

Direct Preference Density Alignment for Conversational Audio Equalization

Ioannis Stylianou

Department of Electronic Systems
Aalborg University
Aalborg, Denmark
iost@es.aau.dk

Sven Ewan Shepstone

Bang & Olufsen A/S
Struer, Denmark
ssh@bang-olufsen.dk

Jon Francombe

Bang & Olufsen A/S
Struer, Denmark
jofr@bang-olufsen.dk

Pablo Martinez Nuevo

Bang & Olufsen A/S
Struer, Denmark
pmn@bang-olufsen.dk

Zheng-Hua Tan

Department of Electronic Systems
Aalborg University
Aalborg, Denmark
zt@es.aau.dk

Abstract

Large Language Model alignment typically relies on learned proxy reward models, which significantly increase the memory footprint during training and are notoriously prone to instability and reward hacking. While offline methods like Direct Preference Optimization (DPO) bypass the reward model, they lose the ability to perform online exploration. If no optimization constraints are applied, this can lead to format collapse in bounded, continuous spaces. To resolve this, we propose Direct Preference Density Alignment: An alternative framework that removes the need for a learned proxy reward model while strictly preserving the benefits of online reinforcement learning. We leverage large-scale user data (approximately 90,000 samples) to construct non-parametric preference density maps, establishing an empirical reward surface. In addition to removing the reward model, Direct Preference Density Alignment enables the combination of the online structural grounding of Group Relative Policy Optimization (GRPO) with the targeted offline refinement of DPO. We show that this GRPO+DPO combination achieves the highest performance, and in a blind audio equalization listening test, enables a 1.5B-parameter model to achieve perceptual parity with a carefully prompt-engineered GPT-4o mini baseline, using only a fraction of the inference compute.

1 Introduction

Aligning Large Language Models (LLMs) to subjective continuous-control tasks remains a significant open challenge. Modern Reinforcement Learning from Human Feedback (RLHF) [23] pipelines rely heavily on Proximal Policy Optimization (PPO) and learned proxy Reward Models (RMs). However, RMs are highly susceptible to “reward hacking” [13] and PPO is extremely resource-intensive, requiring four separate LLMs (Trained, Reference, Reward, and Value models) during training.

Recent advancements attempt to mitigate this compute burden. Group Relative Policy Optimization (GRPO) [30] eliminates the value model, while offline methods like Direct Preference Optimization (DPO) [25] eliminate both the value and reward models. However, we demonstrate that by shifting from online Reinforcement Learning (RL) to offline “contrastive” learning, DPO loses active exploration. In compact spaces [12], this lack of environmental (or programmatic) grounding leads to format collapse, where the model forgets valid syntactical constraints.

To resolve this, we propose a proxy-free, online RL framework for continuous parameter estimation. Instead of training an unstable proxy RM, we leverage population data to construct static, non-parametric preference surfaces. The models then navigate the surface like a topographical map, exploring different regions to eventually find the “peaks” of highest human satisfaction. We instantiate this framework on a subjective audio equalization task, mapping abstract natural language (e.g., “make it warmer”) to loudspeaker parameters. Crucially, this empirical reward surface allows us to combine the online structural grounding of GRPO with the targeted offline refinement of DPO, outperforming either of the two algorithms in isolation, as well as Supervised Fine-Tuning (SFT) and In-Context Learning (ICL) baselines.

Our contributions are summarized as follows:

- **Proxy-free online RL framework:** We introduce a paradigm for aligning LLMs that completely eliminates the instability and memory overhead of learned reward models, while preserving the ability for online exploration.
- **Hybrid alignment strategy:** Our approach seamlessly bridges online structural grounding (GRPO) and offline preference sharpening (DPO), overcoming the inherent limitations of each method when used in isolation.
- **Improved distillation:** Through this GRPO+DPO pipeline, we successfully distill subjective population preferences into a 1.5B parameter model, achieving perceptual parity with GPT-4o mini in a blind A/B listening test.

2 Related Work

Our work positions itself at the intersection of semantic audio control, preference alignment, and RL in continuous action spaces. Traditionally, mapping natural language to audio parameters has been treated as a supervised regression problem. Early systems relied on fixed vocabulary-to-parameter mappings [24], while recent approaches employ LLMs to interpret unconstrained text and estimate equalization (EQ) and reverberation (reverb) parameters [7, 8]. Crucially, these regression-based methods implicitly favor a single parameter realization at inference time. Prior work [31] challenged this assumption, proposing that audio descriptors like “warm” map to distributions rather than point estimates. However, that study relied on small-scale supervised learning ($N = 11$ per prompt). Our work scales this distributional premise ($N \approx 3,000$ per prompt) and transitions from supervised matching to RL, enabling on-policy optimization that fully exploits densely annotated data.

The standard LLM RL-tuning pipeline [23] typically relies on PPO [26] optimized against a learned reward model. However, PPO requires training a separate Value Network to estimate the expected return. In subjective multimodal domains like audio preference, fitting a Value Network introduces an additional proxy objective, compounding the risk of reward misspecification [5]. To circumvent the instability of critics and reward models, recent work has shifted toward direct alignment methods. Approaches like DPO, Identity Preference Optimization (IPO) [1], Kahneman-Tversky Optimization (KTO) [10] and other variants [6, 20, 16], optimize the policy directly from preference data without an explicit reward function. However, these methods are blind to the probability mass assigned to completions outside the preference dataset [36, 15]. As a result, the model may increase the probability of malformed or constraint-violating outputs during training, even though such outputs are never explicitly preferred. As we demonstrate, this lack of active exploration makes them brittle when the model must actively learn to maintain strict formatting constraints. While extensions like C-DPO [22] attempt to mitigate this by integrating explicit penalty margins, the lack of active exploration remains a bottleneck in continuous control tasks where the model must inherently learn the structural manifold.

GRPO [30] offers a middle ground between the heavy machinery of PPO and the static nature of DPO. Originally designed for reasoning tasks, GRPO removes the value network and instead uses the average reward of a group of generated rollouts as the baseline. By generating its own data during training, GRPO provides the exploration necessary to learn format constraints. Other variants such as Dr.GRPO [21] and DAPO [37] have proposed advanced advantage estimation and regularization techniques to improve upon this foundation. However, in this study we deliberately evaluate the vanilla instantiations of both DPO and GRPO. By omitting explicitly constrained decoding, customized validity penalties, or specific algorithmic extensions, we isolate the fundamental structural trade-off:

the formatting failures of pure offline contrastive learning versus on-policy exploration of online methods.

To mitigate the individual weaknesses of online and offline methods, recent works in reasoning and multimodal generation have begun exploring hybrid pipelines that sequence online exploration with offline refinement [19, 34, 33]. Building on this emerging paradigm, our work adapts this hybrid stabilization strategy to continuous parameter estimation. Crucially, we do so while completely eliminating the reliance on the learned verifiers or reward models used in these prior works.

3 Preliminaries and reward surface construction

To effectively align an LLM with subjective auditory preferences, we must first define the control space and construct a reliable reward signal that captures the nuance of human perception without relying on unstable proxy networks.

3.1 The Beosonic control space

We ground our study in the “Beosonic” interface [3], a continuous two-dimensional control plane $\mathcal{X} \in [-6, 6]^2$. This space reduces the EQ transfer function to 2 parameters, with each corresponding to the intensity of a distinct equalization filter (± 6 gain), as illustrated in Figure 1. With this approach we effectively represent audio equalization through an interpretable 2D parameter space. Unlike discrete selection tasks, the model must predict a continuous coordinate $x \in [-6, 6]^2$.

3.2 Dataset and motivation

Our reward signal is the product of a large-scale comparison dataset, originally collected to evaluate the framework proposed in [31]. In this deployment, participants were presented with natural language prompts paired with audio variations generated by four distinct strategies: a deterministic Zero-Shot LLM (T2B), two distributional baselines (LoRA, RAG), and a Random agent. For each variation, users provided binary feedback indicating whether the audio satisfied the prompt.

The resulting dataset comprises $N \approx 90,000$ interaction events (consisting of both selections and rejections). An analysis of the positive selections (Figure 2) reveals a critical insight: The prompt-engineered Zero-Shot baseline (T2B) achieved the highest selection count (14,087), significantly outperforming the more complex distributional models. This outcome suggests that the prior distributional approaches, having been trained on extremely sparse data ($N = 11$), failed to generalize effectively.

This observation motivates our RL approach: We require an objective that can explicitly optimize for the peaks of human satisfaction (Exploitation) while maintaining sufficient coverage of the space (Exploration).

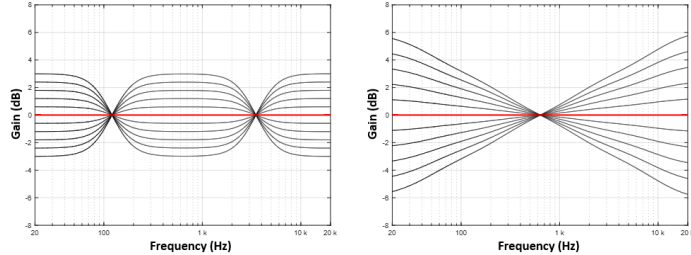


Figure 1: Filters of the EQ controller. Horizontal movement yields a “smile curve” (left); vertical applies a linear adjustment (right).

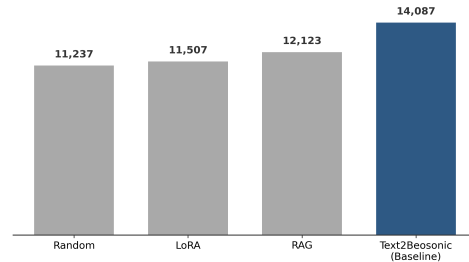


Figure 2: Total user selection counts by model type. The Zero-Shot baseline (T2B) outperforms other models (LoRA, RAG), indicating that previous attempts at distributional modeling failed to capture preference modes.

3.3 Probability of selection

To construct the reward function $R(x|p)$ for a specific prompt p , we cannot simply estimate the density of high-rated points, as this would be biased by the sampling distribution of the generating models (e.g., if a model samples the center frequently, high density there might differ from high preference).

We therefore aim to estimate the Probability of Selection $P(\text{Selected}|x)$ for a given coordinate x . Let $\mathcal{D}_{all}$ be the set of all offered coordinates for prompt p , and $\mathcal{D}_{pref} \subseteq \mathcal{D}_{all}$ be the same set weighted by binary selection events. We estimate two densities using Reflective-KDE (defined in Subsection 3.4)

1. $\hat{f}_{total}(x)$: The density of the proposal distribution (where the points came from; $P(x)$).
2. $\hat{f}_{pref}(x)$: The density of the users' preferred regions ($P(x|\text{Selected})$).

By applying Bayes' theorem, the Probability of Selection is proportional to the ratio of the positive density to the total offered density:

$$S(x) = P(\text{Selected} | x) = \frac{P(x | \text{Selected}) P(\text{Selected})}{P(x)} \propto \frac{\hat{f}_{pref}(x)}{\hat{f}_{total}(x) + \epsilon} \quad (1)$$

where ϵ ensures numerical stability. This ratio effectively cancels out the sampling bias, isolating the underlying population preference.

3.4 Reflective kernel density estimation

Standard KDE suffers from boundary bias in bounded spaces, underestimating density near the edges (where user preferences often cluster, e.g., "Max Bass"). To correct this, we adopt Reflective-KDE [27], augmenting the dataset by mirroring each point x_i across the boundaries of $\mathcal{X}$ into the 8 surrounding neighbor squares, creating a 3×3 grid of "ghost" points. The density estimate becomes:

$$\hat{f}(x) = \frac{1}{9nh^2} \sum_{i=1}^n \sum_{k=0}^8 K\left(\frac{x - T_k(x_i)}{h}\right) \quad (2)$$

where T_k represents the reflection mappings (T_0 is the Identity) and K is a Gaussian kernel. We set the bandwidth $h = 0.25 \cdot h_{\text{Scott}}$. This scaling factor on Scott's Rule [28] accounts for two specific properties of our space: a 0.5 factor to adjust for the extra point mass introduced by the reflective boundary, and an additional 0.5 factor to counteract the known tendency of Scott's Rule to over-smooth non-convex spaces. An example of the effect of different bandwidths is provided in Appendix C.

3.5 Global normalization and augmentation

To ensure the reward signal is consistent across different prompts, we perform a global normalization pass. We compute the global maximum (S_{max}) and minimum (S_{min}) density ratios across the entire dataset. The final reward $R(x)$ used for RL training is scaled to $[-1, 1]$:

$$R(x) = 2 \cdot \frac{S(x) - S_{min}}{S_{max} - S_{min}} - 1 \quad (3)$$

The whole reward-surface construction process is exemplified in Figure 3.

Finally, we employ semantic data augmentation. We assume that synonymous prompts (e.g., "Make it warmer" and "Increase warmth") share an identical preference manifold. We aggregate data for synonymous clusters and map them to the same reward surface.

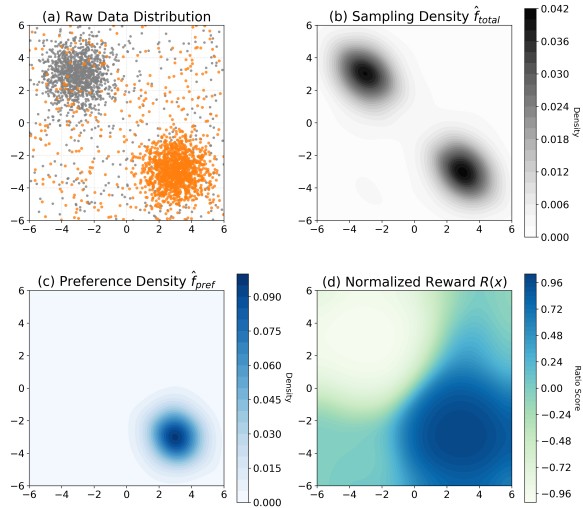


Figure 3: **Non-Parametric Reward.** (a) Sampling bias. (b) Total density $\hat{f}_{total}$. (c) Preferred density $\hat{f}_{pref}$. (d) Final normalized Reward Surface $R(x)$.

3.6 Evaluation splits

To assess generalization, we partition the prompts into three sets; a training set containing approximately 80% of the prompts used for policy updates, a 10% validation set consisting of the randomly held-out prompts drawn from the same distribution, and a manually isolated 10% Out-Of-Distribution (OOD) set containing a cluster of prompts related to ‘‘Vocal Clarity.’’ These OOD concepts are entirely unseen during training, allowing to evaluate the model’s ability to extrapolate its learned preference geometry to novel semantic inputs.

4 Alignment strategies in continuous spaces

Having constructed a non-parametric reward surface $R(x)$, our goal is to align an LLM policy π_θ to maximize the expected return $\mathbb{E}_{x \sim \pi_\theta} [R(x)]$. In this section, we analyze the structural limitations of standard offline alignment in continuous spaces and propose a hybrid pipeline that utilizes an online-aligned model to synthesize its own high-precision preference dataset.

4.1 The challenge of continuous formatting

Unlike standard chat models, our agent must output valid coordinates in a strict format (e.g., $[1.5, -3.2]$). A critical failure mode in this domain is *format collapse*, where the model drifts into generating conversational text or invalid syntax.

We hypothesize that offline methods (i.e., DPO) are particularly susceptible to this. Since DPO is limited to the specific negative examples in the static dataset, it cannot penalize emergent structural failures. As the policy shifts, it may drift into generating novel malformed syntaxes that were not present in the pre-collected data, effectively bypassing the optimization constraints. Conversely, online methods (like GRPO) generate their own predictions during training. This allows the environment to provide immediate negative feedback ($R = -1.0$) for syntax errors, effectively ‘‘grounding’’ the model in the constraints of the control interface.

4.2 Baseline 1: direct preference optimization

DPO trains the policy by increasing the margin between preferred responses and dispreferred ones:

$$\mathcal{L}_{\text{DPO}} = -\mathbb{E}_{(p, x_w, x_l) \sim \mathcal{D}} [\log \sigma(\hat{r}_\theta(p, x_w) - \hat{r}_\theta(p, x_l))] \quad (4)$$

where the implicit reward is defined as: $\hat{r}_\theta(p, x) = \beta \log \frac{\pi_\theta(x|p)}{\pi_{\text{ref}}(x|p)}$.

While efficient, DPO relies heavily on the Kullback-Leibler (KL)-divergence constraint (implicitly controlled by β) to prevent the model from deviating from the reference policy π_{ref} . In our domain, we construct the dataset $\mathcal{D}$ using the reward map $R(x)$, where x_w and x_l are pairs of coordinates such that $R(x_w|p) > R(x_l|p)$. More details are provided in Subsection 4.4.

4.3 Baseline 2: group relative policy optimization

We adopt GRPO as our online framework. Unlike PPO, GRPO eliminates the value model. For each prompt p , the policy generates a group of G outputs $\{x_1, \dots, x_G\}$. By setting $\rho_i(\theta) = \frac{\pi_\theta(x_i|p)}{\pi_{\text{old}}(x_i|p)}$, the objective maximizes:

$$\mathcal{J}_{\text{GRPO}}(\theta) = \mathbb{E}_{p \sim P(p), \{x_i\}_{i=1}^G \sim \pi_{\theta_{\text{old}}}} \left[\underbrace{\frac{1}{G} \sum_{i=1}^G \left(\min \left(\rho_i(\theta) A_i, \text{clip}(\rho_i(\theta), 1 - \epsilon, 1 + \epsilon) A_i \right) \right)}_{\text{Clipped Surrogate Objective}} - \underbrace{\beta \mathbb{D}_{\text{KL}}(\pi_\theta || \pi_{\text{ref}})}_{\text{KL Penalty}} \right] \quad (5)$$

The advantage A_i is computed by normalizing the rewards within the group. Crucially, if the model generates invalid format, the reward function returns $R = -1.0$. Since this penalty is included in the group, the model learns to avoid invalid syntax to achieve a positive advantage relative to its peers.

4.4 Proposed method: Hybrid GRPO+DPO pipeline

We observe that while GRPO excels at structural stability (learning *how* to format), it may fail to find all the local maxima of the preference density (learning exactly *what* different people want to hear). To resolve this, we propose a two-stage pipeline that combines the stability of online RL with the precision of offline RL.

Stage 1: Structural Alignment (GRPO). We first train π_θ using GRPO against the Density Map. This transforms the generalist LLM into a robust control agent (π_{GRPO}) that reliably outputs coordinates in the correct manifold.

Stage 2: Synthetic Preference Mining & Peak Injection. We construct a synthetic DPO dataset $\mathcal{D}_{syn}$ using π_{GRPO} as the generator. To ensure the dataset steers the model toward the local optima, we employ a peak injection strategy:

1. **On-Policy Mining:** We sample completions from π_{GRPO} . Those with high rewards ($R(x) > 0.8$) are added to the positive pool; malformed or low-reward outputs ($R(x) < -0.2$) are added to the negative pool.
2. **Peak Injection:** If π_{GRPO} fails to discover the top-K local maxima of the Density Map $R(x)$ for a given prompt, we analytically find the peak coordinates via grid search on $R(x)$ and inject them into the positive pool as “Golden Truths.”

The process is exactly the same as in standard DPO (Subsection 4.2), with π_{ref} as the generator instead of π_{GRPO} .

Stage 3: Refinement (DPO). We tune the model using DPO on $\mathcal{D}_{syn}$, setting $\pi_{ref} = \pi_{GRPO}$. By initializing with the GRPO model and training on injected expert peaks with relatively high β values, the hybrid model achieves both high format compliance and improved preference matching.

5 Experiments and results

We evaluate our alignment strategies on the OOD test set described in Section 3.6. This set contains semantic concepts unseen during training, requiring the model to generalize the geometric understanding of the preference space.

5.1 Experimental setup

Models: We utilize the **Qwen2.5-Instruct** family (0.5B and 1.5B parameters) [32] as our base models due to their strong instruction-following capabilities and open weights (Apache License 2.0).

Baselines: We compare against ICL baselines using Qwen2.5 and GPT-4o mini. To establish a rigorous performance floor, we inject domain knowledge into these baselines via a sophisticated system prompt containing an explicit semantic-to-coordinate mapping dictionary (e.g., “warm” $\rightarrow [0, -6]$), derived from audio attribute clustering literature [11]. In contrast, our RL-tuned models are agnostic to these expert definitions; they rely on a minimal system prompt and must learn the semantic-to-acoustic relationships purely from the reward signal.

Metrics:

Greedy Mean Relative Reward (GMRR): We evaluate the models’ performance on their most probable (**greedy**) completion for each prompt. Additionally, while training utilizes globally normalized rewards, evaluation requires measuring optimality **relative** to the specific prompt. We therefore revert to local normalization: For each prompt, the reward is rescaled such that the maximum of the density map is 1.0 and the minimum is 0.0.

$$\text{GMRR}(x) = \frac{S(x) - S_{\min, \text{local}}}{S_{\max, \text{local}} - S_{\min, \text{local}}}. \quad (6)$$

Under this metric, a score of 1.0 indicates the model found the absolute peak of human preference for that specific query, while the absolute low or malformed outputs are assigned 0.0.

Format Success Rate (Parse Rate): The percentage of generated outputs that **strictly** adhere to the coordinate format $[x, y]$.

Table 2: Performance comparison of Qwen2.5 models under the GRPO algorithm with varying numbers of rollouts (sample size) and fixed β value (0.5). Metrics shown are Greedy Mean Relative Reward ($\pm$ Standard Deviation) and format success rate (%).

Model	2 Rollouts		4 Rollouts		8 Rollouts		16 Rollouts		32 Rollouts	
	GMRR	Parse Rate	GMRR	Parse Rate	GMRR	Parse Rate	GMRR	Parse Rate	GMRR	Parse Rate
Qwen2.5-0.5B	0.47 $\pm$ 0.17	81%	0.50 $\pm$ 0.14	93%	0.53 $\pm$ 0.10	100%	0.51 $\pm$ 0.10	100%	0.48 $\pm$ 0.14	96%
Qwen2.5-1.5B	0.50 $\pm$ 0.08	100%	0.51 $\pm$ 0.09	100%	0.51 $\pm$ 0.09	100%	0.52 $\pm$ 0.09	100%	0.52 $\pm$ 0.09	100%

5.2 The stability gap: Online vs. offline

Our first experiment investigates the structural stability of the alignment algorithms. We sweep the KL-divergence penalty coefficient $\beta \in \{0.0, 0.1, 0.5, 1.0\}$ for both DPO (Offline) and GRPO (Online) and measure the Format Success Rate on the OOD test set.

As shown in Table 1, a critical failure mode emerges for DPO. A low β value allows the model to deviate more aggressively from its initial state. During training, the objective of the algorithm is to maximize the margin between the positive and negative examples in the pre-developed preference dataset. This approach indeed increases the relative probability of generating a preferred example, *but only compared to generating a dis-preferred one*. It is likely, especially with low KL regularization, that the overall probability of generating a positive example decreases during training. However, the training objective is still met as long as the overall probability of the negative counterpart decreases even faster.

A different way to view this is that offline DPO tends to “tunnel-vision” on the two examples within each preference dataset entry, while the probabilities of all other completions are updated without any supervision. In contrast, GRPO maintains a 100% format success rate even at $\beta = 0.0$. Because the online agent generates its own data during training, any format deviation results in an immediate penalty ($R = -1$), forcing the policy to remain within the valid syntax manifold. Interestingly, excessively large β values can over-constrain the model, hindering its ability to follow strict structure.

5.3 Scaling exploration

We further analyze the impact of the group size G (number of rollouts per prompt) on GRPO performance (Table 2). Increasing G consistently improves the GMRR, saturating between $G = 8$ and $G = 16$. However, this improvement is relatively small, with even the 2-Rollout models outperforming their sophisticated ICL counterparts. This observation aligns with recent studies [35], which question the benefits of larger samples sizes relative to the computational burden.

5.4 Hybrid pipeline performance

Finally, we evaluate the proposed **hybrid GRPO+DPO** pipeline. We select the best GRPO configuration ($\beta = 0.5$) as the Reference and use DPO ($\beta = 1.0$) on the synthetically mined “Peak Injection” dataset (Subsection 4.4) across different rollouts.

Results in Table 3 demonstrate that for the 0.5B model, GMRR jumps from 0.53 (Base GRPO) to **0.56** (hybrid), matching the GPT-4o baseline at the cost of some inconsistency (97% Parse Rate). The 1.5B model achieves a GMRR of **0.60** without compromising format compliance. However, the hybrid approach exhibits performance fluctuations across rollout configurations. To confirm this reflects hyperparameter sensitivity rather than algorithmic instability, we evaluated the optimal configuration across four independent seeds, achieving a consistent mean GMRR of 0.58 (see Appendix E).

Table 1: GMRR ($\pm$ STD) on the OOD test set and format success rate (%). The top section compares the GRPO and DPO methods for different β values. The bottom section provides ICL reference scores.

β	Model	12-GRPO		DPO	
		GMRR	Rate	GMRR	Rate
0.0	Qwen2.5-0.5B	0.44 $\pm$ 0.08	100%	0.00	0%
	Qwen2.5-1.5B	0.52 $\pm$ 0.10	100%	0.00	0%
0.1	Qwen2.5-0.5B	0.49 $\pm$ 0.14	100%	0.31 $\pm$ 0.27	47%
	Qwen2.5-1.5B	0.53 $\pm$ 0.09	100%	0.00	0%
0.5	Qwen2.5-0.5B	0.52 $\pm$ 0.10	100%	0.33 $\pm$ 0.22	81%
	Qwen2.5-1.5B	0.53 $\pm$ 0.09	100%	0.32 $\pm$ 0.27	69%
1.0	Qwen2.5-0.5B	0.50 $\pm$ 0.10	94%	0.30 $\pm$ 0.25	98%
	Qwen2.5-1.5B	0.51 $\pm$ 0.09	100%	0.07 $\pm$ 0.23	10%
ICL Reference Models		GMRR		Rate	
		Qwen2.5-0.5B		0.25 $\pm$ 0.24	
		Qwen2.5-1.5B		0.49 $\pm$ 0.08	
		GPT-4o mini		0.56 $\pm$ 0.07	
				54%	
				100%	
				100%	

Table 3: Performance comparison of Qwen2.5 models using GRPO (Baseline) versus the combined GRPO + DPO algorithm. The β values for GRPO and DPO are fixed at 0.5 and 1.0 respectively.

Model	Rollouts	Approach	GMRR	Parse Rate (%)	Observation
Qwen2.5-0.5B	2	GRPO	0.47 $\pm$ 0.17	81	
		GRPO+DPO	0.44 $\pm$ 0.32	88	Drop in GMRR, improvement in parsing rate
	4	GRPO	0.50 $\pm$ 0.14	93	
		GRPO+DPO	0.56 $\pm$ 0.26	73	Preference sharpened, but stability drops
	8	GRPO	0.53 $\pm$ 0.10	100	
		GRPO+DPO	0.48 $\pm$ 0.25	100	Drop in GMRR
	16	GRPO	0.51 $\pm$ 0.10	100	
		GRPO+DPO	0.49 $\pm$ 0.32	74	Drop in both metrics
	32	GRPO	0.48 $\pm$ 0.14	96	
		GRPO+DPO	0.56 $\pm$ 0.18	97	Improvement in both metrics
Qwen2.5-1.5B	2	GRPO	0.50 $\pm$ 0.08	100	
		GRPO+DPO	0.27 $\pm$ 0.24	100	GMRR Drop
	4	GRPO	0.51 $\pm$ 0.09	100	
		GRPO+DPO	0.56 $\pm$ 0.18	78	Improvement but stability degradation hurts GMRR
	8	GRPO	0.51 $\pm$ 0.09	100	
		GRPO+DPO	0.40 $\pm$ 0.17	100	GMRR Drop
	16	GRPO	0.52 $\pm$ 0.09	100	
		GRPO+DPO	0.60 $\pm$ 0.17	100	Strongest Improvement (GMRR & Rate)
	32	GRPO	0.52 $\pm$ 0.09	100	
		GRPO+DPO	0.48 $\pm$ 0.13	100	GMRR Drop

To assess if the preference manifold could be learned via supervision, we also trained an SFT baseline with Low Rank Adaptation [14] (LoRA) on the “peaks” of the training prompts. This model achieved a GMRR of 0.45, performing worse than the base model with In-Context Learning (0.49). This suggests that SFT leads to overfitting, degrading the model’s ability to extrapolate to OOD semantic concepts. Table 4 provides a performance summary of the different methods for Qwen2.5-1.5B.

6 Subjective evaluation

While our objective metrics demonstrate the convergence of the proposed RL approach, the ultimate validation of an audio system lies in listener perception. We conducted a blind A/B listening test to compare our best-performing hybrid model (Qwen2.5-1.5B) against the baseline (GPT-4o mini via In-Context Learning). Both models controlled the exact same equalization system, ensuring that any perceived differences result purely from the parameters selected by the LLMs.

6.1 Unconstrained data collection and protocol

To ensure the evaluation reflected real-world interaction, we utilized a two-phase study with human listeners. First, we recruited 12 participants for a prompt-collection phase. We employed an unconstrained elicitation protocol: Participants listened to audio clips and were asked to request *any* changes they desired without being informed of the system’s technical limitations (i.e., that it was an equalizer). This design choice was deliberate to avoid *priming bias*, ensuring that users did not subconsciously tailor their requests to perceived system capabilities (e.g., only asking for “more bass”). As expected, this yielded highly Out-Of-Distribution requests (e.g., volume control, removing instruments, or changing speaker identity). After filtering out these functional system requests, a set of 30 prompts was retained.

In the second phase, 11 of the original participants returned to evaluate the models’ responses on these 30 prompts. For each prompt, participants listened to audio processed by both the hybrid RL model and the GPT-4o mini baseline in a randomized, blinded interface. They rated how well the

Table 4: Performance summary of Qwen2.5-1.5B on the OOD Test Set. **GRPO** improves generalization compared to **SFT** via online exploration, while the **hybrid GRPO+DPO** pipeline achieves the highest performance.

Model	Approach	GMRR	Rate
1.5B	SFT (Peak)	0.45 $\pm$ 0.12	100%
1.5B	ICL	0.49 $\pm$ 0.08	100%
1.5B	GRPO	0.53 $\pm$ 0.09	100%
1.5B	Hybrid	0.60 $\pm$ 0.17	100%

resulting audio satisfied the original request on a 5-point Likert scale (1=“Very Poorly” to 5=“Very Well”).

6.2 Quantitative results

The aggregate results (Table 5) indicate that our specialized RL model performs at parity with the previously-best generalist baseline. More specifically:

Mean Preference Score: The 1.5B RL model achieved a mean rating of **3.04**, outperforming the GPT-4o mini baseline (2.92). This suggests that the RL model handles ambiguous, OOD prompts gracefully.

Win/Tie Rate: In head-to-head comparisons, the RL model was preferred or rated as equivalent to the baseline in **77.9%** of the 330 trials (11 participants $\times$ 30 prompts).

Statistical Analysis: Because a total of 660 evaluations (two per trial) were presented individually in randomized order, we analyzed the results using a Linear Mixed-Effects Model (LMM) with crossed random effects for both participant and prompt to account for repeated measures [2]. The LMM revealed a non-significant main effect between the models ($p = 0.095$). To formally establish perceptual parity between the 1.5B model and the baseline, we subsequently conducted an equivalence test using the Two One-Sided Tests (TOST) procedure [18]. Detailed methodology, the full LMM summary matrix, and the TOST confidence intervals demonstrating statistical parity are provided in Appendix A.

The critical implication of these results is efficiency: The proposed Density-Based RL pipeline successfully distilled the necessary audio control capabilities into a model that is not utilizing expert knowledge and is orders of magnitude smaller than its competitor, achieving equivalent user satisfaction without the need for large inference compute or proprietary APIs.

Table 5: Subjective evaluation on Likert score ($\pm$ Standard Error). The p -value is calculated via the Linear Mixed-Effects Model.

Method	Score (1–5)	Win/Tie %	p -value
GPT-4o	2.92 $\pm$ 0.06	72.1%	0.095
Hybrid	3.04 $\pm$ 0.06	77.9%	

7 Discussion and limitations

Our results suggest a paradigm shift for LLM alignment: Rather than training unstable proxy reward models, we can leverage the “wisdom of the crowd” directly via non-parametric density maps. However, this approach entails specific constraints that merit discussion.

The curse of dimensionality Our Reflective-KDE approach is highly effective for low-dimensional spaces (e.g., the 2D Beosonic interface). However, KDE suffers from the curse of dimensionality [4]. As the control space expands (e.g., additional acoustic parameters or a 6-DoF robotic arm), the data required to populate a dense reward map grows exponentially. Scaling this framework to high-dimensional action spaces will likely require latent variable models (e.g., Variational Auto Encoders [17]) to compress the control space into a tractable lower-dimensional manifold before applying our density-based RL.

Text-only conditioning The current system aligns audio parameters solely based on the semantic intent of the text prompt (e.g., “Make it warm”). It does not analyze the input audio signal ($P(\text{Selected}|x, \text{Audio}_{ID})$). While this enables a lightweight, sparse (responds when requested), privacy-preserving inference (no audio upload required), it limits the system’s context awareness. A “warm” adjustment for a podcast could differ significantly from a “warm” adjustment for techno music. Future iterations could integrate audio embeddings (e.g., CLAP [9] features) into the policy to enable content-adaptive equalization.

Consensus vs. personalization Finally, while our density map method effectively models the population’s preference space, it does not yet solve the problem of individual personalization. By optimizing for the peaks of the density map, the hybrid model acts as a “Mode-Seeking” agent—it generates the setting that could satisfy *any* user. It does not yet have the capacity to adapt to user-specific hearing profiles. Extending this framework to conditioned density estimation ($P(\text{Selected}|x, \text{User}_{ID})$) remains a promising and natural avenue for future research.

8 Conclusion

In this work, we presented a proxy-free reinforcement learning framework for continuous parameter estimation. By applying Reflective Kernel Density Estimation to large-scale user data ($N \approx 90,000$), we constructed non-parametric preference density maps that serve as a stable, empirical reward surface. This approach completely eliminates the memory overhead and reward-hacking vulnerabilities of learned proxy reward models, while preserving the ability to perform true online exploration.

Crucially, this continuous reward surface provides the flexibility to employ and combine distinct alignment methodologies. It seamlessly enables both the active, on-policy exploration of GRPO and the targeted, contrastive optimization of DPO. By leveraging this capability to build a hybrid pipeline, we successfully aligned a highly efficient 1.5B-parameter model, achieving perceptual parity with a heavily prompt-engineered GPT-4o mini baseline in a blind A/B listening test.

References

- [1] Mohammad Gheshlaghi Azar, Mark Rowland, Bilal Piot, Daniel Guo, Daniele Calandriello, Michal Valko, and Rémi Munos. A general theoretical paradigm to understand learning from human preferences, 2023. URL <https://arxiv.org/abs/2310.12036>.
- [2] R Harald Baayen, Douglas J Davidson, and Douglas M Bates. Mixed-effects modeling with crossed random effects for subjects and items. *Journal of memory and language*, 59(4):390–412, 2008.
- [3] Bang&Olufsen. Beosound beovision theatre: Technical sound guide, 2023.
- [4] Richard Bellman. Dynamic programming, princeton univ. *Press Princeton, New Jersey*, 39, 1957.
- [5] Stephen Casper, Xander Davies, Claudia Shi, Thomas Krendl Gilbert, Jérémy Scheurer, Javier Rando, Rachel Freedman, Tomasz Korbak, David Lindner, Pedro Freire, Tony Wang, Samuel Marks, Charbel-Raphaël Segerie, Micah Carroll, Andi Peng, Phillip Christoffersen, Mehul Damani, Stewart Slocum, Usman Anwar, Anand Siththaranjan, Max Nadeau, Eric J. Michaud, Jacob Pfau, Dmitrii Krasheninnikov, Xin Chen, Lauro Langosco, Peter Hase, Erdem Biyik, Anca Dragan, David Krueger, Dorsa Sadigh, and Dylan Hadfield-Menell. Open problems and fundamental limitations of reinforcement learning from human feedback, 2023. URL <https://arxiv.org/abs/2307.15217>.
- [6] Sayak Ray Chowdhury, Anush Kini, and Nagarajan Natarajan. Provably robust dpo: Aligning language models with noisy feedback. *arXiv preprint arXiv:2403.00409*, 2024.
- [7] Annie Chu, Patrick O’Reilly, Julia Barnett, and Bryan Pardo. Text2fx: Harnessing clap embeddings for text-guided audio effects. In *ICASSP 2025-2025 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 1–5. IEEE, 2025.
- [8] Seunghoon Doh, Junghyun Koo, Marco A Martínez-Ramírez, Wei-Hsiang Liao, Juhan Nam, and Yuki Mitsufuji. Can large language models predict audio effects parameters from natural language? *arXiv preprint arXiv:2505.20770*, 2025.
- [9] Benjamin Elizalde, Soham Deshmukh, Mahmoud Al Ismail, and Huaming Wang. Clap learning audio concepts from natural language supervision. In *ICASSP 2023-2023 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 1–5. IEEE, 2023.
- [10] Kawin Ethayarajh, Winnie Xu, Niklas Muennighoff, Dan Jurafsky, and Douwe Kiela. Kto: Model alignment as prospect theoretic optimization, 2024. URL <https://arxiv.org/abs/2402.01306>.
- [11] Jon Francombe, Timothy Brookes, and Russell Mason. Automatic text clustering for audio attribute elicitation experiment responses. In *AES 143rd Convention*. Audio Engineering Society, 2017.
- [12] M Maurice Fréchet. Sur quelques points du calcul fonctionnel. *Rendiconti del Circolo Matematico di Palermo (1884-1940)*, 22(1):1–72, 1906.

- [13] Leo Gao, John Schulman, and Jacob Hilton. Scaling laws for reward model overoptimization. In *International Conference on Machine Learning*, pages 10835–10866. PMLR, 2023.
- [14] Edward J. Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. Lora: Low-rank adaptation of large language models, 2021. URL <https://arxiv.org/abs/2106.09685>.
- [15] Hamish Ivison, Yizhong Wang, Jiacheng Liu, Zeqiu Wu, Valentina Pyatkin, Nathan Lambert, Noah A Smith, Yejin Choi, and Hannaneh Hajishirzi. Unpacking dpo and ppo: Disentangling best practices for learning from preference feedback. *Advances in neural information processing systems*, 37:36602–36633, 2024.
- [16] Haozhe Ji, Cheng Lu, Yilin Niu, Pei Ke, Hongning Wang, Jun Zhu, Jie Tang, and Minlie Huang. Towards efficient exact optimization of language model alignment. *arXiv preprint arXiv:2402.00856*, 2024.
- [17] Diederik P Kingma and Max Welling. Auto-encoding variational bayes. *arXiv preprint arXiv:1312.6114*, 2013.
- [18] Daniël Lakens. Equivalence tests: A practical primer for t tests, correlations, and meta-analyses. *Social psychological and personality science*, 8(4):355–362, 2017.
- [19] Yunxin Li, Xinyu Chen, Zitao Li, Zhenyu Liu, Longyue Wang, Wenhan Luo, Baotian Hu, and Min Zhang. Veripo: Cultivating long reasoning in video-llms via verifier-guided iterative policy optimization. *arXiv preprint arXiv:2505.19000*, 2025.
- [20] Tianqi Liu, Yao Zhao, Rishabh Joshi, Misha Khalman, Mohammad Saleh, Peter J Liu, and Jialu Liu. Statistical rejection sampling improves preference optimization. *arXiv preprint arXiv:2309.06657*, 2023.
- [21] Zichen Liu, Changyu Chen, Wenjun Li, Penghui Qi, Tianyu Pang, Chao Du, Wee Sun Lee, and Min Lin. Understanding r1-zero-like training: A critical perspective, 2025. URL <https://arxiv.org/abs/2503.20783>.
- [22] Zixuan Liu, Xiaolin Sun, and Zizhan Zheng. Enhancing llm safety via constrained direct preference optimization, 2024. URL <https://arxiv.org/abs/2403.02475>.
- [23] 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.
- [24] Bryan Pardo, David Little, and Darren Gergle. Building a personalized audio equalizer interface with transfer learning and active learning. In *Proceedings of the second international ACM workshop on Music information retrieval with user-centered and multimodal strategies*, pages 13–18, 2012.
- [25] 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.
- [26] John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. Proximal policy optimization algorithms, 2017. URL <https://arxiv.org/abs/1707.06347>.
- [27] Eugene F Schuster. Incorporating support constraints into nonparametric estimators of densities. *Communications in Statistics-Theory and methods*, 14(5):1123–1136, 1985.
- [28] David W Scott. Scott’s rule. *Wiley Interdisciplinary Reviews: Computational Statistics*, 2(4): 497–502, 2010.
- [29] Skipper Seabold, Josef Perktold, et al. Statsmodels: econometric and statistical modeling with python. *scipy*, 7(1):92–96, 2010.

- [30] 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.
- [31] Ioannis Stylianou, Jon Francombe, Pablo Mart´inez Nuevo, Sven Ewan Shepstone, and Zheng-Hua Tan. One prompt, many sounds: Modeling listener variability in llm-based equalization. *IEEE Journal of Selected Topics in Signal Processing*, page 1–13, 2026. ISSN 1941-0484. doi: 10.1109/jstsp.2026.3710145. URL <http://dx.doi.org/10.1109/JSTSP.2026.3710145>.
- [32] Qwen Team. Qwen2.5: A party of foundation models, September 2024. URL <https://qwenlm.github.io/blog/qwen2.5/>.
- [33] Chengzhuo Tong, Ziyu Guo, Renrui Zhang, Wenyu Shan, Xinyu Wei, Zhenghao Xing, Hongsheng Li, and Pheng-Ann Heng. Delving into rl for image generation with cot: A study on dpo vs. grpo. *arXiv preprint arXiv:2505.17017*, 2025.
- [34] Siye Wu, Jian Xie, Yikai Zhang, Aili Chen, Kai Zhang, Yu Su, and Yanghua Xiao. Arm: Adaptive reasoning model. *arXiv preprint arXiv:2505.20258*, 2025.
- [35] Yihong Wu, Liheng Ma, Lei Ding, Muzhi Li, Xinyu Wang, Kejia Chen, Zhan Su, Zhanguang Zhang, Chenyang Huang, Yingxue Zhang, Mark Coates, and Jian-Yun Nie. It takes two: Your grpo is secretly dpo, 2025. URL <https://arxiv.org/abs/2510.00977>.
- [36] Shusheng Xu, Wei Fu, Jiaxuan Gao, Wenjie Ye, Weilin Liu, Zhiyu Mei, Guangju Wang, Chao Yu, and Yi Wu. Is dpo superior to ppo for llm alignment? a comprehensive study. *arXiv preprint arXiv:2404.10719*, 2024.
- [37] Qiyang Yu, Zheng Zhang, Ruofei Zhu, Yufeng Yuan, Xiaochen Zuo, Yu Yue, Weinan Dai, Tiantian Fan, Gaohong Liu, Lingjun Liu, Xin Liu, Haibin Lin, Zhiqi Lin, Bole Ma, Guangming Sheng, Yuxuan Tong, Chi Zhang, Mofan Zhang, Wang Zhang, Hang Zhu, Jinhua Zhu, Jiaze Chen, Jiangjie Chen, Chengyi Wang, Hongli Yu, Yuxuan Song, Xiangpeng Wei, Hao Zhou, Jingjing Liu, Wei-Ying Ma, Ya-Qin Zhang, Lin Yan, Mu Qiao, Yonghui Wu, and Mingxuan Wang. Dapo: An open-source llm reinforcement learning system at scale, 2025. URL <https://arxiv.org/abs/2503.14476>.

A Extended statistical analysis of perceptual parity

To evaluate the subjective listening test results presented in Section 6.2, we transitioned from a naive paired hypothesis test to a Linear Mixed-Effects Model (LMM). The 660 user evaluations were presented individually and in randomized order to prevent anchoring bias. Because each participant rated both model outputs for each prompt, the dataset contains crossed random effects.

We fit an LMM using the `statsmodels` library [29], defining the subjective score (1–5 Likert scale) as the dependent variable and the model variation as the fixed effect. We included crossed random intercepts for both the `user_id` and the `prompt` to explicitly model the variance introduced by individual rater harshness and the inherent difficulty of specific audio requests. The full regression results are provided in Table 6.

Table 6: Linear Mixed-Effects Model regression results

Model: MixedLM		Dependent Variable: score				
No. Observations: 660		Method: REML				
Log-Likelihood: -905.9961		Converged: Yes				
Scale: 0.7866						
	Coef.	Std.Err.	z	P> z	[0.025	0.975]
Intercept (Baseline)	2.921	0.190	15.364	0.000	2.549	3.294
Variation [RL Model]	0.115	0.069	1.668	0.095	-0.020	0.250
prompt Var	0.247	0.086				
user_id Var	0.281	0.150				

The model converged successfully, yielding a non-significant main effect for the model variation ($p = 0.095$). The variance component estimates indicate that prompt-level variability ($\sigma^2 = 0.247$) was comparable in magnitude to participant-level variability ($\sigma^2 = 0.281$), validating our choice to use a crossed random-effects structure.

A.1 Two One-Sided Tests (TOST) for Equivalence

A non-significant p -value indicates a failure to reject the null hypothesis of no difference, but it does not formally prove that the two models are equivalent. To evaluate statistical equivalence, we conducted an equivalence test using the Two One-Sided Tests (TOST) framework [18].

We defined the margin of practical equivalence (Smallest Effect Size of Interest, or Δ) as ± 0.25 points on our 5-point Likert scale. The empirical standard deviation of the subjective ratings was approximately 0.88 ($SD \approx 0.88$). Under the common half-standard-deviation heuristic for practical significance, this corresponds to a margin of approximately ± 0.44 rating points. Our chosen equivalence margin of $\Delta = 0.25$ is therefore more conservative.

In the TOST procedure at an $\alpha = 0.05$ significance level, equivalence is established if the 90% Confidence Interval (CI) of the mean difference falls entirely within the $[-\Delta, +\Delta]$ bounds. Using the estimates for the Mean Difference (+0.115), Standard Error (0.069) and Critical value for the 90% CI ($z_{0.95} = 1.645$), the 90% Confidence Interval is calculated as:

$$CI_{90\%} = 0.115 \pm (1.645 \times 0.069) = [+0.002, +0.229] \quad (7)$$

The entire 90% Confidence Interval $[+0.002, +0.229]$ falls within our equivalence bounds of $[-0.25, +0.25]$, we reject the presence of a practically meaningful difference ($p_{equivalence} < 0.05$). These results support that the perceptual quality of the 1.5B RL model is statistically equivalent to the GPT-4o mini baseline within the predefined equivalence margin of ± 0.25 rating points.

B Subjective evaluation user interface

To ensure the transparency of our subjective evaluation (Section 6), we provide screenshots of the custom user interface presented to the participants. The study was conducted in two distinct phases: an initial unconstrained prompt elicitation phase (Figure 4) and a subsequent blind A/B model comparison phase (Figure 5).

Welcome to the User-Driven Audio Reproduction Survey

Thank you for your interest in our study. Your participation will help us understand how people communicate with future audio systems.

Instructions

- You will listen to a series of audio clips.
- For each clip, a text box will be available.
- In the text box, please type a command describing how you would want the audio to change. Think of it like talking to a smart speaker. There are no right or wrong answers — just type what feels natural to you!
- Optional: After submitting your command, you can listen to how the audio changes based on your request. If the change matches your expectation, you can click the 👍 button; if not, you can click the 👎 button.

Note: The audio changes may not always perfectly reflect your command.

Time Commitment

The entire experiment should take approximately 20-30 minutes to complete.

Data Collection & Privacy (Informed Consent)

We respect your privacy. We will *NOT* collect any personal information about you (like your name, email, or IP address). The data we will collect includes:

- The text commands (prompts) you write.
- The timestamp of when you submitted each command.
- A randomly generated user ID to help us analyze the data.
- Your feedback on whether the audio changes matched your expectations. (👍 or 👎)

This anonymous data will be used for research and development purposes. Your participation is completely voluntary, and you can close the page at any time to withdraw from the study.

By clicking 'I Agree and Begin', you confirm that you have read the information above and consent to participate in the study.

I Agree and Begin

(a) Introduction screen for the prompt elicitation phase, explaining the task to the user.

User-Driven Audio Reproduction

Play **Stop** **Next Audio**

Track Progress

1/30

Please enter a desired audio adjustment

Prompt

There are no wrong answers :)

Submit

Was the change in the audio close to what you would expect?

👍 **👎**

Quick guide

1. Use the **Next Audio** button to change clip. At least 1 prompt must be submitted per clip.
2. Click **Play** to start. Use **Stop** to halt. The change will be more audible while audio is playing.
3. Enter a prompt and press **Submit**.
4. Listen to the changes. If satisfied, press the 👍 button; if not, press the 👎 button.
5. You can submit multiple prompts per clip if desired. The response to each prompt will be logged.

(b) Main user interface for the prompt elicitation phase. Participants listened to an audio clip and could type any desired change into the text box.

Figure 4: Screenshots from Phase 1 (Prompt Elicitation) of the subjective evaluation study.

Welcome to the Audio Evaluation Survey

Thank you for your interest in our study. Your participation will help us evaluate the quality of different audio processing techniques.

Instructions

- You will be presented with a series of prompts and two audio clips.
- The first clip is the original audio.
- The second clip is an altered version of the original.
- Please listen to the clips and select one of the five options (from "Very Far" to "Very Close").
- The feedback you provide should reflect whether the altered audio meets your expectations based on the prompt.

Time Commitment

The entire experiment should take approximately 30 minutes to complete.

Data Collection & Privacy (Informed Consent)

We respect your privacy. The data we will collect includes:

- A randomly generated user ID to help us analyze the data.
- The timestamp of when you submitted your feedback.
- The audio clip and prompt you evaluated.
- Your feedback for the altered audio version.

This anonymous data will be used for research and development purposes. Your participation is completely voluntary, and you can close the page at any time to withdraw from the study.

By clicking 'I Agree and Begin', you confirm that you have read the information above and consent to participate in the study.

I Agree and Begin

(a) Introduction screen for the blind A/B comparison phase.

Audio Evaluation Survey

Progress
Prompt 1 of 60

Instructions: Listen to both audio clips below. The 'Original' clip is the reference, and the 'Altered' clip is the one to evaluate.

make it feel more lo-fi and echoey, with chill vibes

Original

0:00 0:10

Altered

0:00 0:10

How "well" has the altered audio achieved the prompt?

☐ Very Poorly ☐ Poorly ☐ Neutral ☐ Well ☐ Very Well

Previous **Next**

(b) Main user interface for the A/B comparison. For each of the previously elicited prompts, users listened to two versions ("Original" and "Altered") and rated them on a 5-point Likert scale.

Figure 5: Screenshots from Phase 2 (Blind A/B Comparison) of the subjective evaluation study.

C Ablation on KDE bandwidth selection

To validate our choice of the bandwidth scaling factor for the Reflective-KDE ($h = 0.25 \cdot h_{\text{Scott}}$) introduced in Section 3.4, we provide a visual ablation. Standard density estimators, such as Scott’s Rule, typically assume a unimodal Gaussian distribution. While effective for simple datasets, applying an unscaled Scott’s Rule to the highly multimodal and subjective preference data found in continuous audio control results in excessive smoothing. This smoothing destroys the local topological features representing specific user preferences.

Figure 6 demonstrates the effect of varying this scaling factor on the final non-parametric reward surface for a representative prompt. To contextualize the density estimation, the raw data heatmap (bottom-right) displays both the density of user evaluations (integers within cells) and the average preference score (cell color) across the discretized space. By scaling the bandwidth by 0.25, we successfully counteract over-smoothing, interpolating high-confidence preference clusters while avoiding overfitting to isolated, low-sample noise.

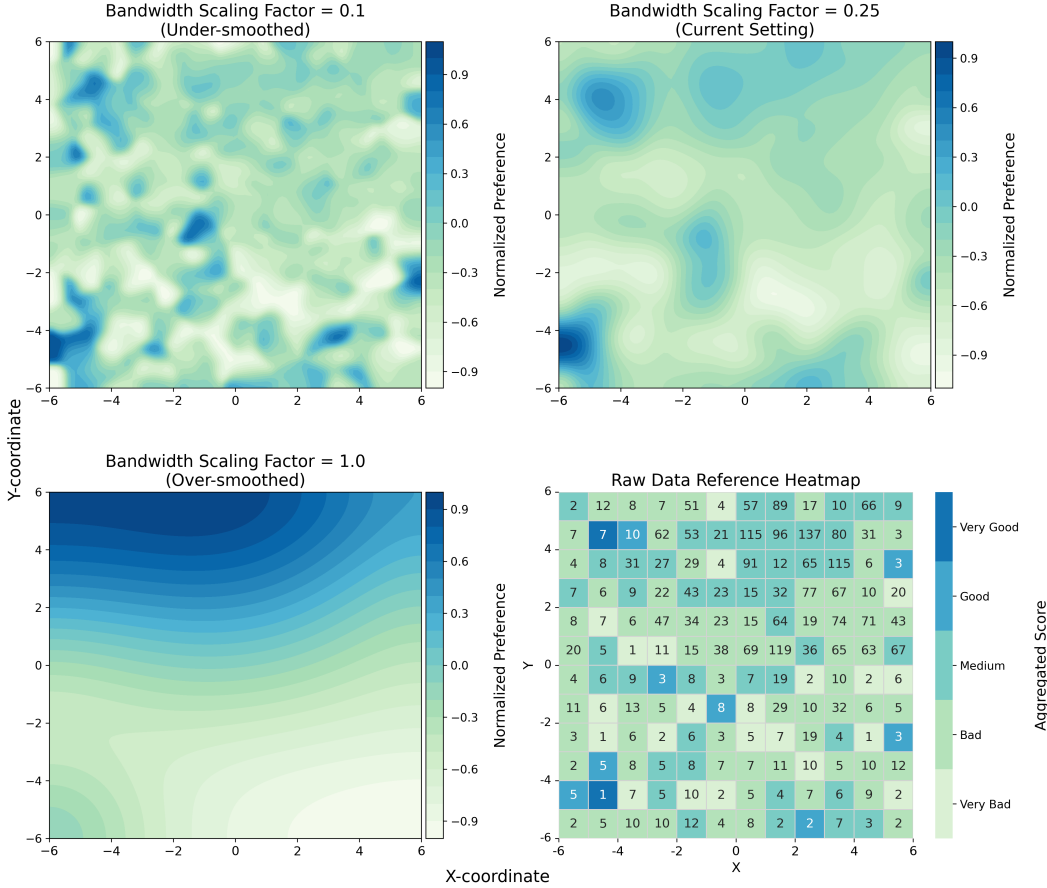


Figure 6: **Effect of KDE bandwidth scaling on the preference density.** The bottom-right panel displays a heatmap of the raw data for a representative prompt; cell colors indicate the aggregated preference score, while the integers inside the cells denote the number of user evaluations in that region. The remaining three panels show the resulting normalized reward surfaces using different bandwidth scaling factors. An under-smoothed factor (0.1, top-left) overfits to individual low-count data points, creating a noisy and fragmented reward signal. An over-smoothed factor (1.0, bottom-left) assumes unimodality, merging distinct preference modes into a single, uninformative gradient. Our chosen setting (0.25, top-right) provides an optimal balance, capturing the essential topological features and local maxima of the underlying preference manifold without overfitting to sparse noise.

D Experimental Details

To facilitate full reproducibility of our alignment pipeline, Table 7 provides a comprehensive overview of the exact hyperparameters, hardware configurations, and methodological settings used throughout the training and evaluation phases.

Table 7: Hyperparameters and Methodological Details for Reproducibility

Parameter	Value / Description
General & Infrastructure	
Base Models	Qwen/Qwen2.5-{0.5B, 1.5B}-Instruct
Precision	bfloat16 (Model weights and gradients)
Hardware	1 × NVIDIA L40 (48GB VRAM)
Distributed Training	None (Single GPU)
Optimizer	AdamW (adamw_torch)
LR Scheduler	Linear decay (0 warmup steps)
Training Duration (GRPO)	≈ 6 hours per model
Training Duration (DPO)	≈ 1 hour per model
Reward Surface & Reflective-KDE	
Global Normalization Range	$[-1.0, 1.0]$
Bandwidth Scaling (h)	$0.25 \cdot h_{Scott}$
Numerical Stability Epsilon (ϵ)	10^{-7}
Malformed Format Penalty	-1.0
Stage 1: GRPO Training	
Learning Rate	10^{-5}
Max Training Steps	1000
Per-Device Train Batch Size	4
Gradient Accumulation Steps	32
KL Penalty (β)	0.5 (Optimal), Swept $\{0.0, 0.1, 0.5, 1.0\}$
Number of Rollouts (G)	16 (Optimal), Swept $\{2, 4, 8, 16, 32\}$
Stage 2: Synthetic Data & DPO Refinement	
Injected Peaks (K)	10
Total DPO Pairs per Prompt	20 (10 Positive, 10 Negative)
Learning Rate	5×10^{-6}
Max Training Steps	1000
Per-Device Train Batch Size	1
Gradient Accumulation Steps	12
KL Penalty (β)	1.0 (Optimal), Swept $\{0.0, 0.1, 0.5, 1.0\}$
Evaluation & Inference	
Decoding Strategy	Greedy (do_sample=False, num_beams=1)
Max New Tokens	32

E Multi-seed experiments

As noted in Section 5.4, the performance of the hybrid pipeline fluctuates depending on the number of rollouts (G) used during the initial GRPO phase. To determine whether this stems from fundamental algorithmic instability or mere hyperparameter sensitivity, we evaluated our best configuration ($G = 16$) across four independent training seeds. The model achieved a relatively consistent mean GMRR of 0.58 ± 0.016 with a 100% parse rate.

Table 8: Multi-seed performance of the optimal Qwen2.5-1.5B Hybrid model ($G = 16$). The low standard deviation (± 0.016) demonstrates high training stability. Main model (GMRR = 0.60) not included.

Training Run	GMRR	Parse Rate
Seed 1	0.60	100%
Seed 2	0.58	100%
Seed 3	0.56	100%
Seed 4	0.58	100%
Mean ($\pm$ Std. Dev.)	0.58 ± 0.016	100%