ADAPTIVE REWARD ROUTING: DYNAMIC MULTI-REWARD OPTIMIZATION FOR JOINT AUDIO-VIDEO DIFFUSION VIA FORWARD-PROCESS RL

Anonymous authors
Paper under double-blind review

ABSTRACT

Multi-reward guided reinforcement learning (*i.e.*, RL) offers a promising way to improve joint audio-video diffusion models along several objectives, including modality-specific quality, cross-modal semantic alignment, and temporal synchronization. Its effectiveness, however, depends on two quantities that change during training: *where reward-driven updates should act, and how competing rewards should be coordinated*. Existing methods tend to rely on fixed routing and reward weights, failing to track evolving model functions. To address these limitations, we propose **Adaptive Reward Routing** to jointly adapt update locations and reward coordination during forward-process RL (*i.e.*, DiffusionNFT) of joint audio-video diffusion models. Our method consists of two components. (i) Cross-Modal Influence-Guided Routing (Localizing Updates): We use bidirectional cross-attention responses as an efficient proxy for evolving cross-modal influence, dynamically reweighting token-aware losses and scaling gradients across cross-modal layers without additional model interventions. (ii) Preference-Preserving Modality-Aware Reweighting (Coordinating Rewards): We preserve predefined weights as preference priors and use branch-specific reward-gradient interactions as residual corrections after warm-up. This resolves evolving conflicts without letting dominant rewards suppress weak but essential objectives. Extensive experiments demonstrate consistent improvements in modality quality, semantic consistency, and audio-video synchronization over strong RL baselines. Ablations and mechanism analyses further validate the complementary benefits of adaptive update routing and reward coordination.

1 INTRODUCTION

Recent advances in joint audio-video diffusion models (HaCohen et al., 2026) have enabled the generation of visual and audio content from text prompts. However, high-quality joint generation must simultaneously satisfy modality-specific visual and audio quality, cross-modal semantic alignment, and temporal synchronization, which are difficult to capture with a single supervised objective. Reward-guided diffusion reinforcement learning (RL), including GRPO-based methods (Guo et al., 2025; Liu et al., 2026a) and DiffusionNFT (Zheng et al., 2025), therefore provides a promising paradigm by expressing these requirements through multiple reward signals.

However, reward-guided RL of joint audio-video diffusion models remains challenging, as it involves dynamic multi-reward optimization along two coupled dimensions: *where should reward-driven updates act, and how should multiple rewards be coordinated?* **(i) Dynamic Reward Routing: Where to Optimize.** Joint audio-video models contain modality-specific branches coupled through cross-attention. Reward routers first determine the responsible modality branches, while the resulting branch-level updates must be localized across tokens and cross-modal interaction layers. OmniNFT (Zhang et al., 2026) recognizes these but fixes its layer routing based on the base model. However, our probing of the base and OmniNFT-trained checkpoints in Fig. 1(a) and (b) shows that cross-modal functions and gradient flows evolve during fine-tuning, making static routing progressively stale. **(ii) Dynamic Reward Coordination: How to Balance.** Rewards frequently disagree on the same sample, as shown in Fig. 1(c), and their appropriate balance changes throughout optimization. GDPO (Liu et al., 2026e) normalizes each reward but combines them through fixed

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

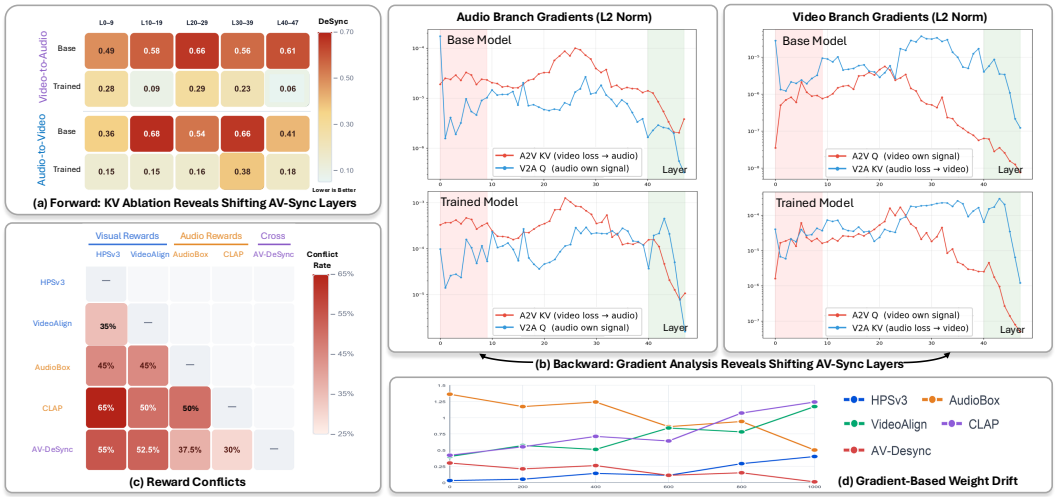

Figure 1: **Joint audio-video post-training requires adaptive update routing and reward coordination. (a) Forward KV Ablation.** We ablate cross-modal K/V outputs by layer range and measure DeSync (lower is better; larger degradation indicates greater synchronization importance). Divergent base/trained profiles reveal shifting synchronization-critical layers. **(b) Backward Gradient Analysis.** We measure layer-wise ℓ_2 norms of Q (within-branch) and K/V (cross-modal) gradients in both branches; shifted base/trained peaks show evolving cross-modal gradient paths. **(c) Reward Conflicts.** Rewards conflict when their per-prompt normalized preferences have opposite signs, with larger off-diagonal values indicating more disagreement. The conflicts show that fixed branch assignment is insufficient. **(d) Gradient-Based Weight Drift.** We track weights derived solely from reward-gradient geometry, where near-zero trajectories indicate objective suppression. The vanishing AV-DeSync weight shows that geometry-only weighting can discard essential objectives. Together, these results motivate dynamic routing and preference-anchored reward coordination. Detailed experimental procedures and analyses are provided in Appendix A.

weights, leaving conflicts unadapted. MARBLE (Zhao et al., 2026) adjusts weights using gradient geometry, but its coefficients reflect gradient compatibility rather than importance aligned with user preferences, as shown in Fig. 1(d). Effective post-training therefore requires conflict-aware adaptation anchored by user-defined priorities.

Together, these challenges call for an approach that adapts both reward coordination and update routing as the model evolves. We therefore propose **Adaptive Reward Routing** for forward-process RL (*i.e.*, DiffusionNFT (Zheng et al., 2025)) of joint audio-video diffusion models. It has two components. **(i) Cross-Modal Influence-Guided Routing** uses bidirectional cross-attention responses to locate reward-driven updates. Layer aggregation yields token weights that emphasize cross-modally influential locations, while token aggregation yields layer scales that preserve gradients through influential cross-modal pathways. **(ii) Preference-Preserving Modality-Aware Reweighting** estimates reward conflicts within the branch responsible for each objective. After warm-up, it uses the resulting coefficients as residual corrections to predefined reward weights, adapting to changing conflicts without overriding user priorities.

Our experiments establish three findings. First, Adaptive Reward Routing consistently improves modality quality, semantic alignment, and audio-video synchronization of joint audio-video diffusion models. Second, controlled ablations verify the complementary contributions of token- and layer-level routing, branch-aware conflict estimation, residual preference correction, and warm-up. Third, mechanism analyses with direct path interventions confirm that the cross-attention response proxy identifies functionally important layers and tokens, while routes frozen at initialization become stale as training progresses.

Contributions. (i) We formulate joint audio-video diffusion RL as dynamic multi-reward optimization over two coupled dimensions (*i.e.*, where modality-conditioned reward updates should act and how multiple rewards should be coordinated) and empirically reveal the limitations of static solutions. (ii) We propose Adaptive Reward Routing, which unifies cross-modal response-guided token/layer localization with preference-preserving, modality-aware reward coordination. (iii) We

provide comprehensive comparisons, ablations, and mechanism analyses demonstrating that adapting both dimensions enables more stable and effective joint audio-video post-training.

2 Related Work

Joint Audio-Video Generation. Video generation has progressed from image diffusion models with temporal modules (Blattmann et al., 2023; Guo et al., 2024) to large diffusion Transformers (Kong et al., 2024), with flow matching (Wan et al., 2025; HaCohen et al., 2024) and compressed latents improving efficiency. Joint audio-video systems connect pretrained experts through cross-modal projections (Wang et al., 2025), use unified diffusion Transformers (Liu et al., 2026c;d), or couple separate streams through bidirectional cross-attention (HaCohen et al., 2026). These heterogeneous branches enable mutual conditioning but make reward responsibility and gradient routing less obvious than in a single-stream model.

Reinforcement Learning for Diffusion Models. GRPO (Guo et al., 2025) estimates relative advantages without a critic. Flow-GRPO (Liu et al., 2026a) and DanceGRPO (Xue et al., 2025) extend online optimization to flow-based generation through stochastic sampling. DiffusionNFT (Zheng et al., 2025) instead optimizes the forward process using implicit positive and negative policies. OmniNFT (Zhang et al., 2026) adds modality-wise credit assignment for joint audio-video generation. We retain its forward-process formulation but replace fixed routing rules with token- and layer-level routes recomputed from the current model.

Multi-Reward Optimization. Fixed scalarization cannot react to changing conflicts. GDPO (Liu et al., 2026e) preserves reward-specific signals through decoupled normalization, but still uses predefined aggregation weights. Multi-task methods instead seek common descent directions (Désidéri, 2012; Sener & Koltun, 2018), project conflicting gradients (Yu et al., 2020), or optimize local agreement (Liu et al., 2021). MARBLE (Zhao et al., 2026) adapts this idea to diffusion RL. Because gradient compatibility alone does not encode objective importance or modality responsibility, we estimate conflicts within each modality branch and use them as residuals to preference priors.

3 Problem Formulation and Preliminaries

We study reward-guided post-training of joint audio-video diffusion models (*i.e.*, LTX-2 (HaCohen et al., 2026)) under DiffusionNFT (Zheng et al., 2025), which can be formulated as multi-modal, multi-reward forward-process reinforcement learning. We use $m \in \mathcal{M} = \{v, a\}$ for modality, n for rollout sample, k for reward, i for token, l for Transformer block, and t for flow-matching timestep.

Joint Audio-Video Flow Matching. LTX-2 (HaCohen et al., 2026) uses separate audio and video streams under a shared timestep. Each latent follows the standard linear interpolation $x_t^m = (1-t)x_0^m + tx_1^m$, with $x_1^m \sim \mathcal{N}(0, I)$, and the model predicts the two velocity fields jointly. The streams exchange information through bidirectional cross-attention:

$$o_{a\to v}^{l,t} = \mathrm{Attn}\Big(Q_v(h_v^{l,t}), K_a(h_a^{l,t}), V_a(h_a^{l,t})\Big), \quad o_{v\to a}^{l,t} = \mathrm{Attn}\Big(Q_a(h_a^{l,t}), K_v(h_v^{l,t}), V_v(h_v^{l,t})\Big). \tag{1}$$

The gated audio-to-video (A2V) and video-to-audio (V2A) outputs are added to the video and audio streams, respectively.

Diffusion Forward-Process Reinforcement Learning. DiffusionNFT (Zheng et al., 2025) constructs implicit positive and negative policies from the updated and trainable velocity predictors:

$$v_\theta^+ = (1-\beta)v^{\text{updated}} + \beta v_\theta, \qquad v_\theta^- = (1+\beta)v^{\text{updated}} - \beta v_\theta. \tag{2}$$

For each prompt, the updated policy generates a group of N samples. The reward of sample n is converted to a group-relative advantage,

$$A^{(n)} = \frac{R^{(n)} - \mu_R}{\sigma_R + \varepsilon}, \qquad r^{(n)} = \frac{1}{2} + \frac{1}{2}\,\mathrm{clip}\left(\frac{A^{(n)}}{A_{\max}}, -1, 1\right), \tag{3}$$

where μ_R and σ_R are computed within the rollout group. Thus $r^{(n)} > 1/2$ favors the positive policy, whereas $r^{(n)} < 1/2$ favors the negative policy. The resulting objective is

$$\mathcal{L}_{\text{NFT}} = \mathbb{E}_{n,t}\Big[r^{(n)}\|v_\theta^+(x_t^{(n)}, c, t) - u^{(n)}\|_2^2 + (1-r^{(n)})\|v_\theta^-(x_t^{(n)}, c, t) - u^{(n)}\|_2^2\Big]. \tag{4}$$

This advantage requires no learned value function: it states only whether a sample performs above or below its peers for the same prompt.

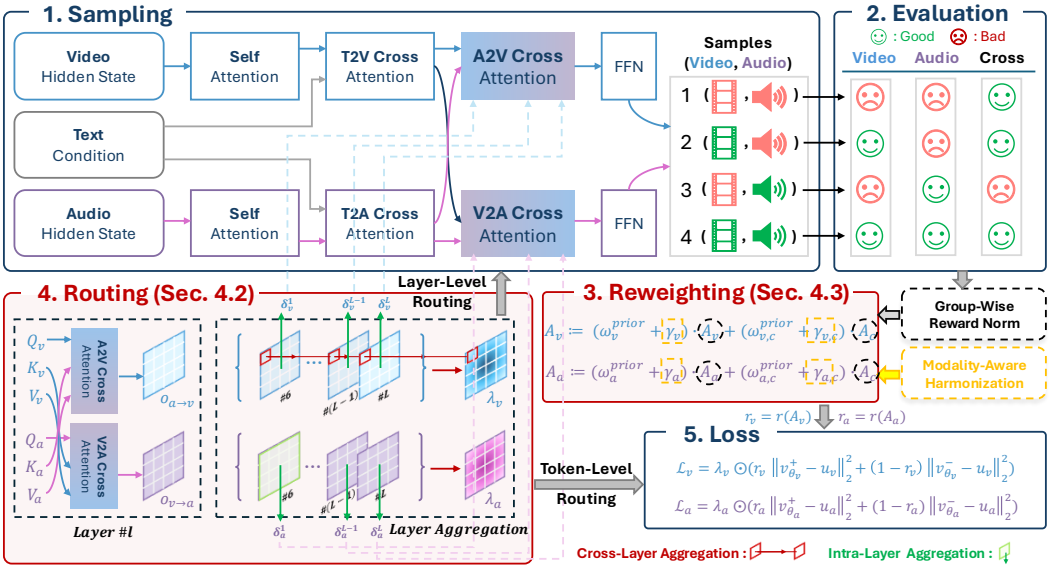

Figure 2: **Overview of Adaptive Reward Routing.** Our framework adapts reward-driven optimization at four levels: reward reweighting, modality-branch assignment, token-level credit allocation, and layer-wise cross-modal gradient routing.

Multi-Reward Optimization. Let $\mathcal{K} = \mathcal{K}_v \cup \mathcal{K}_a \cup \mathcal{K}_c$ denote video, audio, and cross-modal rewards. Eq. 3 is applied independently to each reward, producing $A_k^{(n)}$. GDPO (Liu et al., 2026e) combines them using predefined weights, $A_{\text{GDPO}}^{(n)} = \sum_k \omega_k^{\text{prior}} A_k^{(n)}$. MARBLE (Zhao et al., 2026) instead chooses simplex weights that minimize the norm of the weighted sum of normalized reward gradients. GDPO therefore preserves explicit preferences but cannot adapt to conflicts, while MARBLE adapts to local gradient geometry but does not encode preference or modality responsibility.

4 Method: Adaptive Reward Routing

4.1 Overview

We propose **Adaptive Reward Routing**, a forward-process RL framework that adapts both reward priorities and routing locations, as shown in Fig. 2. It contains two components. First, **Cross-Modal Influence-Guided Routing** (Sec. 4.2) determines where the update should act by adapting token weights and layer-wise cross-modal gradient flow. Second, **Preference-Preserving Modality-Aware Reweighting** (Sec. 4.3) determines how rewards should be combined within the video and audio branches. As shown in Algorithm 1, the complete optimization flow is

$$\{A_k\} \xrightarrow[\text{(Sec. 4.3)}]{\text{reward reweighting}} \{\omega_{m,k} A_k\} \xrightarrow[\text{(Sec. 4.3)}]{\text{branch routing}} (A_v, A_a) \xrightarrow[\text{(Sec. 4.2)}]{\text{token routing}} \mathcal{L} \xrightarrow[\text{(Sec. 4.2)}]{\text{layer routing}} \nabla_\theta \mathcal{L}. \tag{5}$$

Intuitively, reward reweighting decides how strongly each objective contributes, branch routing assigns objectives to target modalities, token routing selects where each modality loss is emphasized, and layer routing controls how the resulting gradient crosses modality boundaries.

4.2 Cross-Modal Influence-Guided Routing

A direct measure of directional influence would disable A2V or V2A and compare the velocity predictions. Repeating this intervention during training would require extra model evaluations. We instead use a quantity already produced by the forward pass: the pre-gate response of the corresponding cross-attention path. For target token i,

$$d_{v,i}^{l,t} = \|o_{a\to v,i}^{l,t}\|_2, \qquad d_{a,i}^{l,t} = \|o_{v\to a,i}^{l,t}\|_2. \tag{6}$$

These directional responses are collected over an intermediate-to-late denoising window and detached before policy optimization. Sec. 5.4 validates their relationship to direct interventions.

Token-Level Routing. For each target token, we average its responses over the selected timesteps and cross-modal blocks. After percentile-clipped min–max normalization (Norm_{99}), the score be-

Algorithm 1 Adaptive Reward Routing

Require: Policy v_θ, updated policy v^{updated}, and smoothed coefficients $\bar{\gamma}_m$
1: **for** each training round e **do**
2: Generate a group of samples with v^{updated} and collect pre-gate A2V/V2A responses during sampling
3: Evaluate $\{R_k^{(n)}\}$ and independently normalize each reward to obtain $\{A_k^{(n)}\}$
4: Compute token weights $\{\lambda_{m,i}^{(n)}\}$ using Eq. 7 and layer routes $\{\alpha_m^l\}$ using Eq. 8
5: Compute $\{\omega_{m,k}\}$ from the cached $\bar{\gamma}_m$ using Eq. 10
6: Construct $A_m^{(n)}$ using Eq. 11 and set $r_m^{(n)} = r(A_m^{(n)})$
7: **if** $e \geq e_{\text{warm}}$ and e is a refresh round **then**
8: Probe reward gradients through their responsible branches with uniform token weights
9: Solve the branch-wise MARBLE problem and update $\bar{\gamma}_m$ for the next round
10: **end if**
11: Compute $\ell_{m,i}^{(n)}$, $\mathcal{L}_m^{\text{policy}}$, and $\mathcal{L}(\theta)$ using Eqs. 12–14
12: Backpropagate through the routed KV paths in Eq. 9 and update θ
13: Update v^{updated} according to the DiffusionNFT policy-update schedule
14: **end for**

comes a positive loss weight:

$$\lambda_{m,i} = 1 + (\lambda_{\max} - 1)\,\text{Norm}_{99}\left(\frac{1}{|\mathcal{B}||\mathcal{T}|}\sum_{l\in\mathcal{B}}\sum_{t\in\mathcal{T}} d_{m,i}^{l,t}\right). \tag{7}$$

Audio responses are normalized globally, while video responses are normalized within each frame to prevent frame-level magnitude differences from dominating the weights. These weights are applied to the token-level negative-aware loss in Sec. 4.4. The exact normalization is given in Appendix B.

Layer-Level Routing. For each layer, we instead average the same response over tokens and selected timesteps. Let $\widetilde{\delta}_m^l$ denote this layer score after min–max normalization across blocks. We convert it to a soft detachment coefficient

$$\alpha_m^l = (1 - \widetilde{\delta}_m^l)^{1/\tau}. \tag{8}$$

For a source key or value tensor $X \in \{K, V\}$, the routed representation is

$$\widetilde{X}_{\tilde{m}\to m}^{l,t} = \alpha_m^l\,\text{sg}(X_{\tilde{m}}^{l,t}) + (1 - \alpha_m^l)X_{\tilde{m}}^{l,t}. \tag{9}$$

This operation leaves the forward value unchanged but scales its backward gradient by $1 - \alpha_m^l$. Strongly influential layers retain more gradient, while weakly coupled layers are increasingly detached. A2V and V2A are routed independently.

4.3 Preference-Preserving Modality-Aware Reweighting

Motivation. Predefined reward weights express what the user wants, but they cannot react to reward conflicts. Gradient-based coefficients react to conflicts, but may suppress a weak objective because its early gradient is noisy or incompatible. We combine the two rather than choosing one.

Implementation. Each reward is probed only through the branch it supervises: video and audio rewards use their respective branches, while cross-modal rewards use both. MARBLE then produces a conflict-aware coefficient $\gamma_{m,k}$ within each branch. Token routing is disabled during these probes so that the measured geometry is not biased by the current token weights. After warm-up, the smoothed coefficient provides a residual correction to the prior:

$$\omega_{m,k} = \begin{cases} \omega_{m,k}^{\text{prior}}, & e < e_{\text{warm}}, \\ (1-\kappa)\omega_{m,k}^{\text{prior}} + \kappa C_m \bar{\gamma}_{m,k}, & e \geq e_{\text{warm}}. \end{cases} \tag{10}$$

Here C_m rescales the simplex coefficients to preserve the total prior weight within branch m, and $\bar{\gamma}_m \leftarrow \rho\bar{\gamma}_m + (1-\rho)\gamma_m^*$ smooths successive estimates. The prior therefore sets a nonzero floor, while the residual term adapts to current conflicts.

4.4 Training Objective

The adaptive reward weights first produce a separate advantage for each modality branch:

$$A_m^{(n)} = \sum_{k\in\mathcal{K}_m\cup\mathcal{K}_c} \omega_{m,k} A_k^{(n)}, \qquad m \in \{v, a\}. \tag{11}$$

Cross-modal rewards are included in both branches. We then map each branch advantage to an optimality probability $r_m^{(n)} = r(A_m^{(n)})$ using Eq. 3. For token i of sample n, the negative-aware loss is

$$\ell_{m,i}^{(n)} = r_m^{(n)} \frac{\|v_{\theta,m,i}^{+} - u_{m,i}\|_2^2}{w_m^{+,(n)} + \varepsilon} + \left(1 - r_m^{(n)}\right) \frac{\|v_{\theta,m,i}^{-} - u_{m,i}\|_2^2}{w_m^{-,(n)} + \varepsilon}. \tag{12}$$

Here $w_m^{\pm,(n)}$ is the detached mean absolute residual of the corresponding policy, averaged over all tokens and feature dimensions of modality m. The token routing weights then form the modality loss

$$\mathcal{L}_m^{\text{policy}} = \mathbb{E}_n \left[\frac{\sum_{i \in \mathcal{I}_m} \lambda_{m,i}^{(n)} \ell_{m,i}^{(n)}}{\sum_{i \in \mathcal{I}_m} \lambda_{m,i}^{(n)}} \right]. \tag{13}$$

Finally, we combine the two branches and regularize them toward the fixed reference policy:

$$\mathcal{L}(\theta) = \sum_{m \in \mathcal{M}} \mathcal{L}_m^{\text{policy}} + \lambda_{\text{KL}} \sum_{m \in \mathcal{M}} \mathcal{L}_{\text{KL},m}(\theta). \tag{14}$$

5 Experiments

5.1 Experimental Setup

Backbones and Training Data. We evaluate Adaptive Reward Routing on two joint audio-video diffusion backbones, LTX-2 (19B) and LTX-2.3 (22B) (HaCohen et al., 2026). Both models employ separate audio and video streams connected through bidirectional cross-attention, making them suitable for studying adaptive reward localization across modalities, tokens, and layers. For reward-guided post-training, we use 19,487 audio-video prompts collected from a VGGSound-derived (Chen et al., 2020) corpus. Each record contains modality-specific audio and video descriptions together with a joint audio-video prompt.

Reward Models. Following the multi-objective evaluation dimensions of joint audio-video generation, we optimize five complementary reward signals: (i) Video Quality: VideoAlign (Liu et al., 2026b) and HPSv3 (Ma et al., 2025); (ii) Audio Quality: AudioBox Aesthetics (Tjandra et al., 2025); (iii) Text-Audio Alignment: CLAP (Wu et al., 2023); and (iv) Audio-Video Synchronization: DeSync (Iashin et al., 2024), which we convert into a higher-is-better AV-DeSync during training, while reporting the original lower-is-better DeSync metric at evaluation time.

Baselines. We compare against complementary reference settings that cover the principal dimensions of multimodal, multi-reward optimization. (i) No Post-Training: the pretrained backbone establishes the performance before reward-guided optimization. (ii) Fixed Reward Coordination: GDPO (Liu et al., 2026e) independently normalizes reward-wise advantages but aggregates them using fixed weights. (iii) Conflict-Aware Reward Coordination: MARBLE (Zhao et al., 2026) dynamically adjusts reward weights according to global gradient conflicts, without modality-specific routing. (iv) Static Multimodal Routing: OmniNFT (Zhang et al., 2026), designed specifically for DiffusionNFT-based joint audio-video post-training, introduces modality-wise credit assignment, layer-wise gradient surgery, and region-wise reweighting, but fixes its routing strategy using the base model. OmniNFT* denotes the checkpoint released by the original authors.

Evaluation. We evaluate on the full JavisBench benchmark (Liu et al., 2026d), which contains 10,140 prompts spanning diverse audio-video generation scenarios. Generated outputs are normalized to the benchmark protocol of four seconds, 24 FPS, and 16-kHz audio. We report four groups of metrics: (i) AV-Quality (Liu et al., 2026b), including Visual Quality (VQ) and Audio Quality (AQ); (ii) Text-Consistency, including Text-Video and Text-Audio ImageBind similarity (TV-IB and TA-IB) (Girdhar et al., 2023), CLIP (Radford et al., 2021), and CLAP; (iii) AV-Consistency (Girdhar et al., 2023), including AV-IB and AVHScore; and (iv) AV-Synchrony, including JavisScore (Liu et al., 2026c) and DeSync.

5.2 Main Results and Training Dynamics

Fig. 3, Tab. 1, and Fig. 4 summarize the generation quality, benchmark performance, and optimization behavior of our method, respectively. **(i) Qualitative Results.** Fig. 3 covers diverse audio-video scenarios, including multilingual speech, a stylized speaking character, a two-speaker exchange, and

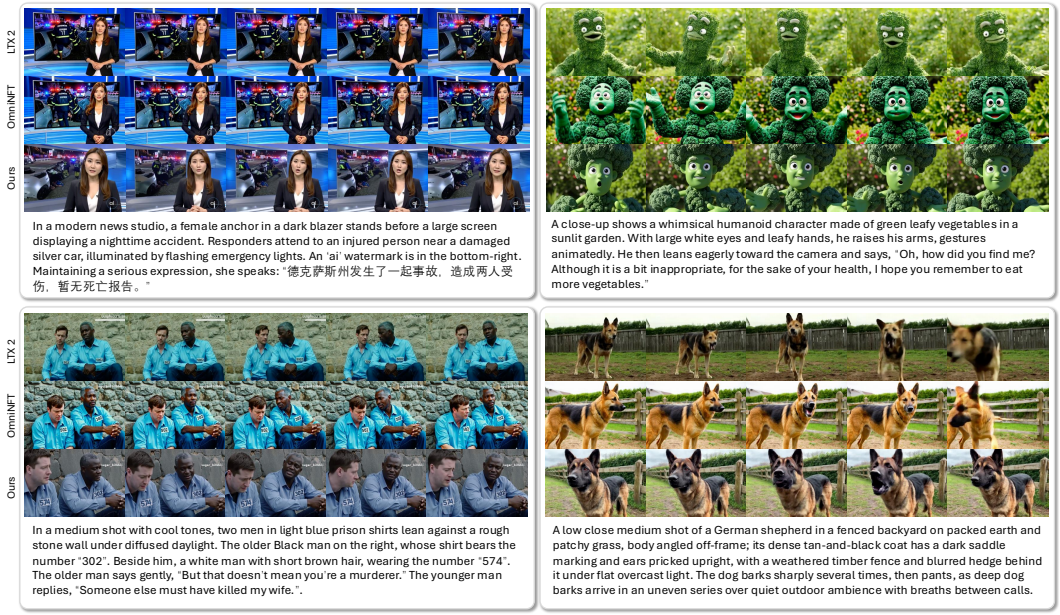

Figure 3: **Qualitative comparison of joint audio-video generation.** For each prompt, we show five temporally ordered frames generated by LTX-2, OmniNFT, and Ours. The examples cover Chinese news delivery, stylized English speech, two-speaker dialogue, and dog barking. Our method better preserves the requested subjects and scene composition while reducing identity and appearance drift throughout the generated sequence. More video results with audio are provided in the HTML of supplementary material.

Table 1: **Main results on JavisBench.** (Mean ± std over 3 seeds.)

Backbone	Method	AV-Quality		Text-Consistency				AV-Consistency		AV-Synchrony	
		VQ↑	AQ↑	TV-IB↑	TA-IB↑	CLIP↑	CLAP↑	AV-IB↑	AVHScore↑	JavisScore↑	DeSync↓
LTX-2	Base Model	1.883	5.201	0.265	0.143	0.312	0.358	0.180	0.177	0.153	0.604
	+ GDPO	2.722 ±0.0136	5.450 ±0.0217	0.261 ±0.0020	0.138 ±0.0018	0.312 ±0.0014	0.347 ±0.0031	0.174 ±0.0025	0.175 ±0.0022	0.155 ±0.0020	0.671 ±0.0104
	+ MARBLE	2.384 ±0.0124	5.100 ±0.0193	0.265 ±0.0016	0.138 ±0.0019	0.311 ±0.0013	0.365 ±0.0028	0.182 ±0.0023	0.182 ±0.0026	0.158 ±0.0021	0.618 ±0.0092
	+ OmniNFT	3.136 ±0.0108	5.614 ±0.0176	0.265 ±0.0018	0.145 ±0.0016	0.312 ±0.0011	0.416 ±0.0024	0.222 ±0.0019	0.219 ±0.0021	0.195 ±0.0017	0.390 ±0.0075
	+ OmniNFT*	3.278	5.609	0.254	0.165	**0.314**	0.422	0.220	0.219	0.193	0.378
	+ Ours	**3.336** ±0.0094	**5.868** ±0.0158	**0.268** ±0.0014	**0.167** ±0.0013	0.314 ±0.0011	**0.425** ±0.0020	**0.235** ±0.0017	**0.234** ±0.0019	**0.206** ±0.0015	**0.341** ±0.0062
LTX-2.3	Base Model	2.032	5.218	0.271	0.151	0.308	0.387	0.205	0.202	0.175	0.504
	+ GDPO	2.929 ±0.0142	5.251 ±0.0225	0.272 ±0.0021	0.144 ±0.0020	0.309 ±0.0015	0.376 ±0.0033	0.218 ±0.0027	0.199 ±0.0025	0.178 ±0.0022	0.560 ±0.0113
	+ MARBLE	2.582 ±0.0130	5.176 ±0.0206	0.271 ±0.0018	0.147 ±0.0017	0.309 ±0.0014	0.394 ±0.0029	0.217 ±0.0022	0.207 ±0.0028	0.182 ±0.0020	0.496 ±0.0098
	+ OmniNFT	3.489 ±0.0115	5.693 ±0.0181	0.271 ±0.0016	0.163 ±0.0015	**0.316** ±0.0010	0.449 ±0.0025	0.250 ±0.0021	0.238 ±0.0023	0.224 ±0.0018	0.369 ±0.0079
	+ OmniNFT*	3.537	5.702	0.260	0.174	0.311	0.456	0.252	0.249	0.221	0.335
	+ Ours	**3.599** ±0.0089	**5.979** ±0.0151	**0.274** ±0.0013	**0.177** ±0.0012	0.314 ±0.0011	**0.460** ±0.0019	**0.267** ±0.0016	**0.266** ±0.0017	**0.236** ±0.0014	**0.302** ±0.0058

animal vocalization. LTX-2 exhibits noticeable subject and appearance drift, particularly in the character and animal examples, while OmniNFT improves prompt fidelity but retains temporal inconsistencies. Our method maintains more stable identities and scene structures while preserving the visual actions associated with speech, dialogue, and barking. **(ii) Quantitative Results.** As shown in Tab. 1, our method achieves the strongest overall performance on both LTX-2 and LTX-2.3, obtaining the best result on nine of the ten metrics under each backbone. For each backbone, GDPO, MARBLE, OmniNFT, and Ours are independently trained with three random seeds under the same data, LoRA, and optimization budgets, and the table reports their arithmetic means. The Base Model and OmniNFT* are fixed checkpoints evaluated under the same generation and evaluation protocol, with OmniNFT* denoting the checkpoint released by its original authors. GDPO improves visual quality but degrades several audio and synchronization metrics, revealing the imbalance caused by fixed reward aggregation. MARBLE alleviates reward conflicts globally, and OmniNFT introduces modality-aware optimization, but neither adapts both reward coordination and update routing to the evolving model. Our method consistently improves modality quality, semantic consistency, cross-modal consistency, and synchronization, with the same trend across both backbones. Full training and evaluation details are provided in Appendix C. **(iii) Training Dynamics.** Fig. 4(a) shows that our method reaches the highest average reward while maintaining favorable trajectories across all five component rewards. In contrast, the baselines make less balanced progress across audio, video, and synchronization objectives. This indicates that the final gains arise from coordinated multi-reward optimization rather than improving one objective at the expense of others.

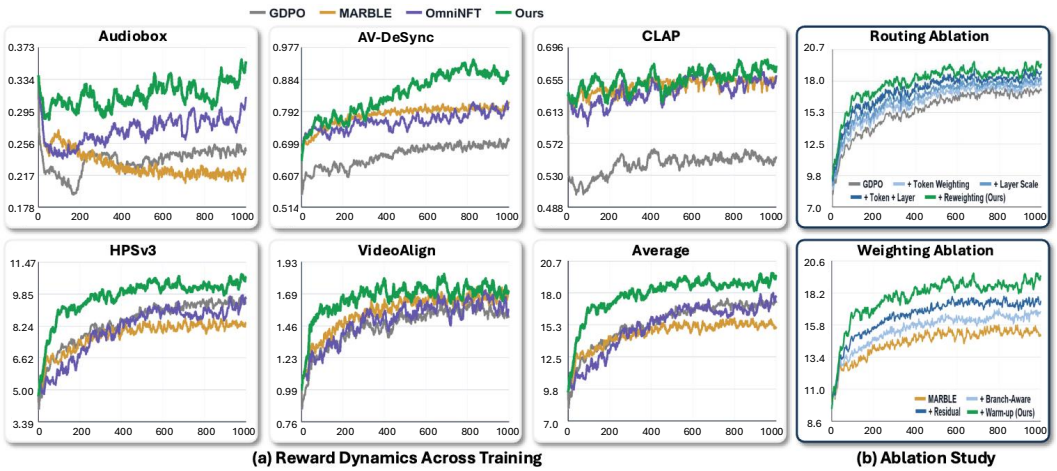

Figure 4: **Reward dynamics and component ablations.** (a) Training trajectories of five individual rewards (AudioBox, AV-DeSync, CLAP, HPSv3, and VideoAlign) and their average for GDPO, MARBLE, OmniNFT, and Ours. Average denotes the arithmetic mean of the five normalized rewards. (b) Ablation studies of the routing and weighting designs. Starting from GDPO or MARBLE, respectively, each added component yields progressive gains, while the complete method achieves the strongest overall performance.

Table 2: **Results of the ablation studies on JavisBench using the LTX-2 backbone.** Gray-shaded rows isolate the reward-coordination components from the MARBLE baseline and do not inherit the routing stack. (Mean ± std over 3 seeds.)

Study	Configuration	AV-Quality		Text-Consistency				AV-Consistency		AV-Synchrony	
		VQ↑	AQ↑	TV-IB↑	TA-IB↑	CLIP↑	CLAP↑	AV-IB↑	AVHScore↑	JavisScore↑	DeSync↓
Routing	+ Token Weighting	3.008 ±0.0127	5.663 ±0.0202	0.263 ±0.0018	0.156 ±0.0016	0.313 ±0.0013	0.388 ±0.0027	0.210 ±0.0022	0.201 ±0.0024	0.179 ±0.0019	0.482 ±0.0088
	+ Layer Scale	3.192 ±0.0111	5.784 ±0.0188	0.264 ±0.0017	0.161 ±0.0015	0.313 ±0.0012	0.409 ±0.0025	0.221 ±0.0020	0.227 ±0.0022	0.189 ±0.0018	0.366 ±0.0072
	+ Token + Layer	3.315 ±0.0103	5.839 ±0.0171	0.266 ±0.0015	0.162 ±0.0014	0.313 ±0.0011	0.411 ±0.0023	0.226 ±0.0018	0.229 ±0.0020	0.190 ±0.0016	0.343 ±0.0065
Weighting	+ Branch-Aware	2.612 ±0.0131	5.290 ±0.0211	0.264 ±0.0019	0.149 ±0.0017	0.311 ±0.0014	0.384 ±0.0030	0.198 ±0.0024	0.188 ±0.0027	0.177 ±0.0021	0.492 ±0.0096
	+ Residual	2.891 ±0.0118	5.526 ±0.0196	0.264 ±0.0016	0.158 ±0.0018	0.312 ±0.0013	0.393 ±0.0026	0.199 ±0.0021	0.199 ±0.0023	0.180 ±0.0019	0.455 ±0.0084
	+ Warm-Up	2.999 ±0.0106	5.791 ±0.0179	0.265 ±0.0017	0.157 ±0.0014	0.312 ±0.0012	0.404 ±0.0024	0.209 ±0.0019	0.201 ±0.0021	0.183 ±0.0017	0.369 ±0.0070
Routing + Weighting (Ours)		**3.336** ±0.0094	**5.868** ±0.0158	**0.268** ±0.0014	**0.167** ±0.0013	**0.314** ±0.0011	**0.425** ±0.0020	**0.235** ±0.0017	**0.234** ±0.0019	**0.206** ±0.0015	**0.341** ±0.0062

5.3 Ablation Studies

(i) Cross-Modal Influence-Guided Routing. The routing ablation in Tab. 2 progressively introduces token weighting and layer scaling over GDPO. Token weighting improves local credit assignment by emphasizing tokens with stronger cross-modal responses, while layer scaling further improves consistency and synchronization by preserving gradients through influential interaction layers. Combining the two produces the strongest routing-only configuration, confirming that token- and layer-level adaptation are complementary. **(ii) Preference-Preserving Reward Coordination.** The gray rows isolate the weighting components from the MARBLE baseline without inheriting the routing stack. Branch-aware balancing assigns reward interactions to their responsible modality branches, while residual mixing preserves the predefined preference prior instead of replacing it with gradient-derived coefficients. Warm-up further stabilizes this adaptation by delaying dynamic reweighting until the estimated gradient relationships become reliable. **(iii) Complementarity and Reward Trade-Offs.** Individual components may favor different objectives, so intermediate configurations do not necessarily improve every metric monotonically. Nevertheless, progressively incorporating routing and weighting produces a stronger overall balance across quality, semantic consistency, and synchronization, as also reflected in Fig. 4(b). Because the routing chain keeps the GDPO weighting fixed and the weighting chain inherits no routing, each chain isolates a single axis. The complete model performs best overall, demonstrating that adaptive update localization and preference-preserving reward coordination address distinct but complementary failure modes.

5.4 Validating the Cross-Modal Influence Proxy

We verify whether the response proxy identifies the cross-modal paths that actually affect the model output. At four training checkpoints, we disable each of the 48 A2V or V2A blocks separately while fixing the prompt, noisy latent, and timestep; a larger change in the final prediction indicates a more influential path. **(i) Layer-Level Fidelity:** The proxy closely recovers the intervention-based layer ranking, with Spearman correlations of 0.98 for A2V and 0.97 for V2A. Since a value close to 1

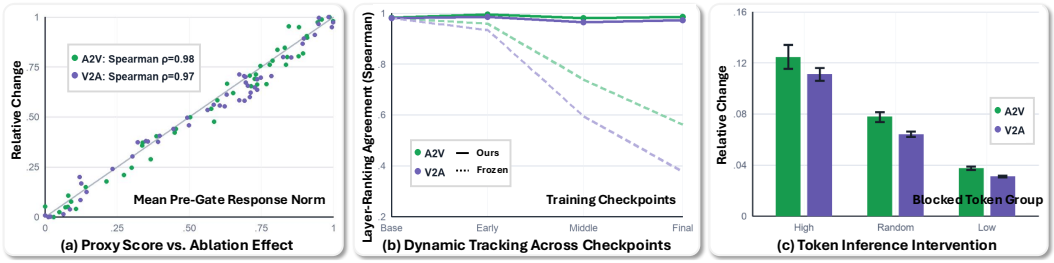

Figure 5: **The proxy correctly identifies influential cross-modal layers and tokens.** (a) Each point is one layer. The x-axis is the layer's mean pre-gate A2V/V2A response norm, while the y-axis is the relative change in the corresponding final video/audio velocity after disabling that layer. Both axes are min–max normalized across the 48 layers for visualization. Points near the diagonal indicate agreement between the proxy and direct intervention. (b) Higher correlation means better agreement with the current model. The current proxy stays accurate, whereas the proxy fixed at initialization becomes inaccurate as training changes the model. (c) We block the cross-modal information received by the top-scoring, random, or bottom-scoring 10% of target tokens. A larger final prediction change means that the blocked tokens were more influential. The top-scoring tokens consistently cause the largest change, confirming that the proxy correctly identifies important tokens.

means nearly identical rankings, the proxy reliably identifies which layers matter (Fig. 5(a)). **(ii) Dynamic Tracking:** The proxy recomputed from the current model remains above 0.96 throughout fine-tuning, whereas the proxy frozen at initialization falls to 0.56 and 0.38. Thus, influential layers shift during training, and a fixed routing map becomes stale (Fig. 5(b)). **(iii) High-Score Tokens Matter More:** We block the cross-modal responses of the top-scoring, random, or bottom-scoring 10% of target tokens. Blocking the top-scoring group changes the final prediction $1.62\times$ more for A2V and $1.74\times$ more for V2A than blocking an equally sized random group. Since the ablation size is identical, this result directly shows that higher proxy scores identify tokens with greater functional influence. Full details are provided in Appendix D.

6 Conclusion

This work reframes multi-reward post-training for joint audio-video generation as a dynamic credit-assignment problem. As optimization reshapes the model's cross-modal functions, both the token/layer localization of modality-conditioned updates and the relative strengths of reward signals should evolve accordingly. Adaptive Reward Routing embodies this principle by tracking changing cross-modal influence and reward conflicts throughout training, while preserving user-defined preferences. Its consistent gains across backbones, metrics, and ablations demonstrate the importance of adapting the optimization process in step with the model itself. More broadly, our findings point toward multimodal learning systems in which feedback is not routed by a fixed recipe, but continually reorganized as the model acquires new capabilities.

Limitations and Future Work. **(i) Reward Models:** No established unified reward jointly captures modality quality, semantic consistency, and temporal synchronization. Human-preference models (Huang et al., 2026) provide a complementary overall signal but do not replace fine-grained, modality-specific supervision. Developing a comprehensive audio-video reward remains an important direction. **(ii) RL Framework:** We adopt DiffusionNFT for stable, direct supervision of sampled flow-matching timesteps without reverse-process likelihood estimation. Our routing can extend to other diffusion objectives through their token losses and gradient paths, while reward coordination requires only reward-wise gradients. **(iii) Architectural Scope:** We validate our method on both dual-stream LTX backbones and the unified single-stream JavisDiT++ backbone (Appendix J). Routing applies when modality tokens are identifiable and directional interaction responses can be isolated, whereas reward coordination is architecture-independent. Models with inseparable modality representations or inaccessible interaction responses remain future work.

Supplementary Material. The appendix details the preliminary studies in the teaser figure, and further provides implementation details, human evaluation, computational-efficiency analysis, failure-case discussions, and experiments on another single-stream audio-video diffusion backbone. A separate HTML project page included in the supplementary material organizes all qualitative comparisons and failure cases for interactive viewing with audio.

AI USE STATEMENT

Large language models were not used to generate research ideas, design methods, produce or analyze experimental results, or write code. All technical content and reported results were produced and verified by the authors, who take full responsibility for the paper.

ETHICS STATEMENT

We have read and adhere to the ICLR Code of Ethics. The human evaluation in Appendix F involved adult participants who gave informed consent, were compensated at a locally appropriate rate, and provided only anonymous preference judgments; no personal or identifying data were collected. Our training prompts derive from a publicly available VGGSound-based corpus, and all backbones and reward models are used in accordance with their licenses. As with any generative audio-video model, reward-guided post-training can amplify biases in the pretrained prior and could be misused to produce misleading synthetic media. We therefore release no new generative weights or datasets, and Appendix I documents failure modes to discourage over-reliance on such outputs.

REPRODUCIBILITY STATEMENT

Reproducibility details are provided in the appendix rather than repeated here. Appendix A details the motivation analysis; Appendix B gives the exact objective and normalization used by our routing and reweighting components; Appendix C specifies backbones, training data, reward models, hyperparameters, hardware, and the multi-seed evaluation protocol; Appendix D describes the intervention, including checkpoints, samples, and aggregation; Appendix F and Appendix G report the human-study design and computational cost. The supplementary material additionally includes an HTML page organizing all qualitative audio-video comparisons.

REFERENCES

Andreas Blattmann, Tim Dockhorn, Sumith Kulal, Daniel Mendelevitch, Maciej Kilian, Dominik Lorenz, Yam Levi, Zion English, Vikram Voleti, Adam Letts, et al. Stable video diffusion: Scaling latent video diffusion models to large datasets. *arXiv preprint arXiv:2311.15127*, 2023.

Honglie Chen, Weidi Xie, Andrea Vedaldi, and Andrew Zisserman. Vggsound: A large-scale audio-visual dataset. In *ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 721–725. IEEE, 2020.

Jean-Antoine Désidéri. Multiple-gradient descent algorithm (mgda) for multiobjective optimization. *Comptes Rendus. Mathématique*, 350(5-6):313–318, 2012.

Rohit Girdhar, Alaaeldin El-Nouby, Zhuang Liu, Mannat Singh, Kalyan Vasudev Alwala, Armand Joulin, and Ishan Misra. Imagebind one embedding space to bind them all. In *2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 15180–15190. IEEE, 2023.

Daya Guo, Dejian Yang, Haowei Zhang, Junxiao Song, Peiyi Wang, Qihao Zhu, Runxin Xu, Ruoyu Zhang, Shirong Ma, Xiao Bi, et al. Deepseek-r1: Incentivizing reasoning capability in llms via reinforcement learning. *arXiv preprint arXiv:2501.12948*, 2025.

Yuwei Guo, Ceyuan Yang, Anyi Rao, Zhengyang Liang, Yaohui Wang, Yu Qiao, Maneesh Agrawala, Dahua Lin, and Bo Dai. Animatediff: Animate your personalized text-to-image diffusion models without specific tuning. *International Conference on Learning Representations*, 2024.

Yoav HaCohen, Nisan Chiprut, Benny Brazowski, Daniel Shalem, Dudu Moshe, Eitan Richardson, Eran Levin, Guy Shiran, Nir Zabari, Ori Gordon, et al. Ltx-video: Realtime video latent diffusion. *arXiv preprint arXiv:2501.00103*, 2024.

Yoav HaCohen, Benny Brazowski, Nisan Chiprut, Yaki Bitterman, Andrew Kvochko, Avishai Berkowitz, Daniel Shalem, Daphna Lifschitz, Dudu Moshe, Eitan Porat, et al. Ltx-2: Efficient joint audio-visual foundation model. *arXiv preprint arXiv:2601.03233*, 2026.

Yinming Huang, Shuyuan Tu, Xi Yan, Zihan Yang, Jianhua Han, Xu Hang, Yu-Gang Jiang, and Zuxuan Wu. Va-judger: Reward modeling from human preference feedback for joint video-audio generation, 2026. URL https://arxiv.org/abs/2608.18607.

Vladimir Iashin, Weidi Xie, Esa Rahtu, and Andrew Zisserman. Synchformer: Efficient synchronization from sparse cues. In *ICASSP 2024-2024 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 5325–5329. IEEE, 2024.

Weijie Kong, Qi Tian, Zijian Zhang, Rox Min, Zuozhuo Dai, Jin Zhou, Jiangfeng Xiong, Xin Li, Bo Wu, Jianwei Zhang, et al. Hunyuanvideo: A systematic framework for large video generative models. *arXiv preprint arXiv:2412.03603*, 2024.

Bo Liu, Xingchao Liu, Xiaojie Jin, Peter Stone, and Qiang Liu. Conflict-averse gradient descent for multi-task learning. *Advances in neural information processing systems*, 34:18878–18890, 2021.

Jie Liu, Gongye Liu, Jiajun Liang, Yangguang Li, Jiaheng Liu, Xintao Wang, Pengfei Wan, Di Zhang, and Wanli Ouyang. Flow-grpo: Training flow matching models via online rl. *Advances in neural information processing systems*, 38:40783–40818, 2026a.

Jie Liu, Gongye Liu, Jiajun Liang, Ziyang Yuan, Xiaokun Liu, Mingwu Zheng, Xiele Wu, Qiulin Wang, Menghan Xia, Xintao Wang, et al. Improving video generation with human feedback. *Advances in Neural Information Processing Systems*, 38:82155–82192, 2026b.

Kai Liu, Wei Li, Lai Chen, Shengqiong Wu, Yanhao Zheng, Jiayi Ji, Fan Zhou, Jiebo Luo, Ziwei Liu, Hao Scofield Fei, et al. Javisdit: Joint audio-video diffusion transformer with hierarchical spatio-temporal prior synchronization. In *International Conference on Learning Representations*, volume 2026, pp. 139160–139194, 2026c.

Kai Liu, Yanhao Zheng, Kai Wang, Shengqiong Wu, Rongjunchen Zhang, Jiebo Luo, Dimitrios Hatzinakos, Ziwei Liu, Hao Scofield Fei, and Tat-Seng Chua. Javisdit++: Unified modeling and optimization for joint audio-video generation. In *International Conference on Learning Representations*, volume 2026, pp. 150592–150618, 2026d.

Shih-Yang Liu, Xin Dong, Ximing Lu, Shizhe Diao, Peter Belcak, Mingjie Liu, Min-Hung Chen, Hongxu Yin, Yu-Chiang Frank Wang, Kwang-Ting Cheng, Yejin Choi, Jan Kautz, and Pavlo Molchanov. GDPO: Group reward-decoupled normalization policy optimization for multi-reward RL optimization. In *Forty-third International Conference on Machine Learning*, 2026e.

Yuhang Ma, Xiaoshi Wu, Keqiang Sun, and Hongsheng Li. Hpsv3: Towards wide-spectrum human preference score. In *2025 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 15086–15095. IEEE, 2025.

Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *International conference on machine learning*, pp. 8748–8763. PmLR, 2021.

Ozan Sener and Vladlen Koltun. Multi-task learning as multi-objective optimization. *Advances in Neural Information Processing Systems*, 31, 2018.

Andros Tjandra, Yi-Chiao Wu, Baishan Guo, John Hoffman, Brian Ellis, Apoorv Vyas, Bowen Shi, Sanyuan Chen, Matt Le, Nick Zacharov, et al. Meta audiobox aesthetics: Unified automatic assessment for speech, music and sound. In *2025 IEEE Automatic Speech Recognition and Understanding Workshop (ASRU)*, pp. 1–8. IEEE, 2025.

Team Wan, Ang Wang, Baole Ai, Bin Wen, Chaojie Mao, Chen-Wei Xie, Di Chen, Feiwu Yu, Haiming Zhao, Jianxiao Yang, et al. Wan: Open and advanced large-scale video generative models. *arXiv preprint arXiv:2503.20314*, 2025.

Duomin Wang, Wei Zuo, Aojie Li, Ling-Hao Chen, Xinyao Liao, Deyu Zhou, Zixin Yin, Xili Dai, Daxin Jiang, and Gang Yu. Universe-1: Unified audio-video generation via stitching of experts. *arXiv preprint arXiv:2509.06155*, 2025.

Yusong Wu, Ke Chen, Tianyu Zhang, Yuchen Hui, Taylor Berg-Kirkpatrick, and Shlomo Dubnov. Large-scale contrastive language-audio pretraining with feature fusion and keyword-to-caption augmentation. In *ICASSP 2023-2023 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 1–5. IEEE, 2023.

Zeyue Xue, Jie Wu, Yu Gao, Fangyuan Kong, Lingting Zhu, Mengzhao Chen, Zhiheng Liu, Wei Liu, Qiushan Guo, Weilin Huang, et al. Dancegrpo: Unleashing grpo on visual generation. *arXiv preprint arXiv:2505.07818*, 2025.

Tianhe Yu, Saurabh Kumar, Abhishek Gupta, Sergey Levine, Karol Hausman, and Chelsea Finn. Gradient surgery for multi-task learning. *Advances in neural information processing systems*, 33: 5824–5836, 2020.

Guohui Zhang, XiaoXiao Ma, Jie Huang, Hang Xu, Hu Yu, Siming Fu, Yuming Li, Zeyue Xue, Lin Song, Haoyang Huang, et al. Omninft: Modality-wise omni diffusion reinforcement for joint audio-video generation. *arXiv preprint arXiv:2605.12480*, 2026.

Canyu Zhao, Hao Chen, Yunze Tong, Yu Qiao, Jiacheng Li, and Chunhua Shen. Marble: Multi-aspect reward balance for diffusion rl. *arXiv preprint arXiv:2605.06507*, 2026.

Kaiwen Zheng, Huayu Chen, Haotian Ye, Haoxiang Wang, Qinsheng Zhang, Kai Jiang, Hang Su, Stefano Ermon, Jun Zhu, and Ming-Yu Liu. Diffusionnft: Online diffusion reinforcement with forward process. *arXiv preprint arXiv:2509.16117*, 2025.

APPENDIX OVERVIEW

This appendix provides the experimental evidence and technical details supporting the main paper. Appendix A expands the four motivation studies in Fig. 1. Appendices B–C describe the training configuration, objective implementation, and response normalization. Appendix D reports controlled interventions validating the cross-modal influence proxy, while Appendix E measures the end-task cost of using a stale route. Appendices F–H present human evaluation, computational cost, and preference controllability. Appendix I analyzes representative failure modes, and Appendix J extends the method to the unified single-stream JavisDiT++ backbone. **Additional qualitative results, including extensive audio-video comparisons and failure cases, are provided in a separate HTML project page included with the supplementary material.**

Appendix Contents

Appendix A: Motivation Analysis: Experimental Details Behind the Teaser Figure 13
Appendix B: Details of Cross-Modal Influence-Guided Routing 17
Appendix C: Implementation Details 18
Appendix D: Validation of the Cross-Modal Influence Proxy 20
Appendix E: Route Refresh Frequency 22
Appendix F: Human Evaluation 23
Appendix G: Computational Cost and Training Efficiency 24
Appendix H: Controllability under Different Preference Priors 25
Appendix I: Failure Case Analysis 25
Appendix J: Generalization to Unified Single-Stream Backbone 27

A MOTIVATION ANALYSIS: EXPERIMENTAL DETAILS BEHIND THE TEASER FIGURE

The teaser figure in the main paper examines two fundamental questions in multi-reward reinforcement learning for joint audio-video diffusion models: *where should reward-driven updates act, and how should multiple, potentially conflicting rewards be combined?* Existing approaches generally answer these questions using static decisions. For update localization, they identify cross-modal layers from the base model and apply fixed gradient scaling throughout post-training. For reward coordination, they either use predefined scalar weights or determine dynamic weights solely from instantaneous gradient geometry. These strategies implicitly assume that the functional organization of the model and the relationships among rewards remain stable during training. This assumption is particularly restrictive for joint audio-video models. Their audio and video branches interact through bidirectional cross-attention, and reinforcement learning continuously changes how information and gradients propagate through these pathways. Meanwhile, rewards for visual quality, audio quality, semantic alignment, and temporal synchronization may disagree on the same generated sample. The strength and direction of these conflicts can also evolve as the policy changes.

We therefore conduct four preliminary experiments. The first two examine, from complementary forward and backward perspectives, whether the locations responsible for cross-modal synchronization remain fixed during training. The other two study whether fixed or purely gradient-based reward coordination can reliably handle reward disagreement while preserving all essential objectives. Together, these experiments motivate jointly adapting *update routing* and *reward coordination* throughout post-training.

A.1 FORWARD ANALYSIS: KV ABLATION REVEALS SHIFTING AV-SYNCHRONIZATION LAYERS

Necessity. A common strategy for controlling cross-modal optimization is to analyze the base model once, identify layers associated with audio-video interaction, and assign fixed gradient scales to these layers during the entire training process. However, the intervention itself changes the model. Strengthening or suppressing gradients at particular layers can redistribute cross-modal functions to other layers. Consequently, a routing rule obtained from the base model may become stale after

optimization begins. We use forward interventions to test whether the cross-modal layers responsible for audio-video synchronization remain stable between the base and trained models.

Experimental Procedure. The joint audio-video diffusion model contains 48 bidirectional cross-attention layers. We partition them into five contiguous ranges: L0–L9, L10–L19, L20–L29, L30–L39, and L40–L47. We separately analyze the two cross-modal directions: video-to-audio (V2A), in which video features provide keys and values to the audio branch, and audio-to-video (A2V), in which audio features provide keys and values to the video branch. For each direction and layer range, we disable the corresponding cross-modal K/V contribution during inference. Equivalently, the output of the selected cross-attention pathway is set to zero while the remainder of the model is left unchanged. We perform the same intervention on the pretrained base model and the model after reinforcement-learning post-training. The generated samples are evaluated using the Synchformer DeSync score, for which a lower value indicates better audio-video synchronization. If ablating a layer range substantially worsens DeSync, that range provides an important synchronization pathway. Conversely, if ablation improves DeSync, the corresponding pathway interferes with synchronization. We evaluate the released LTX-2 base checkpoint and the corresponding official OmniNFT checkpoint post-trained from LTX-2 on the same 1,000 evaluation prompts, and report the results averaged over all prompts.

Results. The resulting DeSync scores differ substantially between the base and trained models. In the base model, ablating several middle and deep regions causes substantial degradation, suggesting that these regions carry important cross-modal information. After training, this pattern changes markedly. For example, L40–L47 had previously been treated as an important synchronization region and assigned an increased cross-attention gradient scale. In the trained model, however, blocking the V2A pathway in this region improves DeSync from approximately 0.175 without ablation to 0.063. Thus, this region is no longer a beneficial synchronization pathway and may instead interfere with synchronization. In contrast, L30–L39 becomes the clearest functional region after training.

Analysis. These results show that audio-video synchronization is not permanently associated with a fixed set of cross-attention layers. Reinforcement learning reshapes the functional organization of the model, causing synchronization-sensitive regions to move across depth. This exposes a self-referential limitation of static routing: (i) the base model is analyzed to define a fixed route; (ii) that route changes the gradient received by each layer; (iii) training reorganizes the model in response to the route; and (iv) the resulting organization invalidates the observation used to define the route. A layer-routing strategy should therefore be recomputed from the evolving model rather than frozen at initialization.

A.2 Backward Analysis: Gradient Paths Shift During Training

Necessity. The forward ablation experiment identifies layers that causally affect synchronization during inference, but it does not directly show how reward-driven gradients are distributed during optimization. A static routing strategy could still be justified if the backward gradient paths remained stable even when the forward representations changed. We therefore complement the forward intervention with a layer-wise gradient analysis.

Experimental Procedure. We reproduce the backward computation of a training step and record gradient norms for the projection parameters in every bidirectional cross-attention layer. In a cross-attention operation, the query is produced by the destination branch, whereas the keys and values come from the source modality. We therefore distinguish Q gradients, which mainly characterize how the destination branch processes its own hidden state, from K/V gradients, which characterize how information from the other modality influences the destination branch. We record the layer-wise ℓ_2 gradient norms of the Q and K/V projections in both the audio and video branches, covering the V2A and A2V information paths. The same procedure is applied to the base model and the trained model. This produces layer-wise gradient profiles that reveal how the dominant gradient pathways move during post-training. We analyze the same released LTX-2 and official OmniNFT checkpoints on the same 1,000 prompts used in Appendix A.1. For each prompt, we backpropagate the full five-reward DiffusionNFT objective, compute the layer-wise gradient norms, and then average them over all 1,000 prompts.

Results. The layer-wise distributions change considerably after training. Most notably, the peak of the V2A Q-gradient moves from approximately L0 in the base model to approximately L43 in the trained model. The shallow-to-middle ratio of the A2V K/V gradient increases from approximately 0.67 to 0.93. More generally, the peaks of multiple Q and K/V gradient paths move toward deeper layers after training, indicating a redistribution of gradient mass rather than uniform rescaling. These observations are consistent with the forward ablation results: the trained model uses a different set of cross-modal pathways from the base model.

Analysis. Fixed gradient scaling can suppress or amplify a predefined layer range, but it cannot prevent the model from reorganizing its computation around that constraint. For example, reducing a gradient pathway in shallow layers may cause the model to move the corresponding interaction into deeper layers rather than eliminate it. The statistics obtained at initialization therefore describe only the current model state, not an invariant architectural property. The forward and backward experiments provide complementary evidence. KV ablation measures the causal effect of a cross-modal pathway on generated synchronization, whereas gradient analysis measures the optimization pressure passing through that pathway. Both indicate that update locations evolve during reinforcement learning. This motivates estimating cross-modal influence online and dynamically controlling token- and layer-level updates.

A.3 Pairwise Reward-Conflict Analysis

Necessity. Joint audio-video post-training optimizes several complementary objectives. Assigning visual rewards to the video branch and audio rewards to the audio branch may appear to isolate their effects, but this does not necessarily eliminate reward conflicts. Multiple rewards assigned to the same modality can disagree; the two branches remain coupled through bidirectional cross-attention; and synchronization is inherently cross-modal. We therefore quantify how frequently the rewards disagree on the relative quality of the same generated sample.

Rewards. We consider five objectives: HPSv3 for visual quality, VideoAlign for text-video alignment, AudioBox for audio quality, CLAP for text-audio alignment, and AV-DeSync for audio-video synchronization. Before comparison, all metric directions are aligned so that a larger normalized value consistently represents a better sample. In particular, the raw DeSync direction is reversed when constructing its reward signal because a lower raw DeSync score is better.

Experimental Procedure. We evaluate 1,000 prompts using the released LTX-2 checkpoint and generate four samples per prompt with different fixed seeds, yielding 4,000 samples in total. Because raw reward values are not directly comparable across prompts, we normalize every reward independently within each prompt group. For sample n, prompt p, and reward k, we compute

$$z_{p,k}^{(n)} = \frac{s_{p,k}^{(n)} - \mu_{p,k}}{\sigma_{p,k} + \epsilon}, \tag{15}$$

where $\mu_{p,k}$ and $\sigma_{p,k}$ are the mean and standard deviation of reward k across the four samples generated for prompt p. This per-prompt normalization removes differences in prompt difficulty and retains only each reward's relative preference among samples generated from the same prompt.

For every pair of rewards i and j, a sample is considered conflicting when their normalized scores have opposite signs:

$$z_{p,i}^{(n)} z_{p,j}^{(n)} < 0. \tag{16}$$

The pairwise conflict rate is

$$C_{i,j} = \frac{1}{N} \sum_{n=1}^{N} \mathbb{I}\left[z_i^{(n)} z_j^{(n)} < 0 \right]. \tag{17}$$

A conflict therefore means that one reward ranks a sample above its prompt-group average while the other ranks the same sample below its prompt-group average.

Results. Cross-modal disagreement is frequent, with several reward pairs conflicting on approximately half of the samples. More importantly, substantial conflicts also occur within a single modality: HPSv3 and VideoAlign conflict on 35% of samples, while AudioBox and CLAP conflict on 50%.

Analysis. The results reveal two limitations of fixed modality-level assignment. First, branch separation cannot resolve conflicts between rewards assigned to the same branch. Visual quality and visual semantic alignment are not interchangeable objectives, nor are audio quality and audio semantic alignment. Second, the assumption that a modality-specific reward affects only its corresponding branch is inaccurate for a joint model. Through A2V and V2A cross-attention, a nominally visual reward can affect audio-related representations, and an audio reward can alter the video branch. Branch assignment is therefore useful for expressing modality responsibility, but it is not a complete conflict-resolution mechanism. Reward weights must also adapt to the evolving relationships among objectives.

A.4 Gradient-Based Reward-Weight Drift

Necessity. Fixed reward weights preserve explicit preferences but cannot react to conflicts that change during training. Gradient-based multi-objective methods such as MARBLE provide a natural alternative: they estimate reward-specific gradients and determine weights that reduce local gradient conflict. However, a gradient-compatibility coefficient is not equivalent to reward importance. An objective may be essential while producing a weak, noisy, or locally redundant gradient. Directly treating geometry-derived coefficients as reward weights can therefore suppress an important objective. We examine this failure mode by tracking gradient-based reward weights throughout training.

Experimental Procedure. For every reward, we compute a reward-specific probe loss and its corresponding gradient. Pairwise inner products form a Gram matrix that describes the local gradient geometry. A minimum-norm multi-objective solver determines simplex coefficients

$$\boldsymbol{\alpha}^* = \arg\min_{\boldsymbol{\alpha}\in\Delta} \left\| \sum_{k=1}^{K} \alpha_k \widehat{\mathbf{g}}_k \right\|_2^2, \qquad \Delta = \left\{ \boldsymbol{\alpha} \mid \alpha_k \geq 0,\ \sum_k \alpha_k = 1 \right\}, \tag{18}$$

where $\widehat{\mathbf{g}}_k$ is the normalized gradient of reward k. In the unanchored gradient-based baseline, these coefficients directly determine the relative contribution of the five rewards to the training objective. We record the dynamic coefficients at multiple checkpoints throughout training. We run the unanchored MARBLE baseline on the LTX-2 backbone with three independent training seeds. At each training checkpoint, the reward coefficients are aligned by training round and averaged across the three runs; Fig. 1(d) reports these arithmetic means.

Results. The coefficients exhibit substantial drift. VideoAlign and CLAP receive increasingly large weights, HPSv3 remains relatively weak for much of training, AudioBox decreases in later stages, and, most importantly, the AV-DeSync weight progressively approaches zero. Thus, an objective essential to joint audio-video generation—temporal synchronization—can be nearly removed from training solely because its gradient is considered locally redundant or incompatible. Additional early-training diagnostics show that weak quality-related rewards can also receive zero coefficients when gradient estimates are noisy. Averaged across the three runs, HPSv3 and AudioBox receive near-zero weights at the initial round. At a later checkpoint, the mean worst cosine between the merged direction and an individual video-reward gradient reaches approximately -0.617.

Analysis. MARBLE coefficients answer which convex combination of the current gradients is most compatible according to local geometry; they do not encode which objectives are important and must remain active. This distinction is particularly important early in reinforcement learning, when fine-grained perceptual rewards may produce weaker or noisier gradients than broad semantic rewards. A minimum-norm solver can interpret these gradients as redundant and assign them very small coefficients even though their objectives remain essential.

This can create a harmful feedback loop: (i) a reward initially produces a weak or noisy gradient; (ii) the solver assigns it a small weight; (iii) the reward receives less optimization signal; and (iv) its gradient remains weak and is suppressed further. Dynamic reward coordination should therefore not overwrite predefined preferences. Gradient geometry should instead provide a bounded correction around a nonzero preference prior.

This observation motivates the residual formulation used by our method:

$$w_k = (1-\kappa) w_k^{\text{prior}} + \kappa\, \widetilde{\alpha}_k, \tag{19}$$

where w_k^{prior} encodes the predefined importance of reward k, $\widetilde{\alpha}_k$ is an appropriately scaled geometry-based correction, and κ controls the adaptation strength. We additionally delay gradient-based correction until after a warm-up stage, when the policy and reward-gradient estimates are more stable. This prevents noisy early estimates from permanently suppressing weak but necessary objectives.

A.5 Summary of Findings

The four preliminary experiments reveal that joint audio-video reinforcement learning is dynamic along two coupled dimensions. First, the forward and backward analyses show that synchronization-sensitive cross-modal layers and reward-gradient pathways both change during post-training, causing routes derived from the base model to become stale. Second, the reward analyses show that disagreement is frequent both across and within modalities: fixed weights cannot react to evolving conflicts, whereas purely geometry-based weights can suppress essential objectives.

These findings support the central design of **Adaptive Reward Routing**: jointly adapting *where reward-driven updates act* and *how competing rewards are combined*. Cross-modal influence-guided routing dynamically localizes token- and layer-level updates, while preference-preserving modality-aware reweighting adapts to current reward-gradient interactions without discarding predefined task priorities.

B Details of Cross-Modal Influence-Guided Routing

B.1 Token-Level Response Normalization

The aggregated response $d_{m,i}$ in Eq. 7 is clipped at its 99th percentile before min–max normalization. We use the numerical floor

$$\varepsilon_{\text{route}} = 10^{-8} \tag{20}$$

for all routing normalizations.

For audio tokens, the statistics are computed globally:

$$\widehat{d}_{a,i} = \frac{\text{clip}\left(d_{a,i}, d_a^{\min}, d_a^{99}\right) - d_a^{\min}}{d_a^{99} - d_a^{\min} + \varepsilon_{\text{route}}}, \tag{21}$$

where $d_a^{\min}$ and d_a^{99} are computed over all audio tokens.

Video response magnitudes can vary substantially across frames. We therefore normalize video tokens independently within each frame. Let $f(i)$ denote the frame containing token i. We compute

$$\widehat{d}_{v,i} = \frac{\text{clip}\left(d_{v,i}, d_{v,f(i)}^{\min}, d_{v,f(i)}^{99}\right) - d_{v,f(i)}^{\min}}{d_{v,f(i)}^{99} - d_{v,f(i)}^{\min} + \varepsilon_{\text{route}}}. \tag{22}$$

This prevents frames with uniformly larger response magnitudes from dominating the token weights. In Eq. 7, $\text{Norm}_{99}(d_{m,i})$ refers to Eq. 21 for audio and Eq. 22 for video. If the response range of an audio sequence or video frame is numerically degenerate, the normalized response is set to zero; Eq. 7 then assigns uniform unit weights rather than creating an artificial routing preference.

B.2 Layer-Level Response Normalization

For layer routing, the unnormalized score is the response averaged over selected timesteps and target tokens:

$$\delta_m^l = \frac{1}{|\mathcal{T}||\mathcal{I}_m|} \sum_{t\in\mathcal{T}} \sum_{i\in\mathcal{I}_m} d_{m,i}^{l,t}. \tag{23}$$

We normalize these scores across the cross-modal block set $\mathcal{B}$:

$$\widetilde{\delta}_m^l = \frac{\delta_m^l - \min_{q\in\mathcal{B}} \delta_m^q}{\max_{q\in\mathcal{B}} \delta_m^q - \min_{q\in\mathcal{B}} \delta_m^q + \varepsilon_{\text{route}}}. \tag{24}$$

If all layers have the same response up to numerical precision, we set $\widetilde{\delta}_m^l = 0$ for every l, avoiding a spurious ranking among indistinguishable layers. The effective gradient scale through the source-modality KV path is

$$s_m^l = 1 - \alpha_m^l = 1 - \left(1 - \widetilde{\delta}_m^l\right)^{1/\tau}. \tag{25}$$

A larger response therefore preserves a larger fraction of the gradient. A2V and V2A scores and normalization statistics are computed independently.

C Implementation Details

Forward-Process RL Objective. We train under the DiffusionNFT objective (Zheng et al., 2025). Each prompt is associated with a group of four jointly generated audio-video samples. Training rollouts use 20 denoising steps, and we randomly retain 40% of the flow-matching timesteps for policy optimization. Evaluation uses 40 denoising steps. Advantages are clipped to $[-5, 5]$, and we use the same value as the optimality-probability scale, $A_{\max} = 5$. The policy extrapolation coefficient is set to $\beta = 1.0$, and the reference-regularization coefficient is 10^{-4}.

Parameter-Efficient Optimization. We use LoRA with rank 32 and scaling factor 64 on the self-attention, text cross-attention, feed-forward, and bidirectional audio-video attention projections. We optimize the LoRA parameters with AdamW using a learning rate of 1×10^{-4}, $(\beta_1, \beta_2) = (0.9, 0.999)$, an optimizer numerical constant of $\epsilon_{\text{Adam}} = 10^{-8}$, weight decay 10^{-4}, and gradient clipping at 1.0. Training uses bfloat16 precision, activation checkpointing, and PyTorch FSDP. Training samples are generated at 768×512 resolution with 121 frames at 24 FPS (≈ 5.04 seconds).

Policy Update. The updated policy is initialized by copying the old policy. After every sampling-and-optimization round, its trainable parameters are updated by

$$\theta_{\text{updated}} \leftarrow d(s)\,\theta_{\text{updated}} + \big(1 - d(s)\big)\theta, \qquad d(s) = \min(0.001s, 0.5), \tag{26}$$

where s is the optimizer step. The updated policy is used to generate the rollout group in the next round.

Routing and Reward Coordination. For influence-guided routing, we set the maximum token weight to $\lambda_{\max} = 3.0$ and the layer-routing temperature to $\tau = 0.5$. Routing statistics are detached from the computation graph and reused in the corresponding policy update. The block set $\mathcal{B}$ contains all 48 bidirectional cross-attention blocks. For a 20-step rollout, responses are collected over the intermediate-to-late denoising window spanning 50%–90% of the trajectory; under zero-based indexing, this corresponds to

$$\mathcal{T} = \{10, 11, \ldots, 18\}. \tag{27}$$

Thus, routing statistics exclude the earliest highly noisy states and the final low-noise steps, where cross-modal responses may be less stable or less informative for policy optimization.

For reward coordination, branch-specific MARBLE coefficients are refreshed every 20 sampling rounds using two probe timesteps and are smoothed with an EMA coefficient of $\rho = 0.7$. We use a reward-coordination warm-up of $e_{\text{warm}} = 50$ sampling rounds. Afterwards, the residual correction strength is $\kappa = 0.3$. The prior weights for VideoAlign, HPSv3, AudioBox Aesthetics, CLAP, and AV-DeSync are 0.9, 1.2, 0.9, 0.5, and 1.5, respectively. VideoAlign and HPSv3 supervise the video branch, AudioBox Aesthetics and CLAP supervise the audio branch, and AV-DeSync supervises both branches.

For branch m, we define

$$\mathcal{K}_m^+ = \mathcal{K}_m \cup \mathcal{K}_c, \qquad C_m = \sum_{k \in \mathcal{K}_m^+} \omega_{m,k}^{\text{prior}}, \tag{28}$$

so that scaling the simplex-valued MARBLE coefficients by C_m preserves the total prior reward mass within that branch. On each refresh round, we compute one loss per reward using only the target-modality loss associated with that reward. Video rewards are probed through the video-token loss, audio rewards through the audio-token loss, and AV-DeSync through both. Token-routing weights are set uniformly during probing so that the measured gradient geometry is not conditioned on the current token route.

Let

$$g_{m,k} = \nabla_{\theta_{\text{LoRA}}} \mathcal{L}_{m,k}, \qquad \widehat{g}_{m,k} = \frac{g_{m,k}}{\|g_{m,k}\|_2 + \varepsilon_{\text{route}}}. \tag{29}$$

Within each branch, MARBLE solves

$$\gamma_m^* = \arg \min_{\gamma_m \in \Delta^{|\mathcal{K}_m^+|}} \left\| \sum_{k \in \mathcal{K}_m^+} \gamma_{m,k} \widehat{g}_{m,k} \right\|_2^2, \tag{30}$$

where

$$\Delta^{|\mathcal{K}_m^+|} = \left\{ \gamma_m : \gamma_{m,k} \geq 0, \sum_{k \in \mathcal{K}_m^+} \gamma_{m,k} = 1 \right\}. \tag{31}$$

The resulting coefficients are EMA-smoothed and cached for subsequent rounds before being combined with the preference priors through Eq. 10.

The lower-is-better DeSync score is converted to the higher-is-better AV-DeSync reward by

$$R_{\text{AV-DeSync}} = \frac{1}{1 + D_{\text{sync}}}, \tag{32}$$

where $D_{\text{sync}} \geq 0$ is the absolute temporal offset predicted by Synchformer. The original D_{sync} is retained as the lower-is-better DeSync metric during evaluation.

Generation and Evaluation. All post-training runs are conducted on three nodes, each equipped with eight NVIDIA H200 GPUs. JavisBench outputs are standardized to four seconds, 24 FPS, and 16-kHz audio. Evaluation uses video and audio guidance scales of 1.5 and 3.0, respectively, with 40 inference steps. Training large joint audio-video diffusion models and repeatedly evaluating all generated samples on JavisBench are computationally expensive. We therefore adopt a multi-seed protocol that balances evaluation reliability with computational cost. For the main comparison in Table 1, each reproduced post-training method (GDPO, MARBLE, OmniNFT, and Ours) is independently trained with three random seeds, and each resulting checkpoint is evaluated on the full JavisBench under the same protocol. We report the arithmetic mean across these three runs. The Base Model and OmniNFT* are fixed checkpoints evaluated under the same generation and evaluation settings, with OmniNFT* denoting the checkpoint released by its original authors. For the ablation study in Tab. 2, ours reuses the corresponding three-seed averages from Tab. 1, whereas each remaining ablation configuration is independently trained and evaluated with three different random seeds. The reported ablation results are likewise averaged across these runs, rather than selected from a single favorable seed.

Residual Normalization. The token-level objective in Eq. 12 normalizes each positive- and negative-policy residual by its per-sample magnitude. Let D_m denote the feature dimension of modality m. We compute

$$w_m^{\pm,(n)} = \text{sg}\left(\frac{1}{|\mathcal{I}_m| D_m} \sum_{i \in \mathcal{I}_m} \sum_{d=1}^{D_m} \left| v_{\theta,m,i,d}^{\pm,(n)} - u_{m,i,d}^{(n)} \right| \right), \tag{33}$$

where $\text{sg}(\cdot)$ denotes stop-gradient. In implementation, each residual normalizer is lower-bounded by

$$\widetilde{w}_m^{\pm,(n)} = \max\left(w_m^{\pm,(n)}, \varepsilon_{\text{res}} \right), \qquad \varepsilon_{\text{res}} = 10^{-5}, \tag{34}$$

and $\widetilde{w}_m^{\pm,(n)}$ is used in Eq. 12. All reproduced baselines use the same residual normalization, so differences between methods are not caused by this scaling.

Reference Regularization. The regularization term denoted by $\mathcal{L}_{\text{KL},m}$ in Eq. 14 is implemented as a squared prediction-matching penalty toward the frozen reference model. For modality m and sample n,

$$\mathcal{L}_{\text{KL},m}^{(n)} = \frac{1}{|\mathcal{I}_m|} \sum_{i \in \mathcal{I}_m} \left\| \widehat{x}_{0,\theta,m,i}^{(n)} - \widehat{x}_{0,\text{ref},m,i}^{(n)} \right\|_2^2, \tag{35}$$

where $\widehat{x}_{0,\theta}$ and $\widehat{x}_{0,\text{ref}}$ are the denoised predictions of the trainable and frozen reference policies at the same noisy latent and timestep. We retain the notation $\mathcal{L}_{\text{KL},m}$ for consistency with the DiffusionNFT objective, while Eq. 35 states its operational implementation.

Training Procedure. Algorithm 1 summarizes the complete optimization loop. The A2V and V2A responses are collected during rollout generation and do not require additional ablated sampling passes. Reward-gradient probing is performed only on refresh rounds. The resulting smoothed coefficients are cached and used to construct the reward weights in the next round, which avoids using a coefficient estimated from the same update that it controls.

D Validation of the Cross-Modal Influence Proxy

D.1 Connection to the Main Method and Validation Objectives

Cross-Modal Influence-Guided Routing uses $d^{l,t}_{m,i}$ in Eq. 6 at two resolutions. Averaging the responses over layers and timesteps produces the token score behind $\lambda_{m,i}$ in Eq. 7. Averaging them over tokens and timesteps produces the layer score δ^l_m and the corresponding gradient scale s^l_m in Eqs. 23–25. Both routes therefore rely on the same assumption: a larger response indicates a token or layer with a greater effect on the final target-modality prediction.

Activation magnitude alone does not guarantee downstream influence. The response is subsequently transformed by the AdaLN gate, residual addition, and later Transformer blocks. We therefore compare the response-based scores with controlled interventions that directly measure their downstream effect. The validation follows the two aggregation paths used by the main method and corresponds to the three panels of Fig. 5:

1. **Layer-Level Fidelity.** We compare the response-derived layer profile with the prediction change caused by disabling each layer. This evaluates whether the ordering induced by δ^l_m, and therefore by the monotonic scale s^l_m, reflects functional layer importance (Fig. 5(a)).
2. **Dynamic Tracking.** We repeat the layer comparison at multiple checkpoints and contrast the proxy recomputed from the current model with the profile frozen at initialization. This evaluates whether recomputation is necessary as cross-modal computation changes (Fig. 5(b)).
3. **Token-Level Fidelity.** We use the across-layer token score to select high-response, random, and low-response token groups, then remove the cross-modal information received by each group under an equal budget. Since $\lambda_{m,i}$ is monotonic in this score, the experiment directly validates the token ordering used by the policy loss (Fig. 5(c)).

The notation below distinguishes the two quantities compared in this evaluation. We use P for a response-derived proxy and C for the downstream effect measured by direct intervention. These are evaluation-specific quantities rather than additional routing variables.

D.2 Experimental Protocol

D.2.1 Layer-Level Fidelity

Models, Checkpoints, and Inputs. LTX-2 contains 48 `BasicAVTransformerBlock`s indexed from 0 to 47. We evaluate the base, early, middle, and final checkpoints on 128 prompts held out from reward-guided post-training. Prompt order and initial noise seeds are fixed across checkpoints. For each prompt, we uniformly select three denoising states from the same intermediate-to-late window $\mathcal{T}$ used by the routing module. Guidance is disabled during probing. Clean and ablated evaluations use identical prompt embeddings, noisy audio-video latents, timesteps, and positional embeddings.

Response Proxy. Let $m \in \{v, a\}$ denote the target modality and $\bar{m}$ the source modality. A2V targets the video prediction, while V2A targets the audio prediction. For prompt n, timestep t, and block l, the pre-gate layer proxy is

$$P^{(n,t,l)}_m = \frac{1}{|\mathcal{I}_m|} \sum_{i \in \mathcal{I}_m} \left\| o^{(n,t,l)}_{\bar{m}\to m,i} \right\|_2 . \tag{36}$$

This is the same response collected by the routing module before the block-level gated residual addition, so obtaining P_m requires no additional forward pass.

Single-Layer Path Ablation. We first cache the clean velocity prediction, then replay the same denoising step while disabling exactly one directional cross-attention block. A2V interventions use `PerturbationType.SKIP_A2V_CROSS_ATTN` with `blocks=[l]`; V2A uses the corresponding `SKIP_V2A_CROSS_ATTN` configuration. This sets the selected directional attention response to zero before gated residual addition and leaves all other blocks unchanged. We measure the resulting target-modality effect as

$$C_m^{(n,t,l)} = \frac{\left\| v_\theta^m(x_t^{(n)}, c^{(n)}, t) - v_{\theta,\neg(l,\bar{m}\to m)}^m(x_t^{(n)}, c^{(n)}, t) \right\|_2}{\left\| v_\theta^m(x_t^{(n)}, c^{(n)}, t) \right\|_2 + \varepsilon}. \tag{37}$$

The norm covers all target-modality tokens and feature dimensions. Although the intervention is applied at one block, its effect is measured at the final velocity prediction and therefore includes all downstream computation. The 48 interventions are evaluated in memory-compatible chunks and require no video or audio decoding.

Aggregated Layer Profiles. For checkpoint θ_e, we average the proxy and intervention effects over prompts and selected timesteps:

$$\begin{aligned} \bar{P}^{\,l}_{m,\theta_e} &= \frac{1}{N|\mathcal{T}|} \sum_{n=1}^{N} \sum_{t\in\mathcal{T}} P_{m,\theta_e}^{(n,t,l)}, \\ \bar{C}^{\,l}_{m,\theta_e} &= \frac{1}{N|\mathcal{T}|} \sum_{n=1}^{N} \sum_{t\in\mathcal{T}} C_{m,\theta_e}^{(n,t,l)}. \end{aligned} \tag{38}$$

Each profile contains one entry for each of the 48 blocks.

Layer-Ranking Metric and Uncertainty. The primary statistic is Spearman's rank correlation across the two 48-entry profiles:

$$\rho_{m,\text{current}}^{(e)} = \text{Spearman}_l\left(\bar{P}^{\,l}_{m,\theta_e}, \bar{C}^{\,l}_{m,\theta_e}\right). \tag{39}$$

A larger value indicates stronger agreement between the proxy and intervention rankings; $\rho = 1$ denotes identical rankings. We also report top-8 overlap.

Figure 5(a) visualizes the same final-checkpoint profiles after independently min–max normalizing each profile across layers. This monotonic transformation does not change Spearman correlation. Each point represents one block, with its horizontal position determined by the proxy and its vertical position determined by direct intervention.

We estimate uncertainty with 1,000 prompt-level bootstrap replicates. Each replicate resamples the 128 prompts with replacement, recomputes both aggregated profiles, and then recomputes their correlation. The reported central values, scatter plots, and 95% confidence intervals therefore use the same aggregation and statistical definition. A2V and V2A are always evaluated independently.

D.2.2 Dynamic Tracking

At training stage e, we compare the current proxy with a profile frozen at the base model:

$$\begin{aligned} \rho_{m,\text{current}}^{(e)} &= \text{Spearman}_l\left(\bar{P}^{\,l}_{m,\theta_e}, \bar{C}^{\,l}_{m,\theta_e}\right), \\ \rho_{m,\text{frozen}}^{(e)} &= \text{Spearman}_l\left(\bar{P}^{\,l}_{m,\theta_0}, \bar{C}^{\,l}_{m,\theta_e}\right), \end{aligned} \tag{40}$$

where θ_0 is the base model. The intervention profile $\bar{C}_{m,\theta_e}$ is always recomputed from the evaluated checkpoint. The two curves in Fig. 5(b) differ only in whether the proxy follows the current checkpoint or remains fixed at initialization. Their confidence intervals use paired prompt-level bootstrap resampling.

D.2.3 Token-Group Interventions

For each prompt and checkpoint, we compute the token score with the same aggregation used during training:

$$d_{m,i} = \frac{1}{|\mathcal{B}||\mathcal{T}|} \sum_{l\in\mathcal{B}} \sum_{t\in\mathcal{T}} d_{m,i}^{l,t}. \tag{41}$$

Video tokens are ranked independently within each frame, while audio tokens are ranked globally. We construct equally sized masks from the highest-response 10%, a uniformly sampled 10%, and the lowest-response 10%. Random video masks preserve the same number of selected tokens in each frame as the high- and low-response masks.

Because the token score is aggregated across layers, a selected mask is applied to the corresponding directional response in all 48 blocks:

$$\tilde{o}^{l,t}_{\bar{m}\to m,i} = \begin{cases} 0, & i \in \mathcal{S}_m, \\ o^{l,t}_{\bar{m}\to m,i}, & i \notin \mathcal{S}_m, \end{cases} \qquad \forall l \in \mathcal{B},\ t \in \mathcal{T}. \tag{42}$$

The mask is applied after the attention output projection and before gated residual addition. It removes only the cross-modal information received by selected target tokens; token representations, self-attention states, and unselected responses remain unchanged.

We evaluate ten independently sampled random masks per prompt. Prediction changes are averaged over selected timesteps and random masks before dataset-level aggregation. Each direction and token group receives an independent 95% confidence interval from 1,000 prompt-level bootstrap replicates. A2V and V2A are compared only within direction because their target velocity spaces have different scales.

D.3 Results and Analysis

D.3.1 Layer-Level Fidelity

At the final checkpoint, the aggregated proxy and intervention profiles reach Spearman correlations of 0.98 for A2V and 0.97 for V2A, with 95% prompt-bootstrap confidence intervals of $[0.96, 0.99]$ and $[0.91, 0.99]$. Their top-8 overlaps are 75% and 100%. A correlation close to one means that the proxy and direct interventions rank the 48 layers almost identically. Thus, high proxy scores identify the layers whose removal most strongly changes the final target-modality prediction. The agreement in both directions also shows that the proxy captures asymmetric A2V and V2A organization rather than a single shared depth prior.

D.3.2 Dynamic Tracking

Current and frozen profiles are identical at initialization, but separate during fine-tuning. The current proxy remains above 0.96, while the frozen correlation falls from 0.98 to 0.56 for A2V and from 0.98 to 0.38 for V2A. Its final top-8 overlap is 0% in both directions. Since the intervention profile is recomputed at every checkpoint, this decline reflects a real change in where cross-modal influence is carried. The base-model route therefore becomes stale, while recomputing the proxy tracks the evolving model.

D.3.3 Token-Level Fidelity

Under the same 10% token budget and across the same 48 blocks, high-, random-, and low-score ablations produce mean normalized effects of 0.125, 0.077, and 0.037 for A2V, and 0.111, 0.064, and 0.030 for V2A. High-score ablation is therefore $1.62\times$ and $1.74\times$ stronger than random ablation. Because all groups affect the same number of tokens and layers, the difference reflects token selection rather than ablation size. These results confirm that the response score identifies target tokens through which cross-modal information has a greater effect on the final prediction.

This validation establishes the proxy as a measure of functional cross-modal influence on the current prediction. However, it does not assume that response magnitude is an exact causal estimate of reward-specific update utility.

E Route Refresh Frequency

Appendix D shows that a response profile frozen at initialization gradually diverges from the current model. Here we measure the end-task consequence of using a stale route. All rows use the identical proxy of Eq. 6, the same token- and layer-routing formulation, and the fixed GDPO weighting. They

Table 3: **Effect of route refresh frequency on LTX-2.** All routed configurations share the same proxy, routing formulation, and fixed reward weights, differing only in the recomputation schedule. No routing (GDPO) is included as a reference baseline. (Mean ± std over 3 seeds.)

Route schedule	AV-Quality		Text-Consistency				AV-Consistency		AV-Synchrony	
	VQ↑	AQ↑	TV-IB↑	TA-IB↑	CLIP↑	CLAP↑	AV-IB↑	AVHScore↑	JavisScore↑	DeSync↓
No routing (GDPO)	2.722 $\pm$0.0136	5.450 $\pm$0.0217	0.261 $\pm$0.0020	0.138 $\pm$0.0018	0.312 $\pm$0.0014	0.347 $\pm$0.0031	0.174 $\pm$0.0025	0.175 $\pm$0.0022	0.155 $\pm$0.0020	0.671 $\pm$0.0104
Frozen at initialization	2.664 $\pm$0.0140	5.398 $\pm$0.0222	0.259 $\pm$0.0022	0.135 $\pm$0.0021	0.311 $\pm$0.0015	0.341 $\pm$0.0033	0.170 $\pm$0.0027	0.171 $\pm$0.0025	0.151 $\pm$0.0023	0.698 $\pm$0.0118
Refresh every 50 rounds	2.913 $\pm$0.0120	5.579 $\pm$0.0191	0.262 $\pm$0.0018	0.146 $\pm$0.0017	0.319 $\pm$0.0013	0.371 $\pm$0.0027	0.192 $\pm$0.0022	0.194 $\pm$0.0024	0.167 $\pm$0.0019	0.512 $\pm$0.0086
Per-round recomputation	**3.315** $\pm$0.0103	**5.839** $\pm$0.0171	**0.266** $\pm$0.0015	**0.162** $\pm$0.0014	**0.313** $\pm$0.0011	**0.411** $\pm$0.0023	**0.226** $\pm$0.0018	**0.229** $\pm$0.0020	**0.190** $\pm$0.0016	**0.343** $\pm$0.0065

therefore extend the routing chain of Tab. 2 along a single axis: how often the route is recomputed from the current model. We train each configuration on LTX-2 with three random seeds under the protocol of Appendix C and report the mean.

As shown in Tab. 3, a route frozen at initialization performs no better than not routing at all, and is slightly worse on the synchronization metrics. This is consistent with Appendix D: once the initial profile no longer matches the current model, a frozen route persistently detaches gradients in layers that have become influential and emphasizes tokens that have become uninformative. A stale route therefore misallocates supervision rather than merely losing its benefit.

Performance then improves as the refresh interval shortens. A 50-round interval recovers only part of the gap, because the route remains outdated for most of training. Since the responses are collected during rollout sampling and require no additional model evaluations, we adopt per-round recomputation by default. Overall, the benefit of influence-guided routing depends on tracking the evolving model closely, rather than on any particular fixed routing pattern.

F Human Evaluation

Automatic metrics provide broad and reproducible coverage, but several of them are related to the reward models used during post-training. We therefore complement the automatic evaluation with a human study of perceptual quality and audio–video correspondence.

F.1 Stimuli and compared methods

We stratify 10 prompts from JavisBench across its event, visual-style, sound-type, spatial-composition, and temporal-composition categories. For every prompt, we evaluate outputs from both LTX-2 and LTX-2.3, yielding 20 backbone–prompt groups. Within each group, we compare five methods trained under the controlled setting used in the main paper: the base model, GDPO, MARBLE, OmniNFT, and Ours. All clips are normalized to four seconds, 24 FPS, and 16-kHz audio, as in the automatic evaluation.

F.2 Participants and protocol

We recruit 20 participants who are at least 18 years old, report normal or corrected-to-normal vision and no known hearing impairment. Participants provide informed consent and are compensated at a locally appropriate hourly rate. Before the main study, each participant completes a headphone check, an audio-level calibration trial, and a short practice block.

Each trial displays the prompt together with two anonymized clips in randomized left–right order. Participants may replay either clip and are asked to select the better output, or *Tie / no perceptible difference*, across five dimensions: (i) visual quality and temporal stability, (ii) audio quality and naturalness, (iii) text-to-audio-video semantic consistency—whether both the visible content and the audible content correctly realize the prompt, including the requested subjects, actions, speech, and sound sources, (iv) audio–video semantic alignment and temporal synchrony, and (v) overall preference. Method names are never shown. We use a balanced incomplete-block design: for each of the 20 backbone–prompt groups, four of the ten unordered method pairs are selected such that every pair occurs equally often within each backbone. This produces 80 unique pairwise trials, each judged by three independent participants, for 240 scored trial assignments in total.

Table 4: **Human evaluation.** Scores are pooled balanced pairwise preference rates (%), where 50 denotes parity against the average competing method. Higher is better. Best and second-best results are marked in bold and underlined, respectively.

Backbone	Method	Visual Quality ↑	Audio Quality ↑	Text–AV Consistency ↑	AV Alignment ↑	Overall Preference ↑
LTX-2	Base Model	27.1	32.3	37.5	33.3	30.2
	GDPO	44.8	46.9	33.3	31.2	42.7
	MARBLE	35.4	31.2	35.4	37.5	35.4
	OmniNFT	<u>66.7</u>	<u>64.6</u>	<u>64.6</u>	<u>68.8</u>	<u>65.6</u>
	Ours	**76.0**	**75.0**	**79.2**	**79.2**	**76.0**
LTX-2.3	Base Model	25.0	33.3	36.5	33.3	29.2
	GDPO	45.8	38.5	34.4	31.2	39.6
	MARBLE	38.5	31.2	34.4	39.6	36.5
	OmniNFT	<u>66.7</u>	<u>67.7</u>	<u>65.6</u>	<u>66.7</u>	<u>67.7</u>
	Ours	**74.0**	**79.2**	**79.2**	**79.2**	**77.1**

Table 5: **Resource allocation for one independent post-training run.** The reward and training pools execute concurrently, so the end-to-end wall-clock time is 48 hours rather than the sum of the two rows.

Worker role	H200 GPUs	Memory/GPU	Wall-clock	GPU-hours
Policy rollout and optimization	16	141 GB	48 h	768
Reward-model evaluation	8	141 GB	48 h	384
End-to-end run	24	141 GB	48 h	1,152

F.3 Preference score

For method m and criterion c, let $W_{m,c}$, $L_{m,c}$, and $T_{m,c}$ be its numbers of wins, losses, and ties across all valid pairwise judgments. We report the balanced pairwise preference score

$$S_{m,c} = 100 \times \frac{W_{m,c} + 0.5T_{m,c}}{W_{m,c} + L_{m,c} + T_{m,c}}. \tag{43}$$

A score of 50 indicates equal preference against the average competing method. Because every unordered pair receives the same number of judgments, the score is not biased by unequal opponent frequency.

F.4 Analysis

As shown in Tab. 4, the human preferences follow a consistent pattern across both backbones. Ours is preferred on all five criteria, with the clearest advantages in text–AV consistency and AV alignment, while also retaining strong visual and audio quality. This suggests that the gains in cross-modal coordination do not come at the expense of either individual modality. GDPO improves visual quality but remains weaker in AV alignment, whereas MARBLE provides a more balanced yet limited improvement. OmniNFT is consistently stronger than these global multi-reward baselines, and Ours further improves upon it by adapting both routing locations and reward coordination. Overall, the human judgments agree with the trends in the automatic evaluation and support the effectiveness of Adaptive Reward Routing.

G Computational Cost and Training Efficiency

All reproduced post-training methods use the same fixed resource envelope on NVIDIA H200 GPUs. Sixteen GPUs run policy rollout and optimization with PyTorch FSDP, bfloat16 precision, activation checkpointing, and the LoRA configuration described in Appendix C. Eight additional GPUs evaluate the five reward models and return reward values to the training workers. The two pools operate concurrently. Each independent run is trained for approximately 48 hours. The 141-GB memory of each H200 is sufficient to hold its training or reward-evaluation shard without CPU or NVMe parameter offloading, although the training workers have little unused device-memory headroom.

Table 6: **Compute implied by the multi-seed evaluation protocol.** GPU-hours count H200 device time and assume the fixed 24-GPU, 48-hour budget for every independently trained configuration.

Evaluation unit	Runs	GPU-hours/run	Total GPU-hours
One configuration, one seed	1	1,152	1,152
Main-table method	3	1,152	3,456
Ablation configuration	3	1,152	3,456

Table 7: **Prior reward profiles used to test preference controllability.** VA denotes VideoAlign, AB denotes AudioBox Aesthetics, and AV-DeSync denotes the higher-is-better training reward derived from the lower-is-better DeSync metric.

Profile	VA	HPSv3	AB	CLAP	AV-DeSync
Visual-oriented	1.25	1.75	0.35	0.65	1.00
Audio-oriented	0.65	0.75	1.25	1.35	1.00
Cross-modal-oriented	0.75	1.00	0.50	0.75	2.00

H Controllability under Different Preference Priors

The residual reweighting in Eq. 10 is intended to adapt to gradient conflicts without erasing user-specified priorities. We directly test this claim by changing only the prior reward profile while keeping the total prior mass, training data, random-seed protocol, routing hyperparameters, residual strength $\kappa = 0.3$, and optimization budget fixed. We train three profiles (as shown in Tab. 7) on LTX-2 and evaluate the resulting checkpoints on the full JavisBench.

As shown in Tab. 8, the visual-oriented prior primarily improves VQ, TV-IB, and CLIP; the audio-oriented prior primarily improves AQ, TA-IB, and CLAP; and the cross-modal-oriented prior primarily improves AV-IB, AVHScore, and JavisScore while reducing DeSync. This is the behavior of Eq. 10: the prior changes the preferred operating point, while the $\kappa = 0.3$ conflict-aware residual, shared routing mechanism, and reference regularization prevent a heavily weighted objective from overwhelming the other modalities.

I Failure Case Analysis

Beyond the qualitative comparisons in the main paper, we document *three* representative failure modes that persist across all three methods (LTX-2, OmniNFT, and Ours). Crucially, these failures are not specific to a particular reward-routing strategy; rather, they reflect systemic limitations of the underlying joint audio–video generation backbone, and therefore highlight directions for *future* work rather than differences among the compared methods. The three cases are as follows. **(i) Fine-grained audio fidelity:** the model mis-pronounces polyphonic Chinese characters—for instance, reading *dé* instead of the contextually correct *děi* in the line 得加钱—a symptom rooted in the lack of fine-grained, context-aware audio supervision; the on-screen text and visuals remain correct, yet the spoken audio is wrong. **(ii) Long-tail content (anime effects):** highly stylised visual effects (glowing cards, elemental blasts, stylised particles) are poorly rendered, because such content lies on the long tail of the pre-training distribution and the ceiling of post-training is bounded by the pre-trained prior. **(iii) Complex multi-person fast motion:** fast multi-subject interactions such as a table-tennis rally break temporal consistency—the ball is dropped or smeared mid-flight and limbs fuse or ghost at the moment of contact—because the reward stack lacks a fine-grained physical plausibility term and the backbone's temporal prior caps what post-training can recover. All three failure modes affect LTX-2, OmniNFT, and Ours alike.

I.1 Failure 1: Fine-Grained Audio Fidelity

The clip in this case depicts a confrontation in a hazy encampment: a robed figure holding a long weapon speaks the line "…得加钱" (roughly, "…we must raise the price"), to which a mounted, armoured rider replies. The character 得 is a polyphone: in this context it should be read **děi**

Table 8: **Effect of the preference prior on LTX-2.** (Mean ± std over 3 seeds.)

Prior profile	AV-Quality		Text-Consistency				AV-Consistency		AV-Synchrony	
	VQ↑	AQ↑	TV-IB↑	TA-IB↑	CLIP↑	CLAP↑	AV-IB↑	AVHScore↑	JavisScore↑	DeSync↓
Visual-oriented	**3.425** $\pm$0.0107	5.817 $\pm$0.0172	**0.270** $\pm$0.0013	0.165 $\pm$0.0015	**0.316** $\pm$0.0010	0.421 $\pm$0.0022	0.232 $\pm$0.0019	0.231 $\pm$0.0020	0.203 $\pm$0.0017	0.352 $\pm$0.0067
Audio-oriented	3.292 $\pm$0.0114	**5.962** $\pm$0.0160	0.266 $\pm$0.0016	**0.172** $\pm$0.0013	0.312 $\pm$0.0012	**0.434** $\pm$0.0020	0.233 $\pm$0.0021	0.232 $\pm$0.0018	0.204 $\pm$0.0016	0.348 $\pm$0.0064
Cross-modal-oriented	3.301 $\pm$0.0109	5.835 $\pm$0.0177	0.267 $\pm$0.0015	0.166 $\pm$0.0014	0.313 $\pm$0.0011	0.423 $\pm$0.0023	**0.242** $\pm$0.0017	**0.243** $\pm$0.0019	**0.216** $\pm$0.0015	**0.316** $\pm$0.0059

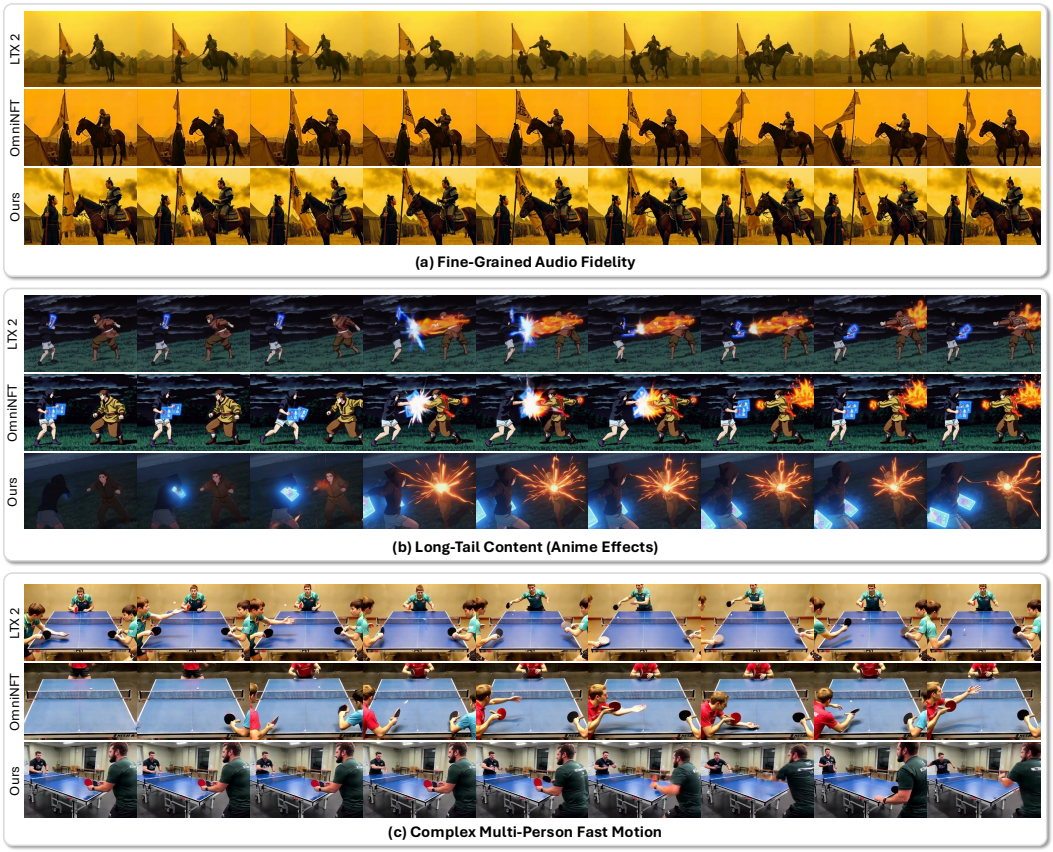

Figure 6: **Persistent failure modes. (a) Fine-grained audio fidelity**: an encampment confrontation where all three methods mis-pronounce the polyphonic character 得 (*dé* instead of contextually correct *děi*); the on-screen text and visual composition are correct. **(b) Long-tail content (anime effects)**: a stylised anime battle where glowing cards, elemental blasts and particle effects are under-rendered by every method due to insufficient pre-training coverage. **(c) Complex multi-person fast motion**: a table-tennis rally where the small, fast-moving ball is dropped or smeared mid-flight and rapid limb exchanges produce ghosted or fused arms across all three methods. In each sub-panel, three rows show sampled frames from LTX-2, OmniNFT and Ours respectively, illustrating that these failures are backbone-level rather than routing-specific.

(expressing necessity, "must / have to"), yielding the correct reading **děi jiā qián**. However, the generated audio instead renders it as **dé** ("to obtain"), i.e. **dé jiā qián**.

This error is not a visual or textual failure — the on-screen characters, scene composition, and lip-sync timing are all correct. The mistake is purely phonetic and context-dependent: the model fails to disambiguate the correct reading of a polyphonic character from the surrounding semantics. We attribute this to the absence of fine-grained audio supervision that would teach context-aware pronunciation disambiguation. Because the defect lies in the audio-generation supervision rather than in the reward-routing mechanism, it appears identically in LTX-2, OmniNFT, and Ours. A promising remedy is to augment the training corpus with Chinese speech data carrying explicit phonetic (*pinyin* / grapheme) annotations and to introduce phoneme- or character-level alignment supervision.

I.2 Failure 2: Long-Tail Content (Anime Effects)

The second case features a highly stylised anime battle: two anime-style figures face off on a grassy field under a dark, cloudy night sky, one wielding two glowing blue rectangular cards bearing the characters 风 (wind) and 雷 (thunder), the other emitting fiery orange energy; the two elemental forces collide in a stylised particle explosion, while wind howls and the background music intensifies in tempo.

Such strongly stylised visual effects reside on the long tail of the pre-training data distribution. Post-training alone cannot close this gap: the ceiling of post-training is fundamentally bounded by the pre-trained prior, so a style that the base model has rarely seen cannot be recovered through reward optimisation at fine-tuning time. The symptom — washed-out or inconsistent card glow, muted particle effects, and weak alignment between the escalating audio and the on-screen action — is therefore shared by all three methods.

Substantially improving these cases requires changing the data mix rather than the routing strategy: adding stylised / anime-domain data and performing supervised fine-tuning (SFT) on the base model before applying adaptive reward routing.

I.3 Failure 3: Complex Multi-Person Fast Motion

The third case captures a table-tennis rally: two players exchange fast drives across a blue table in a bright indoor hall, framed from a side-on, table-height medium shot that shows only the players' forearms, bats and the small white ball against the darker floor. Across the rally every method produces the characteristic symptoms of temporal inconsistency under fast multi-subject motion: the small, fast-moving ball is frequently dropped or smeared between frames and frequently disappears mid-rally altogether, and the rapid limb exchanges at the moment of contact yield ghosted, fused, or temporarily re-attached arms even when the surrounding character identity is well preserved (most visibly so in the Ours row of panel (c)).

This error is not a reward-routing failure either: the routing stack we optimise targets modality quality, cross-modal alignment and temporal synchronisation, but contains no fine-grained term for physical plausibility of transient, high-dynamic, multi-subject events. The ceiling of post-training is therefore set by the base backbone's temporal prior—specifically, the difficulty of jointly modelling small, high-frequency objects and fast inter-limb exchanges in a few-step video diffusion.

Closing this gap likely requires two complementary changes, neither of which is a routing modification: (i) longer-frame or hierarchical temporal models (e.g. video diffusion with longer or multi-scale temporal receptive fields, or latent flow/explicit motion-token designs) that can carry a small fast object across many denoising steps; and (ii) targeted SFT data covering high-speed, multi-subject interactions (sports rallies, dance, hand-offs) so that the backbone itself is exposed to this regime before reward routing is applied.

J Generalization to Unified Single-Stream Backbone

J.1 Adapting the routing signal

Our method does not fundamentally require two physically separate Transformer streams; it requires only identifiable audio and video token sets and an interaction operator whose directional cross-modal contribution can be isolated. JavisDiT++ (Liu et al., 2026d) provides this structure in a unified backbone. It concatenates flattened video and audio tokens, $X = [X_v; X_a]$, and processes them with shared full self-attention, followed by modality-specific FFN experts. Because token boundaries are known, the attention matrix of every layer can be partitioned as

$$A^l = \begin{bmatrix} A^l_{vv} & A^l_{va} \\ A^l_{av} & A^l_{aa} \end{bmatrix}, \qquad o^l_{a\to v} = A^l_{va} V^l_a, \quad o^l_{v\to a} = A^l_{av} V^l_v. \tag{44}$$

The two off-diagonal responses provide an architecture-specific single-stream realization of the same directional interaction signal used by the explicit A2V and V2A paths in LTX-2. Their token-wise norms can be aggregated over layers and timesteps exactly as in Eq. 7. For layer routing, we

Table 9: **Generalization to the unified single-stream JavisDiT++ backbone.** (Mean ± std over 3 seeds.)

Method	AV-Quality		Text-Consistency				AV-Consistency		AV-Synchrony	
	VQ↑	AQ↑	TV-IB↑	TA-IB↑	CLIP↑	CLAP↑	AV-IB↑	AVHScore↑	JavisScore↑	DeSync↓
Base Model	1.462	5.049	0.282	0.164	0.316	0.424	0.198	0.184	0.159	0.832
+ GDPO	2.237 ±0.0145	5.183 ±0.0231	0.279 ±0.0022	0.159 ±0.0020	0.314 ±0.0015	0.413 ±0.0034	0.194 ±0.0028	0.181 ±0.0026	0.160 ±0.0023	0.886 ±0.0124
+ MARBLE	1.836 ±0.0133	5.026 ±0.0210	0.282 ±0.0019	0.161 ±0.0018	0.315 ±0.0013	0.429 ±0.0030	0.201 ±0.0025	0.190 ±0.0024	0.164 ±0.0021	0.806 ±0.0101
+ Ours	**2.318** ±0.0098	**5.437** ±0.0164	**0.284** ±0.0014	**0.173** ±0.0013	**0.317** ±0.0010	**0.448** ±0.0021	**0.220** ±0.0018	**0.214** ±0.0019	**0.184** ±0.0016	**0.627** ±0.0069

apply an architecture-specific response-level adaptation of the soft-routing principle to the isolated off-diagonal response,

$$\widetilde{o}^{l}_{\bar{m}\to m} = \alpha^{l}_{m}\,\mathrm{sg}(o^{l}_{\bar{m}\to m}) + (1-\alpha^{l}_{m})o^{l}_{\bar{m}\to m}, \tag{45}$$

while leaving the diagonal within-modality response unchanged. This preserves the complete forward attention output and scales only the backward gradient carried by the corresponding cross-modal submatrix. It avoids globally detaching shared K/V tensors, which would incorrectly modify within-modality paths in a unified attention block.

J.2 Adapting reward coordination

The reweighting component is architecture-independent because it operates on reward-wise gradients rather than on a particular attention implementation. In the unified backbone, video and audio token masks define two target-modality losses: visual rewards are probed through the video-token loss, audio rewards through the audio-token loss, and cross-modal rewards through both. JavisDiT++ further provides modality-specific FFN experts, but separate parameter branches are not required; gradients through the shared attention remain part of the corresponding token-conditioned probe. We apply the same preference prior, branch-conditioned MARBLE residual, warm-up, EMA smoothing, and refresh schedule as in the dual-stream experiments.

J.3 Experimental protocol

We initialize all post-training methods from the same released JavisDiT++ checkpoint and use the same 19,487 training prompts, five reward models, LoRA parameterization, rollout group size, and optimization budget. GDPO, MARBLE, and Ours are independently trained with three random seeds and evaluated on the full JavisBench under its four-second, 24-FPS, 16-kHz protocol. The published JavisDiT++ checkpoint is reported as the base model. Tab. 9 follows the same ten-metric evaluation used in the main paper. Note that OmniNFT's analysis of LTX, particularly its layer- and token-level analysis, does not generalize to other architectures.

J.4 Analysis

As shown in Tab. 9, GDPO achieves the second-best visual quality but degrades several consistency and synchronization metrics, whereas MARBLE preserves text–video consistency but provides limited gains elsewhere. Ours achieves the best result, with the clearest improvements in audio quality, AV consistency, and synchronization. These results support that Adaptive Reward Routing depends on identifiable cross-modal interaction paths rather than a particular dual-stream parameterization.