IMPROVING TAIL-CLASS DIVERSITY IN DIFFUSION MODELS VIA TRAINING-FREE SNR-ADAPTIVE BLOCK ROUTING

Anonymous authors
Paper under double-blind review

ABSTRACT

Although diffusion models perform strongly in image generation, their performance degrades on the class-imbalanced datasets common in real applications. Tail classes suffer from lower generation quality and intra-class diversity, and their generated distributions can concentrate on a few common modes. We study how to improve the coverage of real intra-class modes in a class-conditional U-Net diffusion model already trained on long-tailed data while keeping all model parameters frozen. Our analysis shows that lower training frequency is associated with stronger local class-conditional pressure and narrower representational coverage. Local interventions further reveal that class conditioning can affect denoising and final distributional coverage differently, and that moderate modulation in low-resolution encoder blocks can improve tail-class coverage. Based on these findings, we propose SNR-Adaptive Block Routing (SABR), a training-free method that modulates local conditioning strength during sampling according to internal conditional responses and the signal-to-noise ratio. Experiments across multiple long-tailed distributions and generative baselines show that SABR improves generation quality in most evaluated settings, mitigates tail-class mode concentration, and increases coverage of real intra-class modes.

1 INTRODUCTION

Diffusion models generate high-quality images through iterative denoising (Sohl-Dickstein et al., 2015; Ho et al., 2020; Song et al., 2021a;b), yet long-tailed data challenge conditional generation (Liu et al., 2019; Van Horn et al., 2018; Qin et al., 2023; Zhang et al., 2024). Limited tail-class observations restrict within-class variation, concentrating generated samples on a few modes even when classes are sampled equally at inference. Improving long-tailed generation therefore requires both image quality and coverage of real intra-class variation.

Long-tailed recognition uses rebalancing and adaptation (Cao et al., 2019; Cui et al., 2019; Hong et al., 2024; Shi et al., 2024; Wei et al., 2024; Luo et al., 2025), while GAN-based approaches target tail-class mode collapse during training (Rangwani et al., 2022; 2023). For diffusion models, CBDM adds distribution-adjustment regularization (Qin et al., 2023), OC transfers head-class knowledge to tail classes (Zhang et al., 2024), and CM reserves capacity for minority classes (Hong et al., 2026). Other approaches use importance reweighting, contrastive overlap optimization, or mutual-learning regularization (Kim et al., 2024; Yan et al., 2024; Das et al., 2025). Complementing these training-stage methods, we ask whether internal conditional responses can improve tail-class coverage through sampling-time intervention, without updating model parameters.

Our analysis reveals different effects of class conditioning on denoising and generative coverage. We define *conditional pressure* as the normalized magnitude of class-induced feature change under matched block inputs. Our empirical analysis shows that tail classes exhibit higher conditional pressure, with the largest head–tail gap in the low-resolution encoder region, Down-Late. Controlled frequency reassignment further links this gap to training frequency rather than class semantics alone. Removing Down-Late conditioning increases denoising error but improves tail-class Recall and FID while reducing Precision. This coverage–fidelity trade-off motivates selective modulation based on current conditional responses and reference trajectories.

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.

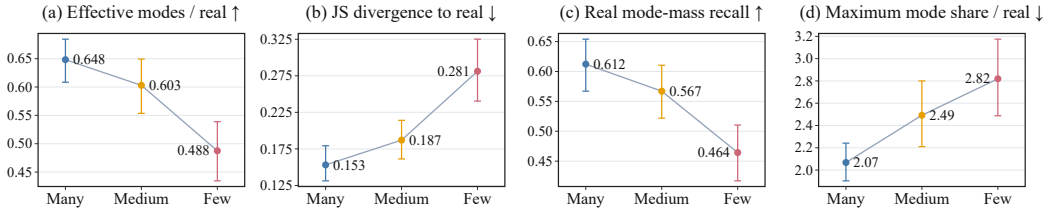


Figure 1: Intra-class mode coverage in long-tailed conditional diffusion. A DDPM is evaluated on CIFAR-100-LT using DDIM50, with 500 images generated per class. The four panels show the effective mode ratio, JS divergence, Mode-mass Recall, and the largest mode mass ratio.

We propose training-free SNR-Adaptive Block Routing (SABR). Fully conditioned trajectories from data-rich classes provide a frozen reference. At each step, accumulated squared pressure is compared with its reference budget, and normalized pressure ranks candidate blocks. Class frequency and signal-to-noise ratio (SNR) determine correction availability; remaining excess determines gates on selected conditional residuals. Model parameters and the sampler update rule remain unchanged, and no router is trained.

Our contributions are threefold:

- **A new perspective.** To our knowledge, we provide the first systematic characterization of training-frequency-related differences in local conditional pressure in long-tailed diffusion models. We identify stronger local conditional pressure in tail classes and link it to training frequency through controlled frequency reassignment. Local interventions further reveal distinct effects of conditioning on denoising and coverage, motivating selective modulation.
- **A new method.** We propose SABR, a training-free adaptive block-routing method. SABR uses frozen references to identify excess conditional pressure during sampling and adaptively selects local blocks and modulates their conditional residuals based on the current sample's pressure distribution and class frequency. Unlike fixed attenuation strategies, SABR adjusts where and how strongly to intervene as generation evolves, improving tail-class coverage while keeping model parameters frozen.
- **Broad empirical evaluation.** Across long-tailed datasets and baselines, SABR improves generation quality in most settings. Real-mode and semantic analyses on CIFAR-100-LT additionally demonstrate reduced tail-class concentration and improved intra-class coverage.

2 Insufficient Tail-Class Coverage in Long-Tailed Diffusion Models

2.1 Problem Formulation

Let $F_\theta(x_t, t, c)$ be a denoiser trained on long-tailed data with N_c examples of class c. The forward process is $x_t = \sqrt{\bar{\alpha}_t}x_0 + \sqrt{1-\bar{\alpha}_t}\epsilon$, where $\epsilon \sim \mathcal{N}(0, I)$ and $\bar{\alpha}_t = \prod_{s=1}^{t}(1-\beta_s)$. Reverse-sampling index k increases as diffusion timestep t_k decreases. We aim to improve tail-class intra-class coverage by modulating local conditioning during sampling, with θ frozen.

2.2 Insufficient Intra-Class Mode Coverage in Tail-Class Generation

We construct class-wise modes by clustering Inception features (Szegedy et al., 2016) from balanced real images, then assign DDPM-generated samples to these fixed modes. Effective mode ratio, JS divergence, Mode-mass Recall, and largest mode mass ratio measure diversity, distribution matching, real-probability-mass coverage, and concentration, respectively. Appendix A.1 provides their definitions, aggregation, and statistical procedures.

Figure 1 shows that effective mode ratio and Mode-mass Recall decrease from Many to Few, while JS divergence and largest mode mass ratio increase. All four trends are statistically significant:

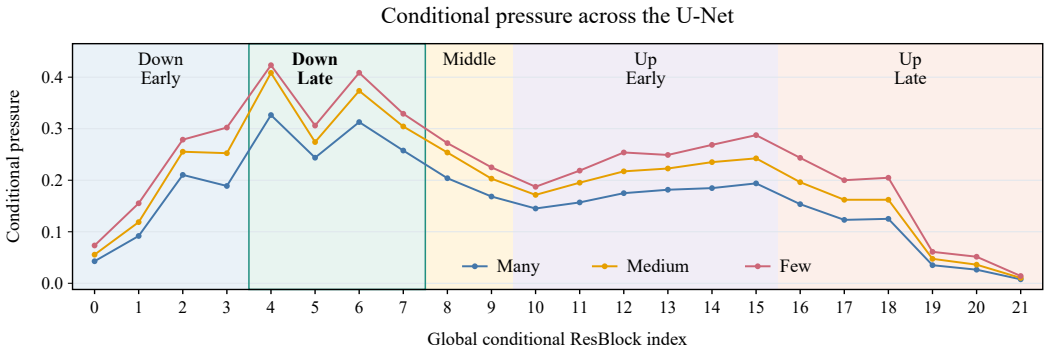


Figure 2: Distribution of conditional pressure across the U-Net and localization of the low-resolution encoding region. The figure shows the average pressure of the 22 residual blocks for the Many, Medium, and Few groups; background shading indicates the five network regions.

lower training frequency is associated with poorer real-mode coverage and greater concentration on a few modes.

2.3 Network Localization of Abnormal Conditional Responses in Tail Classes

To localize conditional responses, we hold the input $h_{l,t}$ to residual block l fixed and compare its outputs with time conditioning alone and with both time and class conditioning:

$$\begin{aligned} h^0_{l,t} &= R_l(h_{l,t}; e_t, 0; \theta_l), \\ h^c_{l,t} &= R_l(h_{l,t}; e_t, e_c; \theta_l), \\ p_l(x_t, t, c) &= \frac{\mathrm{RMS}\left(h^c_{l,t} - h^0_{l,t}\right)}{\mathrm{RMS}\left(h^0_{l,t}\right) + \varepsilon}. \end{aligned} \tag{1}$$

Here, R_l is the residual block, e_t and e_c are time and class embeddings, and $\varepsilon = 10^{-8}$. Conditional pressure p_l is the magnitude of the direct class-induced feature change, normalized by the time-conditioned output magnitude. The fixed input may already contain upstream class information; this comparison isolates the direct class-conditioning effect added by the current block.

The 22 blocks are partitioned by path and resolution, independently of pressure: high-/low-resolution encoding (Down-Early/Down-Late), bottleneck (Middle), and low-/high-resolution decoding (Up-Early/Up-Late).

Tail-class pressure exceeds head-class pressure in all five regions (Figure 2). Down-Late has the largest Few–Many gap (0.08154). Its mean pressure is strongly negatively correlated with class training frequency (Spearman $\rho = -0.8728$): less frequent classes tend to have higher conditional pressure. We therefore use residual blocks in this low-resolution encoding region as the candidate set.

2.4 Effect of Training Frequency on Conditional Response and Coverage Contraction

To separate training frequency from fixed class semantics, we compare Original, Reverse, and Permuted class–frequency assignments on CIFAR-100-LT, preserving the class set, count multiset, and dataset size. High Down-Late pressure follows the reassigned low-frequency classes (Figure 3). Controlling for class and assignment effects, lower frequency is associated with higher pressure and narrower internal and output representation spectra. Appendix E.1 reports the protocols and regression results (Figures E1–E2).

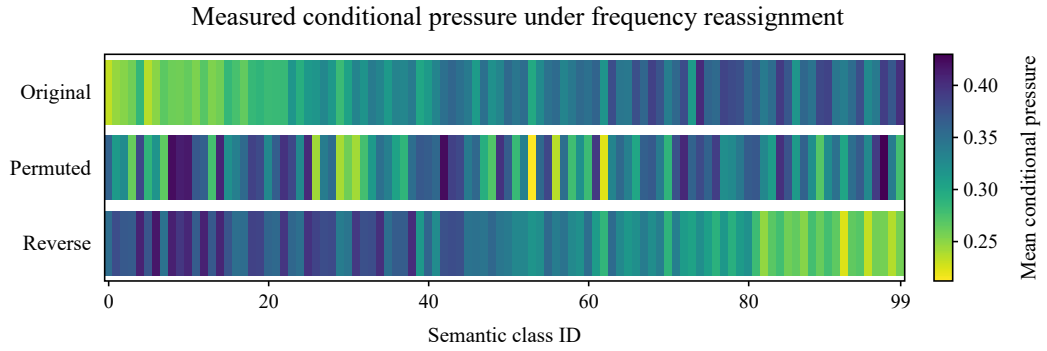


Figure 3: Measured conditional pressure under training-frequency reassignment. The horizontal axis denotes semantic class ID, and color indicates average Down-Late pressure, with darker purple representing higher pressure. Original decreases from 500 to 5 images per class; Reverse increases from 5 to 500; Permuted randomly assigns the same count multiset. All schemes share the total sample size and training protocol.



Figure 4: Relationship between the denoising objective and generative coverage. The panels show the denoising-MSE increase caused by removing the Down-Late residual and the paired changes in Few-class FID, Recall, and Precision. Fixed release reduces FID and raises Recall while lowering Precision. Groups are Many 0–32, Medium 33–65, and Few 66–99.

2.5 Difference Between the Denoising Objective and Final Distribution Coverage

To test conditioning in Down-Late, define the direct residual $\Delta h^c_{l,t} = h^c_{l,t} - h^0_{l,t}$, and intervene using a scale g: $h_{l,t}(g) = h^0_{l,t} + g\Delta h^c_{l,t}$. Here, $g = 1$ retains full conditioning and $g = 0$ removes the current block's direct class-conditional residual.

Holding noisy inputs, timesteps, and noise targets fixed, removing Down-Late conditioning ($g = 0$ versus $g = 1$) increases Few-class denoising MSE by 18.3078%. Yet in paired DDIM50 generation for Few classes 66–99, the same release reduces FID from 31.525 to 23.881 and increases Recall from 27.86% to 42.99%, while Precision falls from 75.11% to 69.01% (Figure 4). Thus, the denoising and coverage responses differ, and fixed release trades fidelity for coverage.

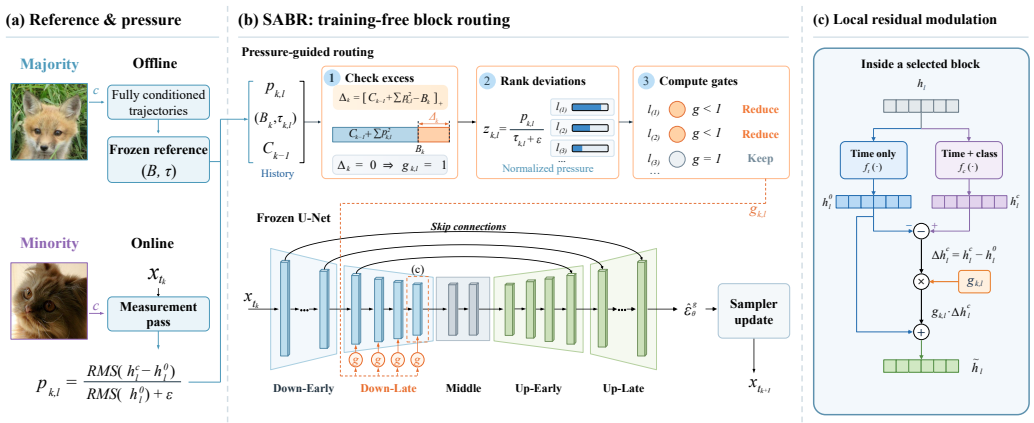


Figure 5: Overview of SABR. Fully conditioned data-rich-class trajectories provide a frozen cumulative squared-pressure budget B and block scales τ. Online, matched-input pressure measurements determine budget excess and block ranking. Block budget, class group, and SNR control correction availability; closed-form gates modulate conditional residuals in a separate forward pass from the same sampling state. The ledger accumulates squared products of measured pressures and gates. No model parameters are updated and no router is trained.

This trade-off motivates SABR's selective modulation according to the current conditional response and sampling state.

3 Method

3.1 Gating and Frozen References

SABR uses a frozen pressure reference to determine whether modulation is needed, normalized pressure to prioritize blocks, and the remaining excess to compute gate values (Figure 5). Model parameters remain frozen.

The main model uses Down-Late residual blocks $\mathcal{L} = \{4, 5, 6, 7\}$, indexed from zero over residual blocks only. Other U-Nets map candidates by resolution structure and conditioning locations.

Measurement pass. At step k, a fully conditioned pass through the frozen encoder measures $p_{k,l} = p_l(x_{t_k}, t_k, c)$ on matched block inputs. Only these pressures inform gate computation; the gated pass restarts from the same x_{t_k}.

Reference from data-rich classes. Let $\mathcal{H}$ contain fully conditioned calibration trajectories from data-rich classes. The cumulative squared pressure of trajectory i through step k is

$$E_i(k) = \sum_{j=1}^{k} \sum_{l \in \mathcal{L}} \left(p_{j,l}^{(i)}\right)^2. \tag{2}$$

With median operator M, define budget $B_k = M(\{E_i(k)\}_{i \in \mathcal{H}})$ and block scale $\tau_{k,l} = M(\{p_{k,l}^{(i)}\}_{i \in \mathcal{H}})$. These control routing activation and normalize block deviations, respectively. References match the model, candidate set, and sampling schedule and remain frozen online.

3.2 Sparse Block Routing Based on Pressure Deviations

Let C_{k-1} accumulate squared products of gates and measured pressures over candidate blocks and preceding steps, with $C_0 = 0$. SABR compares this history plus the current ungated squared pres-

sures with B_k:

$$\Delta_k = \max\left(C_{k-1} + \sum_{l\in\mathcal{L}} p_{k,l}^2 - B_k,\ 0\right). \tag{3}$$

When $\Delta_k = 0$, conditioning is unchanged. Otherwise, blocks are ranked in descending order of $z_{k,l} = p_{k,l}/(\tau_{k,l} + \varepsilon)$, prioritizing larger deviations from their references.

The effective block budget κ_B gives rank r the capacity weight $a_r(\kappa_B) = \mathrm{clip}(\kappa_B - r + 1, 0, 1)$: its integer part permits full correction in that many blocks, while its fractional part controls the next-ranked block. Order the class groups as Few, Medium, and Many, with $j(c) \in \{1, 2, 3\}$. The group activation weight is $w_c(\kappa_G) = \mathrm{clip}(\kappa_G - j(c) + 1, 0, 1)$ for $\kappa_G \in [1, 3]$. Define the basic target set $\mathcal{T}_0 = \text{Medium} \cup \text{Few}$ and first-rank weight $b_c(\kappa_G) = \max(\mathbf{1}[c \in \mathcal{T}_0], w_c(\kappa_G))$.

At step k, $\mathrm{SNR}(t_k) = \bar{\alpha}_{t_k}/(1 - \bar{\alpha}_{t_k})$ and $I_k = \mathbf{1}[\mathrm{SNR}(t_k) \geq \gamma_{\mathrm{SNR}}]$. Block budget, class group, and SNR jointly constrain the available correction strength:

$$s_{k,r,c} = a_r(\kappa_B) \begin{cases} b_c(\kappa_G), & r = 1, \\ w_c(\kappa_G) I_k, & r \geq 2. \end{cases} \tag{4}$$

At the default $\kappa_G = 1$, Medium and Few permit first-ranked-block correction without an SNR restriction; only Few permits additional blocks with sufficient SNR. Many remains unchanged. All correction requires pressure excess. Appendix B.1 details group extensions.

3.3 Continuous Gates and the Gated Forward Pass

At step k, write $q_l = p_{k,l}^2$ and initialize the remaining excess as $d_1 = \Delta_k$. The target gate removes the remaining excess up to the block's measured squared pressure; the operational gate scales the target attenuation amplitude by the available strength. For block $l = l_{(r)}$,

$$g_l^* = \begin{cases} \sqrt{1 - \dfrac{\min(d_r, q_l)}{q_l}}, & q_l > \varepsilon, \\ 1, & q_l \leq \varepsilon, \end{cases} \qquad g_l = 1 - s_l(1 - g_l^*), \quad s_l = s_{k,r,c}. \tag{5}$$

Here $\varepsilon = 10^{-8}$. Each block's achieved reduction is subtracted from the remaining excess, clipped at zero; limited capacity may leave excess (Algorithm A.1, Appendix A.5).

The actual forward pass restarts from x_{t_k} and applies each gate to the direct class-conditional residual at that block's actual input:

$$\widetilde{h}_l = h_l^0 + g_l \left(h_l^c - h_l^0\right). \tag{6}$$

Unselected blocks use $g_l = 1$. The ledger adds $\sum_{l\in\mathcal{L}} g_l^2 p_{k,l}^2$ using measured, not remeasured post-gating, pressures. The unconditional branch and sampler update rule remain unchanged.

Theoretical analysis of pressure guidance. **Proposition 3.1 (selective modulation and pressure ranking).** Under Assumption D.1 in Appendix D, the local conditional response decomposes into orthogonal normal-signal and excess-bias components. The associated positively weighted quadratic distortion has a unique optimal gate, strictly between zero and one for positive bias and equal to one without bias. For small squared-pressure reductions, bias suppression provides a first-order benefit while signal attenuation incurs a second-order cost. If the frozen reference equals the normal-signal scale and bias-response weights are shared across blocks, normalized pressure strictly ranks initial benefits. Appendices D.1 and D.2 give the expressions and proof; Theorem D.2 bounds reference mismatch and finite-budget effects.

Under the mode-readout assumptions and perturbation bounds, Theorem D.3 and Corollary D.4 give sufficient conditions for retaining existing modes, covering additional modes, and reducing JS divergence.

Relation to sampling-time control. Sampling controls modify guidance (Dhariwal and Nichol, 2021; Kynkäänniemi et al., 2024), U-Net features (Si et al., 2024), or attention (Hong et al., 2023; Ahn et al., 2024). CADS (Sadat et al., 2024) anneals conditioning-vector noise for diversity. SABR instead links local conditional pressure to training frequency and modulates residuals using current-sample measurements (Appendix B.2).

4 EXPERIMENTS

4.1 EXPERIMENTAL SETUP

Datasets. We evaluate CIFAR-10/100-LT (Krizhevsky, 2009) at 32×32, ImageNet-LT (Liu et al., 2019) and iNaturalist2018-LT (Van Horn et al., 2018) at 32×32 and 64×64, and ArtBench-10 (Liao et al., 2022) at 256×256. CIFAR and ArtBench use exponential subsampling (Cao et al., 2019) with $\mathrm{IR} = \max_c N_c / \min_c N_c = 100$; ImageNet-LT and iNaturalist retain their benchmark distributions. CIFAR frequency groups follow CM (Hong et al., 2026). Appendix A.2 gives dataset statistics, group definitions, and ArtBench training settings.

Base generative models. Fully conditioned *Baseline* sampling and SABR use identical frozen DDPM (Ho et al., 2020), OC (Zhang et al., 2024), and CBDM (Qin et al., 2023) models (Appendix C); only sampling-time routing differs.

Evaluation metrics. We report FID (Heusel et al., 2017), KID (Bińkowski et al., 2018), IS (Salimans et al., 2016), and Precision/Recall (Kynkäänniemi et al., 2019), with matched real references and evaluation procedures for CIFAR and ArtBench pairs; arrows indicate preferred directions.

Implementation details and hyperparameter settings are provided in Appendix A.2.

4.2 MAIN RESULTS

CIFAR-10-LT and CIFAR-100-LT. SABR improves FID and Recall in all six base-model/dataset pairs (Table 1). For DDPM on CIFAR-100-LT, FID falls from 14.227 to 12.977 and Recall rises from 41.67% to 43.66%. Across all 30 metrics, there are 25 improvements, one displayed tie, and four declines. Group-wise results (Table 2) show improved Few FID in all six pairs and mixed Medium-group responses. Under the main experimental settings, SABR modulates conditioning only for the Medium and Few classes; Many classes retain the original sampling path, so their FID remains unchanged.

Although tail-class pressure remains higher than head-class pressure in CBDM's Down-Late region on CIFAR-100-LT, group-wise pressures and the absolute head–tail gap are lower than in DDPM (Appendix E.4, Table E6). This may limit the benefits of further modulation in this region, partially explaining the limited gains and even deterioration in some metrics observed with SABR in certain CBDM configurations.

Large-scale long-tailed datasets. Across both resolutions, FID improves for DDPM and OC on ImageNet-LT and all three models on iNaturalist2018-LT (Table 3). The benefit is not universal: CBDM's ImageNet-LT FID increases slightly at both resolutions. Thus, transfer across datasets depends partly on the base model.

Artistic image generation. ArtBench uses LoRA-fine-tuned miniSD (Rombach et al., 2022; Hu et al., 2022) with DDIM50; SABR modulates low-resolution encoder text-conditioning residuals without further training. Training settings are in Appendix A.2.

All five metrics improve for DDPM and OC (Table 4); OC FID falls from 15.037 to 13.350. CBDM gains Recall (0.59 to 0.60) but worsens on the other metrics, illustrating that increased coverage need not improve overall generation quality.

Appendix F provides qualitative comparisons for Baroque and Post-Impressionism (Figures F1 and F2) and additional ArtBench samples (Figure F4).

4.3 TAIL-CLASS MODE COVERAGE AND SEMANTIC RETENTION

Real-mode coverage. For the Few classes (66–99) in Section 2.2, SABR improves all four real-mode metrics (Figure 6): effective mode ratio rises from 0.4875 to 0.5249, JS divergence falls from 0.2812 to 0.2535, Mode-mass Recall increases, and maximum concentration decreases. These gains indicate improved coverage of real intra-class modes.

Table 1: Overall generation performance on CIFAR-10-LT and CIFAR-100-LT. The table reports mean values; means and standard deviations are provided in Table E1 in Appendix E. Parenthesized SABR denotes our method applied to the corresponding base model. Bold indicates the better value within each pair; displayed ties are both bold. SABR rows are shaded in light red.

Dataset	Method	FID ↓	KID ↓	Precision ↑	Recall ↑	IS ↑
CIFAR-10-LT	DDPM	15.342	**0.0047**	0.8480	0.4326	**9.42**
	DDPM (SABR)	**14.825**	**0.0047**	**0.8541**	**0.4429**	9.40
	OC	11.802	0.0036	0.8288	0.5227	9.63
	OC (SABR)	**10.959**	**0.0033**	**0.8310**	**0.5450**	**9.66**
	CBDM	8.466	0.0018	0.8227	0.6013	**9.68**
	CBDM (SABR)	**7.558**	**0.0017**	**0.8230**	**0.6178**	9.65
CIFAR-100-LT	DDPM	14.227	0.0037	0.8020	0.4167	13.38
	DDPM (SABR)	**12.977**	**0.0034**	**0.8055**	**0.4366**	**13.63**
	OC	9.256	0.0030	0.7686	0.5116	13.45
	OC (SABR)	**8.695**	**0.0025**	**0.7705**	**0.5333**	**13.50**
	CBDM	7.339	**0.0014**	0.7707	0.5426	**13.42**
	CBDM (SABR)	**7.276**	0.0015	**0.7720**	**0.5533**	13.27

Table 2: FID results on the Many, Medium, and Few class groups of CIFAR-10-LT and CIFAR-100-LT. Overall FID is included for reference. Bold indicates the better value within each pair; displayed ties are both bold. SABR rows are shaded in light red.

Dataset	Method	Many FID ↓	Med. FID ↓	Few FID ↓	Overall FID ↓
CIFAR-10-LT	DDPM	**14.687**	**27.258**	25.155	15.342
	DDPM (SABR)	**14.687**	27.364	**22.780**	**14.825**
	OC	**13.576**	**19.936**	20.212	11.802
	OC (SABR)	**13.576**	20.067	**17.013**	**10.959**
	CBDM	**9.795**	14.895	16.396	8.466
	CBDM (SABR)	**9.795**	**14.859**	**12.673**	**7.558**
CIFAR-100-LT	DDPM	**18.950**	22.217	31.525	14.227
	DDPM (SABR)	**18.950**	**20.484**	**27.017**	**12.977**
	OC	**13.362**	15.032	21.510	9.256
	OC (SABR)	**13.362**	**14.715**	**19.493**	**8.695**
	CBDM	**10.700**	12.899	17.845	7.339
	CBDM (SABR)	**10.700**	**12.813**	**17.062**	**7.276**

Semantic retention. Semantic retention increases from 52.85% to 53.55%, accompanied by a higher semantic margin and a lower target-centroid distance (Appendix E.2, Table E5).

4.4 Necessity of Adaptive Routing

No tested fixed gate matches SABR on both Few FID and Precision (Appendix E.3). Under the evaluated configuration, SABR outperforms CADS in mean FID, KID, IS, Precision, and Recall for Overall, Medium, and Few (Appendix E.5). With gates, timing, and active-block count fixed, pressure-guided assignment improves Few FID and Recall over random assignment (Table E4). Against frozen group–step–block mean gates, SABR improves Few FID (28.836 → 27.017) and Recall (30.89% → 32.96%) at comparable Precision (Appendix E.6).

4.5 Robustness and Inference Cost

Sensitivity and generalization. On CIFAR-100-LT, SABR lowers overall FID at all tested DDIM step counts (50, 100, 200, and 500) and across three U-Net configurations, each with matched candidate blocks and frozen references. Appendix A.6 includes these results and SNR-threshold

Table 3: FID (↓) and KID (↓) on ImageNet-LT and iNaturalist2018-LT at 32×32 and 64×64 resolutions. KID is unscaled. Bold indicates the better value within each baseline–SABR pair; displayed ties are both bold. SABR rows are shaded in light red.

Method	ImageNet-LT 32×32		ImageNet-LT 64×64		iNaturalist 32×32		iNaturalist 64×64	
	FID ↓	KID ↓	FID ↓	KID ↓	FID ↓	KID ↓	FID ↓	KID ↓
DDPM	15.681	0.0062	10.042	0.0030	16.807	0.0071	9.263	0.0039
DDPM(SABR)	**15.106**	**0.0061**	**9.345**	**0.0025**	**16.085**	**0.0068**	**8.618**	**0.0035**
OC	12.998	0.0052	8.734	0.0023	13.141	0.0053	7.343	0.0027
OC(SABR)	**12.278**	**0.0048**	**8.235**	**0.0021**	**12.878**	**0.0051**	**6.780**	**0.0023**
CBDM	**12.879**	**0.0053**	**7.350**	**0.0025**	10.819	**0.0036**	6.273	0.0019
CBDM(SABR)	12.886	**0.0053**	7.360	**0.0025**	**10.783**	**0.0036**	**6.235**	**0.0018**

Table 4: Overall generation performance on imbalanced ArtBench-10. The table reports means; means and standard deviations are provided in Table E2 in Appendix E. Precision and Recall are proportions on a 0–1 scale. Bold indicates the better value within each pair.

Method	FID ↓	KID ↓	Precision ↑	Recall ↑	IS ↑
DDPM	18.189	0.0070	0.661	0.51	8.46
DDPM (SABR)	**16.797**	**0.0062**	**0.662**	**0.53**	**8.49**
OC	15.037	0.0064	0.682	0.50	7.49
OC (SABR)	**13.350**	**0.0054**	**0.685**	**0.52**	**7.62**
CBDM	**10.588**	**0.0059**	**0.669**	0.59	**6.71**
CBDM (SABR)	10.893	0.0061	0.662	**0.60**	6.67

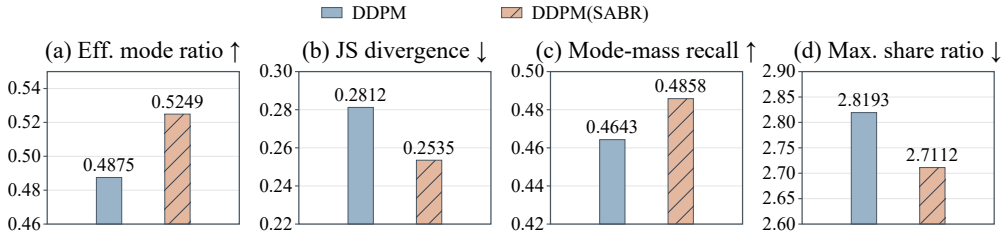


Figure 6: Real-mode coverage for Few classes 66–99. Panels show effective mode ratio, JS divergence, Mode-mass Recall, and largest mode mass ratio. Raw values appear above bars; y-axis ranges are adjusted by panel to show the differences.

and block-budget sweeps (Figure A1); Appendix E reports overall and tail-group results with 25-step DPM-Solver (Table E3).

Inference cost. Under DDIM50, latency per 100 images rises from 2.7823 to 3.3022 seconds (18.68%), throughput falls by 15.74%, and peak memory increases by approximately 4.50 MiB. Appendix A.3 gives the protocol and measurements (Table A2).

5 Conclusion

Tail classes exhibit stronger local conditional pressure, linked to training frequency by controlled reassignment. Local interventions reveal distinct effects on denoising and coverage. SABR selectively modulates conditioning using pressure, class frequency, and SNR without retraining. It improves generation quality in most settings and DDPM tail-class real-mode coverage on CIFAR-100-LT. Our analysis characterizes initial pressure-ranking benefits and conditions for preserving mode coverage and reducing JS divergence. These findings support sampling-time conditioning control as a complement to long-tailed training.

Ethics Statement

SABR aims to improve tail-class generation and intra-class diversity in diffusion models trained on imbalanced data. Better coverage of underrepresented classes may support more representative synthetic datasets and benefit applications with uneven data availability. However, improved class-level coverage does not by itself establish fairness across demographic or social groups. Since SABR operates on frozen pretrained models, it does not directly correct biases or privacy risks inherited from their training data. As with other image-generation methods, enhanced generation capabilities may be misused to produce misleading content. Moreover, improving the representation of underrepresented classes requires attention to the reliability of the underlying data, as biased or manipulated data may introduce harmful patterns into generated outputs. Responsible use therefore calls for appropriate data monitoring, validation, and governance, compliance with applicable data licenses and privacy requirements, and clear identification of synthetic content where it could be mistaken for authentic imagery.

Reproducibility Statement

Experiments were implemented using Python 3.12.3 and PyTorch 2.7.0 (CUDA 12.8) and conducted on an NVIDIA RTX PRO 6000 Blackwell Server Edition GPU. Experimental settings and main results are reported in Sections 4.1 and 4.2. Appendix A documents the implementation and evaluation protocols, while Appendix D states the assumptions and proofs for the theoretical results. An anonymized implementation will be made available to reviewers and area chairs during the discussion period.

Use of Large Language Models (LLMs)

For transparency, we disclose that GPT-6 was used to assist with coding and manuscript language polishing. It was not involved in generating research ideas, designing research methods, analyzing data, or drawing conclusions. The authors take full responsibility for the code, experimental results, references, and conclusions presented in this work.

References

Donghoon Ahn, Hyoungwon Cho, Jaewon Min, Wooseok Jang, Jungwoo Kim, Seonhwa Kim, Hyunhee Park, Kyonghwan Jin, and Seungryong Kim. Self-rectifying diffusion sampling with perturbed-attention guidance. In *ECCV*, 2024.

Mikołaj Bińkowski, Danica J. Sutherland, Michael Arbel, and Arthur Gretton. Demystifying MMD GANs. In *ICLR*. OpenReview.net, 2018.

Kaidi Cao, Colin Wei, Adrien Gaidon, Nikos Arechiga, and Tengyu Ma. Learning imbalanced datasets with label-distribution-aware margin loss. In *NeurIPS*, 2019.

Yin Cui, Menglin Jia, Tsung-Yi Lin, Yang Song, and Serge Belongie. Class-balanced loss based on effective number of samples. In *CVPR*, pages 9268–9277, 2019.

Pranoy Das, Kexin Fu, Abolfazl Hashemi, and Vijay Gupta. Principled long-tailed generative modeling via diffusion models. In *NeurIPS*, 2025.

Prafulla Dhariwal and Alexander Nichol. Diffusion models beat GANs on image synthesis. In *NeurIPS*, 2021.

Martin Heusel, Hubert Ramsauer, Thomas Unterthiner, Bernhard Nessler, and Sepp Hochreiter. GANs trained by a two time-scale update rule converge to a local Nash equilibrium. In *NeurIPS*, 2017.

Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. In *NeurIPS*, 2020.

Feng Hong, Jiangchao Yao, Yueming Lyu, Zhihan Zhou, Ivor W. Tsang, Ya Zhang, and Yanfeng Wang. On harmonizing implicit subpopulations. In *ICLR*. OpenReview.net, 2024.

Feng Hong, Jiangchao Yao, Yifei Shen, Dongsheng Li, Ya Zhang, and Yanfeng Wang. Improving diffusion models for class-imbalanced training data via capacity manipulation. In *ICLR*. OpenReview.net, 2026.

Susung Hong, Gyuseong Lee, Wooseok Jang, and Seungryong Kim. Improving sample quality of diffusion models using self-attention guidance. In *ICCV*, pages 7462–7471, 2023.

Edward J. Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. LoRA: Low-rank adaptation of large language models. In *ICLR*. OpenReview.net, 2022.

Yeongmin Kim, Byeonghu Na, Minsang Park, JoonHo Jang, Dongjun Kim, Wanmo Kang, and Il-Chul Moon. Training unbiased diffusion models from biased dataset. In *ICLR*. OpenReview.net, 2024.

Alex Krizhevsky. Learning multiple layers of features from tiny images. Technical report, University of Toronto, 2009.

Tuomas Kynkäänniemi, Tero Karras, Samuli Laine, Jaakko Lehtinen, and Timo Aila. Improved precision and recall metric for assessing generative models. In *NeurIPS*, 2019.

Tuomas Kynkäänniemi, Miika Aittala, Tero Karras, Samuli Laine, Timo Aila, and Jaakko Lehtinen. Applying guidance in a limited interval improves sample and distribution quality in diffusion models. In *NeurIPS*, 2024.

Peiyuan Liao, Xiuyu Li, Xihui Liu, and Kurt Keutzer. The ArtBench dataset: Benchmarking generative models with artworks. *CoRR*, abs/2206.11404, 2022.

Ziwei Liu, Zhongqi Miao, Xiaohang Zhan, Jiayun Wang, Boqing Gong, and Stella X. Yu. Large-scale long-tailed recognition in an open world. In *CVPR*, 2019.

Jiaan Luo, Feng Hong, Qiang Hu, Xiaofeng Cao, Feng Liu, and Jiangchao Yao. Long-tailed recognition with model rebalancing. In *NeurIPS*, 2025.

Yiming Qin, Huangjie Zheng, Jiangchao Yao, Mingyuan Zhou, and Ya Zhang. Class-balancing diffusion models. In *CVPR*, pages 18434–18443, 2023.

Harsh Rangwani, Lavish Bansal, Kartik Sharma, Tejan Karmali, Varun Jampani, and R. Venkatesh Babu. NoisyTwins: Class-consistent and diverse image generation through StyleGANs. In *CVPR*, pages 5987–5996, 2023.

Harsh Rangwani, Naman Jaswani, Tejan Karmali, Varun Jampani, and R. Venkatesh Babu. Improving GANs for long-tailed data through group spectral regularization. In *ECCV*, pages 426–442, 2022.

Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *CVPR*, pages 10684–10695, 2022.

Seyedmorteza Sadat, Jakob Buhmann, Derek Bradley, Otmar Hilliges, and Romann Weber. CADS: Unleashing the diversity of diffusion models through condition-annealed sampling. In *ICLR*. OpenReview.net, 2024.

Tim Salimans, Ian Goodfellow, Wojciech Zaremba, Vicki Cheung, Alec Radford, and Xi Chen. Improved techniques for training GANs. In *NeurIPS*, 2016.

Jiang-Xin Shi, Tong Wei, Zhi Zhou, Jie-Jing Shao, Xin-Yan Han, and Yu-Feng Li. Long-tail learning with foundation model: Heavy fine-tuning hurts. In *ICML*, 2024.

Chenyang Si, Ziqi Huang, Yuming Jiang, and Ziwei Liu. FreeU: Free lunch in diffusion U-Net. In *CVPR*, pages 4733–4743, 2024.

Jascha Sohl-Dickstein, Eric Weiss, Niru Maheswaranathan, and Surya Ganguli. Deep unsupervised learning using nonequilibrium thermodynamics. In *ICML*, pages 2256–2265, 2015.

Jiaming Song, Chenlin Meng, and Stefano Ermon. Denoising diffusion implicit models. In *ICLR*. OpenReview.net, 2021a.

Yang Song, Jascha Sohl-Dickstein, Diederik P. Kingma, Abhishek Kumar, Stefano Ermon, and Ben Poole. Score-based generative modeling through stochastic differential equations. In *ICLR*. OpenReview.net, 2021b.

Christian Szegedy, Vincent Vanhoucke, Sergey Ioffe, Jonathon Shlens, and Zbigniew Wojna. Rethinking the Inception architecture for computer vision. In *CVPR*, pages 2818–2826, 2016.

Grant Van Horn, Oisin Mac Aodha, Yang Song, Yin Cui, Chen Sun, Alex Shepard, Hartwig Adam, Pietro Perona, and Serge Belongie. The iNaturalist species classification and detection dataset. In *CVPR*, 2018.

Tong Wei, Zhen Mao, Zi-Hao Zhou, Yuanyu Wan, and Min-Ling Zhang. Learning label shift correction for test-agnostic long-tailed recognition. In *ICML*, 2024.

Divin Yan, Lu Qi, Vincent Tao Hu, Ming-Hsuan Yang, and Meng Tang. Training class-imbalanced diffusion model via overlap optimization. *CoRR*, abs/2402.10821, 2024.

Tianjiao Zhang, Huangjie Zheng, Jiangchao Yao, Xiangfeng Wang, Mingyuan Zhou, Ya Zhang, and Yanfeng Wang. Long-tailed diffusion models with oriented calibration. In *ICLR*. OpenReview.net, 2024.

APPENDIX: IMPROVING TAIL-CLASS DIVERSITY IN DIFFUSION MODELS VIA TRAINING-FREE SNR-ADAPTIVE BLOCK ROUTING

CONTENTS

A Experimental Details **15**

A.1 Metrics and Statistical Procedures 15

A.2 Implementation Details 16

A.3 Inference Cost 16

A.4 Shallow U-Net Evaluation 17

A.5 Algorithm 17

A.6 Sensitivity and Generalization 17

B Method Details **19**

B.1 Group Activation 19

B.2 Comparison with Related Methods 19

C Objectives of CBDM and OC **20**

D Theoretical Analysis **20**

D.1 Local-Response Model 20

D.2 Proof of Proposition 3.1 21

D.3 Robustness to Reference and Response Scales 21

D.4 Mode-Distribution Model 22

D.5 Mode Coverage and JS Improvement 23

E Additional Experiments **24**

E.1 Training-Frequency Reassignment 25

E.2 Semantic Retention 26

E.3 Fixed-Gate Comparison 27

E.4 Cross-Model Pressure 27

E.5 Comparison with Condition-Annealed Sampling 28

E.6 Adaptive versus Frozen Mean Gates 29

F Generated Samples **29**

G Limitations and Future Work **34**

References **34**

A Experimental Details

A.1 Metrics and Statistical Procedures

Real-mode, spectral-diversity, and semantic metrics evaluate different aspects of the generated samples. Real-mode metrics use within-class modes constructed from real images to assess mode coverage and probability allocation. Effective rank (Roy and Vetterli, 2007) and Vendi Score (Friedman and Dieng, 2023) characterize feature diversity, while semantic metrics assess consistency with the target class. Our primary evidence concerns real-mode coverage, complemented by spectral-diversity and semantic analyses.

Mode construction. For each class, we extract 2,048-dimensional Inception pool3 features (Szegedy et al., 2016) from real and generated images and apply L_2 normalization. MiniBatchK-Means (Sculley, 2010) is fitted only to the balanced real images of that class. Generated features are assigned to the nearest real cluster center by Euclidean distance. All methods use the same real-mode definitions.

Probability distributions. Let K denote the number of modes. Suppressing class and clustering-setting subscripts, let p_j and q_j denote the proportions of real and generated samples assigned to mode j. Each distribution is obtained by dividing its mode counts by its own sample total, so $\sum_{j=1}^{K} p_j = \sum_{j=1}^{K} q_j = 1$. Zero-probability terms contribute zero to entropy and divergence.

Effective mode ratio. We first define the effective number of modes as the exponential of entropy and then normalize the generated value by the real value:

$$R_{\text{eff}} = \frac{\exp(H(q))}{\exp(H(p))}, \qquad H(p) = -\sum_{j=1}^{K} p_j \ln p_j. \tag{A.1}$$

Entropy uses the natural logarithm. This metric measures the effective breadth of the mode-probability distribution and is not the fraction of occupied clusters. It can exceed one when the generated distribution is more uniform than the real distribution.

Jensen–Shannon divergence. With $m_j = (p_j + q_j)/2$, we compute (Lin, 1991)

$$D_{\text{JS}}(p, q) = \frac{1}{2} \sum_{j=1}^{K} p_j \log_2 \frac{p_j}{m_j} + \frac{1}{2} \sum_{j=1}^{K} q_j \log_2 \frac{q_j}{m_j}. \tag{A.2}$$

The base-2 divergence lies in $[0, 1]$, with smaller values indicating closer mode distributions. The implementation squares the Jensen–Shannon distance, yielding the divergence above.

Mode-mass Recall. We accumulate the real probability mass of modes for which the generated proportion reaches at least half of the corresponding real proportion:

$$R_{0.5} = \sum_{j=1}^{K} p_j \, \mathbf{1}[q_j \geq 0.5 p_j]. \tag{A.3}$$

The 50% threshold is relative to each mode's real proportion; it does not select the largest half of the modes. This metric is distinct from the aggregate generation Recall reported elsewhere.

Largest-mode mass ratio. We compare the peak probabilities of the generated and real distributions:

$$R_{\max} = \frac{\max_j q_j}{\max_j p_j}. \tag{A.4}$$

The two maxima may correspond to different modes; this is not the maximum of the per-mode probability ratios. It measures peak concentration relative to the real distribution and should be interpreted together with coverage and distribution-matching metrics.

Group aggregation. Within each class, we average metrics equally across clustering settings and repeated evaluations, then average equally across classes in each Many, Medium, or Few group. These groups contain 33, 33, and 34 classes, respectively, on CIFAR-100-LT. We do not pool mode histograms across classes or treat different clustering settings for the same class as independent classes.

Confidence intervals and tests. We construct percentile bootstrap confidence intervals (Efron and Tibshirani, 1993). For Figure 1, classes are resampled with replacement within each group, retaining the original group size. We use 10,000 resamples and the 2.5th and 97.5th percentiles of the mean to obtain 95% confidence intervals. These quantify between-class variation, not the standard deviation across repeated generation runs. Head–tail differences are computed after resampling the Many and Few groups separately. Section 2.2 uses two-sided class-label permutation tests with fixed group sizes and 10,000 permutations; Holm correction (Holm, 1979) is applied across the four metric tests.

A.2 Implementation Details

Base-model training. For CIFAR-100-LT, the diffusion process uses $T = 1000$ timesteps, with noise variance increasing linearly from 0.0001 to 0.02. The DDPM base model is trained using Adam (Kingma and Ba, 2015) with betas $(0.9, 0.999)$, a learning rate of 0.0002, dropout 0.1, batch size 64, and 5,000 learning-rate warm-up steps. Sampling uses its EMA model at training step 300,000.

SABR inference. We apply SABR to pretrained base models and freeze all model parameters during sampling. Unless otherwise specified, $\kappa_B = 2$, $\kappa_G = 1$, and $\gamma_{\mathrm{SNR}} = 1$. SABR is applied only to the low-resolution encoder of the U-Net. Calibration references are computed from data-rich-class trajectories as defined in Section 3.1, constructed separately for the corresponding model, candidate set, and sampling schedule, and kept fixed during inference. Models are in evaluation mode with dropout disabled, and we use 50-step DDIM by default. The generation batch size is 256 for CIFAR-100-LT evaluation and 100 for the overhead measurements in Table A2. Candidate mappings and sensitivity analyses are provided in Appendices A.4 and A.6.

Dataset statistics and class groups. For CIFAR-10, CIFAR-100, and ArtBench-10, class-dependent exponential subsampling gives $\mathrm{IR} = 100$ and 12,406, 10,847, and 12,406 training images, respectively. ImageNet-LT retains its benchmark distribution of 1,000 classes and 115,846 images ($\mathrm{IR} = 256$), and iNaturalist2018-LT contains 8,142 classes and 437,513 images ($\mathrm{IR} = 500$). CIFAR uses 32×32 images, ImageNet-LT and iNaturalist use 32×32 and 64×64, and ArtBench uses 256×256. With classes ordered by decreasing training count, Many, Medium, and Few are classes 0–2, 3–5, and 6–9 for CIFAR-10-LT and ArtBench-10, and 0–32, 33–65, and 66–99 for CIFAR-100-LT.

ArtBench training. We fine-tune pretrained miniSD at 256×256 using rank-128 LoRA and prompts of the form "a {class} painting." LoRA dropout is 0.1. Training uses 100 epochs, batch size 64, and AdamW (Loshchilov and Hutter, 2019) with learning rate 3×10^{-4} and weight decay 10^{-6}. Each Baseline–SABR pair uses the same frozen fine-tuned model with 50-step DDIM. SABR modulates only local text-conditioning residuals in the low-resolution encoder.

A.3 Inference Cost

Many batches follow the original model path, with a measured latency change of 0.29%. Medium and Few batches require an additional measurement pass and local gating, increasing latency by 28.18% and 27.35%, respectively.

We measure full sampling cost using the DDPM EMA300k checkpoint. Each class group receives one warm-up run followed by three timed complete sampling runs, weighted by the 33/33/34 class-group proportions. Model loading, Inception evaluation, and file I/O are excluded. SABR increases the mean time per 100 images from 2.7823 s to 3.3022 s, an 18.68% latency increase and a 15.74% throughput decrease. Peak GPU memory increases by 0.00439 GiB, approximately 4.50 MiB. The

Table A1: Full sampling latency by class group, using the hardware, sampling settings, and timing protocol of Table A2. Changes are relative to the corresponding baseline; the SABR row is shaded in light red.

Method (s/batch)	Many	Medium	Few
Baseline	2.7856	2.7804	2.7810
SABR	2.7937	3.5640	3.5415
Change	+0.29%	+28.18%	+27.35%

Table A2: Full sampling cost under the evaluated protocol. Measurements use an NVIDIA RTX PRO 6000 Blackwell Server, DDIM50, and batch size 100, weighted by class-group proportions.

Method	Seconds/100 ↓	Images/s ↑	Peak memory (GiB) ↓
Baseline	2.7823	35.9409	1.0230
SABR	3.3022	30.2831	1.0274

overhead is concentrated in Medium and Few batches, which require pressure measurement and routing.

A.4 Shallow U-Net Evaluation

The shallow network uses channel multipliers $[1, 2, 2]$, 128 base channels, and two residual blocks per stage. Following its resolution structure, we map the low-resolution encoder candidates to residual blocks $[4, 5]$ at 8×8 resolution and use the default routing parameters in Appendix A.2. This setting tests whether SABR depends on the specific structure of the original 22-block U-Net.

We construct a separate reference using the shallow model's 33 Many classes, collecting four fully conditioned trajectories per class to recompute cumulative squared-pressure budgets and block scales. Paired evaluation uses DDIM50, with 500 images generated per class for each method. The overall, Many, Medium, and Few evaluations contain 50,000, 16,500, 16,500, and 17,000 generated images, respectively.

A.5 Algorithm

Algorithm A.1 summarizes the per-sample procedure in Section 3. Local pressure is measured under matched module inputs as in Section 2.3. Each reference must match the model, candidate set, and sampling schedule.

Here $j(c)$ indexes Few, Medium, and Many as 1, 2, and 3. The first-ranked block has no SNR restriction, and unselected gates remain one. The actual sampling pass does not reuse measurement-pass features. The ledger uses measured pressures and their gates, rather than network pressure remeasured after gating. Model parameters and the original sampler rule remain unchanged.

A.6 Sensitivity and Generalization

We examine the number of sampling steps, the SNR threshold, the effective block budget, and the U-Net configuration on CIFAR-100-LT. Figure A1 summarizes overall FID for the four experiments. Appendix A.4 gives the shallow U-Net mapping and evaluation settings. Table E3 further reports overall and tail-group results with 25-step DPM-Solver.

Sampling steps. We compare DDIM sampling with 50, 100, 200, and 500 steps, using a frozen reference matched to each schedule. SABR lowers overall FID at every tested step count, from 14.227 to 12.977 at 50 steps and from 12.367 to 11.264 at 500 steps. Its benefit is therefore not limited to short sampling schedules.

SNR threshold. For $\gamma_{\text{SNR}} \in \{0, 0.5, 1, 2\}$, overall FID is 13.038, 12.983, 12.977, and 12.914, respectively, all below the original DDPM value of 14.227. Under this budget, delaying the activation

Algorithm A.1 Offline calibration and online sampling with SABR

Input: Frozen model θ; candidates $\mathcal{L}$; schedule $t_1, \ldots, t_K$; data-rich calibration trajectories $\mathcal{H}$; target class c; $\kappa_B, \kappa_G, \gamma_{\text{SNR}}$.

Set $\varepsilon = 10^{-8}$ and $\mathcal{T}_0 = \text{Medium} \cup \text{Few}$; S_k is the original sampler update.

▷ Offline reference construction; frozen model

for each trajectory $i \in \mathcal{H}$ **do**

Sample with full conditioning; record $p_{k,l}^{(i)}$ at each step and candidate block.

$E_i(k) \leftarrow \sum_{j=1}^{k} \sum_{l \in \mathcal{L}} (p_{j,l}^{(i)})^2$ for all k.

end for

$B_k \leftarrow \text{median}_{i \in \mathcal{H}} E_i(k)$; $\tau_{k,l} \leftarrow \text{median}_{i \in \mathcal{H}} p_{k,l}^{(i)}$ for all k, l.

Freeze and save (B, τ).

▷ Online sampling; sample-wise

Sample $x \sim \mathcal{N}(0, I)$; $C_0 \leftarrow 0$.

$w_c \leftarrow \text{clip}(\kappa_G - j(c) + 1, 0, 1)$; $b_c \leftarrow \max(\mathbf{1}[c \in \mathcal{T}_0], w_c)$.

for $k = 1, \ldots, K$ **do**

$g_l \leftarrow 1$ for all $l \in \mathcal{L}$; $C_k \leftarrow C_{k-1}$.

if $b_c > 0$ **then**

Run a fully conditioned encoder measurement pass from x to obtain $p_{k,l}$.

$q_l \leftarrow p_{k,l}^2$; $z_{k,l} \leftarrow p_{k,l} / (\tau_{k,l} + \varepsilon)$ for all l.

$d_1 \leftarrow \max(C_{k-1} + \sum_{l \in \mathcal{L}} q_l - B_k, 0)$.

Sort $z_{k,l}$ in descending order to obtain $l_1, \ldots, l_{|\mathcal{L}|}$.

$I_k \leftarrow \mathbf{1}[\text{SNR}(t_k) \geq \gamma_{\text{SNR}}]$.

for $r = 1, \ldots, |\mathcal{L}|$ **do**

$l \leftarrow l_r$; $a_r \leftarrow \text{clip}(\kappa_B - r + 1, 0, 1)$.

$s_{k,r,c} \leftarrow a_r b_c$ if $r = 1$, else $s_{k,r,c} \leftarrow a_r w_c I_k$.

if $d_r > 0$ and $s_{k,r,c} > 0$ and $q_l > \varepsilon$ **then**

$g_l^\star \leftarrow \sqrt{1 - \min(d_r, q_l)/q_l}$; $g_l \leftarrow 1 - s_{k,r,c}(1 - g_l^\star)$.

end if

$d_{r+1} \leftarrow \max(d_r - (1 - g_l^2) q_l, 0)$.

end for

$C_k \leftarrow C_{k-1} + \sum_{l \in \mathcal{L}} g_l^2 q_l$.

end if

Run one actual sampling pass from x:

if $b_c = 0$, use the original model; otherwise apply

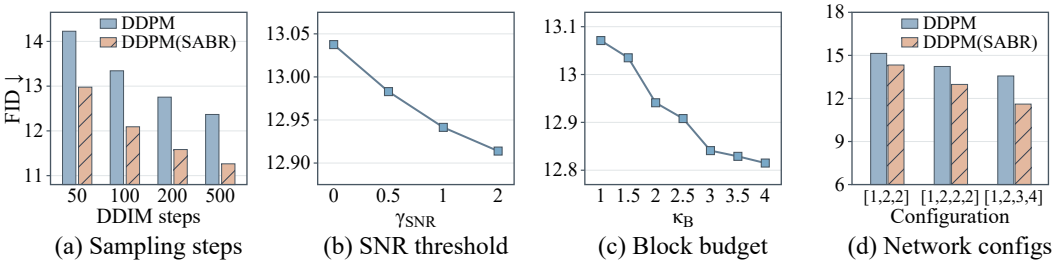
$\widetilde{h}_l = h_l^0 + g_l(h_l^c - h_l^0)$ at each candidate's actual input.

Denote the resulting noise prediction by $\hat{\varepsilon}_\theta^g$.

$x \leftarrow S_k(x, \hat{\varepsilon}_\theta^g)$.

end for

return x.



(a) Sampling steps (b) SNR threshold (c) Block budget (d) Network configs

Figure A1: Sensitivity and cross-architecture performance of SABR on CIFAR-100-LT. Panels report overall FID for (a) DDIM step count, (b) SNR threshold, (c) effective block budget, and (d) three U-Net configurations. Lower is better. Other routing parameters use the defaults in Appendix A.2. Settings on each horizontal axis are equally spaced.

of additional corrections can lower overall FID. We retain $\gamma_{\mathrm{SNR}} = 1$ as the standard operating point because it corresponds to equal signal and noise energy.

Effective block budget. We sweep $\kappa_B \in \{1, 1.5, 2, 2.5, 3, 3.5, 4\}$ with all other hyperparameters fixed. Overall FID decreases from 13.071 to 12.815 as the budget increases, with smaller changes in the higher-budget range. The standard configuration $\kappa_B = 2$ yields 12.977, providing gains with relatively sparse correction support, although it is not the minimum in this sweep.

U-Net architectures. We compare channel-multiplier configurations $[1, 2, 2]$, $[1, 2, 2, 2]$, and $[1, 2, 3, 4]$. SABR reduces overall FID from 15.138, 14.227, and 13.566 to 14.330, 12.977, and 11.607, respectively. All three evaluated configurations improve, indicating that the benefit is not limited to the standard U-Net. Each architecture nevertheless requires an architecture-specific candidate mapping and matched frozen reference.

B Method Details

B.1 Group Activation

This subsection expands the class-group activation parameter used in Section 3.2. Order the groups as Few, Medium, and Many, and let $j(c)$ denote the group index of class c. For $\kappa_G \in [1, 3]$, define

$$w_c(\kappa_G) = \mathrm{clip}\big(\kappa_G - j(c) + 1, 0, 1\big), \qquad b_c(\kappa_G) = \max\big(\mathbf{1}[c \in \mathcal{T}_0], w_c(\kappa_G)\big),$$

where $\mathcal{T}_0 = \mathrm{Medium} \cup \mathrm{Few}$. With $I_k = \mathbf{1}[\mathrm{SNR}(t_k) \geq \gamma_{\mathrm{SNR}}]$, the available correction strength is

$$s_{k,r,c} = a_r(\kappa_B) \begin{cases} b_c(\kappa_G), & r = 1, \\ w_c(\kappa_G) I_k, & r \geq 2. \end{cases} \tag{B.1}$$

When $\kappa_G = 1$, Medium and Few are eligible for base attenuation at the first-ranked block, while additional blocks are opened only to Few. When $\kappa_G = 2$, additional blocks are also opened to Medium. For $2 < \kappa_G \leq 3$, both the base and additional attenuation weights for Many equal $\kappa_G - 2$, with additional blocks still subject to the SNR threshold. For example, at $\kappa_G = 2.5$, the class-group attenuation weight for Many is 0.5. Actual gate values remain jointly determined by block capacity, SNR, and the remaining pressure excess; conditioning is unchanged when there is no excess.

B.2 Comparison with Related Methods

Distinction from training-stage methods. CBDM and OC modify training objectives; SABR instead modulates sampling with the model frozen. We construct a pressure reference for each base model and compare original and SABR sampling using the same configuration of each model.

Distinction from fixed gating. Fixed gating applies the same attenuation coefficient to predefined modules without adapting the intervention to the current sample's pressure deviation. SABR first checks whether cumulative squared pressure exceeds the frozen reference, ranks blocks by relative abnormality, and combines class group, block capacity, and SNR to determine available correction strengths. Gates are computed sequentially from the remaining excess; conditioning is retained when no excess or eligibility exists. The distinction concerns how intervention locations and magnitudes are determined, rather than merely choosing another fixed coefficient. Section 4.4 evaluates these selection mechanisms using fixed-gating and random-assignment controls.

Meaning of routing and deployment. SABR selects local class-conditional residuals within existing U-Net modules rather than separate expert models, and it does not skip entire residual blocks. Gates follow pressure measurements, frozen references, and closed-form rules without training a router or introducing adapter parameters. The offline stage constructs and freezes pressure references without updating the generative model parameters. The online stage performs pressure measurement and local gating; computational costs are reported in Appendix A.3.

C Objectives of CBDM and OC

We use $x_t = \sqrt{\bar{\alpha}_t}x + \sqrt{1-\bar{\alpha}_t}\epsilon$, with $\epsilon \sim \mathcal{N}(0, I)$. Expectations below include training samples, timesteps, noise, and the corresponding reference sampling.

CBDM. Qin et al. (2023) add distribution-adjustment regularization to conditional denoising. Writing $e_c = \epsilon_\theta(x_t, t, c)$, the objective is

$$\mathcal{L}_{\text{CBDM}} = \mathbb{E}\Bigg[\|e_c - \epsilon\|_2^2 + \frac{\tau t}{|\mathcal{Y}|}\sum_{c'\in\mathcal{Y}}\Big(\|e_c - \text{sg}(e_{c'})\|_2^2 + \gamma\|\text{sg}(e_c) - e_{c'}\|_2^2\Big)\Bigg]. \tag{C.1}$$

Here, $\text{sg}(\cdot)$ stops gradients; τ and γ weight the regularizer and commitment term. The reference labels $\mathcal{Y}$ are sampled from a target label distribution, and $|\mathcal{Y}|$ is their number.

OC. Zhang et al. (2024) transfer head-class information to tail classes by calibrating denoising targets. Conditional generation uses the T2H rule and batch resampling to construct a class-balanced reference batch $\mathcal{B}$. The objective is

$$\mathcal{L}_{\text{OC}} = \mathbb{E}\left[\|\epsilon_\theta(x_t, t, c) - \hat{\epsilon}\|_2^2\right]. \tag{C.2}$$

For (x, c) and its noisy state x_t, a reference $(\hat{x}, \hat{c})$ is selected from $\mathcal{B}$ using the forward Gaussian transition density q_t:

$$p_{\text{sel}}(\hat{x}, \hat{c} \mid x_t, \mathcal{B}) = \frac{q_t(x_t \mid \hat{x})}{\sum_{(u,v)\in\mathcal{B}} q_t(x_t \mid u)}. \tag{C.3}$$

Let $\pi(c)$ denote training-set class frequency. Transfer is accepted only when the reference class is at least as frequent as the original class. Under the standard noise parameterization above, the calibrated target is

$$\hat{\epsilon} = \begin{cases} \epsilon, & \pi(\hat{c}) < \pi(c), \\ \dfrac{x_t - \sqrt{\bar{\alpha}_t}\hat{x}}{\sqrt{1-\bar{\alpha}_t}}, & \pi(\hat{c}) \geq \pi(c). \end{cases} \tag{C.4}$$

D Theoretical Analysis

D.1 Local-Response Model

Assumption D.1 (conditional-response decomposition at fixed inputs). Fix a sample, sampling state, and finite candidate set $\mathcal{L}$. Hold each block input and normalization denominator fixed in this local analysis. For block l's d_l-dimensional response, define the mean-square inner product by $\langle v, w\rangle_R = d_l^{-1}\sum_i v_i w_i$ and suppose

$$r_l = \frac{h_l^c - h_l^0}{\text{RMS}(h_l^0) + \varepsilon} = v_l + b_l u_l, \qquad \langle v_l, u_l\rangle_R = 0, \quad \|v_l\|_R = t_l > 0, \quad \|u_l\|_R = 1, \quad b_l \geq 0. \tag{D.1}$$

Here v_l is the target normal signal and $b_l u_l$ is an excess bias, giving $q_l = p_l^2 = t_l^2 + b_l^2 > 0$. This decomposition forms the basis of the local benefit–cost analysis. Gating gives the fixed-input response $g_l r_l$, with $0 \leq g_l \leq 1$. Its squared-pressure reduction is

$$x_l = (1 - g_l^2)q_l, \qquad g_l = \sqrt{1 - x_l/q_l}, \qquad 0 \leq x_l \leq q_l. \tag{D.2}$$

Frequency reassignment (Appendix E.1) and fixed gating (Appendix E.3) empirically motivate distinguishing frequency-related conditional-response bias from the fidelity cost of conditioning attenuation.

Full statement of Proposition 3.1. Under Assumption D.1, define the local distortion by

$$F_l(g) = \nu_l t_l^2 (1-g)^2 + \kappa_l b_l^2 g^2, \qquad \nu_l, \kappa_l > 0. \tag{D.3}$$

Its unique optimal gate is

$$g_l^{\text{oracle}} = \frac{\nu_l t_l^2}{\nu_l t_l^2 + \kappa_l b_l^2}. \tag{D.4}$$

Let $x = (1-g^2)q_l$ and $B_l = \nu_l t_l^2 / q_l^2$. The utility relative to full conditioning is

$$U_l(x) = F_l(1) - F_l\left(\sqrt{1 - x/q_l}\right) = a_l x - C_l(x), \qquad a_l = \frac{\kappa_l b_l^2}{q_l}, \tag{D.5}$$

where

$$C_l(x) = \nu_l t_l^2 \left(1 - \sqrt{1 - x/q_l}\right)^2, \qquad \frac{B_l x^2}{4} \le C_l(x) \le B_l x^2.$$

If the normal-signal reference is t_l and $\kappa_l = \kappa_0$, then $a_l = \kappa_0(1 - z_l^{-2})$, where $z_l = p_l/t_l$. The initial-benefit coefficient is therefore strictly increasing in normalized pressure.

D.2 Proof of Proposition 3.1

Differentiating equation D.3 gives

$$F_l'(g) = 2(\nu_l t_l^2 + \kappa_l b_l^2) g - 2\nu_l t_l^2, \qquad F_l''(g) = 2(\nu_l t_l^2 + \kappa_l b_l^2) > 0.$$

Strict convexity establishes the unique minimizer equation D.4, interior when $b_l > 0$ and equal to one when $b_l = 0$. Orthogonality gives $q_l = t_l^2 + b_l^2$. Substitution yields

$$U_l(x) = \kappa_l \frac{b_l^2}{q_l} x - \nu_l t_l^2 \left(1 - \sqrt{1 - x/q_l}\right)^2.$$

Rationalizing,

$$1 - \sqrt{1 - x/q_l} = \frac{x}{q_l(1 + \sqrt{1 - x/q_l})}.$$

Since $1 \le 1 + \sqrt{1 - x/q_l} \le 2$, both cost bounds follow, including the endpoints $x = 0, q_l$. For fixed $q_l > 0$, $C_l(x) = B_l x^2/4 + O(x^3)$, hence $U_l'(0) = a_l$. With a matched reference and common κ_0, the derivative of $\kappa_0(1 - z_l^{-2})$ is $2\kappa_0 z_l^{-3} > 0$; unequal scores strictly rank initial benefits, and ties remain ties. □

The oracle gate provides an analytic optimum for local modulation; SABR computes its online gates using the pressure-guided rule. The weights ν_l, κ_l quantify signal attenuation and bias in the analysis and require no online setting. This result concerns fixed-input local responses; the following allocation bound and mode-distribution analysis connect it to operational gating and mode coverage, respectively.

D.3 Robustness to Reference and Response Scales

Let $\hat{t}_l > 0$ be the effective normalization denominator and $\hat{z}_l = p_l/\hat{t}_l$; in the implementation, $\hat{t}_l = \tau_{k,l} + \varepsilon$. For a representative scale $\kappa_0 > 0$, assume

$$|\kappa_l - \kappa_0| \le \delta_\kappa, \qquad \frac{|\hat{t}_l^2 - t_l^2|}{q_l} \le \epsilon_\tau. \tag{D.6}$$

Define $\hat{a}_l = \kappa_0(1 - \hat{t}_l^2/q_l)$, $\eta_a = \delta_\kappa + \kappa_0 \epsilon_\tau$, and $B_{\max} = \max_l B_l$. The calibration median need not equal the normal-signal scale; ϵ_τ includes this mismatch and the numerical stabilizer.

Theorem D.2 (finite-intervention utility bound). On a nonempty compact feasible set

$$\mathcal{F} \subseteq \left\{ x : 0 \le x_l \le q_l, \ \sum_l x_l = D \right\},$$

let $x^\star \in \operatorname{argmax}_{x\in\mathcal{F}} U(x)$ and $\hat{x} \in \mathcal{F}$, where $U(x) = \sum_l U_l(x_l)$. Define

$$L = \max_{x\in\mathcal{F}} \hat{a}^\mathsf{T} x - \hat{a}^\mathsf{T}\hat{x}.$$

Continuity guarantees existence of both maxima. The allocation-sensitive bound and its total-amount relaxation are

$$\begin{aligned} 0 \le U(x^\star) - U(\hat{x}) &\le L + 2\eta_a D + \sum_l B_l \hat{x}_l^2 \\ &\le L + 2\eta_a D + B_{\max} D^2. \end{aligned} \tag{D.7}$$

For $\mathcal{F} = \{x : 0 \le x_l \le c_l,\ \sum_l x_l = D\}$ with fixed capacities $0 \le c_l \le q_l$ and $0 \le D \le \sum_l c_l$, descending $\hat{z}_l$ filling attains the linear-surrogate optimum, hence $L = 0$, with arbitrary tie breaking. Bound equation D.7 controls its gap to the optimum of the nonlinear utility U.

Proof. The coefficient identity

$$a_l - \hat{a}_l = (\kappa_l - \kappa_0)\frac{b_l^2}{q_l} + \kappa_0 \frac{\hat{t}_l^2 - t_l^2}{q_l}$$

and $b_l^2/q_l \le 1$ imply $|a_l - \hat{a}_l| \le \eta_a$. Equal total amounts give $\|x^\star - \hat{x}\|_1 \le 2D$, so

$$a^\mathsf{T}(x^\star - \hat{x}) = \hat{a}^\mathsf{T}(x^\star - \hat{x}) + (a - \hat{a})^\mathsf{T}(x^\star - \hat{x}) \le L + 2\eta_a D.$$

Subtracting the nonnegative cost $C(x^\star)$ and using $C(\hat{x}) \le \sum_l B_l \hat{x}_l^2 \le B_{\max} D^2$ proves equation D.7. Under fixed capacities, moving mass from an occupied lower-score block to an unsaturated higher-score block cannot decrease the linear objective. Descending filling therefore reaches a maximizer. Its ordering agrees with pressure because $\hat{a}_l = \kappa_0(1 - \hat{z}_l^{-2})$ is strictly increasing. □

The bound separates allocation suboptimality L, reference and response-scale mismatch $2\eta_a D$, and the second-order attenuation cost. The gap is $O(D^2)$ with matched references and scales and $L = 0$.

Connection to operational SABR. Freeze the measured pressures, ordering, and strengths $s_l \in [0, 1]$ at the current step. For an effective block, let d be the remaining excess, $u_l = \min(d, q_l)$, and $y_l = 1 - \sqrt{1 - u_l/q_l}$. Substitution of $g_l = 1 - s_l y_l$ gives the exact accounting reduction

$$\hat{x}_l = q_l(1 - g_l^2) = s_l u_l + s_l(1 - s_l) q_l y_l^2, \qquad 0 \le \hat{x}_l \le u_l.$$

Monotonicity in s_l gives $\hat{x}_l \le u_l$, and $y_l \le 1$ gives the capacity $c_l = q_l s_l(2 - s_l)$; numerically excluded blocks have zero capacity. Thus $D = \sum_l \hat{x}_l \le \Delta_k$ and $C_k = C_{k-1} + \sum_l q_l - D$. The realized one-step reduction D is distinct from the cumulative reference budget B_k. Theorem D.2 bounds the utility gap at the same capacities and realized total, with L given by the linear-objective gap to descending filling. Binary effective strengths recover that filling rule; soft strengths do not automatically give $L = 0$. These identities exactly describe SABR's measured-pressure accounting.

D.4 Mode-Distribution Model

Assumption D.2 (mode readout of bias and signal attenuation). Use a finite, mutually exclusive, and exhaustive mode partition with target masses $\pi_j > 0$, $\sum_j \pi_j = 1$, and nonconstant costs $v_j \ge 0$. Define

$$\beta_g = \beta_{\text{rest}} + \sum_l \kappa_l b_l^2 g_l^2, \qquad \beta_{\text{rest}} \ge 0,$$

$$S_j(g) = \sum_l \nu_l t_l^2 (1 - g_l)^2 c_{lj}, \qquad 0 \le c_{lj} \le 1.$$

The mode readout is

$$P_g(j) = \frac{\pi_j \exp[-\beta_g v_j - S_j(g)]}{\sum_i \pi_i \exp[-\beta_g v_i - S_i(g)]}. \tag{D.8}$$

Here β_g is the shared bias strength and $S_j(g)$ is a mode-dependent signal-attenuation cost. Assumption D.2 adds a mode readout to the local-response model, linking bias suppression and signal

attenuation to mode masses and distributional matching. Mode concentration in Section 2.2 and semantic retention in Appendix E.2 empirically motivate this mechanism.

The variational form follows from the identity

$$\mathrm{KL}(P\|\pi) + \sum_j P_j[\beta_g v_j + S_j(g)] = \mathrm{KL}(P\|P_g) - \ln Z_g$$

for every probability vector P, with $Z_g = \sum_j \pi_j e^{-\beta_g v_j - S_j(g)}$ and natural-log KL. Nonnegativity of KL, with equality only at $P = P_g$, establishes the unique minimizer. This variational form characterizes the trade-off between deviation from the target mode distribution and the costs of bias and signal attenuation.

Let $\mathbf{1}$ denote full conditioning and $\beta_0 = \beta_{\mathbf{1}}$, so $S_j(\mathbf{1}) = 0$. For $\beta_0 > 0$, define

$$P_\beta(j) = \frac{\pi_j e^{-\beta v_j}}{\sum_i \pi_i e^{-\beta v_i}}, \qquad Q_g = P_{\beta_g},$$

$$\lambda_g = 1 - \frac{\beta_g}{\beta_0}, \qquad \rho_g = \max_j S_j(g) - \min_j S_j(g).$$

Since $0 \le c_{lj} \le 1$, $0 \le S_j(g) \le \sum_l C_l(x_l)$ and hence $\rho_g \le \sum_l C_l(x_l)$. For $0 < \alpha < 1$, define

$$M_\alpha(P;\pi) = \sum_j \pi_j \mathbf{1}\{P(j) \ge \alpha\pi_j\}. \tag{D.9}$$

All logarithms are natural except in JS_2, which uses base two. The thresholded mode mass M_α is not empirical feature-space Recall. If $\beta_0 = 0$, all bias terms vanish and $P_{\mathbf{1}} = \pi$; its zero JS cannot decrease strictly.

D.5 Mode Coverage and JS Improvement

Theorem D.3 (benefits of mode modulation). Under Assumptions D.1–D.2 with $\beta_0 > 0$, if

$$\rho_g \le \lambda_g \log(1/\alpha), \tag{D.10}$$

then $M_\alpha(P_g;\pi) \ge M_\alpha(P_{\mathbf{1}};\pi)$. The inequality is strict if an initially undercovered mode additionally satisfies $\log(Q_g(j)/\pi_j) \ge \log\alpha + \rho_g$. Let $J(\beta) = \mathrm{JS}_2(P_\beta, \pi)$. Then $J'(\beta) > 0$ for $\beta > 0$, and

$$|\mathrm{JS}_2(P_g,\pi) - J(\beta_g)| \le \rho_g/2. \tag{D.11}$$

Hence JS decreases strictly whenever $J(\beta_0) - J(\beta_g) > \rho_g/2$.

Proof: coverage. The normalizer in $P_g(j)/Q_g(j) = e^{-S_j}/\mathbb{E}_{Q_g} e^{-S}$ lies between $e^{-\max S}$ and $e^{-\min S}$, proving $|\log(P_g(j)/Q_g(j))| \le \rho_g$. The path $\ell_j(\beta) = \log(P_\beta(j)/\pi_j)$ satisfies $\ell_j(0) = 0$ and $\ell_j''(\beta) = -\mathrm{Var}_{P_\beta}(v) \le 0$. Concavity and $\beta_g = (1-\lambda_g)\beta_0$ give, for every initially covered mode,

$$\log\frac{P_g(j)}{\pi_j} \ge \ell_j(\beta_g) - \rho_g \ge (1-\lambda_g)\log\alpha - \rho_g \ge \log\alpha.$$

The final step is equation D.10. Thus the covered set is preserved. Any initially undercovered mode satisfying the additional margin enters this set with positive mass, giving strict improvement.

Proof: JS divergence. The base-2 JS divergence uses the strictly convex generator

$$f(r) = \frac{r\log(2r/(1+r)) + \log(2/(1+r))}{2\log 2}, \qquad f''(r) = \frac{1}{2\log 2\, r(1+r)} > 0.$$

Write $r_j = P_\beta(j)/\pi_j$ and $w_j = -v_j$. Differentiating $J = \sum_j \pi_j f(r_j)$ yields

$$J'(\beta) = \frac{1}{2}\sum_{i,j} P_\beta(i)P_\beta(j)[f'(r_i) - f'(r_j)](w_i - w_j) > 0.$$

For $\beta > 0$, both r_j and $f'(r_j)$ increase strictly with w_j. Every summand is nonnegative, and nonconstant costs with positive masses ensure a positive summand. Along $R_t(j) = Q_g(j)e^{-tS_j}/\sum_i Q_g(i)e^{-tS_i}$,

$$\frac{d}{dt}\,\mathrm{JS}_2(R_t,\pi) = \frac{1}{2\log 2}\mathbb{E}_{R_t}\left[(S-\mathbb{E}_{R_t}S)\log(1+\pi/R_t)\right].$$

The centered cost is bounded by ρ_g, and Jensen's inequality gives $\mathbb{E}_{R_t}\log(1+\pi/R_t) \le \log(1+\sum_j \pi_j) = \log 2$. The absolute derivative is therefore at most $\rho_g/2$. Integration from $R_0 = Q_g$ to $R_1 = P_g$ proves the two-sided perturbation bound equation D.11, which implies the stated strict-improvement condition. □

Corollary D.4 (a multi-block small-budget interval). Under Theorem D.3, choose a direction $w_l \ge 0$, $\sum_l w_l = 1$, satisfying $A_w > 0$, and set $x_l = Dw_l$. Define

$$A_w = \sum_l a_l w_l > 0, \qquad B_w = \sum_l B_l w_l^2 > 0, \qquad D_{\max} = \min_{l:w_l>0}\frac{q_l}{w_l}.$$

For $0 \le D \le D_{\max}$, $\beta_g = \beta_0 - A_w D$, $\rho_g \le B_w D^2$, and as $D \downarrow 0$,

$$\mathrm{JS}_2(P_g,\pi) = J(\beta_0) - J'(\beta_0)A_w D + O(D^2). \tag{D.12}$$

Let $m = \min_{\beta\in[\beta_0/2,\beta_0]} J'(\beta) > 0$. Every D satisfying

$$0 < D < \min\left\{D_{\max},\ \frac{A_w\ln(1/\alpha)}{\beta_0 B_w},\ \frac{\beta_0}{2A_w},\ \frac{2mA_w}{B_w}\right\} \tag{D.13}$$

preserves all initially covered modes and strictly reduces JS. When the margin condition of Theorem D.3 also holds, an initially undercovered mode enters the covered set, strictly increasing its total target mass.

Proof. Substituting $g_l^2 = 1 - Dw_l/q_l$ gives the exact bias identity; the cost bound gives $\rho_g \le B_w D^2$. Finitely many modes with positive probabilities make J smooth. Taylor expansion of $J(\beta_0 - A_w D)$, together with the two-sided perturbation bound, proves equation D.12. Continuity and strict positivity of J' on $[\beta_0/2, \beta_0]$ give $m > 0$. The first three bounds in equation D.13 ensure gate feasibility, the coverage condition, and $\beta_g \ge \beta_0/2$, respectively. Hence

$$J(\beta_0) - \mathrm{JS}_2(P_g,\pi) \ge \int_{\beta_g}^{\beta_0} J'(u)\,du - \frac{B_w D^2}{2} \ge mA_w D - \frac{B_w D^2}{2} > 0,$$

where the last step follows from the fourth bound. All thresholds are positive, so the interval is nonempty; choosing a direction supported on one block recovers the single-block case. □

The improved mode coverage and reduced JS in Figure 6 agree with the predictions of this section, while the block-assignment control in Table E4 supports the practical value of pressure-guided allocation.

E Additional Experiments

Integration with DPM-Solver. We further evaluate SABR with 25-step DPM-Solver (Lu et al., 2022) on CIFAR-100-LT. SABR reduces overall FID and Few FID and improves Few Recall compared to the baseline.

Pressure-guided block assignment. We compare pressure-guided and random assignment on CIFAR-100-LT using the same model and sampling settings. For each sample and sampling step, the random control preserves the gate values, intervention timing, and number of active blocks, but randomly assigns the gate values to candidate blocks. Pressure-guided assignment achieves lower Few FID and higher Few Recall, supporting the effectiveness of pressure-based block selection.

Table E1: Full results (mean ± std.) on CIFAR-10-LT and CIFAR-100-LT. Precision/Recall use a 0–1 scale. Bold marks paired results with better or tied means; SABR rows are shaded. †: KID SD ≤ 0.0001 (upper bound).

Dataset	Method	FID ↓	KID ↓	Precision ↑	Recall ↑	IS ↑
CIFAR-10-LT IR = 100	DDPM	15.342 ± 0.012	$\mathbf{0.0047 \pm 0.0001}$	0.8480 ± 0.0007	0.4326 ± 0.0041	$\mathbf{9.42 \pm 0.03}$
	DDPM (SABR)	$\mathbf{14.825 \pm 0.061}$	$\mathbf{0.0047 \pm 0.0001}$	$\mathbf{0.8541 \pm 0.0009}$	$\mathbf{0.4429 \pm 0.0056}$	9.40 ± 0.06
	OC	11.802 ± 0.065	0.0036 ± 0.0001	0.8288 ± 0.0005	0.5227 ± 0.0022	9.63 ± 0.03
	OC (SABR)	$\mathbf{10.959 \pm 0.088}$	$\mathbf{0.0033 \pm 0.0001}$	$\mathbf{0.8310 \pm 0.0005}$	$\mathbf{0.5450 \pm 0.0014}$	$\mathbf{9.66 \pm 0.02}$
	CBDM	8.466 ± 0.008	$0.0018 \pm 0.0001^{\dagger}$	0.8227 ± 0.0008	0.6013 ± 0.0049	$\mathbf{9.68 \pm 0.04}$
	CBDM (SABR)	$\mathbf{7.558 \pm 0.009}$	$\mathbf{0.0017 \pm 0.0001}^{\dagger}$	$\mathbf{0.8230 \pm 0.0010}$	$\mathbf{0.6178 \pm 0.0050}$	9.65 ± 0.04
CIFAR-100-LT IR = 100	DDPM	14.227 ± 0.048	0.0037 ± 0.0001	0.8020 ± 0.0012	0.4167 ± 0.0006	13.38 ± 0.11
	DDPM (SABR)	$\mathbf{12.977 \pm 0.034}$	$\mathbf{0.0034 \pm 0.0001}^{\dagger}$	$\mathbf{0.8055 \pm 0.0016}$	$\mathbf{0.4366 \pm 0.0014}$	$\mathbf{13.63 \pm 0.10}$
	OC	9.256 ± 0.023	$0.0030 \pm 0.0001^{\dagger}$	0.7686 ± 0.0016	0.5116 ± 0.0030	13.45 ± 0.01
	OC (SABR)	$\mathbf{8.695 \pm 0.026}$	$\mathbf{0.0025 \pm 0.0001}^{\dagger}$	$\mathbf{0.7705 \pm 0.0008}$	$\mathbf{0.5333 \pm 0.0028}$	$\mathbf{13.50 \pm 0.03}$
	CBDM	7.339 ± 0.035	$\mathbf{0.0014 \pm 0.0001}^{\dagger}$	0.7707 ± 0.0003	0.5426 ± 0.0012	$\mathbf{13.42 \pm 0.02}$
	CBDM (SABR)	$\mathbf{7.276 \pm 0.002}$	$0.0015 \pm 0.0001^{\dagger}$	$\mathbf{0.7720 \pm 0.0001}$	$\mathbf{0.5533 \pm 0.0008}$	13.27 ± 0.01

Table E2: Full results (mean ± std.) on imbalanced ArtBench-10. Precision/Recall use a 0–1 scale. Bold marks paired results with better or tied means; SABR rows are shaded. †: KID SD ≤ 0.0001 (upper bound).

Dataset	Method	FID ↓	KID ↓	Precision ↑	Recall ↑	IS ↑
ArtBench-10 IR = 100	DDPM	18.189 ± 0.086	0.0070 ± 0.0001	0.661 ± 0.004	0.51 ± 0.01	8.46 ± 0.05
	DDPM (SABR)	$\mathbf{16.797 \pm 0.067}$	$\mathbf{0.0062 \pm 0.0001}^{\dagger}$	$\mathbf{0.662 \pm 0.003}$	$\mathbf{0.53 \pm 0.01}$	$\mathbf{8.49 \pm 0.05}$
	OC	15.037 ± 0.048	0.0064 ± 0.0001	0.682 ± 0.004	0.50 ± 0.05	7.49 ± 0.01
	OC (SABR)	$\mathbf{13.350 \pm 0.100}$	$\mathbf{0.0054 \pm 0.0001}$	$\mathbf{0.685 \pm 0.004}$	$\mathbf{0.52 \pm 0.01}$	$\mathbf{7.62 \pm 0.02}$
	CBDM	$\mathbf{10.588 \pm 0.012}$	$\mathbf{0.0059 \pm 0.0001}^{\dagger}$	$\mathbf{0.669 \pm 0.001}$	0.59 ± 0.01	$\mathbf{6.71 \pm 0.04}$
	CBDM (SABR)	10.893 ± 0.082	0.0061 ± 0.0001	0.662 ± 0.004	$\mathbf{0.60 \pm 0.01}$	6.67 ± 0.01

Table E3: Generation results with 25-step DPM-Solver on CIFAR-100-LT. Values are mean ± standard deviation; Few Recall uses a 0–1 scale. Better results are bold, and the SABR row is shaded in light red.

Method	Overall FID ↓	Few FID ↓	Few Recall ↑
DDPM	11.927 ± 0.041	28.124 ± 0.230	0.2930 ± 0.0093
DDPM (SABR)	$\mathbf{11.056 \pm 0.071}$	$\mathbf{24.797 \pm 0.135}$	$\mathbf{0.3410 \pm 0.0036}$

Table E4: Ablation of pressure-guided block assignment on CIFAR-100-LT. Random assignment and SABR share gate values, intervention timing, and the number of active blocks, differing only in the gate-to-block assignment. Better results are bold, and the SABR row is shaded in light red.

Block assignment	Few FID ↓	Few Recall (%) ↑
Random	27.668	31.11
Pressure-guided (SABR)	**27.017**	**32.96**

E.1 Training-Frequency Reassignment

Natural long-tailed distributions combine frequency variation with semantic-difficulty variation, so a simple head–tail comparison cannot distinguish their effects. We construct three class–frequency assignments on CIFAR-100-LT, $\mathcal{A} = \{\mathrm{Original}, \mathrm{Permuted}, \mathrm{Reverse}\}$, while preserving the class set, count multiset, and total sample size. Original retains the observed class–count mapping; Permuted reassigns the same counts using a fixed random permutation; Reverse inverts the mapping. All schemes use the same 100 class counts, ranging from 5 to 500 images, and contain 10,847 images in total. Architecture, initialization, optimizer, and training length are identical, and generation uses shared initial noise across the three settings. This controlled experiment analyzes training-frequency

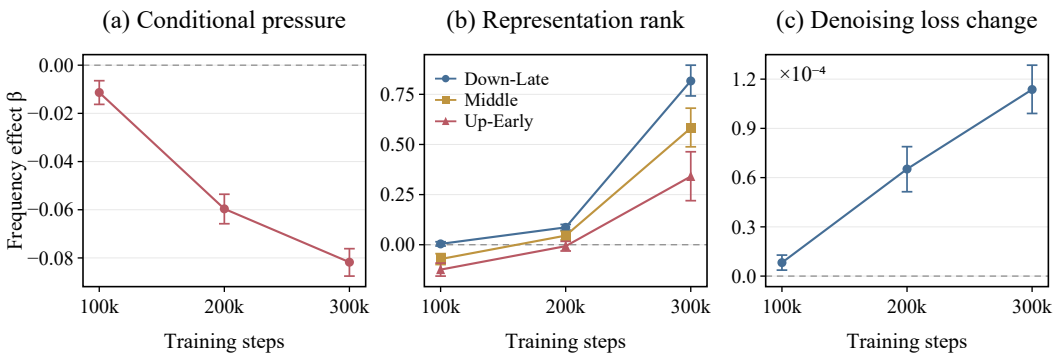


Figure E1: Evolution of the training-frequency effect. The vertical axis reports the regression coefficient β, not the raw metric value. (a) A more negative coefficient indicates higher conditional pressure at lower frequency. (b) A positive coefficient indicates higher internal effective rank at higher frequency. (c) Denoising-loss change is the loss after weakening conditioning minus the fully conditioned loss; a positive coefficient means that this difference increases with frequency. Error bars are 95% class-clustered bootstrap confidence intervals.

effects and is not additional training required to deploy SABR. The design compares each semantic class across different training frequencies.

Figure 3 in the main text shows that conditional pressure generally increases as training counts decrease in Original. In Reverse, high pressure moves with the reversed frequency order to the newly low-frequency classes. Permuted disrupts the ordered trend along semantic class IDs. Thus, high-pressure regions are not tied to a fixed semantic subset but shift with training-frequency reassignment.

We quantify the relationship between training frequency and each analyzed outcome using

$$y_{c,a} = \beta \log n_{c,a} + \alpha_c + \delta_a + \varepsilon_{c,a}, \tag{E.1}$$

where $y_{c,a}$ is conditional pressure, internal effective rank, denoising-loss change, generated-image effective rank (Roy and Vetterli, 2007), or Vendi Score (Friedman and Dieng, 2023) for class c under assignment a, and $n_{c,a}$ is its training count. The class fixed effect α_c controls semantic differences that do not vary across assignments, and δ_a controls assignment-level differences. For generated images, effective rank is the exponential of the entropy of the normalized eigenvalues of the centered feature covariance matrix. We estimate β at the 100k, 200k, and 300k checkpoints to examine how the frequency effect develops during training.

As training progresses, the frequency coefficient for conditional pressure becomes more negative, indicating stronger local conditioning pressure at lower frequency. Meanwhile, effective-rank coefficients in Down-Late, Middle, and Up-Early become positive, indicating narrower internal representation spectra at lower frequency, with the strongest dependence in Down-Late. Figure E1(c) also shows that the denoising-loss change induced by weakened conditioning varies with training frequency after controlling for class and assignment effects.

The contraction extends to final generations. The frequency coefficients for generated-image effective rank and Vendi Score are significantly positive, indicating richer spectral coverage at higher training frequency and consistently narrower output coverage under low-frequency training.

E.2 Semantic Retention

To examine whether improved intra-class mode coverage is accompanied by a shift in target-class semantics, we use L_2-normalized, 2,048-dimensional Inception pool3 features and construct class centroids using only balanced real images. We compare SABR with the original DDPM. Semantic retention is the fraction of samples whose nearest class centroid belongs to the target class. Semantic margin is the mean difference between the squared Euclidean distance to the nearest other-class centroid and that to the target-class centroid. Target-centroid distance is the mean squared Euclidean distance to the target-class centroid. SABR increases semantic retention from 52.85% to 53.55% and

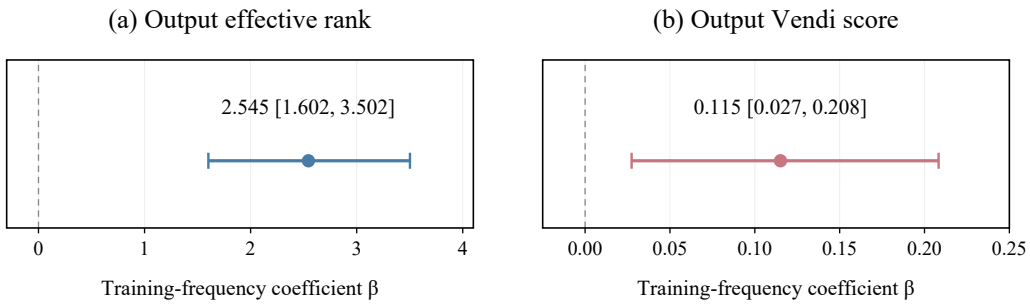


Figure E2: Relationship between training frequency and generated-image diversity. Panels report (a) effective rank and (b) Vendi Score. Dots are frequency coefficients after controlling for class and assignment fixed effects, horizontal lines are 95% confidence intervals, and the vertical dashed line denotes zero.

Table E5: Paired comparison of tail-class semantic retention between DDPM and DDPM (SABR). Bold indicates the better value for each metric, and the SABR row is shaded in light red.

Method	Semantic retention ↑	Semantic margin ↑	Target-centroid distance ↓
DDPM	52.85%	0.00430	0.35790
DDPM (SABR)	**53.55%**	**0.00569**	**0.35686**

Table E6: Down-Late conditional pressure of DDPM and CBDM on CIFAR-100-LT. The absolute head–tail gap is Few−Many.

Model	Many	Medium	Few	Few−Many
DDPM	0.2852	0.3400	0.3667	0.0815
CBDM	0.2063	0.2419	0.2514	0.0451

semantic margin from 0.00430 to 0.00569, while reducing target-centroid distance from 0.35790 to 0.35686.

E.3 Fixed-Gate Comparison

Using DDIM50 sampling on DDPM with a batch size of 256, we apply a constant gate g to all four Down-Late candidates at every timestep for Medium and Few, leaving Many unchanged. Intervention locations are fixed; uniform scaling replaces SABR's sample-, block-, and timestep-dependent gates.

Figure E3 shows results for $g = 0, 0.25, 0.5, 0.75$, and 0.9. Across these tested values, decreasing g monotonically lowers both Few FID and Precision. These metrics are 30.723 and 75.67% at $g = 0.9$, versus 23.881 and 69.01% at $g = 0$, illustrating a trade-off between improved FID and reduced Precision. SABR achieves a Few FID of 27.017 and a Precision of 75.64%, yielding lower FID than $g = 0.9$ at similar Precision. None of the plotted fixed gates matches SABR on both metrics, supporting adaptive modulation.

E.4 Cross-Model Pressure

Although tail-class conditional pressure remains higher than head-class pressure in the Down-Late region of CBDM, both the group-wise pressure levels and the absolute head–tail gap in this region are lower than those of DDPM. The absolute gap decreases from 0.08154 to 0.0451. This may leave less room for further modulation of this region to improve performance, partially explaining the limited gains and even deterioration in some metrics observed with SABR in certain CBDM configurations.

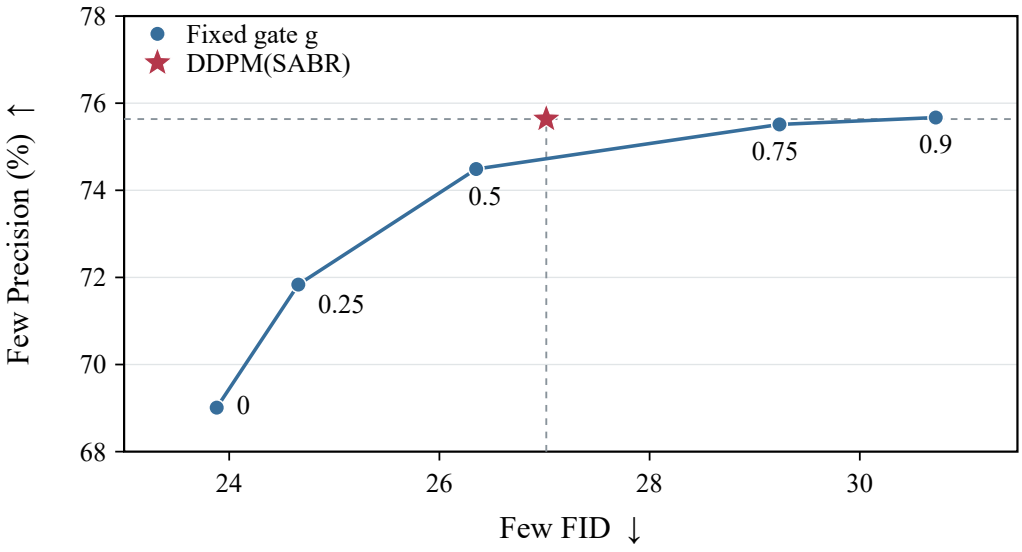


Figure E3: Tail-class FID–Precision comparison between DDPM (SABR) and fixed conditioning attenuation. Lower FID and higher Precision are better. Blue circles and connecting lines show fixed-gate results, with adjacent numbers indicating g; the red star denotes DDPM (SABR). Dashed lines indicate the projections of DDPM (SABR) onto the two axes.

Table E7: Comparison with CADS on CIFAR-100-LT. Values are mean ± sample standard deviation. KID is multiplied by 10^3; Precision and Recall are percentages. Bold marks the best mean within each group, including ties; SABR rows are shaded.

Group	Method	FID ↓	KID $\times 10^3$ ↓	IS ↑	Precision ↑	Recall ↑
Overall	DDPM	14.227 ± 0.048	3.702 ± 0.066	13.380 ± 0.112	80.20 ± 0.12	41.67 ± 0.06
	DDPM + CADS	13.220 ± 0.059	3.768 ± 0.064	13.333 ± 0.075	79.56 ± 0.21	42.92 ± 0.36
	DDPM + SABR	**12.977 ± 0.034**	**3.370 ± 0.026**	**13.631 ± 0.096**	**80.55 ± 0.16**	**43.66 ± 0.14**
Many	DDPM	18.950 ± 0.081	6.003 ± 0.331	**14.691 ± 0.027**	**88.99 ± 0.17**	44.07 ± 0.30
	DDPM + CADS	**17.280 ± 0.095**	**5.231 ± 0.184**	14.539 ± 0.044	88.36 ± 0.11	**46.07 ± 0.31**
	DDPM + SABR	18.950 ± 0.081	6.003 ± 0.331	**14.691 ± 0.027**	**88.99 ± 0.17**	44.07 ± 0.30
Medium	DDPM	22.217 ± 0.039	5.312 ± 0.170	10.587 ± 0.146	79.27 ± 0.01	41.14 ± 0.37
	DDPM + CADS	20.550 ± 0.097	5.280 ± 0.162	10.582 ± 0.068	79.19 ± 0.14	42.59 ± 0.66
	DDPM + SABR	**20.484 ± 0.056**	**4.688 ± 0.178**	**10.705 ± 0.152**	**79.88 ± 0.41**	**43.08 ± 0.13**
Few	DDPM	31.525 ± 0.105	6.288 ± 0.202	9.589 ± 0.060	75.10 ± 0.47	27.74 ± 0.89
	DDPM + CADS	29.301 ± 0.139	6.577 ± 0.177	9.735 ± 0.124	74.35 ± 0.80	28.15 ± 0.69
	DDPM + SABR	**27.017 ± 0.097**	**4.354 ± 0.104**	**10.059 ± 0.055**	**75.39 ± 0.27**	**32.96 ± 0.41**

E.5 Comparison with Condition-Annealed Sampling

Setup. We compare DDPM, CADS (Sadat et al., 2024), and SABR using the CIFAR-100-LT DDPM EMA checkpoint at 300,000 training steps and 50-step DDIM, generating 500 images per class in each evaluation. Many, Medium, and Few contain classes 0–32, 33–65, and 66–99, respectively.

CADS perturbs the global class embedding throughout the U-Net for all classes, following its condition-annealing and rescaling equations. We use normalized diffusion time with $\tau_1 = 0.5$, $\tau_2 = 0.9$, noise scale $s = 0.1$, and rescaling coefficient $\psi = 1$. These parameters are transferred from the published ImageNet-256 settings, rather than optimized for CIFAR-100-LT. A separate random stream for condition perturbations preserves the paired initial-noise stream.

Table E8: Adaptive versus frozen group–timestep–block mean gates on CIFAR-100-LT Few classes. Values are mean ± sample standard deviation; Precision and Recall are percentages. Bold marks the better mean, and the SABR row is shaded.

Method	Few FID ↓	Few Precision ↑	Few Recall ↑
Frozen mean gates	28.836 ± 0.078	$\mathbf{75.41 \pm 0.38}$	30.89 ± 1.27
DDPM + SABR	$\mathbf{27.017 \pm 0.097}$	75.39 ± 0.27	$\mathbf{32.96 \pm 0.41}$

Results and distinction. Under this configuration, SABR achieves better mean FID, KID, IS, Precision, and Recall than CADS for Overall, Medium, and Few. In Few classes, FID decreases from 29.301 to 27.017 and Recall increases from 28.15% to 32.96%, with higher Precision. CADS obtains better Many-class FID, KID, and Recall, whereas SABR retains the original Many-class outputs. The improvements therefore vary across class groups.

CADS addresses generation diversity through scheduled perturbations of conditioning vectors. Our investigation instead concerns training-frequency-related local conditional pressure, and SABR uses current pressure measurements to selectively modulate internal conditional residuals. This comparison evaluates a related training-free sampling method; the pressure-localization and frequency-reassignment experiments establish the empirical basis of our approach.

E.6 Adaptive versus Frozen Mean Gates

Frozen policy. Using the same DDPM and sampling configuration as Appendix E.5, we estimate SABR's mean gates from 64 independent calibration trajectories per Medium/Few class (4,288 trajectories in total). Gates are averaged separately by class group, sampling step, and candidate block, including nonintervention gates equal to one. The resulting lookup table is frozen before evaluation; Many-class gates remain one. At inference, the control uses only this table, without current-sample pressure measurement, block ranking, or budget feedback. It preserves average group-, time-, and block-specific modulation rather than applying a single constant gate.

Both policies are evaluated on 17,000 Few-class images (500 per class) in each evaluation, using samples disjoint from calibration and a shared frozen calibration table. Verified existing SABR samples are reused.

Results. SABR reduces Few FID from 28.836 to 27.017 and increases Recall from 30.89% to 32.96%, while Precision remains close (75.41% versus 75.39%). Thus, freezing the average modulation pattern does not fully recover the benefits of the sample-dependent adaptive policy.

Attenuation and scope. Averaging gates preserves calibration first moments, but not sparsity, second moments, or pressure-weighted attenuation. For the available archived gate traces, mean attenuation $\mathbb{E}[1-g]$ is 26.77% for the frozen policy and 26.83% for SABR. However, $\mathbb{E}[1-g^2]$ is 44.35% versus 27.45%, and the mean number of active blocks per step is 4.00 versus 1.15. The comparison supports the complete adaptive policy over its frozen average, rather than isolating pressure feedback from changes in attenuation and sparsity.

F Generated Samples

Figures F1 and F2 compare subjects, backgrounds, and compositions between OC and OC (SABR) for Baroque and the Post-Impressionism tail class, respectively.



Figure F1: Qualitative comparison for the Baroque class on imbalanced ArtBench-10 (IR $= 100$). From top to bottom, rows show OC, OC (SABR), and real images. Compared with OC, the displayed OC (SABR) samples show more varied subjects and compositions while maintaining visual styles consistent with the real-image examples. The images shown are randomly selected.



Figure F2: Additional visualization of intra-class diversity in the Post-Impressionism tail class on imbalanced ArtBench-10 (IR $= 100$). Left: OC. Right: OC (SABR). The displayed OC samples mainly depict natural landscapes, whereas OC (SABR) includes landscapes, architecture, and still lifes with more varied backgrounds and compositions. Images are randomly selected.

The following figures show additional generation results with OC (SABR) on four datasets.

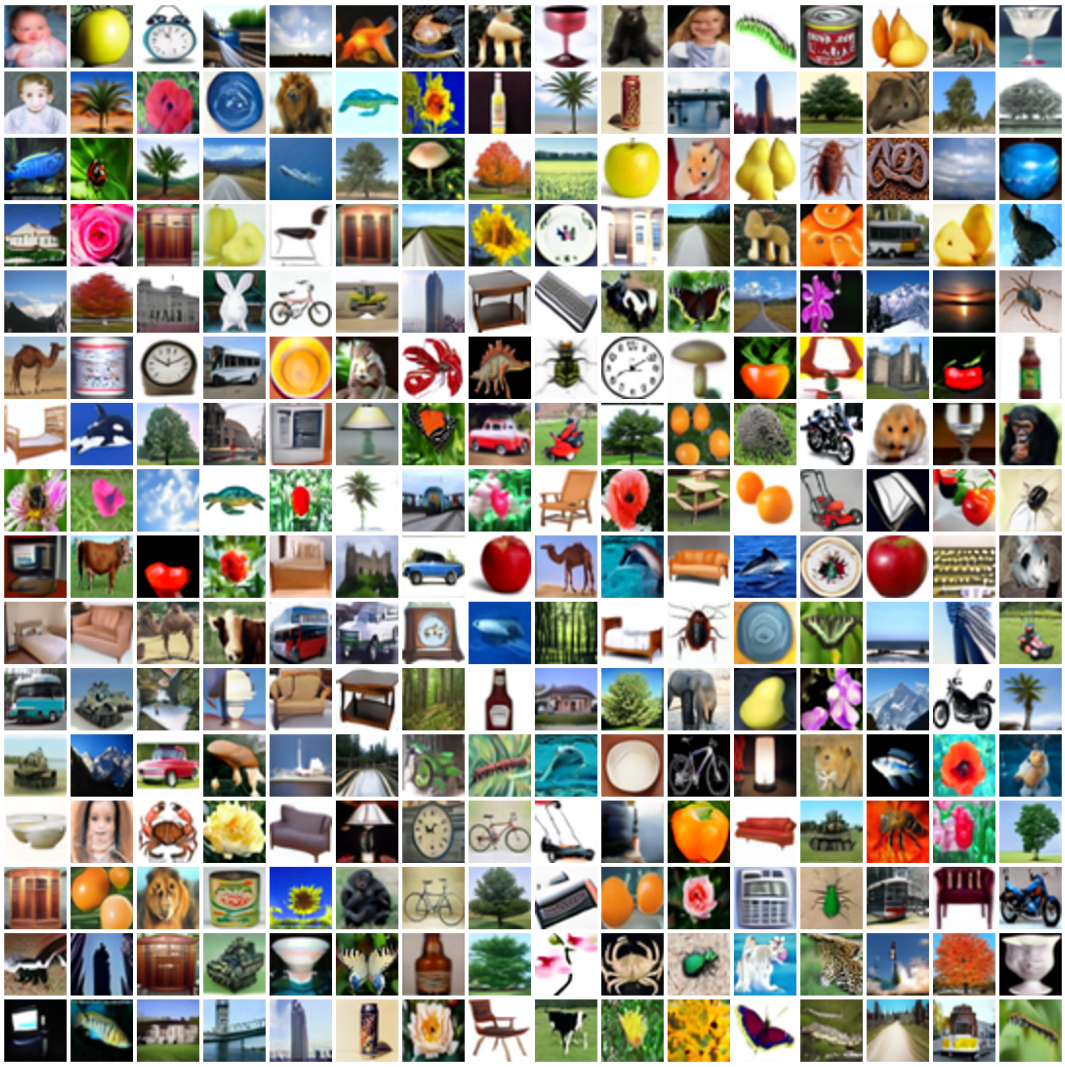


Figure F3: Visualization of generation results on CIFAR-100-LT with OC (SABR). The images shown are randomly selected.

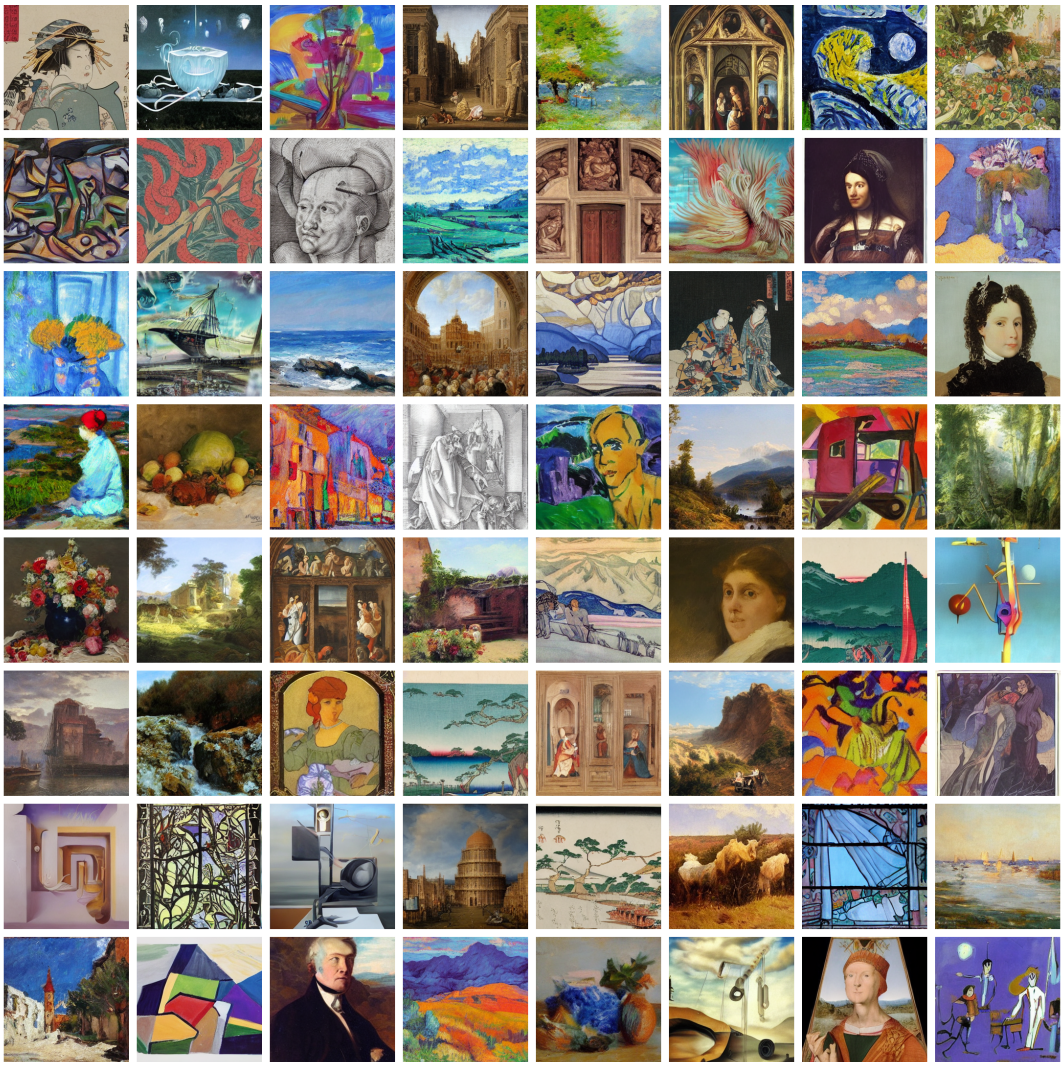


Figure F4: Visualization of generation results on imbalanced ArtBench-10 with OC (SABR). The images shown are randomly selected.

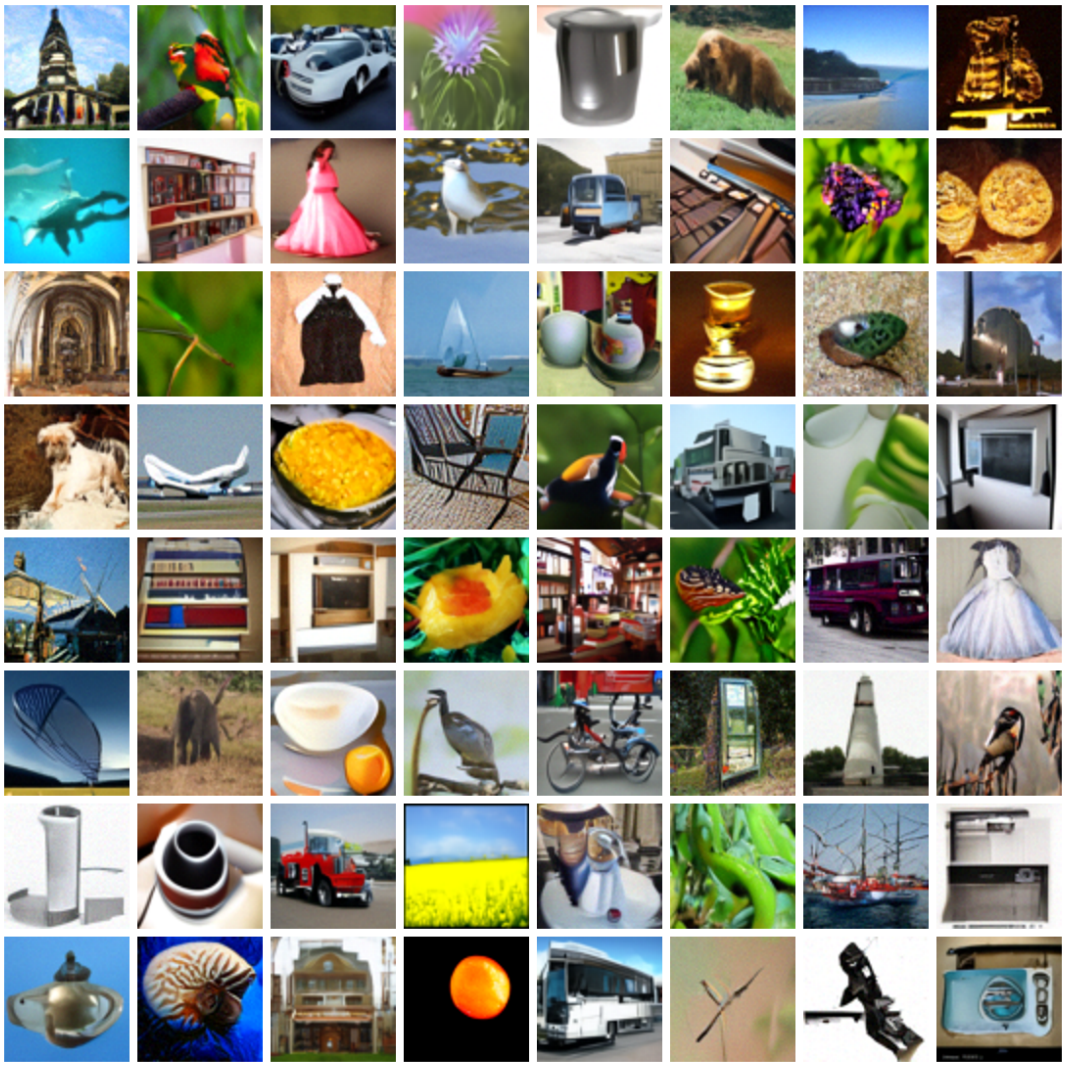


Figure F5: Visualization of generation results on ImageNet-LT with OC (SABR). The images shown are randomly selected.



Figure F6: Visualization of generation results on iNaturalist2018-LT with OC (SABR). The images shown are randomly selected.

G Limitations and Future Work

Candidate-region localization and controlled frequency analyses are conducted mainly on CIFAR-100-LT, while cross-model and cross-architecture experiments further examine the applicability of SABR. Different base models respond differently to conditioning adjustments, and some CBDM configurations exhibit a trade-off between quality and coverage. SABR accumulates the squared products of gate values and measured pressures and compares the cumulative quantity with a frozen budget to determine routing. Online measurement and gating account for its additional inference cost. Future work will extend the validation of internal mechanisms across architectures, further analyze the roles of routing components, and explore lower-overhead pressure measurement and routing implementations.

References

Bradley Efron and Robert J. Tibshirani. *An Introduction to the Bootstrap*. Chapman & Hall, 1993.

Dan Friedman and Adji Bousso Dieng. The Vendi Score: A diversity evaluation metric for machine learning. *Transactions on Machine Learning Research*, 2023.

Sture Holm. A simple sequentially rejective multiple test procedure. *Scandinavian Journal of Statistics*, 6(2):65–70, 1979.

Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In *ICLR*, 2015.

Jianhua Lin. Divergence measures based on the Shannon entropy. *IEEE Transactions on Information Theory*, 37(1):145–151, 1991.

Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. In *ICLR*. OpenReview.net, 2019.

Cheng Lu, Yuhao Zhou, Fan Bao, Jianfei Chen, Chongxuan Li, and Jun Zhu. DPM-Solver: A fast ODE solver for diffusion probabilistic model sampling in around 10 steps. In *NeurIPS*, 2022.

Yiming Qin, Huangjie Zheng, Jiangchao Yao, Mingyuan Zhou, and Ya Zhang. Class-balancing diffusion models. In *CVPR*, pages 18434–18443, 2023.

Olivier Roy and Martin Vetterli. The effective rank: A measure of effective dimensionality. In *EUSIPCO*, pages 606–610, 2007.

D. Sculley. Web-scale k-means clustering. In *WWW*, pages 1177–1178, 2010.

Christian Szegedy, Vincent Vanhoucke, Sergey Ioffe, Jon Shlens, and Zbigniew Wojna. Rethinking the Inception architecture for computer vision. In *CVPR*, pages 2818–2826, 2016.

Tianjiao Zhang, Huangjie Zheng, Jiangchao Yao, Xiangfeng Wang, Mingyuan Zhou, Ya Zhang, and Yanfeng Wang. Long-tailed diffusion models with oriented calibration. In *ICLR*. OpenReview.net, 2024.