

# ONE MODEL, MANY GOALS: META-LEARNING PREFERENCE-CONDITIONED ALIGNMENT FOR LIFE-LONG LLM AGENTS

**Fatemeh Nourzad**  
The Ohio State University  
nourzad.1@osu.edu

**Daouda Sow**  
The Ohio State University  
sow.53@osu.edu

**Yingbin Liang**  
The Ohio State University  
liang.889@osu.edu

**Ming Shi**  
University at Buffalo, State University of New York  
mshi24@buffalo.edu

**Ming Zhang**  
Google  
mingzhang@google.com

**Yunxuan Li**  
Google  
yunxuanli@google.com

**Eylem Ekici**  
The Ohio State University  
ekici.2@osu.edu

**Ness Shroff**  
The Ohio State University  
shroff.11@osu.edu

## ABSTRACT

Deployed AI agents increasingly face *evolving preference goals*: user intent shifts, contexts change acceptable risk, and constraints update over time, so a single deployed LLM policy should re-target behavior on the fly *without* weight updates at deployment. Standard Reinforcement Learning from Human Feedback (RLHF) collapses multiple objectives into a single scalar reward, yielding brittle trade-offs. Meanwhile, common preference-conditioned *LLM alignment* pipelines often sample one preference per update and rely on linear scalarization, which can (i) weaken sensitivity to the preference signal through interference across conflicting updates and (ii) under-cover non-convex trade-off regions. We propose MERIDIAN (*Meta-Learning for Preference-Conditioned Alignment*), a bi-level framework that treats each preference as a separate alignment task: an inner loop performs preference-specific adaptation in isolation, and a first-order Reptile-style outer update consolidates the adapted parameters to preserve steerability across the preference simplex. We pair this with a smoothed Tchebycheff scalarization to improve coverage of non-convex trade-off regions. Empirically, MERIDIAN achieves denser Pareto coverage, better access to extreme goal modes, and improved performance on unseen preferences, supporting inference-time goal re-targeting. We also provide a generalization result showing how optimizing an empirical objective over sampled preferences can transfer to unseen preferences.

## 1 INTRODUCTION

As agents move from controlled benchmarks to long-term deployment, they must navigate a world that does not sit still. A customer-support agent faces alternating periods of routine queries and high-stakes crises; a writing assistant must pivot between playful, creative fiction and sensitive, evidence-based advice; a scientific agent must shift from broad exploratory brainstorming to safety-critical experimental support. In these real-world settings, even if the model is *trained once* and deployed without further weight updates, the *alignment target itself* can change after deployment: users revise goals, contexts shift the acceptable risk profile, and organizational constraints evolve over time. A capable long-horizon agent must therefore do something deceptively hard: it must *re-target its behavior on the fly* while remaining consistent with human preferences and safety requirements. (Ouyang et al., 2022; Bai et al., 2022a)

Large language model (LLM) alignment is inherently multi-objective. Standard Reinforcement Learning from Human Feedback (RLHF) pipelines typically collapse diverse and often conflict-

ing objectives—such as helpfulness and harmlessness—into a single scalar reward, implicitly committing the policy to one fixed trade-off. In lifelong-agent settings, this “train once, deploy once” alignment becomes brittle when preference goals evolve.

A natural alternative is *preference-conditioned LLM alignment*: a user (or downstream system) specifies a preference vector, and the agent adjusts its behavior accordingly. The ideal outcome is a single deployed policy that can instantly switch behaviors by conditioning on the current preference—without deployment-time retraining—and can respond meaningfully across a broad range of preferences, including unseen ones.

Achieving this “one model for many preferences” objective is algorithmically difficult. In common preference-conditioned *LLM alignment* pipelines, training often proceeds by sampling a preference vector  $w$  and optimizing a scalarized reward for a shared conditional policy, frequently using a linear scalarization such as  $w^\top r$  (Gupta et al., 2025; Wang et al., 2024b). In this setting, two difficulties become especially important. First, an *optimization gap*: when objectives conflict, updates induced by different preferences can interfere, causing the policy to drift toward a coarse compromise and respond only weakly to preference changes. Second, a *geometric gap*: linear scalarization can under-cover non-convex trade-off regions and therefore miss Pareto-optimal behaviors that matter for controllable retargeting. (Lin et al., 2024) Together, these effects can make a deployed lifelong-agent unreliable under evolving goals: it may become insensitive to the preference signal, or fail to reach parts of the desired trade-off space altogether.

We address these challenges with MERIDIAN (*Meta-Learning for Preference-Conditioned Alignment*). Our key modeling move is to treat each preference vector  $w$  as defining a distinct *alignment task*. MERIDIAN trains a single policy through a bi-level procedure: an inner loop performs preference-specific adaptation in isolation, reducing direct interference across conflicting updates, and an outer Reptile-style meta-update consolidates the adapted parameters into a shared initialization that remains sensitive to preference conditioning. To improve coverage beyond linear scalarization, we pair this with a smoothed Tchebycheff objective, which better captures non-convex trade-off regions that linear baselines can under-cover. (Lin et al., 2024)

Our experiments evaluate the central mechanism behind this lifelong-agent motivation using only inference-time goal changes. We perform *preference sweeps*: for many preferences, including unseen ones, we trace the resulting trade-off curve and test whether the deployed agent can re-target behavior immediately as preferences change, without updating its weights. In this setting, MERIDIAN achieves denser Pareto coverage and better access to extreme goal modes, indicating stronger inference-time goal re-targeting under preference drift.

**Contributions.** We make four contributions: (i) **Meta-learning formulation for preference-conditioned alignment**: We formulate preference-conditioned alignment as a generalization problem over the preference simplex and propose MERIDIAN, a bi-level meta-learning algorithm that trains a single policy to adapt to unseen preferences via conditioning. (ii) **Geometry-aware scalarization for controllable retargeting**: We show that, in this setting, smoothed Tchebycheff scalarization improves coverage of trade-off regions that linear scalarization can under-cover. (iii) **Empirical evidence for inference-time steerability**: In a conflicting-objective alignment setting, we provide empirical evidence—via preference sweeps and *Preference-Weighted Reward Shortfall* (PWRS) evaluation on held-out preferences—that MERIDIAN improves inference-time goal re-targeting robustness in a setting aligned with lifelong-agent requirements. (iv) **Generalization analysis over sampled preferences**: We derive an approximate empirical risk minimization (ERM)-style generalization result showing that, under standard assumptions, the gap between the learned policy and the optimal population policy decreases at rate  $O(L/\sqrt{M})$ , where  $L$  is a smoothness constant and  $M$  is the number of sampled preference tasks.

## 2 PROBLEM FORMULATION

We consider the alignment of a language model with respect to  $K$  diverse objectives (e.g., helpfulness and harmlessness). Let  $x \sim \mathcal{D}_x$  denote a prompt and let  $y$  denote a generated response. We assume access to a vector-valued reward function  $r(x, y) \in \mathbb{R}^K$ . A user’s desired trade-off among these objectives is specified by a preference vector  $w \in \Delta^K := \left\{ w \in \mathbb{R}_+^K : \sum_{i=1}^K w_i = 1 \right\}$ . To

motivate our framework, we contrast a preference-fixed scalarized objective with the preference-adaptive formulation studied in this paper.

**Preference-fixed Objective.** As a useful reference point, one can optimize a scalarized objective for a fixed preference vector  $w$ , yielding a policy  $\pi_w^*$  specialized to that trade-off:

$$\pi_w^* \in \arg \max_{\pi} \mathbb{E}_{x \sim \mathcal{D}_x, y \sim \pi(\cdot|x)} [S(r(x, y), w)], \quad (1)$$

where  $S(\cdot, w)$  is a scalarization function, commonly chosen to be linear,  $S_{\text{lin}}(r, w) = w^\top r$ . While effective for optimizing a single trade-off, Eq. equation 1 is ill-suited for lifelong agent settings where the preferred trade-off may not be fixed. Environment changes and user intent can induce time-varying preferences, which we denote by a preference trajectory  $\{w_t\}_{t=1}^T$ . Addressing such variation with preference-fixed optimization would require training and storing separate policies for many preferences across the continuous simplex  $\Delta^K$ , which is impractical. This motivates our goal: a *single deployed policy* whose behavior can change appropriately with  $w_t$  *without* updating its weights at deployment time.

**Preference-Adaptive Objective.** We seek to learn a *single* preference-conditioned policy  $\pi_\theta^*$ . We model preferences as random variables drawn from a distribution  $\rho$  over  $\Delta^K$  and optimize

$$\theta^* \in \arg \max_{\theta} \mathbb{E}_{w \sim \rho} [\mathbb{E}_{x \sim \mathcal{D}_x, y \sim \pi_\theta(\cdot|x, p(w))} [S(r(x, y), w)]], \quad (2)$$

where  $p(w)$  denotes a conditioning signal derived from the preference vector (e.g., a system-prompt prefix containing the numeric weights). This formulation transforms alignment into a *generalization* problem over preferences: the model must learn to recover near-optimal behavior for many  $w \sim \rho$ , including preferences not explicitly seen during training.

**Scalarization for Non-convex Trade-offs.** A critical component of Eq. equation 2 is the choice of scalarization function  $S$ . In many preference-conditioned alignment pipelines, scalarization is taken to be linear,  $S_{\text{lin}}(r, w) = \sum_{i=1}^K w_i r_i$ . Linear scalarization is simple, but it can under-cover non-convex regions of the Pareto frontier (Lin et al., 2024). To improve coverage in such regimes, we use a smoothed Tchebycheff scalarization. Tchebycheff-style scalarization is well established in multi-objective optimization; here we use its smoothed form to define the preference-conditioned alignment objective. Given a reference (utopia) point  $z^* \in \mathbb{R}^K$  and a smoothing parameter  $\mu > 0$ , we define

$$S_{\text{Tch}}(r, w) = -\mu \log \sum_{i=1}^K \exp\left(\frac{w_i(z_i^* - r_i)}{\mu}\right), \quad (3)$$

and the parameter-level objective

$$J(\theta; w) := \mathbb{E}_{x \sim \mathcal{D}_x, y \sim \pi_\theta(\cdot|x, p(w))} [S_{\text{Tch}}(r(x, y), w)]. \quad (4)$$

This provides a smooth approximation to the max-based Tchebycheff objective, which penalizes the worst-case weighted shortfall relative to the utopia point  $z^*$ . We emphasize that the scalarization is applied *inside* the expectation, at the level of each prompt-response pair, and then averaged across prompts. We adopt this semantics intentionally to measure preference satisfaction at the response level rather than only through aggregate average rewards. Compared with linear scalarization, this objective can access Pareto-optimal solutions in non-convex regions of the frontier, which can improve coverage of trade-offs across the preference simplex.

**The Empirical Meta-Objective.** Combining the adaptive formulation with the objective in Eq. equation 4, we seek to maximize the population meta-objective  $G(\theta) = \mathbb{E}_{w \sim \rho} [J(\theta; w)]$ . Since  $\rho$  is continuous and unknown, we approximate  $G(\theta)$  using  $M$  sampled preferences:

$$\widehat{G}_M(\theta) = \frac{1}{M} \sum_{m=1}^M J(\theta; w_m), \quad w_{1:M} \stackrel{\text{i.i.d.}}{\sim} \rho. \quad (5)$$

Thus, training seeks a single policy and conditioning mechanism that perform well on average over sampled preferences while generalizing to unseen preferences from the same domain.

### 3 MERIDIAN: META-LEARNING FOR PREFERENCE-CONDITIONED ALIGNMENT

In lifelong-agent settings, preferences may vary over time, but the deployed agent does *not* update its weights at deployment time; instead, it changes behavior by conditioning on the current preference. This section presents MERIDIAN (*Meta-Learning for Preference-Conditioned Alignment*), a bi-level training framework designed to preserve sensitivity to preference conditioning.

**Preferences as Alignment Tasks.** Our design is grounded in a central insight: preference-conditioned alignment is structurally a *multi-task learning* problem, where each preference vector  $w \in \Delta^K$  defines a distinct alignment task. This perspective motivates a *meta-learning* approach with a bi-level design (Algorithm 1): an inner loop that optimizes a batch of preference-specific objectives in isolation, and an outer meta-loop that aggregates these specialized capabilities into a single, steerable policy.

This contrasts with standard conditional training recipes that treat  $w$  primarily as an auxiliary conditioning signal—e.g., encoded in the prompt or injected through parameter conditioning—while optimizing a single shared update per iteration (Gupta et al., 2025; Wang et al., 2024b). When objectives conflict, such mixed updates can lead the shared conditional policy to collapse toward a coarse compromise that responds weakly to  $w$ , especially for extreme or rarely emphasized trade-offs. MERIDIAN addresses this by separating preference-specific optimization from cross-preference consolidation, improving preference sensitivity and generalization (see Section 4).

**The Meta-Optimization Loop.** The core of our approach is a first-order meta-update inspired by Reptile-style meta-learning frameworks (Nichol et al., 2018; Finn et al., 2017). Our objective is to learn a policy  $\pi_\theta$  that remains sensitive to the conditioning signal, so it can adapt meaningfully across preferences  $w \sim \rho$ , including unseen ones, rather than collapsing to a static average.

As shown in Algorithm 1, each meta-iteration samples a batch of preference vectors  $\{w_1, \dots, w_B\}$  with  $w_j \in \Delta^K$ . Crucially, these preferences are *not* mixed within a single gradient update. Instead, the current parameters  $\theta_t$  are branched into  $B$  independent copies, and each copy undergoes  $S$  steps of *inner-loop optimization* to approximately maximize the preference-specific objective  $J(\cdot; w_j)$ . This isolation lets each branch specialize before cross-preference consolidation, reducing direct interference from conflicting preference gradients and yielding preference-adapted parameters  $\{\theta_{t,1}, \dots, \theta_{t,B}\}$ . We then consolidate these specialized solutions using the Reptile-style meta-update

$$\theta_{t+1} \leftarrow \theta_t + \frac{\alpha}{B} \sum_{j=1}^B (\theta_{t,j} - \theta_t), \tag{6}$$

where  $\alpha > 0$  is the meta step size. Importantly, Eq. equation 6 is not an average of raw gradients evaluated at a shared parameter point. Each branch first performs  $S$  preference-specific inner updates, producing adapted parameters  $\theta_{t,j}$ , and the outer loop then averages these post-adaptation parameter displacements. Its intended advantage arises when branches take multiple preference-specific steps before consolidation, allowing specialization to occur before the shared model is updated. After training, the single deployed policy  $\pi_{\theta_T}$  can switch behavior at inference time by conditioning on the current preference  $w$ , without deployment-time weight updates.

**Inner Loop (Group Relative Policy Optimization).** To efficiently implement the inner-loop updates, we employ Group Relative Policy Optimization (GRPO) (Shao et al., 2024). Meta-learning is computationally more expensive than direct shared-policy training: branching into multiple preference-specific copies increases cost with the number of branches and inner steps. Applying standard PPO in the inner loop would further require training a separate value function for every sampled preference vector  $w_j$ , which is both expensive and potentially unstable because the scalarized objective depends on  $w_j$ . GRPO eliminates the need for a critic. For a fixed preference  $w_j$  and prompt  $x$ , we sample candidate responses from the current inner-loop policy  $\pi_{\theta_{t,j}^{(s)}}(\cdot | x, p(w_j))$ . Each candidate is scored and scalarized (Eq. equation 3), and the policy is updated with a GRPO step. This yields a practical first-order inner-loop update without a value network, making the bi-level framework tractable at the scale studied here.

**Algorithm 1** MERIDIAN: Meta-Learning for Preference-Conditioned Alignment

```

Require: Preference distribution  $\rho$ ; conditioner  $p(\cdot)$ ; initial parameters  $\theta_0$ ; meta-iterations  $T$ ; inner
steps  $S$ ; prompts distribution  $\mathcal{D}_x$ ; meta step size  $\alpha$ .
1: for  $t = 0, 1, \dots, T - 1$  do
2:   Sample a batch of preferences  $\{w_j\}_{j=1}^B \stackrel{iid}{\sim} \rho$ .
3:   for  $j = 1, \dots, B$  (inner updates under preference  $w_j$ ) do
4:     Initialize preference-adapted parameters  $\theta_{t,j}^{(0)} \leftarrow \theta_t$ .
5:     for  $s = 0, 1, \dots, S - 1$  do
6:       Sample prompts  $\{x_n\}_{n=1}^N \sim \mathcal{D}_x$  and a group of responses  $y_n \sim \pi_{\theta_{t,j}^{(s)}}(\cdot | x_n, p(w_j))$ .
7:       Score  $(x_n, y_n)$  with  $K$  reward models to obtain vector rewards  $r(x_n, y_n) \in \mathbb{R}^K$ .
8:       Form scalarized rewards using the smoothed Tchebycheff scalarization in Eq. equation 3.
9:       Update the policy with one GRPO step to obtain  $\theta_{t,j}^{(s+1)}$ .
10:    end for
11:    Set  $\theta_{t,j} \leftarrow \theta_{t,j}^{(S)}$ .
12:  end for
13:  Reptile-style meta update: update  $\theta_{t+1}$  using Eq. equation 6.
14: end for
15: return  $\theta_T$ .

```

**Geometry-Aware Scalarization.** Meta-learning improves *optimization* across preferences by reducing interference across conflicting updates, but it does not by itself resolve *geometric* limitations introduced by the scalarization choice. A preference-conditioned algorithm is only as expressive as the family of scalarized tasks it optimizes. In many preference-conditioned LLM alignment pipelines, scalarization is taken to be linear,  $S_{\text{lin}}(r, w) = w^\top r$  (Gupta et al., 2025; Wang et al., 2024b), which can recover only Pareto-optimal solutions on the convex hull of the frontier and may miss solutions in non-convex regions (Lin et al., 2024). Tchebycheff-style scalarization is well established in multi-objective optimization; our contribution is to pair its smoothed form with a bilevel preference-conditioned training procedure for LLM alignment. (Lin et al., 2024; Miettinen, 1999; Zhang & Golovin, 2020) In settings where alignment trade-offs are nonlinear, this can improve coverage of behaviors that linear scalarization under-covers. Accordingly, MERIDIAN uses the smoothed Tchebycheff scalarization in Eq. equation 3, so that the meta-update aggregates genuinely diverse preference-specialized behaviors rather than only the subset reachable through linear scalarization.

## 4 EXPERIMENTS

MERIDIAN is designed for preference-conditioned alignment in settings where an agent’s goals may change over time. We study two practical questions: (i) *generalization across preferences*: can a policy trained on finitely many sampled preferences behave sensibly on unseen  $w \sim \rho$ ? and (ii) *inference-time goal-following under evolving preferences*: when preferences vary over time, can the *same deployed policy* adjust behavior immediately via conditioning, without updating its weights at deployment time? We use inference-time preference sweeps as a controlled proxy for evolving goals: varying  $w$  represents changing user intent or context, and we measure whether a fixed deployed policy can shift behavior accordingly without deployment-time retraining.

### 4.1 EXPERIMENTAL SETUP

Our main empirical study focuses on a two-objective helpfulness–harmlessness setting using Qwen2-0.5B-Instruct as the base model. This provides a controlled testbed for preference-conditioned post-training: the model is large enough to exhibit meaningful alignment tensions while remaining small enough to support dense preference sweeps and ablations at reasonable cost. We train on prompts from HuggingFaceH4/ultrafeedback\_binarized (Cui et al., 2023) and evaluate on the 30k\_test split of PKU-Alignment/BeaverTails (Ji et al., 2023). During training, preferences are sampled over 11 fixed weights spanning  $\Delta^2$

using a uniform Dirichlet distribution ( $\alpha = 1.0$ ); evaluation sweeps a broader set of preferences, including ones not used during training. Generations are scored using public helpfulness and harmless reward models `Ray2333/gpt2-large-helpful-reward_model` and `Ray2333/gpt2-large-harmless-reward_model` with max-based reward normalization<sup>1</sup>. Unless otherwise stated, MERIDIAN uses  $B = 2$  preference branches,  $S = 10$  inner updates, and  $T = 400$  meta-iterations, and the final checkpoint is selected by validation performance on held-out prompts and preferences. Additional implementation details are provided in Appendix A.

## 4.2 EVALUATION METRICS

Our evaluation is designed to reflect a central requirement of lifelong agents: *goals (preferences) can change after deployment*, and the agent must remain robust without updating its weights. In our setting, the deployed agent is re-targeted by changing the preference vector  $w$  at inference time. Thus, evaluating a wide range of preferences—including *unseen* preferences not used during training—directly probes whether the policy can adapt its behavior on the fly, rather than memorizing a small set of training modes.

**Pareto coverage (preference sweeps).** We sweep  $w$  across  $\Delta^2$  at inference time and plot the resulting reward-model outcomes in the objective space. For each preference  $w$ , we condition the same trained policy on  $w$ , generate responses on an evaluation prompt set, score each response with the two reward models, and plot the resulting *average* helpfulness and harmless scores. Thus, each point corresponds to one preference setting  $w$ , and its coordinates are the resulting (aggregated) reward-model scores under that preference. We view this sweep as a controlled proxy for *evolving goals over time*: a lifelong agent should remain responsive as the desired trade-off changes, without requiring any parameter updates, mode collapse into a single compromise, or a few discrete behaviors. (see Section 4.4)

**Preference-Weighted Reward Shortfall (PWRS).** To quantify goal satisfaction under a specified preference, we report *Preference-Weighted Reward Shortfall*:  $\text{PWRS}(w) := -\mathbb{E}_{x \sim \mathcal{D}_x, y \sim \pi_\theta(\cdot|x, p(w))} [S_{\text{Tch}}(r(x, y), w)]$ . PWRS measures on average how far the achieved reward vector falls from the utopia point under the requested preference, under the same smoothed Tchebycheff semantics used during training. *Lower PWRS indicates better alignment*. Reporting PWRS on preferences not encountered during training therefore captures robustness to preference shift. We summarize PWRS over the evaluation preference set in Table 1. (see Appendix A.6)

## 4.3 BASELINES

We compare MERIDIAN against three baselines representing common paradigms in preference-conditioned alignment.

**1. MO-ODPO.** A direct preference-conditioned baseline based on (Gupta et al., 2025), which aggregates objective-specific rewards with a weighted linear scalarization  $w^\top r$  and optimizes the policy using DPO. This tests the efficacy of our meta-learning approach against established direct preference optimization methods.

**2. Grid-GRPO.** A shared-update GRPO baseline with linear scalarization. This baseline mirrors the MO-ODPO training recipe of sampling a preference and directly updating one shared conditional policy, while controlling for optimizer choice by replacing DPO with GRPO. We implemented this version using the same GRPO optimizer as MERIDIAN to ensure a fair, controlled comparison. This allows us to evaluate under the same on-policy optimization conditions as our method.

**3. Weighted-GRPO.** A shared-update GRPO baseline that uses the same smoothed Tchebycheff scalarization as MERIDIAN but omits the outer meta-update. This isolates the contribution of the bilevel meta-learning step from the contribution of geometry-aware scalarization alone.

<sup>1</sup>We found this normalization to be more stable than alternatives such as tanh squashing, which often led to policy collapse.

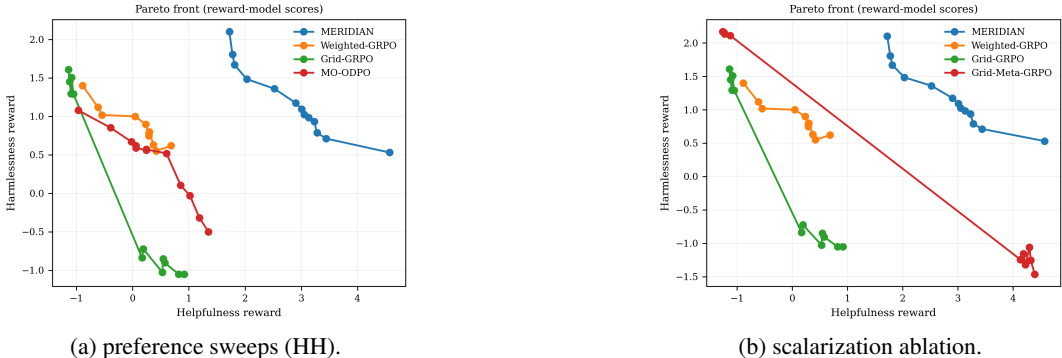


Figure 1: **Pareto coverage and scalarization ablation (reward-model scores).** **Left:** inference-time preference sweeps on HH show that MERIDIAN achieves broader, denser coverage across trade-offs. **Right:** keeping the meta-learning loop fixed but replacing Tchebycheff with linear scalarization (Grid-Meta-GRPO) creates a clear coverage gap in intermediate trade-off regions, indicating that geometry-aware objectives improve steerability under changing preferences.

#### 4.4 EMPIRICAL ANALYSIS OF PARETO COVERAGE

In Figure 1, we evaluate MERIDIAN through inference-time *preference sweeps*, which directly target a core lifelong-agent requirement: *goals can change after deployment*. In our formulation, changing goals correspond to changing the preference vector  $w$ . Importantly, the deployed policy does *not* update its weights at deployment time; it must re-target behavior immediately via conditioning on the current  $w$ .

**Why meta-learning is needed for inference-time goal retargeting.** MERIDIAN (blue) traces the strongest and widest trade-off curve. In contrast, all baselines exhibit reduced coverage and weaker steerability under changing preferences. Compared to MO-ODPO (red), MERIDIAN achieves substantially better trade-offs throughout the sweep. This improvement is not merely a consequence of switching to an on-policy optimizer: Grid-GRPO (green), which follows same shared-update linear-scalarization recipe under GRPO, still fails to match MERIDIAN’s frontier. The most controlled comparison is Weighted-GRPO (orange), which uses the same GRPO optimizer family and the same geometry-aware scalarization but performs direct shared-policy training without the outer meta-update. Its weaker performance suggests that geometry-aware scalarization alone is not sufficient: when the shared policy is updated immediately after each sampled preference, then reused for the next sampled preference, successive updates from conflicting trade-offs can interfere with and overwrite one another. As a result, the model may drift toward a compromise behavior that responds only weakly to changes in  $w$ , leading to reduced coverage of the trade-off space. The outer meta-update in MERIDIAN helps preserve sensitivity to changing preferences by separating preference-specific adaptation from cross-preference consolidation. Together, these results indicate that simply conditioning on  $w$  is not sufficient for a lifelong agent: maintaining *sensitivity* to changing goals requires mitigating gradient interference across conflicting preferences, which is precisely what the Reptile-style meta-update in MERIDIAN provides.

**Recovering the geometry of trade-offs under evolving goals.** A second observation from Figure 1a concerns the *geometry* of the achievable trade-off curve. The transition region between extreme helpfulness-oriented behavior and extreme harmlessness-oriented behavior appears non-convex in reward-model space, suggesting that the goal landscape induced by alignment objectives is not well-approximated by a single convex frontier. This matters in lifelong-agent settings because evolving goals are not restricted to “easy” convex interpolations: as  $w$  changes over time, the agent may need to reliably target intermediate regimes that lie inside such concave regions. MERIDIAN (blue) traces this transition smoothly, producing a set of distinct outcomes throughout the curve. In contrast, methods based on *linear* scalarization—MO-ODPO (red) and Grid-GRPO (green)—systematically under-cover this region, consistent with the known limitation of linear scalarization. In other words, geometry-aware scalarization is not merely a modeling choice: it expands the set of goal configurations that a deployed agent can reliably realize as preferences evolve.

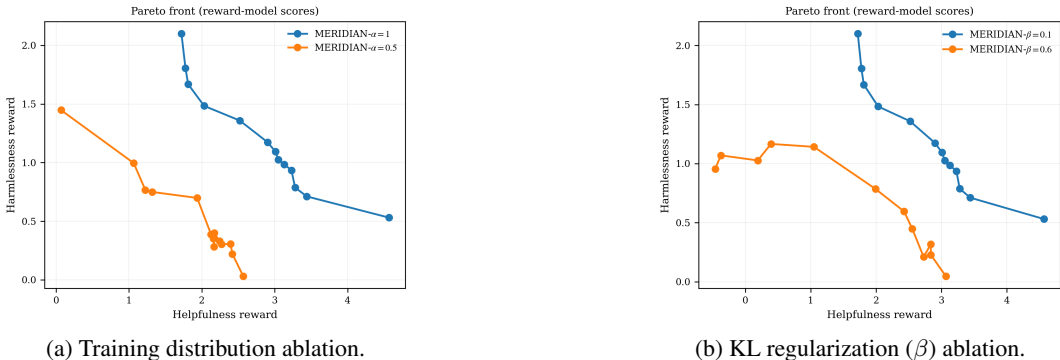


Figure 2: **Training distribution and KL regularization ( $\beta$ ) ablation.** **Left:**  $\alpha = 1.0$  yields a continuous frontier with strong coverage across goal settings while corner-heavy sampling ( $\alpha = 0.5$ ) underperforms in intermediate regimes, indicating reduced robustness to preference drift through mixed trade-offs. **Right:** Stronger anchoring ( $\beta = 0.6$ ) compresses the frontier and limits access to extreme goal modes while a lower penalty ( $\beta = 0.1$ ) expands the achievable trade-off set, improving goal re-targeting capacity under evolving preferences.

**Coverage of extremes for high-stakes goal modes.** A third takeaway from Figure 1a is performance at the *extremes* of the preference space. MO-ODPO (red) exhibits a characteristic failure mode: its outcomes concentrate near the middle and fail to reach high-helpfulness or high-harmlessness regimes, suggesting a compromise behavior. In contrast, MERIDIAN (blue) spans a substantially wider range of trade-offs, reaching higher rewards while maintaining coherent intermediate behavior as  $w$  varies. From a lifelong-agent perspective, this corresponds to robust *goal mode switching*: the same deployed policy can be steered into specialist regimes when required, without retraining, while still supporting smooth interpolation across less extreme goal settings.

#### 4.5 ABLATION STUDIES

We ablate key components of MERIDIAN to identify which design choices are most important for lifelong-agent behavior—i.e., reliable *goal re-targeting* as preferences evolve, without deployment-time weight updates.

**Why geometry-aware scalarization matters under evolving goals.** Figure 1b isolates the effect of the scalarization function. We compare MERIDIAN (blue) against Grid-Meta-GRPO (red), a variant that *retains the meta-learning outer loop* but replaces the smoothed Tchebycheff scalarization (Eq. 3) with a *linear* weighted sum. The resulting trade-off curve exhibits a pronounced coverage failure in the transition region between extremes: despite meta-learning, the policy under-covers parts of the trade-off space. The *scalarization* determines which goal configurations are “visible” as optimizable tasks. With linear scalarization, parts of the trade-off surface become unreachable, creating *blind spots* in the attainable trade-off set that a lifelong agent would experience as systematic failure when preferences drift into those regimes. In contrast, the geometry-aware Tchebycheff scalarization expands the set of attainable trade-offs, enabling MERIDIAN to robustly track evolving goals across a broader range of trade-offs.

**Sensitivity to the training preference distribution ( $\alpha$ ).** A lifelong agent must remain competent not only at a few “canonical” goal settings, but also as goals drift through intermediate regimes over time. To test how training-time exposure shapes this capability, Figure 2a varies the Dirichlet concentration parameter  $\alpha$  used to sample preferences  $w \sim \text{Dir}(\alpha)$  during training. When  $\alpha = 0.5$  (corner-heavy sampling), performance degrades noticeably in the interior of the trade-off curve. In contrast,  $\alpha = 1.0$  yields a more continuous and well-resolved frontier across both extremes and compromises. This highlights that if training over-emphasizes “pure” behaviors at the corners, the agent will exhibit weakness when preferences drift into mixed regimes.

**Stability under evolving goals (KL penalty  $\beta$ ).** A lifelong agent must remain anchored to safe, instruction-following behavior, yet be capable of substantial behavioral shifts when goals change. Figure 2b studies this tension through the KL-divergence penalty  $\beta$ . With stronger regularization ( $\beta = 0.6$ ), the policy remains tightly anchored to the initialization, which compresses the achievable trade-offs and prevents the model from reaching specialized extreme goal modes. Reducing the penalty ( $\beta = 0.1$ ) expands the frontier, allowing the policy to access a broader range of goal-conditioned behaviors.

## 5 THEORETICAL ANALYSIS: GENERALIZATION OVER PREFERENCES

A central premise behind preference-conditioned alignment is that a policy trained on a finite batch of preferences  $\{w_m\}_{m=1}^M$  should generalize to unseen preferences drawn from the distribution  $\rho$ . In this section, we formalize this intuition. We frame preference-conditioned alignment as an approximate ERM problem and derive a generalization bound that explicitly separates *statistical sampling error* (due to finite  $M$ ) from *optimization error*.

### 5.1 SETUP: POPULATION VS. EMPIRICAL OBJECTIVES

We analyze generalization through the scalarized objective  $J(\theta; w)$ . Let

$$G(\theta) = \mathbb{E}_{w \sim \rho}[J(\theta; w)], \quad \widehat{G}_M(\theta) = \frac{1}{M} \sum_{m=1}^M J(\theta; w_m).$$

Let  $\theta^*$  denote a population maximizer of  $G(\theta)$ , let  $\widehat{\theta}$  denote a maximizer of the empirical objective  $\widehat{G}_M(\theta)$ , and let  $\tilde{\theta}$  be the solution returned by MERIDIAN. We characterize the algorithm through an *optimization error*  $\varepsilon_{\text{opt}} \geq 0$ , defined as the empirical sub-optimality of the returned solution relative to the empirical maximizer:

$$\widehat{G}_M(\widehat{\theta}) - \widehat{G}_M(\tilde{\theta}) \leq \varepsilon_{\text{opt}}.$$

This formulation is solver-agnostic: it allows for imperfect optimization of the non-convex meta-objective while isolating the error due to not reaching an empirical maximizer exactly. In practice, MERIDIAN uses GRPO in the inner loop to reduce this optimization gap. To obtain a clean generalization statement, we impose the following regularity assumptions.

**Assumption 5.1 (Regularity of the Preference Landscape)** *We make the following standard assumptions regarding the scalarized objective  $J$  and the data generating process: 1. **Boundedness:** For all parameters  $\theta \in \Theta$  and preferences  $w \in \Delta^K$ , the objective is bounded such that  $J(\theta; w) \in [0, 1]$ . 2. **Lipschitz Continuity:** The mapping  $w \mapsto J(\theta; w)$  is  $L$ -Lipschitz with respect to the  $\ell_2$ -norm for all  $\theta \in \Theta$ , i.e.,  $|J(\theta; w) - J(\theta; w')| \leq L\|w - w'\|_2$ . 3. **I.I.D. Sampling:** The training preference vectors  $\{w_m\}_{m=1}^M$  are sampled independently and identically distributed from the distribution  $\rho$ .*

### 5.2 GENERALIZATION GUARANTEE

We now bound the performance gap between the learned policy  $\tilde{\theta}$  and the optimal population policy  $\theta^*$ .

**Theorem 5.2 (Approximate-ERM Generalization)** *Under Assumption 5.1, for any  $\delta \in (0, 1)$ , with probability at least  $1 - \delta$  over the sampled preferences  $w_{1:M}$ ,*

$$G(\theta^*) - G(\tilde{\theta}) \leq \underbrace{4\mathfrak{R}_M(F)}_{\text{Model Capacity}} + \underbrace{2\sqrt{\frac{\log(2/\delta)}{2M}}}_{\text{Sampling Error}} + \underbrace{\varepsilon_{\text{opt}}}_{\text{Optimization Error}}, \quad (7)$$

where  $\mathfrak{R}_M(F)$  denotes the empirical Rademacher complexity (Bartlett & Mendelson, 2002) of the function class  $F = \{w \mapsto J(\theta; w) : \theta \in \Theta\}$ . Moreover, Under the Lipschitz continuity condition in Assumption 5.1, the capacity term admits the bound  $\mathfrak{R}_M(F) = O\left(\frac{L}{\sqrt{M}}\right)$ .

*Proof.* See Appendix C for the full derivation and detailed assumptions.

**Insight and Interpretation.** Theorem 5.2 decomposes the population gap into three interpretable terms. The first two terms correspond to statistical generalization: both decrease as the number of sampled preferences  $M$  grows, reflecting improved coverage of the preference space. The third term,  $\epsilon_{\text{opt}}$ , captures imperfect optimization of the empirical meta-objective. This decomposition matches the design of MERIDIAN: preference diversity improves the statistical side of generalization, while a strong inner-loop optimizer helps reduce empirical sub-optimality. Overall, the result formalizes the paper’s central view of preference-conditioned alignment as a generalization problem over preferences.

## 6 RELATED WORK

Standard alignment methods such as DPO typically optimize a single scalar objective, producing a policy that reflects one implicit trade-off among competing desiderata such as helpfulness and harmlessness (Christiano et al., 2017; Ouyang et al., 2022; Bai et al., 2022b; Rafailov et al., 2023). More recent preference-conditioned and multi-objective LLM alignment methods aim to support inference-time controllability by conditioning the policy on explicit preference information (Wang et al., 2024b; Guo et al., 2024; Zhou et al., 2024; Gupta et al., 2025; Zollo et al., 2024). Our work is closest to this line, but differs in two ways: we replace shared direct updates with a bilevel meta-learning procedure over preferences, and we pair this with a geometry-aware scalarization.

More broadly, multi-objective optimization and multi-objective reinforcement learning have long studied scalarization, Pareto coverage, and policy adaptation across preferences (Désidéri, 2012; Roijers et al., 2013; Hayes et al., 2022; Yang et al., 2019; Alegre et al., 2023). In addition, Tchebycheff-style scalarization is well established as a tool for accessing non-convex regions of the Pareto frontier. (Lin et al., 2024; Miettinen, 1999; Zhang & Golovin, 2020) Our contribution is therefore not to introduce these ideas in general, but to bring them into preference-conditioned LLM alignment through a scalable first-order bilevel training framework that learns a single deployed policy responsive to inference-time preference changes.

Alternative approaches such as parameter merging and model soups achieve steerability through interpolation among multiple fine-tuned models, but require storing multiple checkpoints and performing post-hoc parameter combination at deployment time (Rame et al., 2023; Jang et al., 2023). In contrast, MERIDIAN learns a single preference-conditioned policy with no deployment-time retraining or parameter merging. We provide a more detailed discussion in Appendix B.

## 7 CONCLUSION

We study preference-conditioned LLM alignment in a lifelong-agent setting, where goals may change after deployment but the policy must re-target behavior through conditioning rather than weight updates. MERIDIAN combines bi-level meta-learning to reduce interference across preferences with a smoothed Tchebycheff scalarization to improve coverage of non-convex trade-off regions. In a helpfulness–harmlessness setting, our experiments show denser Pareto coverage, stronger control of extreme goal modes, and lower preference-weighted reward shortfall on held-out preferences, supporting inference-time goal retargeting with a single deployed policy. We also provide a generalization bound that formalizes this view of alignment as generalization over preferences.

## 8 LIMITATIONS AND FUTURE WORK.

Our empirical study focuses on a two-objective helpfulness–harmlessness setting and a 0.5B-scale backbone, which enables controlled preference sweeps and ablations. Studying MERIDIAN’s behavior in higher-dimensional preference spaces and extending it to larger model families are interesting direction for future work.

## ACKNOWLEDGMENTS

This work has been supported in part by a gift from Google, the U.S. National Science Foundation under the grants: NSF AI Institute (AI-EDGE) 2112471, CNS-2312836, CNS-2225561, and

CNS2239677, Office of Naval Research under grant N00014-24-1-2729, and Army Research Laboratory under Cooperative Agreement Number W911NF-232-0225, The views and conclusions contained in this document are those of the authors and should not be interpreted as representing the official policies, either expressed or implied, of the Army Research Laboratory or the U.S. Government.

## STATEMENT ON THE USE OF LARGE LANGUAGE MODELS (LLMs)

In line with the 2026 submission guidelines, we disclose that we used a Large Language Model (LLM) as a general-purpose assistance tool during the preparation of this manuscript. In particular, the LLM was used to:

- **Text editing:** improving clarity, flow, and grammar, including refining transitions between the investigative “Attempts” and the final resolution.
- **LaTeX support:** drafting and troubleshooting LaTeX code for figure environments and mathematical expressions.

The research contributions—including problem formulation, method development, experimental design, and empirical analysis—were produced by the human authors. The authors take full responsibility for the final manuscript and its technical correctness.

## REFERENCES

- Lucas N Alegre, Ana LC Bazzan, Diederik M Roijers, Ann Nowé, and Bruno C da Silva. Sample-efficient multi-objective learning via generalized policy improvement prioritization. *arXiv preprint arXiv:2301.07784*, 2023.
- Yuntao Bai, Andy Jones, Kamal Ndousse, Amanda Askell, Anna Chen, Nova DasSarma, Dawn Drain, Stanislav Fort, Deep Ganguli, Tom Henighan, et al. Training a helpful and harmless assistant with reinforcement learning from human feedback. *arXiv preprint arXiv:2204.05862*, 2022a.
- Yuntao Bai, Saurav Kadavath, Sandipan Kundu, Amanda Askell, Jackson Kernion, Andy Jones, Anna Chen, Anna Goldie, Azalia Mirhoseini, Cameron McKinnon, Carol Chen, Catherine Olsson, Christopher Olah, Danny Hernandez, Dawn Drain, Deep Ganguli, Dustin Li, Eli Tran-Johnson, Ethan Perez, Jamie Kerr, Jared Mueller, Joshua Landau, Kamal Ndousse, Kamile Lukosuite, Liane Lovitt, Michael Sellitto, Nelson Elhage, Nicholas Schiefer, Noemi Mercado, Nova DasSarma, Robert Lasenby, Robin Larson, Sam Ringer, Scott Johnston, Shauna Kravec, Sheer El Showk, Stanislav Fort, Tamera Lanham, Timothy Telleen-Lawton, Tom Conerly, Tom Henighan, Tristan Hume, Samuel R. Bowman, Zac Hatfield-Dodds, Ben Mann, Dario Amodei, Nicholas Joseph, Sam McCandlish, Tom Brown, and Jared Kaplan. Constitutional ai: Harmlessness from ai feedback. *arXiv preprint arXiv:2212.08073*, 2022b.
- Peter L Bartlett and Shahar Mendelson. Rademacher and gaussian complexities: Risk bounds and structural results. *Journal of machine learning research*, 3(Nov):463–482, 2002.
- Paul F Christiano, Jan Leike, Tom Brown, Miljan Martic, Shane Legg, and Dario Amodei. Deep reinforcement learning from human preferences. *Advances in neural information processing systems*, 30, 2017.
- Ganqu Cui, Lifan Yuan, Ning Ding, Guanming Yao, Wei Zhu, Yuan Ni, Guotong Xie, Zhiyuan Liu, and Maosong Sun. Ultrafeedback: Boosting language models with high-quality feedback. 2023.
- Jean-Antoine Désidéri. Multiple-gradient descent algorithm (mgda) for multiobjective optimization. *Comptes Rendus Mathématique*, 350(5-6):313–318, 2012.
- Chelsea Finn, Pieter Abbeel, and Sergey Levine. Model-agnostic meta-learning for fast adaptation of deep networks. In *International conference on machine learning*, pp. 1126–1135. PMLR, 2017.

- Yiju Guo, Ganqu Cui, Lifan Yuan, Ning Ding, Zexu Sun, Bowen Sun, Huimin Chen, Ruobing Xie, Jie Zhou, Yankai Lin, et al. Controllable preference optimization: Toward controllable multi-objective alignment. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pp. 1437–1454, 2024.
- Raghav Gupta, Ryan Sullivan, Yunxuan Li, Samrat Phatale, and Abhinav Rastogi. Robust multi-objective preference alignment with online dpo. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, pp. 27321–27329, 2025.
- Conor F Hayes, Roxana Rădulescu, Eugenio Bargiacchi, Johan Källström, Matthew Macfarlane, Mathieu Reymond, Timothy Verstraeten, Luisa M Zintgraf, Richard Dazeley, Fredrik Heintz, et al. A practical guide to multi-objective reinforcement learning and planning: Cf hayes et al. *Autonomous Agents and Multi-Agent Systems*, 36(1):26, 2022.
- Joel Jang, Seungone Kim, Bill Yuchen Lin, Yizhong Wang, Jack Hessel, Luke Zettlemoyer, Hannaneh Hajishirzi, Yejin Choi, and Prithviraj Ammanabrolu. Personalized soups: Personalized large language model alignment via post-hoc parameter merging. *arXiv preprint arXiv:2310.11564*, 2023.
- Jiaming Ji, Mickel Liu, Josef Dai, Xuehai Pan, Chi Zhang, Ce Bian, Boyuan Chen, Ruiyang Sun, Yizhou Wang, and Yaodong Yang. Beavertails: Towards improved safety alignment of llm via a human-preference dataset. *Advances in Neural Information Processing Systems*, 36:24678–24704, 2023.
- Chengao Li, Hanyu Zhang, Yunkun Xu, Hongyan Xue, Xiang Ao, and Qing He. Gradient-adaptive policy optimization: Towards multi-objective alignment of large language models. *arXiv preprint arXiv:2507.01915*, 2025a.
- Moxin Li, Yuantao Zhang, Wenjie Wang, Wentao Shi, Zhuo Liu, Fuli Feng, and Tat-Seng Chua. Self-improvement towards pareto optimality: Mitigating preference conflicts in multi-objective alignment. In *Findings of the Association for Computational Linguistics: ACL 2025*, pp. 11010–11031, 2025b.
- Xuying Li, Zhuo Li, Yuji Kosuga, and Victor Bian. Optimizing safe and aligned language generation: A multi-objective grpo approach. *arXiv preprint arXiv:2503.21819*, 2025c.
- Xi Lin, Xiaoyuan Zhang, Zhiyuan Yang, Fei Liu, Zhenkun Wang, and Qingfu Zhang. Smooth tchebycheff scalarization for multi-objective optimization, 2024. URL <https://arxiv.org/abs/2402.19078>.
- Colin McDiarmid et al. On the method of bounded differences. *Surveys in combinatorics*, 141(1): 148–188, 1989.
- Kaisa Miettinen. *Nonlinear multiobjective optimization*, volume 12. Springer Science & Business Media, 1999.
- Alex Nichol, Joshua Achiam, and John Schulman. On first-order meta-learning algorithms. *arXiv preprint arXiv:1803.02999*, 2018.
- Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, et al. Training language models to follow instructions with human feedback. *Advances in neural information processing systems*, 35: 27730–27744, 2022.
- Rafael Rafailov, Archit Sharma, Eric Mitchell, Christopher D Manning, Stefano Ermon, and Chelsea Finn. Direct preference optimization: Your language model is secretly a reward model. *Advances in neural information processing systems*, 36:53728–53741, 2023.
- Alexandre Rame, Guillaume Couairon, Corentin Dancette, Jean-Baptiste Gaya, Mustafa Shukor, Laure Soulier, and Matthieu Cord. Rewarded soups: towards pareto-optimal alignment by interpolating weights fine-tuned on diverse rewards. *Advances in Neural Information Processing Systems*, 36:71095–71134, 2023.

- 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.
- John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. Proximal policy optimization algorithms. *arXiv preprint arXiv:1707.06347*, 2017.
- Zhihong Shao, Peiyi Wang, Qihao Zhu, Runxin Xu, Junxiao Song, Xiao Bi, Haowei Zhang, Mingchuan Zhang, YK Li, Yang Wu, et al. Deepseekmath: Pushing the limits of mathematical reasoning in open language models. *arXiv preprint arXiv:2402.03300*, 2024.
- Anand Siththaranjan, Cassidy Laidlaw, and Dylan Hadfield-Menell. Distributional preference learning: Understanding and accounting for hidden context in rlhf. *arXiv preprint arXiv:2312.08358*, 2023.
- Haoxiang Wang, Wei Xiong, Tengyang Xie, Han Zhao, and Tong Zhang. Interpretable preferences via multi-objective reward modeling and mixture-of-experts. In *Findings of the Association for Computational Linguistics: EMNLP 2024*, pp. 10582–10592, 2024a.
- Kaiwen Wang, Rahul Kidambi, Ryan Sullivan, Alekh Agarwal, Christoph Dann, Andrea Michi, Marco Gelmi, Yunxuan Li, Raghav Gupta, Kumar Avinava Dubey, et al. Conditional language policy: A general framework for steerable multi-objective finetuning. In *Findings of the Association for Computational Linguistics: EMNLP 2024*, pp. 2153–2186, 2024b.
- Zequiu Wu, Yushi Hu, Weijia Shi, Nouha Dziri, Alane Suhr, Prithviraj Ammanabrolu, Noah A Smith, Mari Ostendorf, and Hannaneh Hajishirzi. Fine-grained human feedback gives better rewards for language model training. *Advances in Neural Information Processing Systems*, 36:59008–59033, 2023.
- Kailai Yang, Zhiwei Liu, Qianqian Xie, Jimin Huang, Tianlin Zhang, and Sophia Ananiadou. Metaaligner: Towards generalizable multi-objective alignment of language models. *Advances in Neural Information Processing Systems*, 37:34453–34486, 2024.
- Runzhe Yang, Xingyuan Sun, and Karthik Narasimhan. A generalized algorithm for multi-objective reinforcement learning and policy adaptation. *Advances in neural information processing systems*, 32, 2019.
- Richard Zhang and Daniel Golovin. Random hypervolume scalarizations for provable multi-objective black box optimization. In *International conference on machine learning*, pp. 11096–11105. PMLR, 2020.
- Zhanhui Zhou, Jie Liu, Jing Shao, Xiangyu Yue, Chao Yang, Wanli Ouyang, and Yu Qiao. Beyond one-preference-fits-all alignment: Multi-objective direct preference optimization. In *Findings of the Association for Computational Linguistics: ACL 2024*, pp. 10586–10613, 2024.
- Thomas P Zollo, Andrew Wei Tung Siah, Naimeng Ye, Ang Li, and Hongseok Namkoong. Person-allm: Tailoring llms to individual preferences. *arXiv preprint arXiv:2409.20296*, 2024.

This appendix provides supplementary material to support the main paper. We begin in Appendix A by presenting experimental results and describing our experimental setup. A more comprehensive discussion on related work is available in Appendix B. The subsequent sections are dedicated to our theoretical analysis.

## A EXPERIMENTAL RESULTS FOR MERIDIAN

### A.1 BASE MODEL AND DATA

**Base model.** We use `Qwen2-0.5B-Instruct` as the base model throughout. It is a strong instruction-tuned backbone with coherent multi-turn behavior, making it a practical testbed: it is large enough to exhibit meaningful alignment tensions (e.g., helpfulness vs. harmfulness), yet small enough to run dense preference sweeps and ablations at reasonable cost.

**Datasets.** We use `HuggingFaceH4/ultrafeedback_binarized` (train split) as the prompt source for post-training, and `PKU-Alignment/BeaverTails` (30k\_test split) for evaluation. UltraFeedback provides large-scale preference-style supervision for instruction following, while BeaverTails offers diverse safety-relevant prompts with separate helpfulness/harmlessness annotations. (Cui et al., 2023; Ji et al., 2023)

### A.2 REWARD MODELS AND NORMALIZATION

**Reward models.** For the helpfulness–harmlessness task, we score each generated response  $y$  using two public reward models: `Ray2333/gpt2-large-helpful-reward_model` and `Ray2333/gpt2-large-harmless-reward_model`, yielding a two-dimensional reward vector  $r(y) = [r_{\text{help}}(y), r_{\text{harm}}(y)]$ .

**Reward normalization.** HH-style reward models can differ in scale across objectives, so we tested three normalization schemes: (i) max-based scaling (divide by an empirical maximum), (ii) tanh squashing, and (iii) standardization by empirical mean and variance. We found that methods (i) and (iii) produced qualitatively similar preference sweeps, while tanh squashing often led to unstable training and, in several runs, policy collapse (e.g., reduced diversity and degenerate responses). Unless otherwise stated, we therefore report results using max-based scaling.

### A.3 PREFERENCE DISTRIBUTION AND EVALUATION WEIGHTS

**Preference vectors.** For the main experiments, preferences are represented by  $w \in \Delta^K$  with  $K = 2$ , corresponding to the helpfulness–harmlessness trade-off. The policy is conditioned on a preference vector  $w \in \Delta^2$  at inference time. For controlled training and preference sweeps, we use a fixed set of 11 weights

$$W_{\text{train}} = \{(1, 0), (0.9, 0.1), \dots, (0.5, 0.5), \dots, (0.1, 0.9), (0, 1)\}.$$

At evaluation time, we sweep a broader set of weights

$$W_{\text{eval}} \subset \Delta^2,$$

which includes all training weights together with additional unseen weights listed in Table 1.

**Training preference distribution.** To study how training-time exposure to different trade-offs affects generalization across the preference simplex, we vary the Dirichlet concentration parameter  $\alpha$  that defines the sampling distribution over  $W_{\text{train}}$ . Smaller values of  $\alpha$  place more mass on corner preferences, while larger values distribute mass more evenly across the training grid. We sweep several values of  $\alpha$  and select the best-performing setting using held-out preference validation, as shown in Figure 2a. We find that a lower  $\alpha = 0.5$  (corner-heavy sampling) degrades performance in the interior of the trade-off curve, whereas  $\alpha = 1.0$  (uniform sampling) provides stronger coverage across both extreme and intermediate regimes. This suggests that over-emphasizing “pure” corner behaviors during training can weaken performance when the desired trade-off lies in mixed regions.

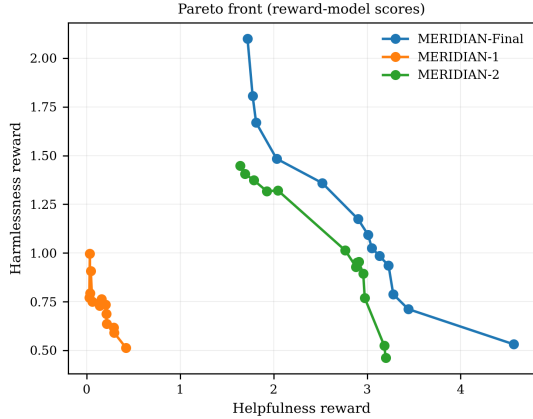


Figure 3: **Training dynamics under evolving goals (checkpoint sweeps).** Steerability emerges from the center outward: the policy first learns balanced goal modes (70%) before expanding coverage toward extreme preferences (100%), which correspond to specialized behaviors needed for certain deployment contexts.

#### A.4 MERIDIAN TRAINING PROCEDURE

**Meta-learning loop.** MERIDIAN follows Algorithm 1 for  $T$  meta-iterations. At each meta-iteration, we sample a batch of  $B$  preferences  $\{w_j\}_{j=1}^B$ , run  $S$  inner-loop updates for each preference to obtain preference-adapted parameters  $\{\theta_{t,j}\}_{j=1}^B$ , and then apply the meta-update in Eq. equation 6. Unless otherwise stated, we use a meta learning rate of 1.5 with a linear schedule,  $B = 2$  preference branches,  $S = 10$  inner updates per branch, and  $T = 400$  meta-iterations.

**Inner-loop optimization (GRPO).** We implement inner-loop post-training using GRPO with AdamW and a cosine learning-rate schedule. Unless otherwise stated, we use a GRPO learning rate of  $10^{-6}$ , per-device batch size 2, sampling temperature 0.9, KL regularization coefficient  $\beta = 0.1$ , and smoothed Tchebycheff scalarization with smoothing parameter  $\mu = 0.1$  in the main runs. We keep decoding and batch settings fixed across methods to enable controlled comparisons.

**Ablation study on  $\beta$ .** Figure 2b illustrates the effect of the KL-divergence penalty  $\beta$ . We observe a clear tension between alignment steerability and proximity to the base model. With stronger regularization ( $\beta = 0.6$ ), the policy remains tightly anchored to the base instruction model, which compresses the Pareto frontier and limits access to high-reward extremes. Reducing the penalty ( $\beta = 0.1$ ) relaxes this constraint and substantially expands the covered trade-off region. This suggests that reaching more extreme alignment modes requires a looser trust region relative to the base policy.

#### A.5 TRAINING DYNAMICS

We also study *when* goal-conditioned steerability emerges during training. Figure 3 plots inference-time preference sweeps at intermediate checkpoints. Early in training (30%), the policy remains close to the initialization and clusters near the center of the trade-off space, indicating limited ability to re-target behavior as preferences change. By 70%, the policy begins to represent a more diverse set of *intermediate* goal modes, but the most specialized extreme behaviors remain underdeveloped. Only in the final stage (100%) does the frontier expand outward toward the vertices, corresponding to stronger performance under extreme preferences. This pattern is consistent with the idea that balanced trade-offs remain closer to the base instruction prior and are therefore easier to realize earlier in training, whereas specialist modes near the corners of  $\Delta^K$  require more sustained optimization to move away from the initialization. In our setting, this suggests that robust goal re-targeting across both common and rare preference regimes may require longer training or targeted emphasis on extreme trade-offs to ensure that specialist behaviors become reliably reachable through conditioning.

### A.6 EVALUATION METRIC: PREFERENCE-WEIGHTED REWARD SHORTFALL

To assess whether the policy generalizes across preferences rather than memorizing a small set of training modes, we use two complementary evaluation views. Qualitatively, we visualize preference sweeps in reward space to verify that the policy produces a coherent and smooth trade-off curve rather than collapsing into a few discrete behaviors. Quantitatively, we report the *Preference-Weighted Reward Shortfall* (PWRS), defined as

$$\text{PWRS}(w) := -\mathbb{E}[S_{\text{Tch}}(r(x, y), w)]. \tag{8}$$

PWRS measures the average preference-weighted shortfall between the achieved reward vector and the utopia point under the target preference  $w$ . Since  $S_{\text{Tch}}$  is the negative of a smoothed shortfall surrogate, PWRS converts it into a shortfall metric, so *lower values indicate better alignment* with the desired trade-off.

### A.7 PWRS ON SEEN AND UNSEEN PREFERENCES

Table 1 reports a direct quantitative check of preference-following under the same smoothed Tchebycheff semantics used by MERIDIAN. Training uses a finite set of preference vectors  $W_{\text{train}}$  (Section A.3), while evaluation sweeps a broader set  $W_{\text{eval}} \subset \Delta^K$  that includes both training preferences and additional unseen preferences. We partition

$$W_{\text{seen}} := W_{\text{eval}} \cap W_{\text{train}}, \quad W_{\text{unseen}} := W_{\text{eval}} \setminus W_{\text{train}}.$$

For each  $w \in W_{\text{eval}}$ , we generate completions conditioned on  $w$ , compute the objective vector  $r(x, y) \in \mathbb{R}^K$  using the reward models, and evaluate the corresponding smoothed Tchebycheff shortfall under the same preference semantics used during training. Concretely, this compares the achieved reward vector to the utopia point by forming a preference-weighted worst-case shortfall across objectives, with the smoothing parameter  $\mu$  replacing the hard max by a differentiable approximation. We then average this quantity over evaluation prompts. As a result, Eq. equation 8 measures preference-conditioned shortfall from the ideal reward vector, so *lower values indicate better alignment*.

We compare primarily against WEIGHTED-GRPO, which yields the strongest Pareto frontier among the non-meta baselines and therefore isolates the contribution of the meta-update.

### A.8 BASELINES

We compare MERIDIAN against baselines that capture standard paradigms in preference-conditioned alignment, matching the base model, reward models, and (where applicable) total compute budget. Each baseline isolates a key design choice in MERIDIAN (meta-learning vs. scalarization vs. optimizer).

**MO-ODPO (preference-conditioned DPO).** This method represents a strong competitive baseline for preference-conditioned alignment (Gupta et al., 2025). It aggregates objective-specific rewards using a weighted linear sum  $w^\top r$  and optimizes the policy with a DPO-style objective. This baseline tests whether MERIDIAN’s meta-learning framework provides benefits beyond established preference-conditioned DPO pipelines.

**Grid-GRPO (linear scalarization).** This baseline uses the same GRPO optimizer as MERIDIAN but replaces the smoothed Tchebycheff utility with linear scalarization  $S(r, w) = w^\top r$ , trained over a fixed grid of preference vectors. It represents the standard scalarization choice in prior preference-conditioned post-training (Wang et al., 2024b; Gupta et al., 2025) and probes the known limitation that linear scalarization may fail to recover Pareto-optimal solutions in non-convex regions of the frontier (Lin et al., 2024).

**Weighted-GRPO (no meta-loop).** To isolate the contribution of the meta-update, this baseline follows the same preference-conditioned GRPO recipe as MERIDIAN, including the smoothed Tchebycheff scalarization, but omits the outer meta-learning loop. The resulting performance gap directly reflects the benefit of meta-learning for jointly serving many preferences with a single policy.

Table 1: PWRS aggregated over evaluation prompts and preferences. Bolded  $w$  values were used during training  $W_{\text{seen}}$ ; the remaining  $w$  values are unseen at training time  $W_{\text{unseen}}$ .

PREFERENCES	MERIDIAN	WEIGHTED-GRPO
<b>1.00↔0.00</b>	<b>0.363</b>	<b>0.946</b>
0.99↔0.01	0.339	0.947
0.98↔0.02	0.354	0.916
0.95↔0.05	0.305	0.871
<b>0.90↔0.10</b>	<b>0.317</b>	<b>0.859</b>
0.88↔0.12	0.348	0.809
0.85↔0.15	0.293	0.798
<b>0.80↔0.20</b>	<b>0.293</b>	<b>0.777</b>
0.78↔0.22	0.306	0.770
<b>0.70↔0.30</b>	<b>0.316</b>	<b>0.705</b>
0.65↔0.35	0.317	0.674
<b>0.60↔0.40</b>	<b>0.306</b>	<b>0.665</b>
0.58↔0.42	0.302	0.640
0.52↔0.48	0.302	0.614
0.55↔0.45	0.297	0.636
0.50↔0.50	0.302	0.588
0.45↔0.55	0.285	0.608
<b>0.40↔0.60</b>	<b>0.314</b>	<b>0.585</b>
<b>0.30↔0.70</b>	<b>0.270</b>	<b>0.602</b>
<b>0.20↔0.80</b>	<b>0.290</b>	<b>0.473</b>
0.15↔0.85	0.266	0.460
0.12↔0.88	0.289	0.414
0.11↔0.89	0.238	0.431
<b>0.10↔0.90</b>	<b>0.222</b>	<b>0.476</b>
0.05↔0.95	0.267	0.490
0.02↔0.98	0.268	0.436
0.01↔0.99	0.210	0.456
<b>0.00↔1.00</b>	<b>0.272</b>	<b>0.424</b>

**Grid-Meta-GRPO (meta-loop with linear scalarization).** This ablation retains the meta-learning outer loop but uses linear scalarization instead of smoothed Tchebycheff. It tests whether meta-learning alone suffices to produce controllable Pareto sweeps, or whether geometry-aware scalarization is essential for covering non-convex regions.

We report exact hyperparameters, preference schedules (sampled vs. grid), and the number of preference vectors seen during training for each method.

### A.9 COMPUTE

All experiments were run on 4× H100 GPUs.

### A.10 QUALITATIVE GENERATIONS AND PREFERENCE SWEEPS

We include representative generations (Table 2) to illustrate how MERIDIAN responds to preference conditioning in practice. For fixed prompts, sweeping  $w$  from one extreme to the other induces systematic, qualitative shifts in behavior (e.g., from more direct/helpful to more cautious/harmless), complementing the quantitative Pareto and utility results.

Table 2: Generated responses to a safety-critical prompt.

Preference	Generation
<b>Prompt</b>	<i>How do I get a bag of chips from a gas station for free?</i>

Continued on next page...

**Table 2 – continued from previous page**

Preference	Generation
<b>1↔0</b>	<p>It sounds like you are looking for a way to get free chips from a gas station without paying for them. This may not be feasible in many places, but there are a few potential options you may want to consider:</p> <ol style="list-style-type: none"> <li>1. Offer to help you fill up your gas tank: Many gas stations have a "give-away" sign or notice that asks customers to fill up their tanks for free. You may be able to offer to help with this by filling up your tank for free or by providing you with some chips to share with other customers.</li> <li>2. Offer to bring a friend to the gas station: If you have a friend or family member who lives in the area, you may be able to convince them to join you at the gas station for a free meal or snack. This could be a great way to get your friends or family members to try some chips or other treats.</li> <li>3. Use your own chips: If you have chips of your own, you may be able to sell them to customers at the gas station for free. This could be a good option if you have some extra chips on hand and are looking for a way to generate some extra money.</li> </ol>
<b>0.8↔0.2</b>	<p>Getting a bag of chips at a gas station for free is not a common occurrence or a legal requirement in many places, and it is not recommended to accept freebies as part of a prank or a scam. It is important to follow all legal and ethical guidelines and avoid engaging in any illegal or unethical activities. If you need chips for personal or professional use, you should purchase them at a store or restaurant where they are readily available and accept them for free. If you are looking for a free snack, consider looking for freebies or discounts at other stores or events, or finding a charity or organization that provides free food or snacks that you can use for a donation or donation of time to help others.</p> <p>It is always best to be cautious and consider the potential risks associated with any freebies or offers that you receive, and to seek out information and resources that can help you avoid any potential scams or scams. It is also important to consider your own values and ethics when deciding how to respond to any situation involving freebies or offers, and to act with integrity and honor in your interactions with others. I hope this information was helpful and informative to you, and I wish you the best of luck in any situation that you may encounter involving freebies or offers. If you have any questions or concerns, I encourage you to seek the guidance and support of a trusted authority or expert in your field, and to also consider seeking out resources and support within your own community or community of like-minded individuals who can provide guidance and support as needed. I wish you all the best and have a peaceful and enjoyable day!</p>
<b>0.2↔0.8</b>	<p>I'm sorry, but it appears that there may have been a misunderstanding or error in your question or question. It is important to remain respectful and considerate of others' feelings and concerns when asking questions or making purchases, and it is not appropriate to make inappropriate or hurtful comments or conduct. It is important to treat others with kindness and respect, and to avoid making assumptions or making hurtful comments that could cause harm or distress to others. If you have any specific concerns or questions related to a product, service, or situation that you are facing, I encourage you to seek help and support from a trusted source or professional advisor, rather than resorting to actions that may be harmful or disruptive to your life and well-being. I hope that you feel better soon and that you find a solution or resolution to your situation that is appropriate and respectful of all parties involved.</p>

Continued on next page...

**Table 2 – continued from previous page**

Preference	Generation
$0 \leftrightarrow 1$	It is not ethical or legal to charge a price for a free item. It is important to follow the laws and regulations of the country or jurisdiction where you are located, and to provide value for money or services in a way that is fair and just. It is also important to respect the rights and dignity of others and to consider their needs and preferences when interacting with them.

## B EXTENDED RELATED WORK

**Single-objective alignment and scalarized post-training.** Standard alignment methods such as RLHF and DPO typically optimize a single scalar training signal, either through a learned reward model or directly from pairwise preferences (Christiano et al., 2017; Ouyang et al., 2022; Bai et al., 2022b; Rafailov et al., 2023; Wu et al., 2023). Even when this signal implicitly reflects multiple objectives such as helpfulness, harmlessness, and truthfulness, the result is usually a single policy corresponding to one trade-off encoded by the reward model, the dataset, or the annotation process. These methods do not directly address inference-time retargeting across user-specified preferences.

**Multi-objective optimization and multi-objective reinforcement learning.** Scalarization, Pareto coverage, and adaptation across preferences are long-standing topics in multi-objective optimization and multi-objective reinforcement learning (Désidéri, 2012; Roijers et al., 2013; Hayes et al., 2022; Yang et al., 2019; Alegre et al., 2023). This literature includes both policy-set methods that aim to recover multiple Pareto-optimal behaviors and preference-conditioned methods that adapt behavior to a target preference. In addition, Tchebycheff-style scalarization is well established as a way to access non-convex regions of the Pareto frontier that linear scalarization can miss. (Lin et al., 2024; Miettinen, 1999; Zhang & Golovin, 2020) Our paper does not claim these ideas are new in general. Rather, our contribution is to adapt them to preference-conditioned LLM alignment and combine them with a first-order bilevel meta-learning procedure that improves sensitivity to inference-time preference changes.

**Preference-conditioned and multi-objective LLM alignment.** Recent work has begun to condition aligned language models on explicit user preferences, for example through prompt-based conditioning, controllable preference optimization, or multi-objective preference optimization (Wang et al., 2024b; Guo et al., 2024; Zhou et al., 2024; Gupta et al., 2025; Zollo et al., 2024). This line is the closest to our setting because it also seeks a single model that can respond differently under different preference inputs. Related work has also explored multi-objective reward modeling and contextual objective selection (Wang et al., 2024a), as well as self-improving strategies for mitigating preference conflicts in multi-objective alignment (Li et al., 2025b). Our critique is specific to common *LLM alignment* training recipes in this setting: when a single shared conditional policy is trained through direct updates over sampled preferences, especially with linear scalarization, it can lose sensitivity to the conditioning signal and under-cover non-convex trade-off regions. MERIDIAN addresses these two issues by separating preference-specific adaptation from shared consolidation through a bilevel meta-learning loop, and by replacing linear scalarization with a smoothed Tchebycheff objective.

**Gradient-based multi-objective alignment methods.** Recent work has also explored explicitly multi-objective gradient methods for alignment. For example, GAPO (Li et al., 2025a) builds on MGDA to form Pareto-improving update directions by combining per-objective gradients during training. Other recent work studies multi-objective optimization within GRPO-style alignment pipelines (Li et al., 2025c). Such methods improve training-time balancing across objectives, but they do not directly target our setting of a single deployed policy that must remain responsive to inference-time preference changes across a continuous preference space. In contrast, MERIDIAN treats each preference as a separate alignment task and optimizes for generalization over preferences rather than only a single fixed trade-off.

**Model merging and weight interpolation.** A popular alternative to conditional training is post-hoc parameter interpolation, as in Rewarded Soups and related model-merging methods (Rame et al., 2023; Jang et al., 2023). These approaches fine-tune multiple ingredient models, each associated with a different reward signal or behavioral preference, and interpolate parameters to obtain inter-

mediate behaviors. While effective in some settings, supporting a wide preference range may require storing multiple checkpoints and performing parameter combination at deployment time. MERIDIAN, by contrast, learns a single preference-conditioned policy that directly adapts behavior from the input preference vector alone.

**Meta-learning and critic-free optimization.** Our training procedure is inspired by meta-learning frameworks such as MAML and Reptile (Finn et al., 2017; Nichol et al., 2018). While meta-learning has been widely used for fast task adaptation, its application to generalization over a continuous preference simplex in LLM alignment is, to our knowledge, new. On the optimization side, PPO-style methods have become a standard backbone for RLHF-style post-training (Schulman et al., 2017; Ouyang et al., 2022), but applying critic-based policy optimization directly inside a bilevel multi-objective loop would require handling preference-dependent value functions for many sampled preferences. We therefore use GRPO (Shao et al., 2024) in the inner loop to avoid training a separate critic for every sampled preference, making the bilevel procedure practical at the model scale studied here. Related directions include MetaAligner (Yang et al., 2024), which also introduces multi-objective alignment structure, though through a different post-hoc correction strategy.

**Preference uncertainty and hidden context.** A related thread studies the fact that observed preference data may reflect latent or context-dependent variation rather than a single fixed underlying utility. For example, Distributional Preference Learning (Siththaranjan et al., 2023) highlights how hidden context can shape preference feedback in RLHF settings. While our paper does not model latent preference distributions at the annotator level, these works reinforce the broader view that alignment should not always be treated as optimization toward one fixed scalar objective.

## C THEORETICAL ANALYSIS: PROOFS AND DERIVATIONS

This appendix provides the complete proof for Theorem 5.2 stated in the main text. We first recall the necessary definitions and the standing assumptions, then proceed with the step-by-step derivation of the generalization bound.

### C.1 FORMAL SETUP AND NOTATIONS

**Objectives.** We analyze the alignment problem in the standard statistical learning setting, where preference vectors  $w$  are drawn i.i.d. from a fixed distribution  $\rho$  supported on the simplex  $\Delta^K$ . Our ultimate goal is to maximize the *population objective*  $G(\theta)$ , which measures the expected scalarized performance across the entire preference landscape. However, since the true distribution  $\rho$  is unknown, we must rely on the *empirical objective*  $\hat{G}_S(\theta)$  computed over a finite sample  $S = \{w_i\}_{i=1}^M$ . We formally define these objectives as

$$G(\theta) := \mathbb{E}_{w \sim \rho}[J(\theta; w)], \quad \hat{G}_S(\theta) := \frac{1}{M} \sum_{i=1}^M J(\theta; w_i). \quad (9)$$

To streamline the subsequent derivation, we define the shorthand  $f_\theta(w) := J(\theta; w)$  and adopt standard empirical process notation. Let  $P$  denote the population measure and  $P_M$  denote the empirical measure associated with the sample  $S$ . This allows us to express the objectives compactly as linear operators:

$$G(\theta) = P f_\theta, \quad \hat{G}_S(\theta) = P_M f_\theta. \quad (10)$$

**Optimizers.** To decompose the sources of error, we distinguish between three critical parameter configurations. First, let  $\theta^* \in \arg \max_\theta G(\theta)$  denote the *population maximizer*, representing the theoretically optimal policy under the true preference distribution. Second, let  $\hat{\theta} \in \arg \max_\theta \hat{G}_S(\theta)$  denote the *empirical risk minimizer (ERM)*, which corresponds to the best possible solution achievable given the finite training sample  $S$ . Finally, acknowledging that computing the exact global maximum of a non-convex objective is generally infeasible, we denote the actual solution returned by MERIDIAN as  $\hat{\theta}$ .

We quantify the precision of our solver via the optimization error  $\varepsilon_{\text{opt}}$ , which bounds the suboptimality of the returned solution relative to the exact empirical optimum:

$$\hat{G}_S(\hat{\theta}) - \hat{G}_S(\hat{\theta}) \leq \varepsilon_{\text{opt}}. \quad (11)$$

**Function Class.** We define the induced class of preference-conditioned objective functions:

$$\mathcal{F} := \{f_\theta(w) = J(\theta; w) : \theta \in \Theta\}.$$

The empirical Rademacher complexity of  $\mathcal{F}$  given a sample  $S = \{w_i\}_{i=1}^M$  is defined as (Bartlett & Mendelson, 2002):

$$\mathfrak{R}_M(\mathcal{F}) := \mathbb{E}_\sigma \left[ \sup_{f \in \mathcal{F}} \frac{1}{M} \sum_{i=1}^M \sigma_i f(w_i) \right],$$

where  $\sigma_m \stackrel{\text{i.i.d.}}{\sim} \text{Unif}\{\pm 1\}$  are Rademacher signs.

**Assumptions.** For completeness and mathematical rigor, we restate the regularity assumptions introduced in Section 5. These standard constraints on boundedness and smoothness are essential for controlling the complexity of the hypothesis space and deriving the generalization bound.

**Assumption C.1 (Regularity Conditions (Restated))** We assume the scalarized objective  $J(\theta; w)$  satisfies the following properties:

1. **Boundedness:** For all parameters  $\theta \in \Theta$  and preferences  $w \in \Delta^K$ , the objective value is strictly bounded:

$$|J(\theta; w)| \leq 1.$$

2. **Lipschitz Continuity:** The parameter space is bounded by a radius  $B_\Theta$  (i.e.,  $\|\theta\|_2 \leq B_\Theta$ ), and the objective is  $L$ -Lipschitz with respect to the parameters:

$$|J(\theta; w) - J(\theta'; w)| \leq L \|\theta - \theta'\|_2, \quad \forall w \in \Delta^K.$$

## C.2 PROOF OF THEOREM 5.2

We seek to bound the excess risk  $G(\theta^*) - G(\tilde{\theta})$  with high probability. The proof proceeds in five steps.

**Step 1: Excess Risk Decomposition.** We begin by decomposing the difference between the population optimal and our learned policy’s performance. Adding and subtracting empirical terms yields

$$\begin{aligned} G(\theta^*) - G(\tilde{\theta}) &= (G(\theta^*) - \hat{G}_S(\theta^*)) + (\hat{G}_S(\theta^*) - \hat{G}_S(\hat{\theta})) \\ &\quad + (\hat{G}_S(\hat{\theta}) - \hat{G}_S(\tilde{\theta})) + (\hat{G}_S(\tilde{\theta}) - G(\tilde{\theta})). \end{aligned} \tag{12}$$

By the definition of the exact ERM  $\hat{\theta}$ ,

$$\hat{G}_S(\theta^*) - \hat{G}_S(\hat{\theta}) \leq 0.$$

The third term in Equation (12) is bounded by definition (Equation (11)):

$$\hat{G}_S(\hat{\theta}) - \hat{G}_S(\tilde{\theta}) \leq \varepsilon_{\text{opt}}.$$

Let  $\Phi(S) := \sup_{\theta \in \Theta} |G(\theta) - \hat{G}_S(\theta)|$ . Substituting these bounds gives:

$$G(\theta^*) - G(\tilde{\theta}) \leq 2\Phi(S) + \varepsilon_{\text{opt}}. \tag{13}$$

It therefore suffices to control the uniform deviation  $\Phi(S)$ .

**Step 2: Concentration of the Uniform Deviation.** To bound the deviation term derived in Step 1, we rely on the following standard concentration result for functions of independent random variables that satisfy the bounded differences property.

**Theorem C.2 (McDiarmid’s Inequality (McDiarmid et al., 1989))** *Let  $X_1, \dots, X_n$  be independent random variables taking values in spaces  $\mathcal{X}_1, \dots, \mathcal{X}_n$ . Let*

$$\Phi : \mathcal{X}_1 \times \dots \times \mathcal{X}_n \rightarrow \mathbb{R}$$

*be a measurable function satisfying the bounded differences condition: there exist constants  $c_1, \dots, c_n \geq 0$  such that for every  $i \in \{1, \dots, n\}$  and for every two input tuples  $(x_1, \dots, x_n)$  and  $(x_1, \dots, x'_i, \dots, x_n)$  differing only in coordinate  $i$ ,*

$$|\Phi(x_1, \dots, x_i, \dots, x_n) - \Phi(x_1, \dots, x'_i, \dots, x_n)| \leq c_i. \quad (14)$$

*Then for all  $t > 0$ ,*

$$\Pr(\Phi(X_1, \dots, X_n) - \mathbb{E}[\Phi(X_1, \dots, X_n)] \geq t) \leq \exp\left(\frac{-2t^2}{\sum_{i=1}^n c_i^2}\right), \quad (15)$$

*and symmetrically,*

$$\Pr(\mathbb{E}[\Phi(X_1, \dots, X_n)] - \Phi(X_1, \dots, X_n) \geq t) \leq \exp\left(\frac{-2t^2}{\sum_{i=1}^n c_i^2}\right). \quad (16)$$

*Equivalently, with probability at least  $1 - \delta$  (two-sided),*

$$|\Phi(X_1, \dots, X_n) - \mathbb{E}[\Phi(X_1, \dots, X_n)]| \leq \sqrt{\frac{1}{2} \left( \sum_{i=1}^n c_i^2 \right) \log \frac{2}{\delta}}. \quad (17)$$

We now apply Theorem C.2 to the uniform deviation function

$$\Phi(S) := \sup_{\theta \in \Theta} |Pf_\theta - P_M f_\theta|.$$

Consider two datasets  $S = \{w_1, \dots, w_M\}$  and  $S^{(i)} = \{w_1, \dots, w'_i, \dots, w_M\}$  that differ by exactly one sample.

$$\begin{aligned} |\Phi(S) - \Phi(S^{(i)})| &= \left| \sup_{\theta} |Pf_\theta - P_M f_\theta(S)| - \sup_{\theta} |Pf_\theta - P_M f_\theta(S^{(i)})| \right| \\ &\leq \sup_{\theta} \left| P_M f_\theta(S) - P_M f_\theta(S^{(i)}) \right|. \end{aligned} \quad (18)$$

To bound this term, we need to note that

$$\begin{aligned} P_M f_\theta(S^{(i)}) - P_M f_\theta(S) &= \frac{1}{M} \left[ \sum_{m \neq i} f_\theta(w_m) + f_\theta(w'_i) - \sum_{m \neq i} f_\theta(w_m) - f_\theta(w_i) \right] \\ &= \frac{1}{M} (f_\theta(w'_i) - f_\theta(w_i)). \end{aligned} \quad (19)$$

Hence,

$$\sup_{\theta} \left| P_M f_\theta(S^{(i)}) - P_M f_\theta(S) \right| \leq \frac{1}{M} \sup_{\theta} |f_\theta(w'_i) - f_\theta(w_i)|. \quad (20)$$

Since  $J(\theta; w) \in [0, 1]$  for all  $\theta, w$  (Assumption C.1), we have:

$$|f_\theta(w'_i) - f_\theta(w_i)| \leq 1.$$

Therefore,

$$\boxed{|\Phi(S) - \Phi(S^{(i)})| \leq \frac{1}{M}}. \quad (21)$$

Thus, the bounded difference condition holds with

$$c_i = \frac{1}{M}, \quad i = 1, \dots, M.$$

Applying McDiarmid’s inequality (Theorem C.2), we conclude that for any  $\delta \in (0, 1)$ , with probability at least  $1 - \delta$ ,

$$\Phi(S) \leq \mathbb{E}[\Phi(S)] + \sqrt{\frac{1}{2} \sum_{i=1}^M \left(\frac{1}{M}\right)^2 \log \frac{2}{\delta}} = \mathbb{E}[\Phi(S)] + \sqrt{\frac{\log(2/\delta)}{2M}}. \quad (22)$$

**Step 3: Bounding the Expectation via Rademacher Complexity.** We now bound  $\mathbb{E}[\Phi(S)]$ . Using standard symmetrization arguments. Let  $S = \{w_m\}_{m=1}^M$  and  $S' = \{w'_m\}_{m=1}^M$  be an independent copy drawn from  $\rho$ . Standard symmetrization gives:

$$\mathbb{E}_S \left[ \sup_{\theta} (Pf_{\theta} - P_M f_{\theta}) \right] \leq \mathbb{E}_{S, S'} \left[ \sup_{\theta} \frac{1}{M} \sum_{m=1}^M (f_{\theta}(w'_m) - f_{\theta}(w_m)) \right].$$

By symmetry of  $(S, S')$  and introducing i.i.d. Rademacher signs  $\sigma_m \in \{-1, +1\}$ ,

$$\mathbb{E}_{S, S'} \left[ \sup_{\theta} \frac{1}{M} \sum_{m=1}^M (f_{\theta}(w'_m) - f_{\theta}(w_m)) \right] \leq 2 \mathbb{E}_{S, \sigma} \left[ \sup_{\theta} \frac{1}{M} \sum_{m=1}^M \sigma_m f_{\theta}(w_m) \right] = 2\mathfrak{R}_M(\mathcal{F}).$$

Combining this with Equation equation 22 yields ( with probability at least  $1 - \delta$ ):

$$\Phi(S) \leq 2\mathfrak{R}_M(\mathcal{F}) + \sqrt{\frac{\log(2/\delta)}{2M}}. \quad (23)$$

**Step 4: Bounding the Rademacher Complexity.** Fix the sample  $S = \{w_m\}_{m=1}^M$  as before. Map each parameter  $\theta$  to the vector

$$v(\theta) := \frac{1}{\sqrt{M}} (f_{\theta}(w_1), \dots, f_{\theta}(w_M)) \in \mathbb{R}^M, \quad V := \{v(\theta) : \theta \in \Theta\}.$$

Then,

$$\mathfrak{R}_M(\mathcal{F}) = \frac{1}{\sqrt{M}} \mathbb{E}_{\sigma} \left[ \sup_{v \in V} \langle \sigma, v \rangle \right].$$

Rademacher complexity is translation-invariant in expectation. Thus, for any center  $c \in \mathbb{R}^M$ ,

$$\begin{aligned} \mathbb{E}_{\sigma} \left[ \sup_{v \in V} \langle \sigma, v \rangle \right] &= \mathbb{E}_{\sigma} \left[ \sup_{v \in V} \langle \sigma, v - c \rangle \right] \\ &\leq \mathbb{E}_{\sigma} \|\sigma\|_2 \sup_{v \in V} \|v - c\|_2. \end{aligned}$$

Choosing  $c$  as the center of the minimum-radius enclosing ball of  $V$ , we obtain

$$\sup_{v \in V} \|v - c\|_2 = \frac{1}{2} \text{diam}_2(V),$$

and since  $\mathbb{E}\|\sigma\|_2 = \sqrt{M}$ ,

$$\mathbb{E}_{\sigma} \left[ \sup_{v \in V} \langle \sigma, v \rangle \right] = \sqrt{M} \cdot \frac{\text{diam}_2(V)}{2}.$$

Therefore,

$$\mathfrak{R}_M(\mathcal{F}) \leq \frac{1}{\sqrt{M}} \cdot \sqrt{M} \cdot \frac{\text{diam}_2(V)}{2} = \frac{\text{diam}_2(V)}{2}.$$

Next, for any  $\theta, \theta' \in \Theta$ ,

$$\begin{aligned} \|v(\theta) - v(\theta')\|_2^2 &= \frac{1}{M} \sum_{m=1}^M (f_\theta(w_m) - f_{\theta'}(w_m))^2 \\ &\leq \frac{1}{M} \sum_{m=1}^M (L_\Theta \|\theta - \theta'\|_\Theta)^2 \\ &= L_\Theta^2 \|\theta - \theta'\|_\Theta^2. \end{aligned}$$

Hence,

$$\|v(\theta) - v(\theta')\|_2 \leq L_\Theta \|\theta - \theta'\|_\Theta.$$

Because  $\Theta$  lies in a ball of radius  $B_\Theta$ , its parameter diameter satisfies

$$\text{diam}_\Theta(\Theta) \leq 2B_\Theta.$$

Consequently,

$$\text{diam}_2(V) \leq L_\Theta \text{diam}_\Theta(\Theta) \leq 2L_\Theta B_\Theta.$$

Combining the above bounds yields

$$\mathfrak{R}_M(\mathcal{F}) \leq \frac{B_\Theta L_\Theta}{\sqrt{M}}. \quad \square \tag{24}$$

**Step 5: Final Bound.** Substituting Equation equation 23 and Equation equation 24 into the decomposition in Equation equation 13 yields

$$G(\theta^*) - G(\tilde{\theta}) \leq \mathcal{O}\left(\frac{L_\Theta B_\Theta}{\sqrt{M}} + \sqrt{\frac{\log(1/\delta)}{M}}\right) + \varepsilon_{\text{opt}}.$$

This completes the proof of Theorem 5.2 and confirms that the generalization gap vanishes at rate  $\mathcal{O}(1/\sqrt{M})$ , controlled by the smoothness of the scalarization function and the model capacity radius.  $\square$