**) Consequently, the max-of-n distribution of X can be viewed as a quantile transform of a distorted uniform distribution. Specifically, we sample $U \sim \text{Unif}(0,1)$, apply the distortion $U^{1/n}$, and pass it to the inverse CDF to obtain $F^{-1}(U^{1/n})$.

Figure 9 further illustrates the intuition of Lemma A.1, where the max-of-n operation can be viewed as a quantile transform of a distorted uniform distribution. Specifically, transforming a standard

uniform sample $U \sim \text{Unif}(0,1)$ into $U^{1/n}$ yields a $\text{Beta}(n,1)$ distribution, which is identically distributed to the maximum of n independent uniforms. More generally, evaluating expectations under a distorted probability space defines a *distortion risk measure* (Yaari, 1987; Wang, 1996). This framework is widely used to model risk sensitivity, such as in cumulative prospect theory (Tversky & Kahneman, 1992), and extensively in risk-aware reinforcement learning (Chow & Ghavamzadeh, 2014; Dabney et al., 2018; Cho et al., 2023; Coache & Jaimungal, 2026). All in all, the implicit use of distributional properties beyond expected utilities, such as the distortion risk measure above, is key to the ROSA framework being able to express objectives with the desired optimal policies.

Differences between ROSA and Distributional RL. We highlight an important distinction between the sources of randomness in ROSA and those in distributional RL (Chung & Sobel, 1987; Bellemare et al., 2023) and risk-sensitive RL. Specifically, there are *two* distinct sources of randomness in ROSA: randomness over sampled outputs ("aleatoric") and randomness over reward functions ("epistemic"). Distributional RL and risk-sensitive RL traditionally focus on aleatoric randomness (Bellemare et al., 2017; Dabney et al., 2018, Section 3.3). By modeling the return distribution, distributional RL can optimize risk-sensitive quantities beyond the mean, such as up-weighting high-return outcomes over low-return ones (Dabney et al., 2018). The *set function* in ROSA operate over this same aleatoric noise: as established in Lemma A.1, the max-of-N operator yields an risk-sensitive summary of the output-induced reward distribution, serving as a Monte Carlo estimator of the same risk-sensitive objectives studied in distributional RL. ROSA then *additionally* considers distributions over *reward functions*—a perspective traditionally emphasized in Bayesian approaches such as PSRL (Osband et al., 2013). It is precisely this combination of an optimistic aleatoric summary over epistemic reward function uncertainty that enables ROSA to produce a diverse policy, allocating action probabilities in a manner calibrated to reward function uncertainty.

A.3 Multi-objective reinforcement learning

In the main paper, we mention comparison with multi-objective reinforcement learning (Roijers et al., 2013; Hayes et al., 2022). Problem formulations in this domain typically focus on finding good policies in settings with a finite collection of reward functions $R_1, \ldots, R_d$. We discuss comparisons to scalarization approaches in the bandit setting below, which are the most closely related variety of MORL problems to our setting, though often in the literature these approaches are expressed more generally in the MDP setting.

A.3.1 Scalarization approaches

The principal approaches to handling multiple reward functions—and those closely related to the ROSA framework—find optimal policies with respect to a scalar objective constructed from the individual rewards. The first approach is termed *expected scalarized return* (ESR), and selects a *scalarization* function $s : \mathbb{R}^d \to \mathbb{R}$ to obtain an objective

$$\mathbb{E}_{Y\sim\pi}[s(R_1(Y), \ldots, R_d(Y))]\,. \tag{8}$$

The second approach is termed *scalarized expected return* (SER), and instead inverts the ordering of the expectation and scalarization function, leading to an objective of the form

$$s(\mathbb{E}_{Y\sim\pi}[R_1(Y)], \ldots, \mathbb{E}_{Y\sim\pi}[R_d(Y)])\,. \tag{9}$$

Both approaches have been applied to a variety of problems in reinforcement learning that are naturally expressed through multiple reward functions. For the problems we are concerned with in this paper, however, these approaches have several drawbacks which the ROSA framework circumvents.

Lack of expressiveness. ESR deals solely with *expected utilities*; properties of distributions that can be expressed as the expectation of a (possibly non-linear function) applied to a random variable. This is a limited class of distributional properties, and is not expressive enough to capture the notions of diversity that we consider in this paper. As a concrete example, we consider the case where $R_1, \ldots, R_d$ are one-hot over distinct actions, and would like an objective that is uniquely maximized by the policy which is uniform over this set of actions. The only possible values that the vector $(R_1(Y), \ldots, R_d(Y))$ can take on are the zero-vector, and one-hot vectors. By symmetry, our non-linearity s must assign equal utility to each one-hot vector. But then the objective simply reduces

to the amount of probability mass allocated to the set of actions that are optimal for one of the $(R_i)_{i=1}^d$, with no incentive for uniformity of the distribution on this set. On the other hand, SER captures quantities that are functions of expected rewards, and thus cannot capture distributional properties of the random variable $R_i(Y)$ that require more than the mean, such as the ones discussed in Section A.2.

Implementation complexity. SER additionally encounters complexities in implementation, owing to the objective applying non-linearities to expectations of rewards; the naive approach to forming a policy gradient for this objective induces bias, requiring more intricate algorithm design, such as two-timescale approaches (Bai et al., 2022; Agarwal et al., 2022; Ganesh & Aggarwal, 2026). By contrast, since the ROSA framework expresses the objective in terms of expectations over set actions, the non-linear function of the reward distribution is made implicit, and unbiased policy gradients for the objective are straightforward to compute.

Accessing all rewards functions. The MORL approaches described above typically assume access to all reward functions when computing an objective estimator, and thus scale computationally with d, the number of reward functions specified in the problem. In particular, they do not generally handle infinitely-many reward functions. By contrast, as the ROSA objective is phrased with an outer expectation over the reward function, unbiased estimates of objective and gradient can be calculated straightforwardly with samples from the reward function distribution.

Standard policy gradient on marginalized reward function as a special case. Given a distribution ρ over a finite number of reward functions $(r_i)_{i=1}^m$, both the ESR and ROSA objectives have standard policy gradient on the reward function $\sum_{i=1}^m \rho(r_i) r_i$ as a special case. For ESR, given a distribution this is obtained by taking $s : \mathbb{R}^d \to \mathbb{R}$ to be

$$s(z_1, \ldots, z_n) = \sum_{i=1}^{n} \rho(r_i) z_i \,.$$

For ROSA, this is obtained for any n by taking the set function f to be the mean function. In the context of Figure 1 in the main paper, illustrating the general ROSA framework, this is mathematically equivalent to collapsing the plot along the reward function axis and applying a standard policy gradient. By contrast, the general ROSA framework applies a non-linear transformation, based on the policy, to each reward function before performing the reduction along this axis.

A.3.2 Pareto optimization

While not explored in this work, we highlight a concurrent work which explores randomized objectives for Pareto optimization (Bahlous-Boldi et al., 2026). Concretely, the authors consider a sequence of vector *utility functions*, $\mathbf{u} = (u_1, \ldots, u_d)$, where $\mathbf{u}(x, y) \in \mathbb{R}^d$ is the d-dimensional utility feedback for inputs (x, y), and aim to find a set of actions which are *Pareto-optimal* for the vector utilities (Ehrgott, 2005; Miettinen, 1999). Note that in our discussion here, we have intentionally used the term *utility functions*, $u_i : \mathcal{X} \times \mathcal{Y} \to \mathbb{R}$, to distinguish from the reward functions, $R : \mathcal{X} \times \mathcal{Y} \to \mathbb{R}$ in the ROSA framework. The primary objective in Bahlous-Boldi et al. (2026) computes the score for some action multiset $(y_1, \ldots, y_n)$ as,

$$\mathbb{E}_{\mathbf{w} \sim \mathrm{Dir}(\mathbf{1})} \left[\max_{1 \le i \le n} \mathbf{w}^\top \mathbf{u}(x, y_i) \right], \tag{10}$$

where $\mathrm{Dir}(\mathbf{1})$ is a uniform distribution over d-dimensional vectors that sum to 1, which linearly combines the d dimensional utilities.

We note that this is in fact a special case of ROSA+Max (Eq. (3)), where the randomized reward function is defined via the following reparameterization,

$$R(x, y) = \mathbf{w}^\top \mathbf{u}(x, y), \qquad \mathbf{w} \sim \mathrm{Dir}(\mathbf{1}) \,. \tag{11}$$

Substituting the above distribution $R \sim \rho$ into the ROSA+Max criterion (Eq. (3)) recovers Equation (10). Thus, since the primary objective in Bahlous-Boldi et al. (2026) is an instance of ROSA, the theoretical understanding we provide in this work can be applied to understand their work as well. From this perspective, Bahlous-Boldi et al. (2026) elegantly apply *distributions over scalarizations* (of utilities $\mathbf{u}$) to induce the reward-function distribution, and demonstrate an additional application of the ROSA framework: to do Pareto-optimization.

It is worth noting that the approach of Bahlous-Boldi et al. (2026) differs operationally from standard action-set methods. Rather than sampling the action multiset i.i.d. in parallel—as done in Tang et al. (2025); Hamid et al. (2026); Orney et al. (2026) and our work—their action multiset is generated *sequentially* within a single rollout. Sequential generation allows later action samples to condition on earlier ones and may, in principle, enjoy greater representation flexibility and improved multiset diversity.[2] Yet, it imposes a number of additional restrictions: (i) prompts have to be modified to specify multiset format and size, (ii) the model must have sufficient instruction-following ability and generate following a rigidly parsable multi-action schema, and (iii) using large multisets requires long autoregressive generations. Conversely, parallel sampling requires no structural assumptions on the prompt and generation beyond compatibility with a reward function, while better exploiting modern parallelized hardware. Credit assignment also differs operationally for the two approaches: because the sequential approach optimizes a single concatenated generation containing multiple actions, its current operationalization lacks action-specific credit assignment. In contrast, parallel sampling naturally admits action-level variance-reduction techniques (Appendix B.1).

Ultimately, the generalized ROSA formulation remains orthogonal and complementary to the choice of scalarization distributions, which are seamlessly absorbed into the reward function distribution ρ. Identifying the right reward function distribution for Pareto-diversity stands as a promising direction for future research. For instance, Bahlous-Boldi et al. (2026) uses linear scalarization, while non-linear scalarization methods can have different coverage properties that better suit Pareto geometries (Ehrgott, 2005; Bowman Jr, 1976; Steuer & Choo, 1983; Lin et al., 2024). Promisingly, any such exploration can be viewed as ROSA with distinct choices of reward distributions, and stand to benefit directly from the theoretical understanding we have established in the current work.

A.4 Diversity bonus approaches

While ROSA induces diversity as naturally emergent from the underlying reward distribution, a separate, distinct paradigm enforces output diversity by pairing a single reward function r with an explicit, external diversity mechanism. Intuitively, these approaches manually alter the optimization objective to "boost" distinct actions and penalize redundant ones. In this section, we describe their properties and draw contrasts with the ROSA framework.

Per-sample multiplicative diversity. A common approach is to scale the reward of each individual action by its average pairwise distance to all other concurrent actions in the multiset (Li et al., 2025). The modified per-sample reward $\bar{r}(Y_i)$ is (again dropping state dependence to lighten notation),

$$\bar{r}(Y_i) = r(Y_i) \cdot \left(\frac{1}{n-1} \sum_{1 \le j \le n,\, j \ne i} d(Y_i, Y_j) \right), \tag{12}$$

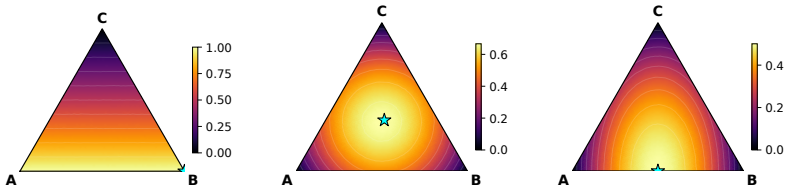
where $d : \mathcal{Y} \times \mathcal{Y} \to \mathbb{R}$ is a pairwise diversity function quantifying the distance between a pair of actions. For instance, d can be a binary classifier encoding whether y and y' are semantically equivalent (Zhang et al., 2025; Li et al., 2025).

While intuitive, this approach suffers from two primary pitfalls. First, optimizing Equation (12) via standard policy gradient (Eq. (2)) introduces *gradient bias*. This is because an individual action's modified reward dynamically depends on all other sampled actions, and standard PG ignores cross-derivatives tracking how changing one action's probability affects the diversity scores of concurrent actions. Second, even if the set policy gradient is correctly computed, per-sample multiplicative diversity warps the optimization landscape such that a worse policy (in terms of reward) can be ranked much higher than a nearly optimal policy with low diversity, therefore complicating policy improvement. This can be observed in Figure 10c: a policy which assign 50-50 probability to actions `A` and `C` ($\mathbb{E}[R(Y)] \approx 0.5$) is preferred over one that puts nearly all probabilities on `A` ($\mathbb{E}[R(Y)] \approx 1$).

Set-level multiplicative diversity. An alternative approach applies a global diversity modifier directly to the collective action set reward (Hamid et al., 2026; Orney et al., 2026). This alters the total multiset objective to the form:

$$\bar{R}(Y_{1:n}) = \left(\frac{1}{n} \sum_{i=1}^{n} r(Y_i) \right) \cdot D(Y_{1:n}), \tag{13}$$

[2] However, whether sequential sampling actually helps in current LLMs is debated, see Gu et al. (2026).



(a) Reward only (b) Diversity only (c) Mult. diversity

Figure 10: Per-sample multiplicative diversity. We numerically simulate the average policy score using (a) reward function only, (b) pairwise diversity function only with diversity function $d(y, y') = \mathbf{1}_{[y \neq y']}$, and (c) per-sample multiplicative between the reward and diversity. Actions `A` and `B` receive a reward of 1, and `C` receives 0. Colours denote the averaged score for different policy on the 3-category simplex. Details in Appendix D.1.

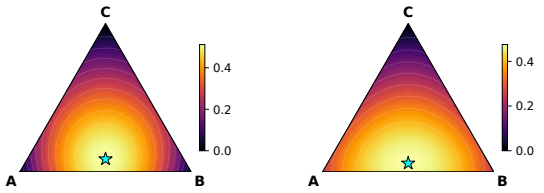
where $D(Y_{1:n})$ is a global set-level metric. A number of diversity metrics are available here, which we again study in the three-category simplex to visualize the average score of Eq. (13) with different action probabilities (Figure 11). One intuitive set diversity metric is the *set average distance* (i.e. set version of Equation (12)),

$$D_{\text{AvgDist}}(Y_{1:n}) = \frac{1}{n(n-1)} \sum_{i=1}^{n} \sum_{1 \leq j \leq n,\, j \neq i} d(Y_i, Y_j)\,,$$

which we visualize in Figure 11a. Another set diversity metric is the *number of distinct clusters* (Hamid et al., 2026; Orney et al., 2026),

$$D_{\text{Clusters}}(Y_{1:n}) = \frac{\text{number of distinct clusters in } Y_{1:n}}{n}\,,$$

which we compute as the number of unique actions, visualized in Figure 11b. Both approaches inherit the landscape-distortion pitfall of multiplicative scaling, and have optimal policies that are reward sub-optimal, as the objective prefers highly diverse sets of incorrect actions over redundant sets of correct actions. In contrast, ROSA+Max (Fig. 2d) and ROSA+Softmax (Fig. 3) naturally preserve sub-optimal policy ordering, ensuring that policy improvement gradients always point in a direction that simultaneously maximizes correctness and diversity.



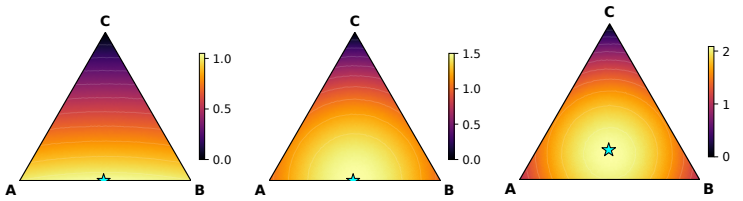
(a) Average distance (b) Unique actions

Figure 11: Set multiplicative diversity. We numerically simulate the average policy score using average reward multiplied with (a) average set distance, and (b) number of distinct actions. Actions `A` and `B` receive a reward of 1, and `C` receives 0. Colours denote the averaged score for different policy on the 3-category simplex. Details in Appendix D.1.

Additive diversity. Finally, we also study *additive* bonuses in Figure 12 of the form,

$$\bar{r}_{\text{additive}}(Y_i) = r(Y_i) + \alpha \left(\frac{1}{n-1} \sum_{1 \leq j \leq n,\, j \neq i} \mathbf{1}_{[Y_i \neq Y_j]} \right).$$

While additive formulations can mitigate some of the policy-ordering distortions found in multiplicative methods, they introduce a separate practical pitfall: extreme sensitivity to the scale of the reward and diversity bonuses. The hyperparameter α requires extensive tuning: a value too small may fail to induce diversity, while a value too large similarly results in low-reward optimal policies.



(a) $\alpha = 0.1$ (b) $\alpha = 1$ (c) $\alpha = 2$

Figure 12: Numerical simulation of additively combined reward and diversity scores. Actions `A` and `B` receive a reward of 1, while `C` receives 0. Colours denote the average action multiset score for different policy distributions on the simplex.

Interpreting diversity bonuses under the ROSA framework. Interestingly, in the case of binary rewards, the multiplicative diversity bonus (Eq. (12)) can be interpreted under the ROSA framework, albeit with a set function not supported by the theoretical guarantees of Theorem 4.1. Specifically, we will assume that actions rewarded under different reward functions are different. Suppose we have taken a multiset of n actions with k of them being rewarded under reward function r, the sum of quality $\times$ diversity scores for this set of correct actions is,

$$k \times \frac{n-k}{n-1}\,.$$

The multiplicative term is because the other $n-k$ actions are different from the k rewarded actions, and therefore contribute to the diversity bonus. In other words, multiplicative diversity can be realized under the ROSA framework with a success-count reward function $\tilde{f}(k) = k(n-k)/(n-1)$. We plot this in Figure 13, alongside the success-count function for ROSA+Max, and ROSA+Softmax. Note that for ROSA+Softmax, the success-count function can be calculated by summing over the k contributions of a reward-1 action in Equation (5), to obtain

$$\frac{ke}{ke + (n-k)}\,.$$

Interestingly, the set function corresponding to multiplicative diversity does not satisfy the criteria of Theorem 4.1; in particular, monotonicity is not satisfied. This can manifest certain issues: if only one unique correct action is observed (k times) in a collection of n actions, values of k around $n/2$ actually receive a higher score than values of k closer to n. This forces optimization to prefer a policy that has low expected reward ($\sim k/n$) in order to be diverse, and is an instance where the diversity-modified objective distorts the ordering of sub-optimal policies. On the other hand, we know from Theorem 4.1 that there exists a family of $\tilde{f}$ that provides both high expected reward and a diverse optimal policy.

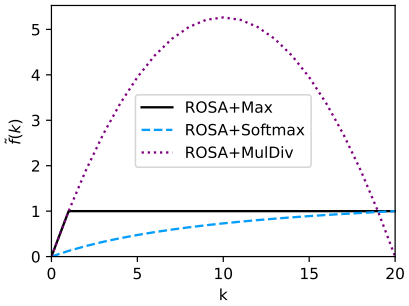


Figure 13: Success-count functions $\tilde{f}$ for ROSA+Max, ROSA+Softmax, and the multiplicative diversity objective appearing in Equation (12), in the binary reward setting.

We conclude this section with Table 1, which summarizes, in the context of our central problem, some key properties of ROSA and a variety of comparator methods discussed in this section.

B PROOFS

In this section, we collect statements and proofs of results from the main paper.

B.1 ROSA POLICY GRADIENT

Proposition B.1. *Let $\mathbf{Y} = (Y_i)_{i=1}^n$ denote a multiset of actions induced by $Y_1, \ldots, Y_n$. We write $\mathbf{Y} \sim \pi_\theta$ to mean the n actions are sampled i.i.d. from policy π_θ. Denote by ρ a distribution over reward functions $R : \mathcal{X} \times \mathcal{Y} \to \mathbb{R}$, and let $R \sim \rho$ be a draw from this distribution. For brevity, we*

Desiderata	Entropy Regularization	Diversity Bonus	MORL (ESR)	MORL (SER)	ROSA
Preserves reward-maximizing optimum	✗	✓ / ✗	✗	✗	✓
Ranks sub-optimal policies correctly	✗	✗	✗	✗	✓
Has stochastic optimal policies (diverse)	✓	✓	✗	✓	✓
Unbiased gradient estimation	✓	✓	✓	✗	✓
Handles arbitrary reward distributions	N/A	N/A	✗	✗	✓

Table 1: Comparison of ROSA to other methods in diversity-promoting and multi-objective RL. ROSA is the only approach that satisfies all desiderata: (i) its objective optimum aligns with an expected reward optimum, (ii) it correctly orders sub-optimal policies by their rewards, (iii) it maintains a stochastic, diversity-preserving optimal policy, (iv) it admits straightforward optimization via unbiased gradient estimators from action samples $Y \sim \pi(\cdot|X)$, and (v) it naturally accommodates arbitrary distributions over potentially infinite reward functions. Note that depending on the type of diversity bonus approach, it may or may not have a reward-maximizing optimal policy (see Appendix A.4).

write $R(X, \mathbf{Y}) = (R(X, Y_i))_{i=1}^n$ as the reward multiset. Let f be a function defined over multisets of actions. The generic ROSA objective is

$$\mathcal{J}_{\text{ROSA}}(\pi_\theta) = \mathbb{E}_{X\sim\mu\,,\,\mathbf{Y}\overset{i.i.d.}{\sim}\pi_\theta(\cdot|X)}\Big[\mathbb{E}_{R\sim\rho}\Big[f(R(X,\mathbf{Y}))\Big]\Big]\,, \tag{14}$$

and the gradient of the objective at a sampled state $X \sim \mu$ is

$$\mathbb{E}_{\mathbf{Y}\overset{i.i.d.}{\sim}\pi_\theta(\cdot|X)}\Big[\mathbb{E}_{R\sim\rho}\Big[\sum_{i=1}^n \Big(f\big(R(X,\mathbf{Y})\big) - b\big(R(X,\mathbf{Y}_{-i})\big)\Big)\nabla_\theta \log\pi_\theta(Y_i|X)\Big]\Big]\,, \tag{15}$$

where b is an arbitrary multiset function that forms an optional control variate, and $\mathbf{Y}_{-i}$ indicates the multiset $\mathbf{Y} = (Y_j)_{j=1}^n$ with element Y_i removed.

Proof. The proof is a straightforward extension of analysis by Tang et al. (2025) in the single-reward setting, to deal with randomization over reward functions. The core technique (in the single-reward setting) is to treat the vector $(Y_1, \ldots, Y_n)$ as a single "macro action", with reward $f(R(X, \mathbf{Y}))$. Generally, for any multiset function g, we have per the usual score function trick (Williams, 1992),

$$\nabla_\theta\, \mathbb{E}_{\mathbf{Y}\overset{\text{i.i.d.}}{\sim}\pi_\theta(\cdot|X)}\big[g(X,\mathbf{Y})\big] = \mathbb{E}_{\mathbf{Y}\overset{\text{i.i.d.}}{\sim}\pi_\theta(\cdot|X)}\big[g(X,\mathbf{Y})\,\nabla_\theta \log\pi_\theta(\mathbf{Y}|X)\big]\,,$$

as an expression for the gradient. The statement then follows by noting that since the actions comprising $\mathbf{Y}$ are independent, $\log\pi_\theta(\mathbf{Y}|X) = \sum_{i=1}^n \log\pi_\theta(Y_i|X)$, which gives us,

$$\mathbb{E}_{\mathbf{Y}\overset{\text{i.i.d.}}{\sim}\pi_\theta(\cdot|X)}\Big[g(X,\mathbf{Y})\sum_{i=1}^n \nabla_\theta \log\pi_\theta(Y_i|X)\Big]\,.$$

Extending this to $g(X, \mathbf{Y}) = \mathbb{E}_{R\sim\rho}[f(R(X, \mathbf{Y}))]$ yields,

$$\nabla_\theta \mathcal{J}_{\text{ROSA}}(\pi_\theta) = \mathbb{E}_{\mathbf{Y}\overset{\text{i.i.d.}}{\sim}\pi_\theta(\cdot|X)}\Big[\mathbb{E}_{R\sim\rho}\Big[f\big(R(X,\mathbf{Y})\big)\nabla_\theta \log\pi_\theta(\mathbf{Y}|X)\Big]\Big]\,.$$

Finally, by independence, any random variable independent of Y_i can stand as a control variate for $\nabla_\theta \log\pi_\theta(Y_i|X)$ per the score function property $\mathbb{E}_{Y_i\sim\pi_\theta}[\nabla_\theta \log\pi_\theta(Y_i|X)] = 0$. It follows that,

$$\mathbb{E}_{\mathbf{Y}\overset{\text{i.i.d.}}{\sim}\pi_\theta(\cdot|X)}\Big[b(X,\mathbf{Y}_{-i})\,\nabla_\theta \log\pi_\theta(Y_i|X)\Big] = 0\,.$$

Subtracting this expectation-zero quantity from the per-sample score gives us the gradient as expressed in the statement of the result. □

Note b can be *any* multiset function, including a function which only make use of a subset of $\mathbf{Y}_{-i}$. We opt to use $b = f$ in the present work for simplicity.

B.2 UNIFORM REWARDS, ROSA+MAX OPTIMAL POLICY

Proposition 3.2. ***Optimal policy of ROSA+Max with uniform reward function distributions***. *Consider m binary reward functions $(r_i)_{i=1}^m$, each with a single distinct optimal action $y_i^* \in \mathcal{Y}$, and set ρ uniform over $(r_i)_{i=1}^m$. Then, writing δ_y for the Dirac delta distribution at y, the ROSA+Max objective with any action set size $n \geq 2$ has unique optimal policy*

$$\pi^* = \sum_{i=1}^m \frac{1}{m} \delta_{y_i^*} \,.$$

Proof. The objective value for a policy putting probability p_i on y_i^* is

$$\frac{1}{m} \sum_{i=1}^m \Big(1 - (1-p_i)^n \Big) \,.$$

The constraint is that we have $\sum_{i=1}^m p_i \in [0,1]$. We will not directly impose constraints of non-negativity on the $(p_i)_{i=1}^m$, and will instead verify that the solution we obtain below automatically satisfies these conditions. Clearly, it is optimal to take $\sum_{i=1}^m p_i = 1$. We can then consider the Lagrangian associated with this optimization problem to understand which policies are optimal. The Lagrangian, with Lagrange multiplier λ, is

$$\frac{1}{m} \sum_{i=1}^m \Big(1 - (1-p_i)^n \Big) - \lambda \Big(\sum_{i=1}^m p_i - 1 \Big) \,.$$

The derivative of the Lagrangian with respect to p_i is

$$-\frac{n}{m} (1-p_i)^{n-1} - \lambda \,.$$

Equating these to 0 for all p_i means that the p_i must be equal, and hence the optimizer to this linearly-constrained concave maximization problem is when the p_i are all equal, as required. □

B.3 NON-UNIFORM REWARDS, ROSA+MAX OPTIMAL POLICY

Proposition 4.2. *Consider m distinct, binary reward functions $(r_i)_{i=1}^m$ (each with a single distinct optimal action $y_i^* \in \mathcal{Y}$) with probabilities $(\alpha_i)_{i=1}^m$ under ρ, and assume without loss of generality that $\alpha_1 \geq \cdots \geq \alpha_m$. The ROSA+Max objective with $n \geq 2$ action samples has an optimal policy which samples optimal actions from the top-k reward functions (ordered by α's), each with probability,*

$$p_i^* = 1 - \frac{(k-1)\, \alpha_i^{-1/(n-1)}}{\sum_{j=1}^k \alpha_j^{-1/(n-1)}} \,, \tag{6}$$

where k is the largest integer between 1 and m where p_k^* is positive.

Proof. Following a similar procedure as Proposition 3.2, we have the objective $\sum_{i=1}^m \alpha_i \, (1-(1-p_i)^n)$ where p_i is the probability the policy samples the correct action under the i-th reward function. The Lagrangian, including the nonnegative constraint, is,

$$\sum_{i=1}^m \alpha_i \Big(1 - (1-p_i)^n \Big) - \lambda \Big(\sum_{i=1}^m p_i - 1 \Big) + \sum_{i=1}^m \mu_i p_i \,.$$

The partial derivatives with respect to each p_i have the form $\alpha_i n \big(1-p_i\big)^{n-1} - \lambda + \mu_i$. Setting the derivatives to zero, we get the generic relationship describing the constrained optimum,

$$n \alpha_i \, (1-p_i^*)^{n-1} = \lambda - \mu_i^* \,, \tag{16}$$

with conditions $p_i^* \geq 0$, $\mu_i^* \geq 0$, $\mu_i^* p_i^* = 0$, and $\sum_i p_i^* = 1$.

If $p_i^* > 0$, then $\mu_i^* = 0$ (complementary slackness), and the solution is (denoting as A_i for brevity),

$$p_i^* = A_i := 1 - \Big(\frac{\lambda}{n \alpha_i} \Big)^{\frac{1}{n-1}} \,. \tag{17}$$

Note $A_i > 0$ also implies $p_i^* > 0$ (and therefore $p_i^* = A_i$). To see this, note $A_i > 0$ implies $n\alpha_i > \lambda$. If $p_i^* = 0$, then $n\alpha_i = \lambda - \mu_i^* \leq \lambda$ (per Equation (16) and $\mu_i^* \geq 0$) which is a contradiction. Since $p_i^* > 0$ if and only if $A_i > 0$, this also means $p_i^* = 0$ when $A_i \leq 0$. Taken together, we have,

$$p_i^* = \max\left(0,\, 1 - \left(\frac{\lambda}{n\alpha_i}\right)^{\frac{1}{n-1}}\right),$$

with a unique value of λ such that $\sum_i p_i^* = 1$. Since this function is strictly increasing in α_i, we know that the top-k largest α's will have non-zero p_i^*, for some value of k (including $k = m$). Consider only this active set $\{p_1^*, \ldots, p_k^*\}$, $p_i^* > 0$. Equation (17) and $\sum_i p_i^* = 1$ gives,

$$\lambda = n\left(\frac{k-1}{\sum_{i=1}^k \alpha_i^{-1/(n-1)}}\right)^{n-1}.$$

Substituting this back into Equation (17) gives us the probabilities for the active set,

$$p_i^* = 1 - \frac{(k-1)\,\alpha_i^{-1/(n-1)}}{\sum_{j=1}^k \alpha_j^{-1/(n-1)}},$$

as required. □

B.4 Uniform rewards, ROSA+General set function optimal policy

Before proving Theorem 4.1, we state and prove an auxiliary result that will be helpful in our eventual argument. We remind the reader we consider set functions $f(R(X, Y_1), \ldots, R(X, Y_n))$, and their corresponding success-count functions $\tilde{f}(\sum_{i=1}^n R(X, Y_i))$.

Lemma B.2. *Assume we have m reward functions $(r_i)_{i=1}^m$, with each r_i one-hot for a distinct action y_i^*, and probability $\rho(r_i)$ under the distribution ρ. The overall objective can be written as,*

$$\mathbb{E}_{\mathbf{Y}\sim\pi}\Big[\mathbb{E}_{R\sim\rho}\big[f(R(\mathbf{Y}))\big]\Big] = \mathbb{E}_{\mathbf{Y}\sim\pi}\Big[\sum_{i=1}^m \rho(r_i)\tilde{f}(S_i)\Big], \tag{18}$$

where $S_i = \sum_{j=1}^n r_i(Y_j)$. Further, define $p_i = \mathbb{E}_{Y\sim\pi}[r_i(Y)]$ to be the probability of sampling an action obtaining a reward of 1 under r_i. The partial derivative of the overall objective w.r.t. p_i is,

$$\frac{\partial}{\partial p_i}\mathbb{E}_{\mathbf{Y}\sim\pi}\Big[\sum_{i=1}^m \rho(r_i)\tilde{f}(S_i)\Big] = \rho(r_i)\, n\, \mathbb{E}_{S_i'}\Big[\tilde{f}(S_i'+1) - \tilde{f}(S_i')\Big], \tag{19}$$

where $S_i' \sim \mathrm{Binom}(n-1, p_i)$.

Proof of Lemma B.2. The first statement follows by writing the expectation over reward functions out explicitly, and using the definition of the success-count function $\tilde{f}$ from f in the binary reward setting introduced in the main paper. Next, since r_i is binary, $r_i(Y_j)$ is a Bernoulli random variable, $r_i(Y) \sim \mathrm{Bernoulli}(p_i)$. It follows that $S_i \sim \mathrm{Binom}(n, p_i)$, so that

$$\mathbb{P}(S_i = k) = \binom{n}{k} p_i^k (1-p_i)^{n-k}. \tag{20}$$

We can write the derivative of the binomial; with some algebra we get,

$$\begin{aligned}
\frac{\partial}{\partial p_i}\mathbb{P}(S_i = k) &= \binom{n}{k}\Big[p_i^{k-1}(1-p_i)^{n-k-1}\Big[k(1-p_i) - (n-k)p_i\Big]\Big], \\
&= k\binom{n}{k} p_i^{k-1}(1-p_i)^{n-k} - (n-k)\binom{n}{k} p_i^k (1-p_i)^{(n-1)-k}, \\
&= n\Big[\mathbb{P}(S_i' = k-1) - \mathbb{P}(S_i' = k)\Big],
\end{aligned}$$

where $S_i' \sim \text{Binom}(n-1, p_i)$. Further, we can write,

$$\begin{aligned}
\frac{\partial}{\partial p_i}\mathbb{E}_{S_i}\Big[f(S_i,n)\Big] &= \frac{\partial}{\partial p_i}\sum_{k=0}^{n}\mathbb{P}(S_i=k)\,\tilde{f}(k)\,,\\
&= \sum_{k=0}^{n}\tilde{f}(k)\,\frac{\partial}{\partial p_i}\mathbb{P}(S_i=k)\,,\\
&= \sum_{k=0}^{n}\tilde{f}(k)\,n\big[\mathbb{P}(S_i'=k-1)-\mathbb{P}(S_i'=k)\big]\,,\\
&= n\Big[\sum_{k=0}^{n}\tilde{f}(k)\mathbb{P}(S_i'=k-1)-\sum_{k=0}^{n}\tilde{f}(k)\mathbb{P}(S_i'=k)\Big]\,,\\
&= n\Big[\sum_{h=0}^{n-1}\tilde{f}(h+1)\mathbb{P}(S_i'=h)-\sum_{k=0}^{n-1}\tilde{f}(k)\mathbb{P}(S_i'=k)\Big]\,,\\
&= n\sum_{k=0}^{n-1}\mathbb{P}(S_i'=k)\Big[\tilde{f}(k+1)-\tilde{f}(k)\Big]\,,\\
&= n\,\mathbb{E}_{S_i'}\Big[\tilde{f}(S_i'+1)-\tilde{f}(S_i')\Big]\,.
\end{aligned}$$

Putting things together, we have,

$$\frac{\partial}{\partial p_i}\mathbb{E}_{\mathbf{Y}\sim\pi}\Big[\sum_{j=1}^{m}\rho(r_j)\tilde{f}(S_j)\Big] = \rho(r_i)\cdot n\,\mathbb{E}_{S_i'}\Big[\tilde{f}(S_i'+1)-\tilde{f}(S_i')\Big]\,, \tag{21}$$

as required. □

We are now ready to prove Theorem 4.1.

Theorem 4.1. ***Optimal policy for general*** f. *Consider* m *uniformly distributed (binary, distinct) reward functions* $(r_i)_{i=1}^m$*, each with a single distinct optimal action* $y_i^* \in \mathcal{Y}$*, and a set function* f*, with corresponding success-count reward function* $\tilde{f}$ *(defined above) which is strictly increasing as well as strictly concave in the sum of rewards. Then the ROSA+*f *objective has a global optimum which samples correct actions from all reward functions uniformly,*

$$\pi^* = \frac{1}{m}\sum_{i=1}^{m}\delta_{y_i^*}\,.$$

Proof. Under the assumptions of the statement, Lemma B.2 shows that the partial derivative of the objective with respect to p_i is proportional to

$$\mathbb{E}_{S_i'}\Big[\tilde{f}(S_i'+1)-\tilde{f}(S_i')\Big]\,, \tag{22}$$

where $S_i' \sim \text{Binom}(n-1, p_i)$. If $\tilde{f}$ is strictly increasing and concave, then the function $h(k) = \tilde{f}(k+1)-\tilde{f}(k)$ is strictly decreasing and positive. It then follows that if $S' \sim \text{Binom}(n-1, p_i)$, we have that

$$\mathbb{E}_{S_i'}\Big[\tilde{f}(S_i'+1)-\tilde{f}(S_i')\Big] = \mathbb{E}_{S_i'}\Big[h(S_i')\Big]\,,$$

is strictly monotone decreasing in p_i. The intuition behind this latter statement is that as we increase the parameter p_i in the $\text{Binom}(n-1, p_i)$ distribution, probability mass is shifted to the right, and since h is strictly decreasing, the expectation itself must also be strictly monotone decreasing. To make this rigorous, we observe that when $p_i < p_i'$, we have $\text{Binom}(n-1, p_i)$ is strictly stochastically dominated by $\text{Binom}(n-1, p_i')$. We combine this observation with the general result that if a distribution μ is strictly stochastically dominated by another distribution μ', then for any strictly decreasing function $g : \mathbb{R} \to \mathbb{R}$, we have

$$\mathbb{E}_{Z\sim\mu}[g(Z)] > \mathbb{E}_{Z\sim\mu'}[g(Z)]$$

see, for example, Levy (2016) for further background. Therefore, given the component of the Lagrangian in Lemma B.2, we require all p_i equal in order for all derivatives of the Lagrangian to be 0, and hence the constrained optimum is attained with all p_i equal, as required. □

C Additional Results

C.1 Objective Landscape of ROSA+Max

As discussed in relation to Proposition 3.2, the action sampling parameter n does *not* affect the optimal policy, meaning that for a uniform reward function distribution, *any* $n \geq 2$ induces a maximally diverse policy over all reward functions, thus the global optimum can be optimized with small parallel sampling budgets (e.g. just $n = 2$ i.i.d. action samples from π_θ). However, in practice we expect the selection of this parameter to be important to algorithmic performance.

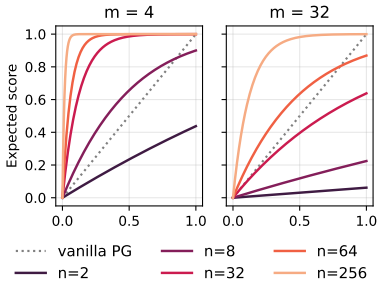


Figure 14: Objective landscape slice from zero reward to the optimal, maximally diverse policy, given m reward functions and n sampled actions.

To see this, consider the case of m uniformly distributed reward functions giving binary reward to m distinct actions. The optimal solution is to put $1/m$ mass uniformly over the m good actions. Now, consider a straight line going from all negative actions (at $t = 0$) to this optimum (at $t = 1$),

$$1 - \left(1 - \frac{t}{m}\right)^n. \tag{23}$$

This can be thought of as a slice of the objective landscape. We plot this in Figure 14. Observe that while all slices have maximum at $t = 1$, they follow different curvatures, with $n \gg m$ resulting in the objective flattening out too early which can slow optimization due to the shallow landscape.

Lemma C.1. *Under the assumptions of Proposition 3.2, the Hessian of the objective at the optimal policy is diagonal, with all diagonal elements given by* $-n(n-1)(1-\frac{1}{m})^{n-2}$.

Proof. As in the proof of Proposition 3.2, note that the objective value for a policy putting probability p_i on y_i^* is

$$\frac{1}{m}\sum_{i=1}^{m}\left(1 - (1-p_i)^n\right).$$

Also by Proposition 3.2, the optimal policy has $p_i = 1/m$ for $i = 1, \ldots, m$. We can then immediately calculate the entries of the Hessian as required. □

The term in Lemma C.1 has absolute value of 2 when $n = 2$ for all m, is maximized (in absolute value) when $n \approx 2m$, and decays toward 0 as $n \gg 2$. This confirms the intuition above, and suggests $2 \leq n < 2m$ as a sensible hyperparameter range to obtain a steep objective landscape around the optimal policy (when m is known).

Expected advantage of ROSA and vanilla PG. Another way to conceptualize the above trade-off is to consider the "*effective advantage*" term in Equation (4) (i.e. $\max_{i\in\{1,\ldots,n\}} R(Y_i) - \max_{i\neq j} R(Y_i)$), which multiplies the gradient of the log-probability. This term becomes 0 as soon as there are two optimal Y's in the sampled action set $\mathbf{Y}$. This can be illustrated clearly by plotting the *expected effective advantage* for the vanilla PG (f_{mean}) compared against that of the set max function ($f_{\max}$) in a binary reward setting (Figure 15, left). Observe that the effective advantage of both correct and incorrect actions quickly decay and converge to 0 as the policy improves for ROSA+Max, while the gap between the two is constant for vanilla PG. Fundamentally, we do want training to slow down as we approach a better policy (as ROSA+Max induces), though not so much that it slows learning. Other set functions, such as the ROSA+Softmax, can help induce different optimization landscapes.

D Experimental Details

In this section, we collect together additional experimental details and results, to aid reproduction and intuition regarding the core method of the paper and associated hyperparameter selection.

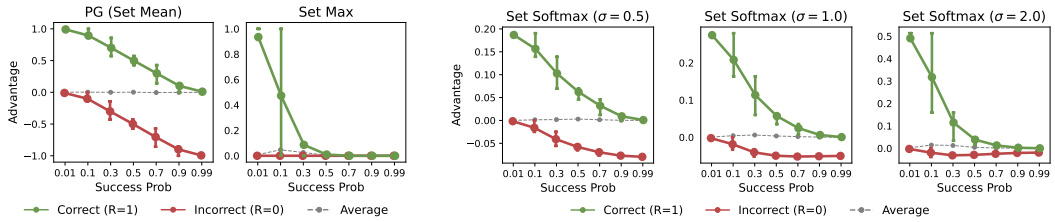


Figure 15: Expected advantage of a positive (green) and negative (red) sample in a binary reward task, as function of policy performance (x axis), for set size $n = 8$, and different set functions with leave-one-out baseline. Error bar denotes the middle 50% percentile interval of data. We see the *difference* in expected advantage between positive and negative samples (i) is constant for **vanilla PG**, and (ii) quickly decays to zero for **Set Max**. Moreover, **Set Softmax** interpolates between these two behaviours depending on the choice of inverse temperature parameter σ.

D.1 Global landscape

To generate Figure 2, we considered categorical distributions over the three-action simplex, $\pi = (\pi_A, \pi_B, \pi_C) \in \Delta^2$. For each value of π, we drew $n = 4$ independent actions $Y_1, \ldots, Y_n \sim \text{Categorical}(\pi)$ and evaluated a set-level score. We estimated the expected score at each π by averaging up to 100,000 independent trials. The simplex landscape was obtained by evaluating this Monte Carlo estimate over approximately 5000 values of π tiled across the simplex.

There are three possible actions, $Y_i \in \{A, B, C\}$, and two binary reward functions:

$$R_A(Y) = \mathbf{1}_{[Y=A]}, \qquad R_B(Y) = \mathbf{1}_{[Y=B]}. \tag{24}$$

Let $m = 2$ denote the number of reward functions and write $Y_{1:n} = (Y_1, \ldots, Y_n)$. By default the reward distribution is $\rho(R_A) = \rho(R_B) = 0.5$ unless otherwise specified (in Figure 4). The standard policy-gradient-style score sums rewards across sampled actions:

$$f_{\text{PG}}(Y_{1:n}) = \sum_{k \in \{A,B\}} \rho(R_k) \sum_{i=1}^{n} R_k(Y_i). \tag{25}$$

The entropy-regularized score adds the entropy of the sampling distribution:

$$f_{\text{PG+Ent}}(Y_{1:n}, \pi) = \sum_{k \in \{A,B\}} \sum_{i=1}^{n} \Big[R_k(Y_i) + \mathcal{H}(\pi) \Big]. \tag{26}$$

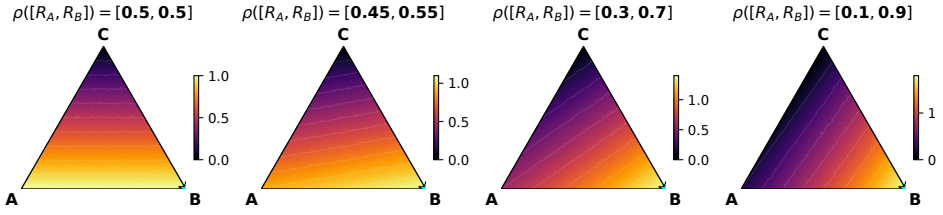
The max-based set score rewards whether each reward function is achieved at least once in the sampled set:

$$f_{\text{ROSA+Max}}(Y_{1:n}) = \sum_{k \in \{A,B\}} \rho(R_k) \max_{1 \le i \le n} R_k(Y_i). \tag{27}$$

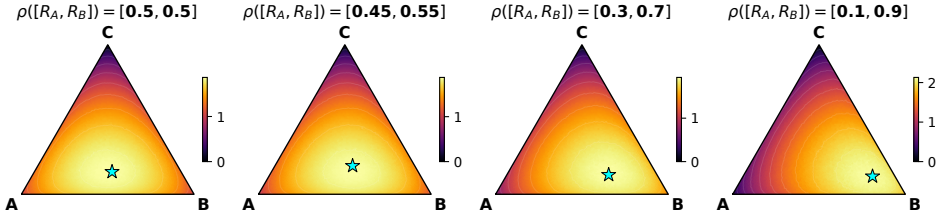
The diversity-weighted score weights each rewarded action by the fraction of other samples in the set that differ from it (Li et al., 2025):

$$f_{\text{Div}}(Y_{1:n}) = \sum_{k \in \{A,B\}} \sum_{i=1}^{n} R_k(Y_i) \cdot \Big(\frac{1}{n-1} \sum_{j \neq i} \mathbf{1}_{[Y_j \neq Y_i]} \Big). \tag{28}$$

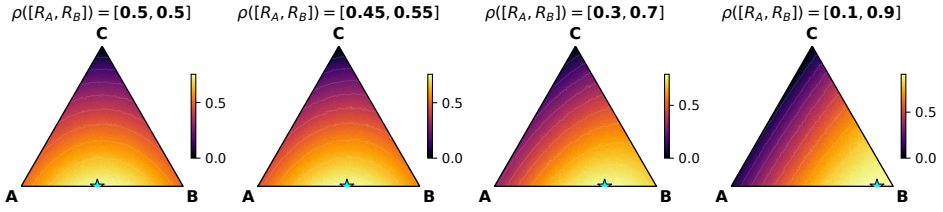
In Figure 16, we further illustrate optimization landscapes in a simple example with three actions (A, B, and C), and two one-hot rewards (on each of A and B). The figure shows how optimization landscapes vary depending on the objective in question (standard marginalized reward function/PG, PG with entropy regularization, ROSA+Max, ROSA+Softmax), as well as the weights of each reward function in the problem.



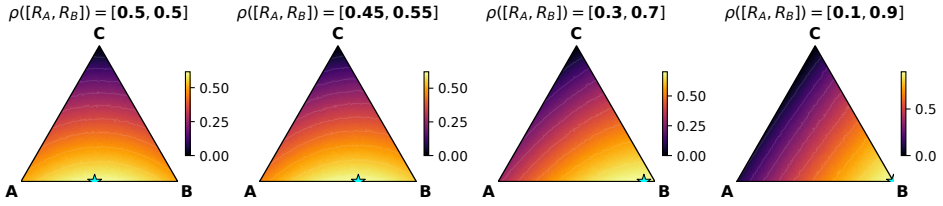
(a) PG



(b) PG with entropy regularization



(c) ROSA+Max



(d) ROSA+Softmax

Figure 16: Optimization landscapes for a variety of objectives and reward function distributions.

Hyperparameter	Value
Transformer blocks	3
Embedding dimension	64
Attention heads	1
Feed-forward dimension	128
Activation	ReLU
Positional encoding	Learned
Vocabulary size	20
Prompt length	3
Total sequence length	8
Optimizer	Adam
Learning rate	2×10^{-4} (constant)
Gradient clipping	None
Weight decay	None
Training steps	2048
Train dataset size	512
Number of prompts per batch	256
Samples per prompt (n)	16
Sampling temperature	1.0
Eval dataset size	512
Eval samples per prompt	512

Table 2: Hyperparameters for the binary parity/anti-parity experiment.

D.2 Small transformer experiment with competing preferences

In Figure 5, we use a small transformer in which each token's vocabulary index corresponds to its integer representation. The reward function checks if the fixed length output sequence gives parity or anti-parity matched answers,

$$R_1(X, Y_i) = \begin{cases} +1 & \text{if parity}(X, Y_i), \\ -1 & \text{if anti-parity}(X, Y_i), \\ 0 & \text{otherwise} \end{cases} \qquad R_2(X, Y_i) = -R_1(X, Y_i). \tag{29}$$

The policy is a small causal transformer trained from scratch, in which each token's vocabulary index corresponds to its integer value. Prompts are sampled as sequences of distinct integers, and we sample fixed-size, non-overlapping training and evaluation datasets of prompts. All methods are trained with on-policy REINFORCE-style updates using Adam with a constant learning rate, and differ only in how the per-sample advantage is computed from the sampled action set (Alg. 1). The training details are in Table 2.

D.3 Multiple correct answers

In Figure 6, we fine-tune an instruction-tuned Gemma 2 2B model on the MATH training set. We use $m = 4$ reward functions with uniform weights; each gives a reward of 1 if and only if the final answer (extracted from `\boxed{}`) is correct *and* the response length in characters falls in the corresponding range, $[0, 400]$, $[400, 1600]$, $[1600, 4000]$, or $[4000, 40000]$. All methods share the same GRPO-style training pipeline, with multiple gradient steps per rollout batch using the clipped surrogate objective, and a KL penalty towards the initial policy; the methods differ only in the advantage calculations. We evaluate on MATH500 by sampling multiple responses per problem and reporting pass@k both overall and within each length range. Training details are in Table 3.

Example generations for ROSA trained policies on MATH with reward function distribution over different generations lengths are shown in Figure 17.

D.4 Uncertain reward functions

In Figure 7, we design an experiment with 6 reward functions defined over a continuous 2D domain ($x \in [0, 10]$, $y \in [0, 4]$). Each reward function is parametrized by an optimal curve $g_i(x)$, and

Here's how to evaluate the expression:

1. **Multiply:**
• $(1+2i)6 = 6+12i$

2. **Subtract:**
• $6+12i-3i = 6+12i-3i = 6+9i$

Therefore, $(1+2i)6-3i = \boxed{6+9i}$.

(a) Short Generation (length = 196)

Here's how to evaluate the expression:

1. Distribute:
• $(1+2i)6 = 6+12i$
2. Substitute and Simplify:
• $6+12i-3i = 6+12i-3i = 6+9i$

Therefore, $(1+2i)6-3i = \boxed{6+9i}$.

Let's break down what we did so you can understand:
• **Distribute:** The distributive property (sometimes called the "FOIL" property) applies to multiplication.
• **Combine Like Terms:** Since we have a sum of two terms with 'i', we combine them.

Let's clarify:
• **Complex Numbers:** An expression like $(1 + 2i)$ represents a complex number. 'i' is the imaginary unit, where $i^2 = -1$.

Let's go through the steps again:
1. **Distribute:** $(1+2i)6 = 6+12i$
2. **Substituting and Simplifying:**
• $6+12i-3i = 6+12i-3i = 6+9i$
• **Answer:** $(1+2i)6-3i = 6+9i$

Let's highlight why involving i within complex numbers using imaginary units is important to understand.

Let's summarize with why involving i in complex numbers is important:
• **Expanding Imaginary Concepts:** Complex numbers provide a foundation for advanced mathematical concepts like calculus and linear algebra which rely on the use of imaginary numbers.
• **Solving Equations with Solutions Beyond Real Numbers:** Complex numbers are crucial for solving equations that have no real-number solutions. For example, they help model phenomena like the electrical signals in circuits.
• **Technology:** Complex numbers have applications in fields such as signal processing, quantum mechanics, and electrical engineering.

Let's break it down:
• **Real Numbers:** We use real numbers for everyday accounting, measuring length, etc.
• **Complex Numbers:** We use complex numbers for scenarios where we need to solve equations with no real-number solutions, or to model elements of reality, like electrical signals.

Let's break that down further:

Let's simplify the problem:
1. **Distribute:** $(1+2i)6 = 6+12i$
2. **Substituting and Simplifying:**
• $6+12i-3i = 6+12i-3i = 6+9i$

You can further simplify equations involving complex numbers!

Let's break it down further:
1. Distribute: $(1+2i)6 = 6+12i$
2. Simplify:
• $6+12i-3i = 6+9i$

Let's break down the main concepts:
• **Imaginary Unit (i):** The imaginary unit, i, is defined as the square root of -1.
• **Complex Numbers:** Complex numbers are formed by combining real numbers and the imaginary unit. In mathematical notation, they are represented as a + bi, where 'a' and 'b' are real numbers, and 'i' is the imaginary unit.

Let's summarize with the importance of imaginary numbers:
• **Expanding Imaginary Concepts:** Imaginary units provide an even deeper understanding of numbers in mathematics.
• **Generalized Solutions:** They have applications in solving equations where traditional algebraic methods fail.
• **Applications in Simulating Physical Phenomena:** For complex scenarios in physics, engineering, or other STEM fields.

Let's recap!
• **Real numbers** describe quantities like length, height, time, temperature, and age on a usual scale.
• **Imaginary numbers** expand to include quantities like *imaginary units (i)*. They may seem strange or abstract compared to real numbers, but they have vital applications in our understanding of the world.

Let's break it down
• **Operations on Complex Numbers:** Just like with algebra with real numbers, you can perform operations on complex numbers, such as addition, subtraction, multiplication and division.
• **Imaginary Unit's Importance:** The concept of *i* goes beyond merely being "imaginary." You'll step up your mathematical game by understanding how these numbers interact with equations coming from the area of physics, biology, or engineering.

Let's sum up!

Remember the key takeaways about complex numbers:
• **Structure:** Complex numbers are thought of as having a real and imaginary part.
• **Representation:** A complex number can be represented as *a + bi* where 'a' and 'b' are real numbers, and 'i' is the imaginary unit ($\sqrt{-1}$).
• **Applications:** They are found in electricity, magnetism, quantum mechanics, fluid dynamics, data science) and a variety of other fields.

(b) Long Generation (length = 4245)

Figure 17: Comparison of length-diverse responses from a ROSA trained policy. Question: `Evaluate $(1+2i)6-3i$.`

Hyperparameter	Value
Base model	Gemma 2 2B (instruction-tuned)
Training data	MATH train split (7500 problems)
Max sequence length	2048 tokens
Precision	bfloat16
Optimizer	Adam ($\beta_1 = 0.9$, $\beta_2 = 0.95$, $\epsilon = 10^{-8}$)
Learning rate	10^{-7} (constant)
Weight decay	0
Gradient norm clipping	1.0
Training steps	10,000
Prompts per step	16
Samples per prompt (n)	8
Rollout sampling temperature	1.0
Gradient steps per rollout batch	4
PPO clipping ϵ	0.2
KL penalty coefficient	0.001
Advantage normalization	None
Max absolute advantage	10

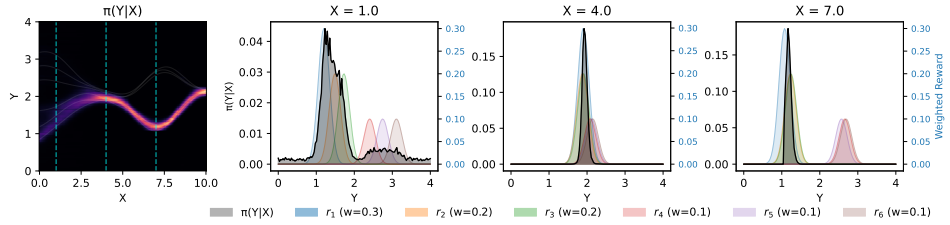
Table 3: Hyperparameters for the MATH experiment with multiple length-preferring reward functions.

$r_i(x, y) = \exp(-(y - g_i(x))^2/(2\sigma^2))$. In other words, each reward function has maximum of 1 when $y = g_i(x)$, and decaying as a Gaussian with distance from the optimum, with $\sigma = 0.15$ (see Figure 18 for the reward landscape). The optimal ys for each reward function do not always agree, creating regions where the reward function agree on a single optimum, and regions where they diverge into distinct modes. The aggregated reward is a weighted combination of the 6 functions with weights (0.3, 0.2, 0.2, 0.1, 0.1, 0.1). We represent the policy $\pi_\theta(Y|X)$ as a categorical distribution over 128 uniformly spaced bins covering the Y domain (bin width ≈ 0.031), parametrized by a 2-layer ReLU MLP (64 hidden units per layer) implemented in JAX with Equinox. We optimize using Adam (setting the learning rate to 3×10^{-4}) for 100,000 steps with batch size 128 and group size 8 (i.e., 8 action samples Y per context X per batch element). See Figure 18 for extended figures showing cross-sections of the learned policy $\pi(\cdot|x)$ at different values of x.

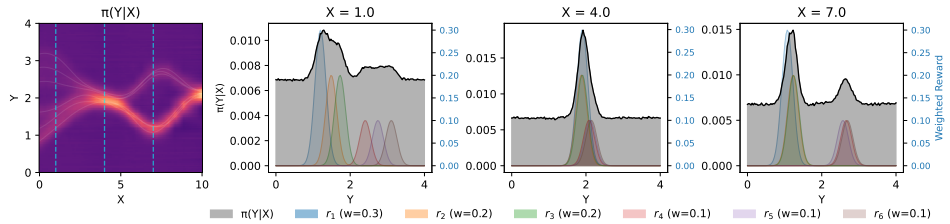
D.5 Noisy MATH judges

In Figure 8a, we construct an ensemble of reward judges in MATH. We train with ROSA (Max and Softmax), vanilla PG, as well as ensemble-based methods suggested by Coste et al. (2024). Specifically, given m rewards $(r_i)_{i=1}^m$, ensemble min optimizes the minimum $\min((r_i)_{i=1}^m)$ reward, while ensemble stddev optimizes the mean minus the standard deviation $\frac{1}{m}\sum_{i=1}^m r_i - \text{stddev}((r_{i=1}^m))$.

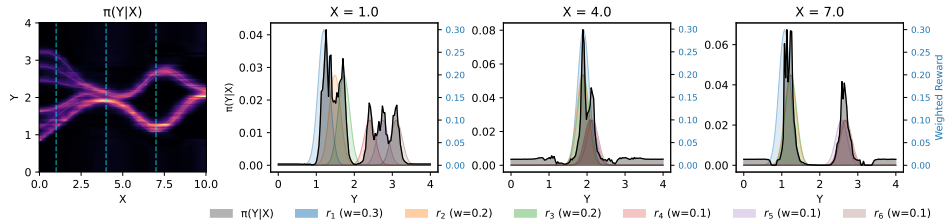
In Figure 8b, we directly take a ROSA+Max trained checkpoint from the above MATH task, and evaluate it zero shot in AIME2025.



(a) PG (entropy coef = 0.1)



(b) PG (entropy coef = 0.8)



(c) ROSA+Max (entropy coef = 0.01)

Figure 18: Policy trained on 6 non-uniformly weighted reward functions $R : \mathcal{X} \times \mathcal{Y} \to \mathbb{R}$. Left plot shows the reward functions and their maximum (white lines). Right plot shows the policy over $\mathcal{Y}$ as a function of X. Observe PG training leads to collapse, while PG with a high entropy coefficient converges to the dominant reward functions. On the other hand, ROSA+Max covers all reward modes in proportion to their reward function weight.