PFGM: Poisson Flow Generative Models

Kaiwen Zheng

2023.03.31

Content

- **Background: Velocity Field (VF) for Generative Modeling**
- PFGM: VF defined by Poisson Flow
- PFGM++: bridging PFGM and Diffusion Models

we only care about continuous case/ODE form

- Summary

Generative Modeling Through Velocity Field

For N-dimensional data $\boldsymbol{x}_0$

$$\frac{d\boldsymbol{x}_t}{dt} = \boldsymbol{v}(\boldsymbol{x}_t, t)$$

q_0 —— degeneration ——→ q_1

q_0 ←—— generation —— q_1

Data distribution | Prior distribution (easy for sampling)

Diffusion models:

- $\boldsymbol{v}$ is the flow of diffusion process
- q_1 is (approximate) Gaussian

Learn to Generate by Matching Velocity Field

Two ODEs:

$$\frac{d\boldsymbol{x}_t}{dt} = \boldsymbol{v}(\boldsymbol{x}_t, t)$$

Pre-defined; Gound-truth

$$\frac{d\boldsymbol{x}_t}{dt} = \boldsymbol{v}_\theta(\boldsymbol{x}_t, t)$$

Model

Objective:

For any point $\boldsymbol{x} \epsilon \mathrm{R}^N$, minimize $E_t[\|\boldsymbol{v}_\theta(\boldsymbol{x}, t) - \boldsymbol{v}(\boldsymbol{x}, t)\|_2^2]$

But $\boldsymbol{v}(\boldsymbol{x}, t)$ is **not tractable**

Transition Kernel and Conditional Velocity

Solution:

1. Regress $\boldsymbol{v}(\boldsymbol{x}, t)$ **unbiasedly**:

   - Suppose we pre-define not the marginal q_t, but the transition kernel q_{0t}, so that

$$q_t(\boldsymbol{x}) = \int q_0(\boldsymbol{x}_0) q_{0t}(\boldsymbol{x}|\boldsymbol{x}_0) d\boldsymbol{x}_0$$

   - And the corresponding conditional velocity v_{0t} in the conditional ODE

$$\frac{d\boldsymbol{x}_t|\boldsymbol{x}_0}{dt} = \boldsymbol{v}_{0t}(\boldsymbol{x}_t|\boldsymbol{x}_0)$$

$$q_{0\epsilon} \rightleftarrows q_{01}$$

   - By the continuity equation, we can prove [3]

$$\boldsymbol{v}_t(\boldsymbol{x}) = \frac{\int q_0(\boldsymbol{x}_0) q_{0t}(\boldsymbol{x}|\boldsymbol{x}_0) \boldsymbol{v}_{0t}(\boldsymbol{x}|\boldsymbol{x}_0) d\boldsymbol{x}_0}{q_t(\boldsymbol{x})}$$

Score Matching and Flow Matching

Usually, we define simple transition q_{0t}

- Sample from q_t is easy
- $\nabla \log q_{0t}, \boldsymbol{v}_{0t}$ is tractable

We have the relation

$$\nabla \log q_t(\boldsymbol{x}) = E_{q_{t0}(\boldsymbol{x}_0|\boldsymbol{x})}[\nabla \log q_{0t}(\boldsymbol{x}|\boldsymbol{x}_0)]$$

$$\boldsymbol{v}(\boldsymbol{x}, t) = E_{q_{t0}(\boldsymbol{x}_0|\boldsymbol{x})}[\boldsymbol{v}_{0t}(\boldsymbol{x}|\boldsymbol{x}_0)]$$

intractable posterior

→

Denoising score matching

$$\min_\theta E_{q_0(\boldsymbol{x}_0)q_{0t}(\boldsymbol{x}_t|\boldsymbol{x}_0)}[\|\boldsymbol{s}_\theta(\boldsymbol{x}_t, t) - \nabla \log q_{0t}(\boldsymbol{x}_t|\boldsymbol{x}_0)\|_2^2]$$

$$\min_\theta E_{q_0(\boldsymbol{x}_0)q_{0t}(\boldsymbol{x}_t|\boldsymbol{x}_0)}[\|\boldsymbol{v}_\theta(\boldsymbol{x}_t, t) - \boldsymbol{v}_{0t}(\boldsymbol{x}_t|\boldsymbol{x}_0)\|_2^2]$$

Conditional flow matching

Biased Estimator Using Finite Points

1. Regress $\boldsymbol{v}(\boldsymbol{x}, t)$ **unbiasedly**
2. Estimate $\boldsymbol{v}(\boldsymbol{x}, t)$ **biasedly** using a batch of reference samples:
   - The data distribution can be viewed as N discrete points

$$q_0(\boldsymbol{x}) = \frac{1}{N}\sum_i \delta(\boldsymbol{x} - \boldsymbol{x}^{(i)})$$

   - Then

Marginal density $$q_t(\boldsymbol{x}) = \frac{1}{N}\sum_i q_{0t}\left(\boldsymbol{x}\middle|\boldsymbol{x}^{(i)}\right)$$ Biased but lower variance

Score $$\nabla \log q_t(\boldsymbol{x}) = \frac{\sum_i q_{0t}\left(\boldsymbol{x}\middle|\boldsymbol{x}^{(i)}\right)\nabla \log q_{0t}\left(\boldsymbol{x}\middle|\boldsymbol{x}^{(i)}\right)}{\sum_i q_{0t}\left(\boldsymbol{x}\middle|\boldsymbol{x}^{(i)}\right)}$$

Velocity field $$\boldsymbol{v}_t(\boldsymbol{x}) = \frac{\sum_i q_{0t}\left(\boldsymbol{x}\middle|\boldsymbol{x}^{(i)}\right)\boldsymbol{v}_{0t}\left(\boldsymbol{x}\middle|\boldsymbol{x}^{(i)}\right)}{\sum_i q_{0t}\left(\boldsymbol{x}\middle|\boldsymbol{x}^{(i)}\right)}$$

Special Case: Diffusion Models

In diffusion models, we pre-define **noise schedule** α_t, σ_t, then

- $q_{0t}(\boldsymbol{x}_t|\boldsymbol{x}_0) = N(\boldsymbol{x}_t; \alpha_t \boldsymbol{x}_0, \sigma_t^2 \boldsymbol{I})$
- $\nabla \log q_{0t}(\boldsymbol{x}_t|\boldsymbol{x}_0) = -\frac{\boldsymbol{x}_t - \alpha_t \boldsymbol{x}_0}{\sigma_t^2} = -\frac{\boldsymbol{\varepsilon}}{\sigma_t}$
- $\boldsymbol{v}_{0t}(\boldsymbol{x}_t|\boldsymbol{x}_0) = \left(\dot{\alpha}_t - \frac{\dot{\sigma}_t}{\sigma_t}\alpha_t\right)\boldsymbol{x}_0 + \frac{\dot{\sigma}_t}{\sigma_t}\boldsymbol{x}_t = \dot{\alpha}_t \boldsymbol{x}_0 + \dot{\sigma}_t \boldsymbol{\varepsilon}$

Are there other types of pre-defined velocity field to match?

Content

- Background: Velocity Field (VF) for Generative Modeling
- **PFGM: VF defined by Poisson Flow**
- PFGM++: bridging PFGM and Diffusion Models

  we only care about continuous case/ODE form
- Summary

Poisson Equation

Poisson equation $$\nabla^2 \varphi(\mathbf{x}) = -\rho(\mathbf{x}),$$

$$\mathbf{E}(\mathbf{x}) = -\nabla \varphi(\mathbf{x})$$

Poisson field

Gauss's law

$$\nabla \cdot \mathbf{E} = \rho$$

density

Instances:

- Electric field $$\nabla \cdot \mathbf{E} = \frac{\rho}{\varepsilon}$$
- Gravitational field $$\nabla \cdot \mathbf{g} = -4\pi G \rho$$

Poisson Field

The Poisson field satisfies

Polynomial decay

$$\mathbf{E}(\mathbf{x}) = -\nabla\varphi(\mathbf{x}) = -\int \nabla_{\mathbf{x}} G(\mathbf{x},\mathbf{y})\rho(\mathbf{y})d\mathbf{y}, \quad \nabla_{\mathbf{x}} G(\mathbf{x},\mathbf{y}) = -\frac{1}{S_{N-1}(1)} \frac{\mathbf{x}-\mathbf{y}}{||\mathbf{x}-\mathbf{y}||^N}.$$

surface area of (N-1)-dim unit sphere

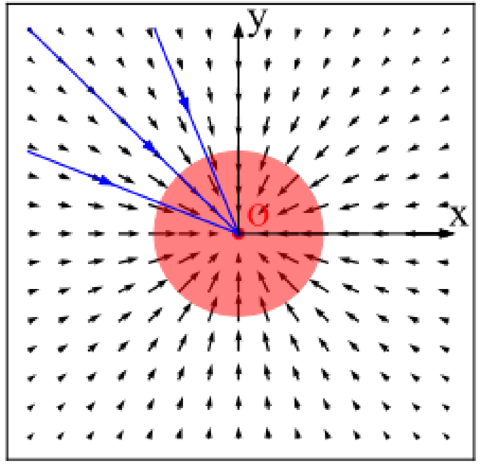
ODE (degeneration direction): $\frac{d\mathbf{x}}{dt} = \mathbf{E}(\mathbf{x})$

- No time anchor t in the field.
- Limit: Uniform distribution on the N-dim hyper-hemisphere of infinite radius

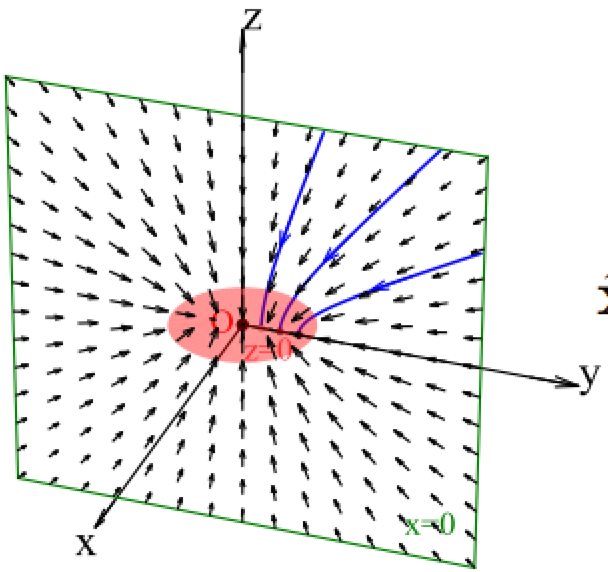
Can we match $\mathbf{E}(\mathbf{x})$ and use it for generating? No!

Learn the (N+1)-dim Field

The N-dim field can't generate.



Add an additional dim z. Stop when z hit 0.



$$\tilde{\mathbf{x}} = (\mathbf{x}, z) \in \mathbb{R}^{N+1}$$

- z also acts as a **time anchor**

$$d(\mathbf{x}, z) = (\frac{d\mathbf{x}}{dt}\frac{dt}{dz}dz, dz) = (\mathbf{v}(\tilde{\mathbf{x}})_{\mathbf{x}}\mathbf{v}(\tilde{\mathbf{x}})_z^{-1}, 1)dz$$

Normalized velocity. But any factor will be cancelled.

Learn Normalized Field

Recall

$$\forall \tilde{\mathbf{x}} \in \mathbb{R}^{N+1}, \mathbf{E}(\tilde{\mathbf{x}}) = -\nabla\varphi(\tilde{\mathbf{x}}) = \frac{1}{S_N(1)} \int \frac{\tilde{\mathbf{x}} - \tilde{\mathbf{y}}}{||\tilde{\mathbf{x}} - \tilde{\mathbf{y}}||^{N+1}} \tilde{p}(\tilde{\mathbf{y}}) d\tilde{\mathbf{y}}$$

- Empirical estimator (**biased**)

$$\hat{\mathbf{E}}(\tilde{\mathbf{x}}) = c(\tilde{\mathbf{x}}) \sum_{i=1}^{n} \frac{\tilde{\mathbf{x}} - \tilde{\mathbf{x}}_i}{||\tilde{\mathbf{x}} - \tilde{\mathbf{x}}_i||^{N+1}}$$

$$c(\tilde{\mathbf{x}}) \downarrow \quad 1/\sum_{i=1}^{n} \frac{1}{||\tilde{\mathbf{x}} - \tilde{\mathbf{x}}_i||^{N+1}}$$

- Further, normalize the field

$$\mathbf{v}(\tilde{\mathbf{x}}) = -\sqrt{N}\hat{\mathbf{E}}(\tilde{\mathbf{x}}) / \| \hat{\mathbf{E}}(\tilde{\mathbf{x}}) \|_2$$

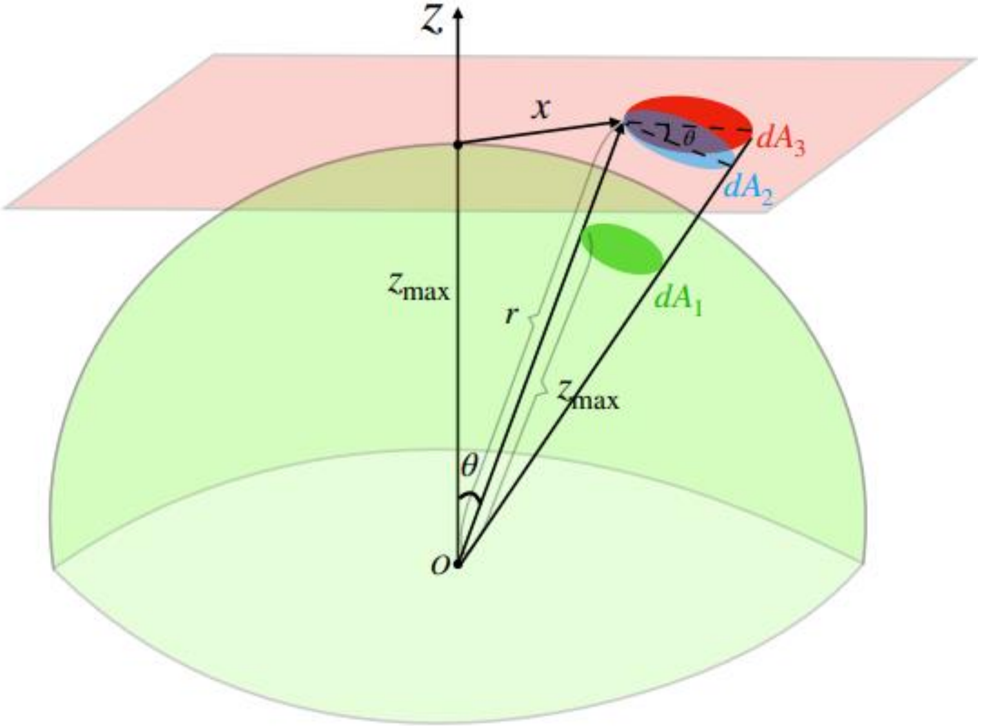
$$\mathcal{L}(\theta) = \frac{1}{|\mathcal{B}|} \sum_{i=1}^{|\mathcal{B}|} \| f_\theta(\tilde{\mathbf{y}}_i) - \mathbf{v}_{\mathcal{B}_L}(\tilde{\mathbf{y}}_i) \|_2^2$$

$$\tilde{\boldsymbol{x}}_i = (\boldsymbol{x}_i, 0) \xrightarrow{\text{perturb}} \tilde{\boldsymbol{y}}_i = (\boldsymbol{y}_i, z)$$

$$\mathbf{y} = \mathbf{x} + \| \epsilon_{\mathbf{x}} \| (1+\tau)^m \mathbf{u}, \quad z = |\epsilon_z|(1+\tau)^m$$

Prior Sampling

Prior sampling on the $z = z_{\mathbf{max}}$ hyperplane



$$p_{\text{prior}}(\mathbf{x}) = \frac{2z_{\text{max}}^{N+1}}{S_N(z_{\text{max}})(\|\mathbf{x}\|_2^2 + z_{\text{max}}^2)^{\frac{N+1}{2}}} = \frac{2z_{\text{max}}}{S_N(1)(\|\mathbf{x}\|_2^2 + z_{\text{max}}^2)^{\frac{N+1}{2}}}$$

1. Sample the radius from $p_{\text{radius}}(\|\mathbf{x}\|_2) \propto \|\mathbf{x}\|_2^{N-1}/(\|\mathbf{x}\|_2^2 + z_{\text{max}}^2)^{\frac{N+1}{2}}$
2. Uniform sample the angle

Experiments

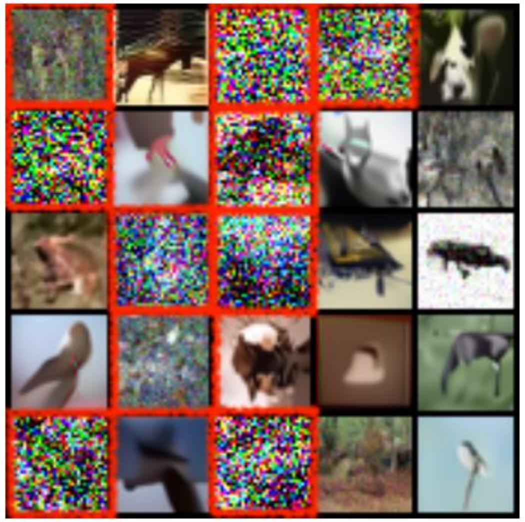
Table 1: CIFAR-10 sample quality (FID, Inception) and number of function evaluation (NFE).

	Invertible?	Inception ↑	FID ↓	NFE ↓
PixelCNN [36]	✗	4.60	65.9	1024
IGEBM [8]	✗	6.02	40.6	60
ViTGAN [24]	✗	9.30	6.66	1
StyleGAN2-ADA [17]	✗	9.83	2.92	1
StyleGAN2-ADA (cond.) [17]	✗	10.14	2.42	1
NCSN [31]	✗	8.87	25.32	1001
NCSNv2 [32]	✗	8.40	10.87	1161
DDPM [16]	✗	9.46	3.17	1000
NCSN++ VE-SDE [33]	✗	9.83	2.38	2000
NCSN++ deep VE-SDE [33]	✗	9.89	2.20	2000
Glow [19]	✓	3.92	48.9	1
DDIM, T=50 [30]	✓	-	4.67	50
DDIM, T=100 [30]	✓	-	4.16	100
NCSN++ VE-ODE [33]	✓	9.34	5.29	194
NCSN++ deep VE-ODE [33]	✓	9.17	7.66	194
DDPM++ backbone				
VP-SDE [33]	✗	9.58	2.55	1000
sub-VP-SDE [33]	✗	9.56	2.61	1000
VP-ODE [33]	✓	9.46	2.97	134
sub-VP-ODE [33]	✓	9.30	3.16	146
PFGM (ours)	✓	**9.65**	**2.48**	**104**
DDPM++ deep backbone				
VP-SDE [33]	✗	9.68	2.41	1000
sub-VP-SDE [33]	✗	9.57	2.41	1000
VP-ODE [33]	✓	9.47	2.86	134
sub-VP-ODE [33]	✓	9.40	3.05	146
PFGM (ours)	✓	**9.68**	**2.35**	**110**

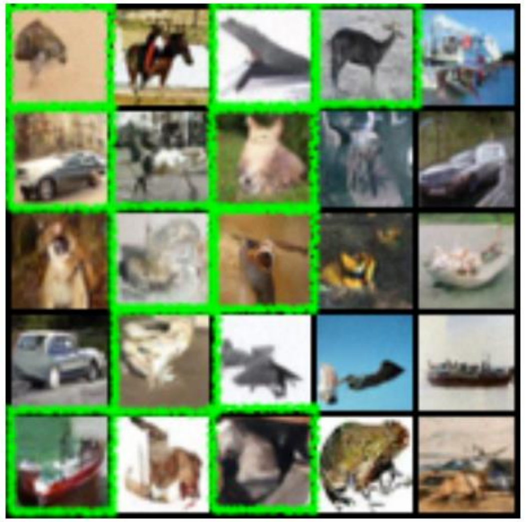
Experiments

Failure of VE/VP-ODEs on NCSNv2 architecture

VE transition: $\mathcal{N}(\mathbf{x}, \sigma(t)^2)$



(a) Samples from VE-ODE (Euler)



(b) Samples from VE-ODE (Euler w/ corrector)

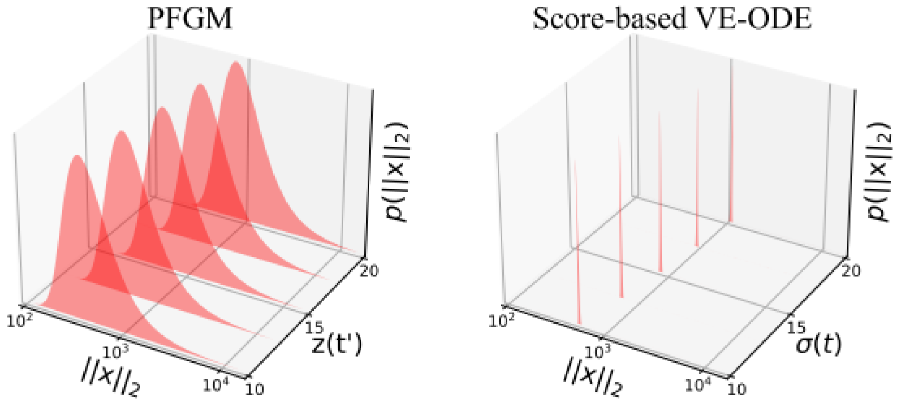


Figure 4: Sample norm distributions with varying time variables (σ for VE-ODE and z for PFGM)

norm-σ correlation

Experiments

Exponential decay on *z* for samping

$$d(\mathbf{x}, z) = (\frac{d\mathbf{x}}{dt}\frac{dt}{dz}dz, dz) = (\mathbf{v}(\tilde{\mathbf{x}})_{\mathbf{x}}\mathbf{v}(\tilde{\mathbf{x}})_z^{-1}, 1)dz \longrightarrow d(\mathbf{x}, z) = (\mathbf{v}(\tilde{\mathbf{x}})_{\mathbf{x}}\mathbf{v}(\tilde{\mathbf{x}})_z^{-1}z, z)dt'$$

Integral form: $\mathbf{x}(\log z_{\max}) = \mathcal{M}(\mathbf{x}(\log z_{\min})) \equiv \mathbf{x}(\log z_{\min}) + \int_{\log z_{\min}}^{\log z_{\max}} \mathbf{v}(\mathbf{x}(t'))_{\mathbf{x}}\mathbf{v}(\tilde{\mathbf{x}}(t'))_z^{-1} e^{t'} dt'$

Results for RK45 solver:

Algorithm	$d(\mathbf{x}, z)/dz$	$d(\mathbf{x}, z)/dt'$
NFE	242	104
FID score	2.53	2.48

Experiments

Likelihood evaluation and latent representation

Table 2: Bits/dim on CIFAR-10

	bits/dim ↓
RealNVP [6]	3.49
Glow [19]	3.35
Residual Flow [3]	3.28
Flow++ [14]	3.29
DDPM (L) [16]	$\leq 3.70^*$
DDPM++ backbone	
VP-ODE [33]	3.20
sub-VP-ODE [33]	**3.02**
PFGM (ours)	3.19

Figure 10: Interpolation on CelebA 64×64 by PFGM

Content

- Background: Velocity Field (VF) for Generative Modeling
- PFGM: VF defined by Poisson Flow
- **PFGM++: bridging PFGM and Diffusion Models**

  → we only care about continuous case/ODE form
- Summary

D-augmented PFGM

Augment with D-dim *z*: $\tilde{\mathbf{x}} = (\mathbf{x}, \mathbf{z}) \in \mathbb{R}^{N+D}$

Poisson field in (N+D)-dim:

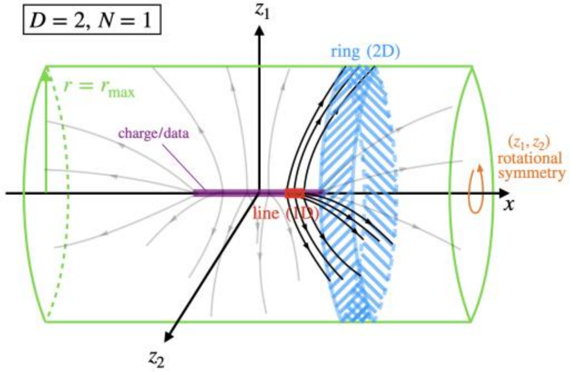
$$\mathbf{E}(\tilde{\mathbf{x}}) = \frac{1}{S_{N+D-1}(1)} \int \frac{\tilde{\mathbf{x}} - \tilde{\mathbf{y}}}{||\tilde{\mathbf{x}} - \tilde{\mathbf{y}}||^{N+D}} p(\mathbf{y}) d\mathbf{y} \quad (3)$$

ODE: $d\tilde{\mathbf{x}} = \mathbf{E}(\tilde{\mathbf{x}})\mathrm{d}t$

N-dim data distribution on $\boldsymbol{z} = 0$ hyperplane → uniform distribution on an infinite (N+D)-dim hemisphere

r-anchor

- Due to symmetry, it's sufficient to track $r(\tilde{\mathbf{x}}) = \|\mathbf{z}\|_2$



- By change-of-variable, we have
  - r-augmented point and field

$$\tilde{\mathbf{x}} = (\mathbf{x}, r) \qquad E(\tilde{\mathbf{x}})_r = \frac{1}{S_{N+D-1}(1)} \int \frac{r}{\|\tilde{\mathbf{x}}-\tilde{\mathbf{y}}\|^{N+D}} p(\mathbf{y}) \mathrm{d}\mathbf{y}.$$

  - ***r*-anchored** ODE

$$\frac{\mathrm{d}\mathbf{x}}{\mathrm{d}r} = \frac{\mathrm{d}\mathbf{x}}{\mathrm{d}t}\frac{\mathrm{d}t}{\mathrm{d}r} = \mathbf{E}(\tilde{\mathbf{x}})_{\mathbf{x}} \cdot E(\tilde{\mathbf{x}})_r^{-1} = \frac{\mathbf{E}(\tilde{\mathbf{x}})_{\mathbf{x}}}{E(\tilde{\mathbf{x}})_r}$$

Training Objective

(Old) biased estimator

$$\mathbb{E}_{\tilde{p}_{\text{train}}(\tilde{\mathbf{x}})}\mathbb{E}_{\{\mathbf{y}_i\}_{i=1}^n \sim p^n(\mathbf{y})}\mathbb{E}_{\mathbf{x}\sim p_\sigma(\mathbf{x}|\mathbf{y}_1)} \left[\left\| f_\theta(\tilde{\mathbf{x}}) - \frac{\sum_{i=0}^{n-1} \frac{\tilde{\mathbf{x}}-\tilde{\mathbf{y}}_i}{\|\tilde{\mathbf{x}}-\tilde{\mathbf{y}}_i\|^{N+D}}}{\left\|\sum_{i=0}^{n-1} \frac{\tilde{\mathbf{x}}-\tilde{\mathbf{y}}_i}{\|\tilde{\mathbf{x}}-\tilde{\mathbf{y}}_i\|^{N+D}}\right\|_2 + \gamma} \right\|_2^2\right]$$

(**New**) unbiased regression

- Transition kernel of r-augmented Poisson field

$$p_r(\mathbf{x}|\mathbf{y}) \propto 1/(\|\mathbf{x}-\mathbf{y}\|_2^2 + r^2)^{\frac{N+D}{2}}.$$

$p_r(\mathbf{x}|\mathbf{y}) \rightarrow q_{0t}$

1. Sample $p_r(R) \propto \frac{R^{N-1}}{(R^2+r^2)^{\frac{N+D}{2}}}$ $R = \|\mathbf{x}-\mathbf{y}\|_2$
2. Uniform sample the angle

Training Objective

(**New**) unbiased regression

- Transition kernel of r-augmented Poisson field

$$p_r(\mathbf{x}|\mathbf{y}) \propto 1/(\|\mathbf{x}-\mathbf{y}\|_2^2 + r^2)^{\frac{N+D}{2}}. \longrightarrow \|\tilde{\boldsymbol{x}} - \tilde{\boldsymbol{y}}\|_2^{N+D}$$

$\downarrow$

q_{0t}

- To obtain the conditional velocity $\boldsymbol{v}_{0t}$

$$\mathbf{E}(\tilde{\mathbf{x}}) = \frac{1}{S_{N+D-1}(1)} \int \frac{\tilde{\mathbf{x}} - \tilde{\mathbf{y}}}{\|\tilde{\mathbf{x}} - \tilde{\mathbf{y}}\|^{N+D}} p(\mathbf{y}) d\mathbf{y} \quad (3)$$

$$\boldsymbol{v}_t(\boldsymbol{x}) = \frac{\int q_0(\boldsymbol{x}_0) q_{0t}(\boldsymbol{x}|\boldsymbol{x}_0) \boldsymbol{v}_{0t}(\boldsymbol{x}|\boldsymbol{x}_0) d\boldsymbol{x}_0}{q_t(\boldsymbol{x})}$$

$$\Rightarrow \quad \boldsymbol{v}_{0r}(\tilde{\boldsymbol{x}}|\tilde{\boldsymbol{y}}) = \frac{p_r(\tilde{\boldsymbol{x}})}{S_{N+D-1}} (\tilde{\boldsymbol{x}} - \tilde{\boldsymbol{y}})$$

- Predict the normalized conditional velocity

$$\mathbb{E}_{r\sim p(r)} \mathbb{E}_{p(\tilde{\mathbf{y}})} \mathbb{E}_{p_r(\tilde{\mathbf{x}}|\tilde{\mathbf{y}})} \left[\left\| f_\theta(\tilde{\mathbf{x}}) - \frac{\mathbf{x}-\mathbf{y}}{r/\sqrt{D}} \right\|_2^2 \right]$$

$$(\tilde{\mathbf{x}} - \tilde{\mathbf{y}})_r / (r/\sqrt{D}) = \sqrt{D}.$$

Diffusion Models as D→∞ Special Cases

$$p_r(\mathbf{x}|\mathbf{y}) \propto 1/(\|\mathbf{x}-\mathbf{y}\|_2^2 + r^2)^{\frac{N+D}{2}}.$$

→

$\nabla \log p_{0r}(\tilde{\boldsymbol{x}}|\tilde{\boldsymbol{y}}) \propto \tilde{\boldsymbol{x}} - \tilde{\boldsymbol{y}}$ conditional score

$\boldsymbol{v}_{0r}(\tilde{\boldsymbol{x}}|\tilde{\boldsymbol{y}}) = \frac{p_r(\tilde{\boldsymbol{x}})}{S_{N+D-1}}(\tilde{\boldsymbol{x}} - \tilde{\boldsymbol{y}}) \propto \tilde{\boldsymbol{x}} - \tilde{\boldsymbol{y}}$ conditional velocity

same direction!

This is similar to **VE schedule** in diffusion models

$$q_{0t}(\boldsymbol{x}_t|\boldsymbol{x}_0) = N(\boldsymbol{x}_t; \boldsymbol{x}_0, \sigma_t^2 \boldsymbol{I})$$

In the limit D→∞ :

$$\lim_{\substack{D\to\infty \\ r=\sigma\sqrt{D}}} \left\| \frac{\sqrt{D}}{E(\tilde{\mathbf{x}})_r}\mathbf{E}(\tilde{\mathbf{x}})_{\mathbf{x}} - \sigma\nabla_{\mathbf{x}} \log p_{\sigma=r/\sqrt{D}}(\mathbf{x}) \right\|_2 = 0$$

VF in PFGM Score in VE DMs

Alignment Relation

In the limit D→∞ :

$$\lim_{\substack{D\to\infty \\ r=\sigma\sqrt{D}}} \left\| \frac{\sqrt{D}}{E(\tilde{\mathbf{x}})_r}\mathbf{E}(\tilde{\mathbf{x}})_{\mathbf{x}} - \sigma\nabla_{\mathbf{x}} \log p_{\sigma=r/\sqrt{D}}(\mathbf{x}) \right\|_2 = 0$$

VF in PFGM Score in VE DMs

Alignment relation: $r = \sigma\sqrt{D}$

- Limit of transition kernel

$$\lim_{D\to\infty, r=\sigma\sqrt{D}} \frac{1}{(\|\mathbf{x}-\mathbf{y}\|_2^2 + r^2)^{\frac{N+D}{2}}}$$

$$\propto \lim_{D\to\infty, r=\sigma\sqrt{D}} e^{-\frac{(N+D)}{2}\ln(1+\frac{\|\mathbf{x}-\mathbf{y}\|^2}{r^2})}$$

$$= \lim_{D\to\infty, r=\sigma\sqrt{D}} e^{-\frac{(N+D)\|\mathbf{x}-\mathbf{y}\|_2^2}{2r^2}} = e^{-\frac{\|\mathbf{x}-\mathbf{y}\|_2^2}{2\sigma^2}}$$

- Limit of training objective

$$\mathbb{E}_{r\sim p(r)}\mathbb{E}_{p(\tilde{\mathbf{y}})}\mathbb{E}_{p_r(\tilde{\mathbf{x}}|\tilde{\mathbf{y}})}\left[\left\| f_\theta(\tilde{\mathbf{x}}) - \frac{\mathbf{x}-\mathbf{y}}{r/\sqrt{D}} \right\|_2^2\right] \xrightarrow{r=\sigma\sqrt{D},\, D\to\infty} \mathbb{E}_{\sigma\sim p(\sigma)}\lambda(\sigma)\mathbb{E}_{p(\mathbf{y})}\mathbb{E}_{p_\sigma(\mathbf{x}|\mathbf{y})}\left[\left\| f_\theta(\mathbf{x},\sigma) - \frac{\mathbf{x}-\mathbf{y}}{\sigma} \right\|_2^2\right]$$

DSM/FM in PFGM DSM/FM under VE in DMs

Alignment Instances

- Align hyperparameters

$$r_{\max} = \sigma_{\max}\sqrt{D},\ p(r) = p(\sigma = r/\sqrt{D})/\sqrt{D}$$

- Align the training and sampling algorithm with EDM [5]

Algorithm 1 EDM training

```
Sample a batch of data {y_i}_{i=1}^B from p(y)
Sample standard deviations {σ_i}_{i=1}^B from p(σ)
Sample noise vectors {n_i ~ N(0, σ_i^2 I)}_{i=1}^B
Get perturbed data {ŷ_i = y_i + n_i}_{i=1}^B
Calculate loss ℓ(θ) = Σ_{i=1}^B λ(σ_i) ||f_θ(ŷ_i, σ_i) − y_i||_2^2
Update the network parameter θ via Adam optimizer
```

Algorithm 2 PFGM++ training with hyperparameter transferred from EDM

```
Sample a batch of data {y_i}_{i=1}^B from p(y)
Sample standard deviations {σ_i}_{i=1}^B from p(σ)
Sample r from p_r: {r_i = σ_i √D}_{i=1}^B
Sample radiuses {R_i ~ p_{r_i}(R)}_{i=1}^B
Sample uniform angles {v_i = u_i / ||u_i||_2}_{i=1}^B, with u_i ~ N(0, I)
Get perturbed data {ŷ_i = y_i + R_i v_i}_{i=1}^B
Calculate loss ℓ(θ) = Σ_{i=1}^B λ(σ_i) ||f_θ(ŷ_i, σ_i) − y_i||_2^2
Update the network parameter θ via Adam optimizer
```

Algorithm 3 EDM sampling (Heun's 2nd order method)

```
x_0 ~ N(0, σ_max^2 I)
for i = 0, ..., T − 1 do
    d_i = (x_i − f_θ(x_i, t_i))/t_i
    x_{i+1} = x_i + (t_{i+1} − t_i) d_i
    if t_{i+1} > 0 then
        d'_i = (x_{i+1} − f_θ(x_{i+1}, t_{i+1}))/t_{i+1}
        x_{i+1} = x_i + (t_{i+1} − t_i)(1/2 d_i + 1/2 d'_i)
    end if
end for
```

Algorithm 4 PFGM++ training with hyperparameter transferred from EDM

```
Set r_max = σ_max √D
Sample radius R ~ p_{r_max}(R) and uniform angle v = u / ||u||_2, with u ~ N(0, I)
Get initial data x_0 = R v
for i = 0, ..., T − 1 do
    d_i = (x_i − f_θ(x_i, t_i))/t_i
    x_{i+1} = x_i + (t_{i+1} − t_i) d_i
    if t_{i+1} > 0 then
        d'_i = (x_{i+1} − f_θ(x_{i+1}, t_{i+1}))/t_{i+1}
        x_{i+1} = x_i + (t_{i+1} − t_i)(1/2 d_i + 1/2 d'_i)
    end if
end for
```

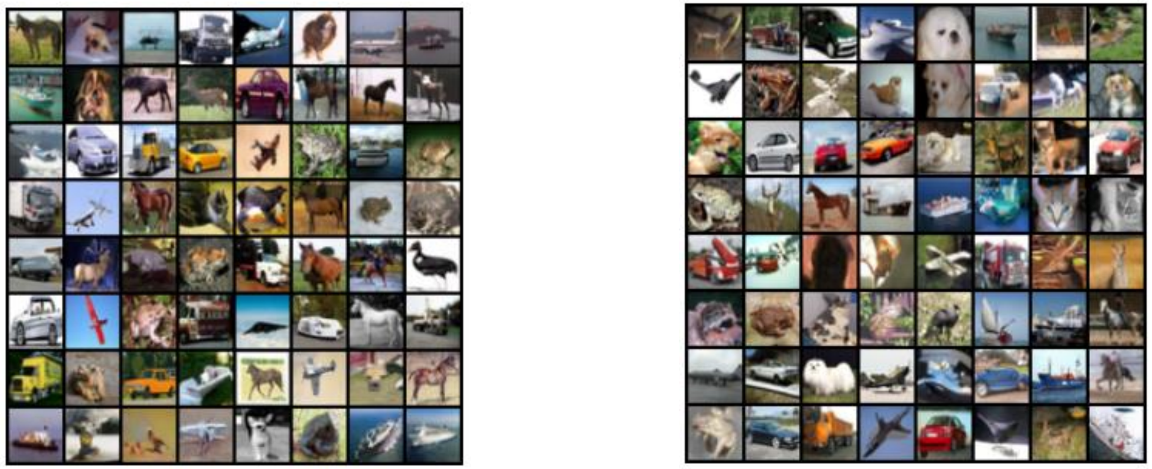
Experiments

Table 1. CIFAR-10 sample quality (FID) and number of function evaluations (NFE).

	Min FID ↓	Top-3 Avg FID ↓	NFE ↓
DDPM (Ho et al., 2020)	3.17	-	1000
DDIM (Song et al., 2021a)	4.67	-	50
VE-ODE (Song et al., 2021b)	5.29	-	194
VP-ODE (Song et al., 2021b)	2.86	-	134
PFGM (Xu et al., 2022)	2.48	-	104
PFGM++ (unconditional)			
$D = 64$	1.96	1.98	35
$D = 128$	1.92	1.94	35
$D = 2048$	**1.91**	**1.93**	35
$D = 3072000$	1.99	2.02	35
$D \to \infty$ (Karras et al., 2022)	1.98	2.00	35
PFGM++ (class-conditional)			
$D = 2048$	**1.74**	-	35
$D \to \infty$ (Karras et al., 2022)	1.79	-	35

Table 2. FFHQ sample quality (FID) with 79 NFE in unconditional setting

	Min FID ↓	Top-3 Avg FID ↓
$D = 128$	**2.43**	2.48
$D = 2048$	2.46	**2.47**
$D = 3072000$	2.49	2.52
$D \to \infty$ (Karras et al., 2022)	2.53	2.54



(g) D=2048, $\alpha = 0$ (FID=1.92) (j) $D \to \infty, \alpha = 0$ (FID=1.98)

(a) $D = 128$ (FID=2.43) (b) EDM ($D \to \infty$) (FID=2.53)

Figure 9. Generated images on FFHQ 64×64 dataset, by **(left)** $D = 128$ and **(right)** EDM ($D \to \infty$).

Balancing Robustness and Rigidity

Robustness

$D = 1$

- Less norm-σ correlation
- Tolerate to errors
- Better on weak networks

Rigidity

$D = \infty$

- Narrow range of norm
- Sensitive to errors
- Better to fit for high-capacity networks

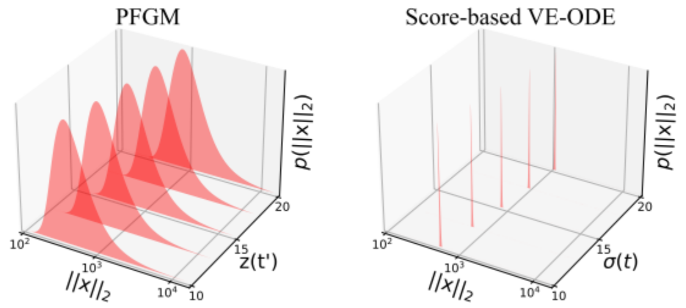


Figure 4: Sample norm distributions with varying time variables (σ for VE-ODE and z for PFGM)

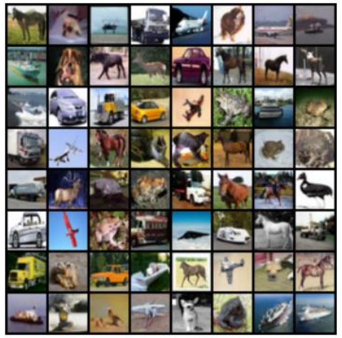
Demonstration of Robustness

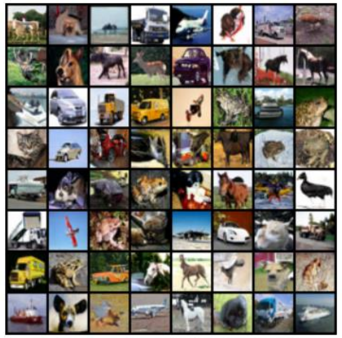
Post-training quantization

Table 3. FID score versus quantization bit-widths on CIFAR-10.

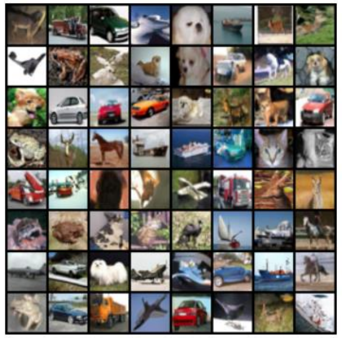
Quantization bits:	9	8	7	6	5
$D = 64$	1.96	**1.96**	**2.12**	**2.94**	**28.50**
$D = 128$	1.93	1.97	2.15	3.68	34.26
$D = 2048$	**1.91**	1.97	**2.12**	5.67	47.02
$D \to \infty$	1.97	2.04	2.16	5.91	50.09

Controlled noise injection in training


(g) D=2048, $\alpha = 0$ (FID=1.92)


(h) D=2048, $\alpha = 0.1$ (FID=1.95)


(i) D=2048, $\alpha = 0.2$ (FID=2.19)


(j) $D \to \infty, \alpha = 0$ (FID=1.98)


(k) $D \to \infty, \alpha = 0.1$ (FID=9.27)


(l) $D \to \infty, \alpha = 0.2$ (FID=92.41)

Content

- Background: Velocity Field (VF) for Generative Modeling
- PFGM: VF defined by Poisson Flow
- PFGM++: bridging PFGM and Diffusion Models

we only care about continuous case/ODE form

- **Summary**

Summary

- D-augmented PGFM
- Balancing robustness and rigidity
- Alignment with DMs, outperform EDM by tuning D

Summary

Under the framework of VF:

	PFGM	DM (VE)
Time Anchor t	r	σ
q_{0t}	$\propto \frac{1}{(\|\boldsymbol{x}_r-\boldsymbol{x}_0\|_2^2+r^2)^{(N+D)/2}}$	$\propto e^{-\frac{\|\boldsymbol{x}_\sigma-\boldsymbol{x}_0\|_2^2}{2\sigma^2}}$
$\boldsymbol{v}_{0t}$	$\frac{q_r(\boldsymbol{x}_r)}{S_{N+D-1}}(\boldsymbol{x}_r-\boldsymbol{x}_0)$	$\frac{\boldsymbol{x}_\sigma-\boldsymbol{x}_0}{\sigma}$
Biased Score/Velocity Estimator	$\propto \sum_i \frac{\boldsymbol{x}_r-\boldsymbol{x}_0^{(i)}}{\left(\left\|\boldsymbol{x}_r-\boldsymbol{x}_0^{(i)}\right\|_2^2+r^2\right)^{(N+D)/2}}$	$\propto \sum_i e^{-\frac{\left\|\boldsymbol{x}_\sigma-\boldsymbol{x}_0^{(i)}\right\|_2^2}{2\sigma^2}} \frac{\boldsymbol{x}_\sigma-\boldsymbol{x}_0^{(i)}}{\sigma^2}$
Normalized Unbiased Score/Flow Matching Target	$\frac{\boldsymbol{x}_r-\boldsymbol{x}_0}{r/\sqrt{D}}$	$\frac{\boldsymbol{x}_\sigma-\boldsymbol{x}_0}{\sigma}$

→ Stable Target Field [4]

References

[1] Xu, Yilun, et al. "Poisson flow generative models." arXiv preprint arXiv:2209.11178 (2022).

[2] Xu, Yilun, et al. "Pfgm++: Unlocking the potential of physics-inspired generative models." arXiv preprint arXiv:2302.04265 (2023).

[3] Lipman, Yaron, et al. "Flow matching for generative modeling." arXiv preprint arXiv:2210.02747 (2022).

[4] Xu, Yilun, Shangyuan Tong, and Tommi Jaakkola. "Stable Target Field for Reduced Variance Score Estimation in Diffusion Models." arXiv preprint arXiv:2302.00670 (2023).

[5] Karras, Tero, et al. "Elucidating the design space of diffusion-based generative models." arXiv preprint arXiv:2206.00364 (2022).