A. Model Specs

Config	#Layers	Hidden dim	#Heads
B/2	12	768	12
L/2	24	1024	16
XL/2	28	1152	16

B. Hyper-parameters

VAE	SD-VAE-f8d4-ft-ema
VAE donwsample	8
latent channel	4
optimizer	AdamW [3]
base learning rate	1e-4
weight decay	0.0
batch size	256
learning rate schedule	constant
augmentation	center crop
diffusion sampler	Euler-ODE
diffusion steps	250
evaluation suite	ADM [2]

C. Pixel Space Class-to-Image Experiments.

We conduct pixel C2I on ImageNet 64×64 to assess the generalization capability. Following standard practice, we use a patch size of 4, our DDT-XL/4 achieved 2.11 FID.

D. Pixel Space Text-to-Image Experiments.

We conduct pixel space T2I without VAE on a publicly collected dataset of 20M images (SAM, CC12M, and JourneyDB). Using a large patch size of 16 for efficiency, our PixDDT-XXL/16 (1.2B Params) trained on 256×256 size achieves an overall score 54.7 on the GenEval Benchmark—improving to 65.7 with prompt rewriting(higher efficiency and better than recent PixelFlow).

E. Linear flow and Diffusion

Given the SDE forward and reverse process:

$$d\boldsymbol{x}_t = f(t)\boldsymbol{x}_t \mathrm{d}t + g(t)\mathrm{d}\boldsymbol{w} \tag{1}$$

$$d\boldsymbol{x}_t = [f(t)\boldsymbol{x}_t - g(t)^2 \nabla_{\boldsymbol{x}} \log p(\boldsymbol{x}_t)]dt + g(t)d\boldsymbol{w} \tag{2}$$

A corresponding deterministic process exists with trajectories sharing the same marginal probability densities of reverse SDE.

$$d\boldsymbol{x}_t = [f(t)\boldsymbol{x}_t - \frac{1}{2} g(t)^2 \nabla_{\boldsymbol{x}_t} \log p(\boldsymbol{x}_t)]dt \tag{3}$$

Given $x_t = \alpha_t x_{data} + \sigma\epsilon$. The traditional diffusion model learns:

$$\nabla_{\boldsymbol{x}_t} \log p(\boldsymbol{x}_t) = -\frac{\epsilon}{\sigma(t)} \tag{4}$$

The flow-matching framework actually learns the following:

$$\boldsymbol{v}_t = \dot{\alpha} x + \dot{\sigma}\epsilon \tag{5}$$

$$= x - \epsilon \tag{6}$$

Here we will demonstrate in flow-matching, the $\boldsymbol{v}_t$ prediction is actually as same as the reverse ode:

$$\dot{\alpha} x + \dot{\sigma}\epsilon \tag{7}$$

$$= f(t)\boldsymbol{x}_t - \frac{1}{2} g(t)^2 \nabla_{\boldsymbol{x}_t} \log p(\boldsymbol{x}_t) \tag{8}$$

Let us start by expanding the reverse ode first.

$$f(t)\boldsymbol{x}_t - \frac{1}{2} g(t)^2 \nabla_{\boldsymbol{x}_t} \log p(\boldsymbol{x}_t) \tag{9}$$

$$= f(t)(\alpha(t)\boldsymbol{x}_{data} + \sigma(t)\epsilon) - \frac{1}{2} g(t)^2 [-\frac{\epsilon}{\sigma(t)}] \tag{10}$$

$$= f(t)\alpha(t)\boldsymbol{x}_{data} + (f(t)\sigma(t) + \frac{1}{2}\frac{g(t)^2}{\sigma(t)})\epsilon \tag{11}$$

To prove Eq. (8), we needs to demonstrate that:

$$\dot{\alpha}(t) = f_t \alpha(t) \tag{12}$$

$$\dot{\sigma}(t) = f_t \sigma(t) + \frac{1}{2}\frac{g_t^2}{\sigma(t)}. \tag{13}$$

Here, let us derive the relation between f_t and $\alpha(t), \dot{\alpha}(t)$. We donate $x_{data}(t) = \alpha(t) x_{data}$ is the remain component of x_{data} in x_t, it is easy to find that:

$$d\boldsymbol{x}_{data}(t) = f_t \boldsymbol{x}_{data}(t) dt \tag{14}$$

$$d(\alpha(t) x_{data}) = f_t \alpha(t) x_{data} dt \tag{15}$$

$$d\alpha(t) = f_t \alpha(t) dt \tag{16}$$

So, Eq. (12) is right.

Based on the above equation, we will demonstrate the relation of g_t, f_t with $\sigma(t)$. Note that Gaussian noise has nice additive properties.

$$a\epsilon_1 + b\epsilon_2 \in \mathcal{N}(0, \sqrt{a^2 + b^2}) \tag{17}$$

Let us start with the gaussian noise component $\epsilon(t)$ calculation, reaching at t, every noise addition at $s \in [0, t]$ while been decayed by a factor of $\frac{\alpha(t)}{\alpha(s)}$. Thus, the mixed Gaussian noise will have a std variance $\sigma(t)$ of:

$$\sigma(t) = \sqrt{(\int_0^t [(\frac{\alpha(t)}{\alpha(s)})^2 g_s^2] ds)} \tag{18}$$

$$\sigma(t) = \alpha(t)\sqrt{(\int_0^t [(\frac{g_s}{\alpha(s)})^2] ds)} \tag{19}$$

After obtaining the relation of f_t, g_t and $\alpha(t), \sigma(t)$, we derive $\dot{\alpha}(t)$ and $\dot{\sigma}(t)$ with above conditions:

$$\dot{\alpha}(t) = f_t \exp[\int_0^t f_s ds] \tag{20}$$

$$\dot{\alpha}(t) = f_t \alpha(t) \tag{21}$$

As for $\dot{\sigma}(t)$, it is quit complex but not hard:

$$\dot{\sigma}(t) = \dot{\alpha}(t)\sqrt{(\int_0^t [(\frac{g_t}{\alpha(s)})^2]ds)} + \alpha(t)\frac{\frac{1}{2}\frac{g_t^2}{\alpha(t)}}{\sqrt{(\int_0^t [(\frac{g_t}{\alpha(s)})^2 g_s^2]ds)}} \tag{22}$$

$$\dot{\sigma}(t) = (f_t\alpha(t))\sqrt{(\int_0^t [(\frac{g_t}{\alpha(s)})^2]ds)} + \alpha(t)\frac{\frac{1}{2}\frac{g_t^2}{\alpha^2(t)}}{\sqrt{(\int_0^t [(\frac{g_t}{\alpha(s)})^2]ds)}} \tag{23}$$

$$\dot{\sigma}(t) = f_t\alpha(t)\sqrt{(\int_0^t [(\frac{g_t}{\alpha(s)})^2]ds)} + \frac{\frac{1}{2}g_t^2}{\alpha(t)\sqrt{(\int_0^t [(\frac{g_t}{\alpha(s)})^2]ds)}} \tag{24}$$

$$\dot{\sigma}(t) = f_t\sigma(t) + \frac{1}{2}\frac{gt}{\sigma(t)} \tag{25}$$

So, Eq. (13) is right.

F. Proof of Spectrum Autoregressive

Lemma 1. *For a linear flow-matching noise scheduler $\{\alpha_t = t, \sigma_t = 1 - t\}$, let us denote K_{freq} as the maximum frequency of the clean data $\boldsymbol{x}_{data}$, $\mathcal{R}(\boldsymbol{x})[f_{max}]$ as the spectral power of f_{max} in $\boldsymbol{x}$. We suppose the power monopoly decreases along with the frequency. The maximum retained frequency satisfies:*

$$R(\boldsymbol{x}_t)[f_{max}(t)] > \left(\frac{t}{1-t}\right)^2 \textit{ and } \; f_{max}(t) < K_{freq}. \tag{26}$$

Given the noise scheduler$\{\alpha_t, \sigma_t\}$, the clean data $\boldsymbol{x}_{\text{data}}$ and Gaussian noise ϵ. Denote K_{freq} as the maximum frequency of the clean data $\boldsymbol{x}_{data}$ The noisy latent x_t at timestep t has been defined as:

$$\boldsymbol{x}_t = \alpha_t \boldsymbol{x}_{data} + \sigma_t \boldsymbol{\epsilon} \tag{27}$$

The spectrum magnitude $\boldsymbol{c}_i$of x_t on DCT basics $\boldsymbol{u}_i$ follows:

$$\boldsymbol{c}_i = \mathbb{E}_\epsilon[\boldsymbol{u}_i^T \boldsymbol{x}_t]^2$$
$$\boldsymbol{c}_i = \mathbb{E}_\epsilon[\boldsymbol{u}_i^T (\alpha_t \boldsymbol{x}_{data} + \sigma_t \boldsymbol{\epsilon})]^2$$

Recall that the spectrum magnitude of Gaussian noise ϵ is uniformly distributed.

Proof.

$$\boldsymbol{c}_i = \mathbb{E}_\epsilon[\boldsymbol{u}_i^T (\alpha_t \boldsymbol{x}_{data} + \sigma_t \boldsymbol{\epsilon})]^2 \tag{28}$$

$$\boldsymbol{c}_i = \alpha_t^2 [\boldsymbol{u}_i^T \boldsymbol{x}_{data}]^2 + \sigma_t^2 \lambda \tag{29}$$

$$\boldsymbol{c}_i = \alpha_t^2 R(\boldsymbol{x}_t)[f_i] + \sigma_t^2 \lambda \tag{30}$$

Note that ϵ is a standard Gaussian noise, so $\lambda = 1$. If $\alpha_t^2 R(\boldsymbol{x}_t)[f_i]$ smaller that σ_t^2, the frequency at c_i will be canceled. Recall that the power monopoly decreases along with the frequency. Thus, the spectral power of f_{max} needs to satisfy:

$$\alpha_t^2 R(\boldsymbol{x}_t)[f_{max}] > \sigma_t^2 \tag{31}$$

$$R(\boldsymbol{x}_t)[f_{max}] > \frac{\sigma_t^2}{\alpha_t^2} \tag{32}$$

$$R(\boldsymbol{x}_t)[f_{max}] > \left(\frac{1-t}{t}\right)^2 \tag{33}$$

□

Lemma 2. *The spectrum magnitude $R(f)$ of a specific frequency f on general nature signals follows:*

$$R(f) = c \cdot f^{-\beta} \tag{34}$$

where, c is a positive numbers and $\beta \approx 2$ in 2-dimension data.

Then, we can obtain the max remaining frequency changes along with timesteps t.

$$c \cdot f_{max}(t)^{-\beta} > \left(\frac{1-t}{t}\right)^2 \tag{35}$$

$$f_{max}(t) > (\frac{1}{c})^{-\frac{1}{\beta}} \left(\frac{t}{1-t}\right) \tag{36}$$

It is clear that when approach to clean data(t to 1.0), the residual frequency $\frac{df_{max}(t)}{dt}$ increase.

G. Linear multisteps method

We conduct targeted experiment on SiT-XL/2 with Adams–Bashforth like linear multistep solver; To clarify, we did not employ this powerful solver for our DDT models in all tables across the main paper.

The reverse ode of the diffusion models tackles the following integral:

$$\boldsymbol{x}_{i+1} = \boldsymbol{x}_i + \int_{t_i}^{t_{i+1}} \boldsymbol{v}_\theta(\boldsymbol{x}_t, t) dt \tag{37}$$

The classic Euler method employs $\boldsymbol{v}_\theta(\boldsymbol{x}_i, t_i)$ as an estimate of $\boldsymbol{v}_\theta(\boldsymbol{x}_t, t)$ throughout the interval $[t_i, t_{i+1}]$

$$\boldsymbol{x}_{i+1} = \boldsymbol{x}_i + (t_{i+1} - t_i)\boldsymbol{v}_\theta(\boldsymbol{x}_i, t_i). \tag{38}$$

The most classic multi-step solver Adams–Bashforth method (deemed as Adams for brevity) incorporates the Lagrange polynomial to improve the estimation accuracy with previous predictions.

$$\boldsymbol{v}_\theta(\boldsymbol{x}_t, t) = \sum_{j=0}^{i} \Big(\prod_{k=0, k\neq j}^{i} \frac{t - t_k}{t_j - t_k} \Big) \boldsymbol{v}_\theta(\boldsymbol{x}_j, t_j)$$

$$\boldsymbol{x}_{i+1} \approx \boldsymbol{x}_i + \int_{t_i}^{t_{i+1}} \sum_{j=0}^{i} \Big(\prod_{k=0, k\neq j}^{i} \frac{t - t_k}{t_j - t_k} \Big) \boldsymbol{v}_\theta(\boldsymbol{x}_j, t_j) dt$$

$$\boldsymbol{x}_{i+1} \approx \boldsymbol{x}_i + \sum_{j=0}^{i} \boldsymbol{v}_\theta(\boldsymbol{x}_j, t_j) \int_{t_i}^{t_{i+1}} \Big(\prod_{k=0, k\neq j}^{i} \frac{t - t_k}{t_j - t_k} \Big) dt$$

Note that $\int_{t_i}^{t_{i+1}} (\prod_{k=0,k\neq j}^{i} \frac{t-t_k}{t_j-t_k}) dt$ of the Lagrange polynomial can be pre-integrated into a constant coefficient, resulting in only naive summation being required for ODE solving.

H. Classifier free guidance.

As classifier-free guidance significantly impacts the performance of diffusion models. Traditional classifier-free guidance improves performance at the cost of decreased diversity. Interval guidance is recently been adopted by REPA[4] and Causalfusion[1], It applies classifier-free guidance only to the high-frequency generation phase to preserve the diversity. We sweep different classifier-free guidance strength with selected intervals. Our DDT-XL/2 achieves the best performance with interval $[0.3, 1]$ with a classifer-free guidance of 2. Recall that we donate $t = 0$ as the pure noise timestep while REPA[4] use $t = 1$, thus this exactly correspond to the $[0, 0.7]$ interval in REPA[4]

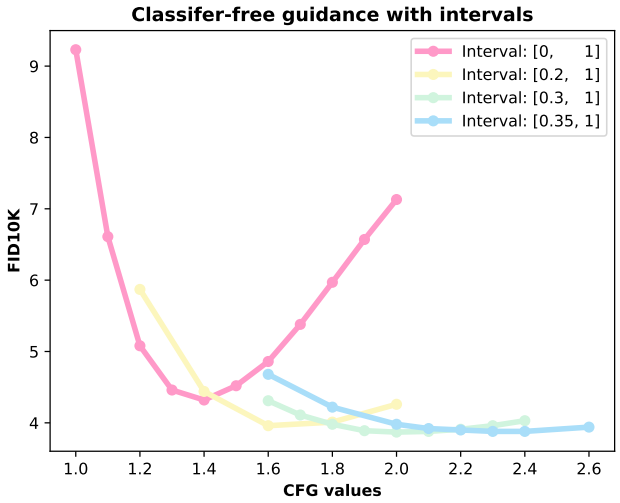

Figure 1. **FID10K of DDT-XL/2 with different Classifer free guidance strength and guidance intervals.** We sweep different classifier-free guidance strength with selected intervals. Our DDT-XL/2 achieves the best performance with interval $[0.3, 1]$ with a classifer-free guidance of 2.

References

[1] Chaorui Deng, Deyao Zh, Kunchang Li, Shi Guan, and Haoqi Fan. Causal diffusion transformers for generative modeling. *arXiv preprint arXiv:2412.12095*, 2024. 3

[2] Prafulla Dhariwal and Alexander Nichol. Diffusion models beat gans on image synthesis. *Advances in neural information processing systems*, 34:8780–8794, 2021. 1

[3] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014. 1

[4] Sihyun Yu, Sangkyung Kwak, Huiwon Jang, Jongheon Jeong, Jonathan Huang, Jinwoo Shin, and Saining Xie. Representation alignment for generation: Training diffusion transformers is easier than you think. *arXiv preprint arXiv:2410.06940*, 2024. 3