A. Appendix

A.1. Experiment Details

Model Configurations and Hyperparameter Settings. Hyperparameter configurations for CIFAR-10 and ImageNet are detailed in Table 5. For ImageNet experiments, we employed a VAE model for initial 8×8 downsampling, converting 256×256 images to 32×32 inputs for the DiT model. We evaluated performance metrics, including FID, IS, Precision, and Recall, on 50K generated samples. During training, we adopted the logit normal sampling strategy as proposed in SD3 [7].

Hyperparameters	CIFAR-10	ImageNet
Image size	32×32	256×256
CFG scale	-	1.25
Model	DDPM++	DiT-B/2
Batch size	512	768
Training epoches	3072	350
Learning rate	1e-3	5e-4

Table 5. Hyperparameter configuration for experiments on CIFAR-10 and ImageNet dataset.

Stochastic Sampling Configurations. We utilize the Stochastic Curved Euler Sampler (SDE Sampler in our paper) [24] as our primary sampling method. The detailed sampling algorithm is presented in Algorithm 1. Additionally, for comparison, we also employ the Overshooting Sampler [14] shown in Algorithm 2. Theoretically, the Stochastic Curved Euler Sampler is equivalent to the Overshooting Sampler with $c = 1.0$ in the limit of infinitesimal step sizes, and can be viewed as a variant of DDPM adapted for Rectified Flow models [24].

Algorithm 1 Stochastic Curved Euler Sampler for Rectified Flow.

1: **procedure** STOCHASTICCURVEDEULERSAMPLER(v, $\{t_i\}_{i=0}^{N}$)
2:     **Initialize** $\tilde{Z}_0 \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$
3:     **for** $i \in \{0, \ldots, N-1\}$ **do**
4:         Calculate velocity $\boldsymbol{v}_i = v(\tilde{Z}_{t_i}; t_i)$
5:         Predict end points:
6:             $\tilde{Z}_{t_i}^{(1)} \leftarrow \tilde{Z}_{t_i} + (1-t_i)\boldsymbol{v}_i$ $\tilde{Z}_{t_i}^{(0)} \leftarrow \tilde{Z}_{t_i} - t_i\boldsymbol{v}_i$
7:         Sample random noise $\boldsymbol{\xi}_i \sim \mathcal{N}(0, \mathbf{I})$
8:         Calculate noise replacement factor $\alpha_i \leftarrow 1 - \frac{t_i(1-t_{i+1})}{t_{i+1}(1-t_i)}$
9:
10:         Refresh end point $\tilde{Z}_{t_i}^{(0),\text{ref}} \leftarrow (1-\alpha_i)\tilde{Z}_{t_i}^{(0)} + \sqrt{1-(1-\alpha_i)^2}\boldsymbol{\xi}_i$
11:
12:         Update sample $\tilde{Z}_{t_{i+1}} \leftarrow t_{i+1}\tilde{Z}_{t_i}^{(1)} + (1-t_{i+1})\tilde{Z}_{t_i}^{(0),\text{ref}}$
13:     **return** $\tilde{Z}_{t_N}$

A.2. Figure 1 Generation Details and Violation of Boundary Condition in T2I Models

To generate Figure 1 (main paper), we analyzed the velocity field predicted by the pre-trained Rectified Flow model, *FLUX.1-dev* [18]. Specifically, we extracted the model's velocity prediction at timestep $t = 1$. To visualize this high-dimensional velocity field as an image, we utilized the pre-trained VAE model from the same FLUX.1-dev model. This VAE decoder was employed to map the predicted velocity vector back into the pixel space, resulting in the visual representations shown in Figure 1. Our analysis reveals that even this state-of-the-art text-to-image diffusion model, *FLUX.1-dev*, exhibits a significant deviation from the expected boundary condition at $t = 1$. This boundary violation shows a practical limitation inherent in current Rectified Flow models, as highlighted in our main paper. In Figure 6, we show more samples generated by FLUX model that violates the boundary conditions.

Algorithm 2 Overshoot Sampling for Rectified Flow.

1: **procedure** OVERSHOOTINGSAMPLER(v, $\{t_i\}_{i=0}^{N}$, $c \in \mathbb{R}^+$)
2:     **Initialize** $\tilde{Z}_0 \sim \mathcal{N}(\mathbf{0}, \, \mathbf{I})$
3:     **for** $i \in \{0, \dots, N-1\}$ **do**
4:         Calculate velocity $\boldsymbol{v}_i = v(\tilde{\boldsymbol{Z}}_{t_i}; t_i)$
5:         Overshooted ODE update:
6:
7:         $\hat{\boldsymbol{Z}}_o = \tilde{\boldsymbol{Z}}_{t_i} + (\boldsymbol{o} - t_i) \circ \boldsymbol{v}_i, \quad \text{with} \quad o = \min(t_{i+1} + c(t_{i+1} - t_i), \, 1).$
8:
9:         Backward update by adding noise:
10:
11:         $\tilde{\boldsymbol{Z}}_{t_{i+1}} \leftarrow a\hat{\boldsymbol{Z}}_o + b\boldsymbol{\xi}_i, \quad \text{where} \ \boldsymbol{\xi}_i \sim \mathcal{N}(\mathbf{0}, \boldsymbol{I}), \text{ and } a = \frac{s}{o} \ \text{ and } \ b = \sqrt{(1-s)^2 - (a(1-o))^2}.$
12:
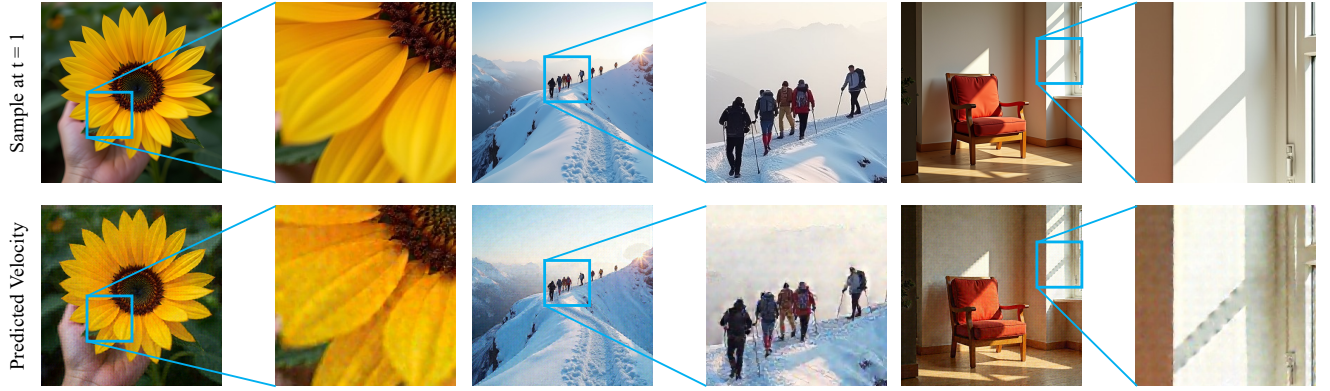13:   **return** $\tilde{\boldsymbol{Z}}_{t_N}$

Figure 6. **Boundary Condition Violation in FLUX.1-dev Model.** Visualization of the predicted velocity field at $t = 1$, demonstrating deviation from the expected data distribution and violation of the right boundary condition $v(\mathbf{x}, t) = \mathbf{x}$.

Figure 7. **Qualitative comparison of image generation results on ImageNet** 256×256 **dataset.** We present paired examples generated by vanilla RF model (first row), Mask-based Boundary RF Model (second column) and Subtraction-based Boundary RF Model (last column). We use the same random seed for all models to ensure a fair visual comparison. Our approach consistently generates images with clearer and more detailed structures, and improved visual fidelity compared to vanilla RF model.

A.3. Additional Qualitative Results

We present an expanded qualitative comparison in Figure 7, providing more diverse examples as an extension of Figure 4 in the main paper. These additional examples further show the visual advantages of Boundary RF Model and Subtraction-based Boundary RF Model over the vanilla RF model.

A.4. Qualitative Comparison of Boundary Functions

We provide qualitative samples generated by the the models that apply different boundary functions, visualized in Figure 8. The default Standard Cosine model exhibits better visual fidelity compared to those from other function choices.
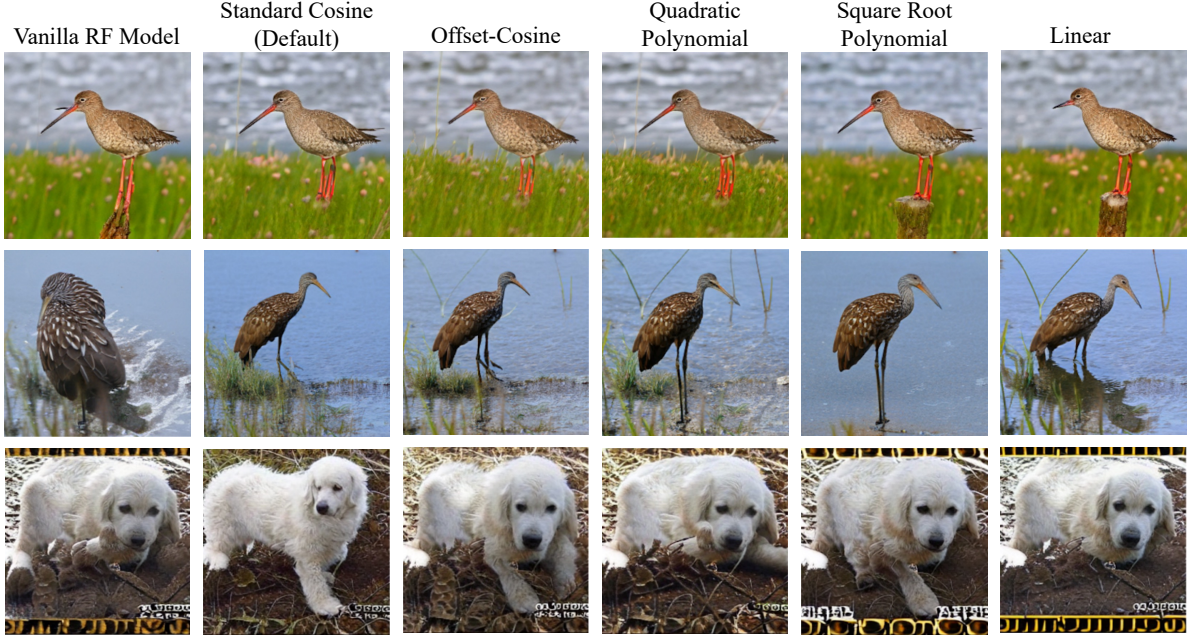

Figure 8. Qualitative comparison of boundary functions. Generated samples from double-boundary models using different choices of functions $(f(t), g(t), h(t))$.

A.5. Qualitative Comparison of Stochastic Sampling with Overshooting Sampler.

We present qualitative differences in Figure 9. We observed that over-smoothed samples from vanilla RF model when using Overshooting sampler, while Boundary RF Model maintains sharper and more finegrained details. The results illustrate the robustness of Boundary RF Model to stochastic sampling and its superior preservation of high-frequency details compared to vanilla RF model under stochasticity sampling.

Figure 9. **Qualitative comparison of stochastic sampling with Overshooting sampler.** Qualitative samples using the overshooting sampler reveal that Boundary RF Model effectively preserves high-frequency details, contrasting with the over-smoothed results from vanilla RF model under stochastic sampling.

A.6. Qualitative Comparison of Boundary RF Model and Vanilla RF Model on DiT-L/2 and DiT-XL/2 Models

We present qualitative comparison in Figure 10, providing visual examples generated using the same random seeds. These examples show the visual advantages of Boundary RF Model and Subtraction-based Boundary RF Model over the vanilla RF model on models with larger sizes.

(a) DiT-L/2

(b) DiT-XL/2

Figure 10. **Qualitative comparison of image generation results on ImageNet** 256×256 **dataset with DiT-L/2 and DiT-XL/2 models.** We present paired examples generated by vanilla RF model (first row), Mask-based Boundary RF Model (second column) and Subtraction-based Boundary RF Model (last column). We use the same random seed for all models to ensure a fair visual comparison. Our approach consistently generates images with clearer and more detailed structures, and improved visual fidelity compared to vanilla RF model.