Elucidating the Design Space of Arbitrary-Noise-Based Diffusion Models

Supplementary Material

A. Derivation of $f(\cdot)$, $g(\cdot)$ and $\phi(\cdot)$ in SDE

Problem Setup

Given the EDA forward process:

$$x_t = s(t)x_0 + s(t)\sigma(t)\sum_{m=1}^{M}\frac{\epsilon_m+\eta}{\eta+1}h_{m,\boldsymbol{x}_0}, \tag{21}$$

and its distribution:

$$P(x_t \mid x_0) = \mathcal{N}\Big(x_t;\, s(t)x_0 + \frac{\eta\, s(t)\,\sigma(t)}{\eta+1}\sum_{m=1}^{M}h_{m,\boldsymbol{x}_0},\\ \frac{s^2(t)\,\sigma^2(t)}{(\eta+1)^2}\Sigma_{x_0}\Big). \tag{22}$$

where $\Sigma_{x_0} = H_{x_0}H_{x_0}^T$ and $H_{x_0} = \begin{bmatrix} h_{1,\boldsymbol{x}_0} & \dots & h_{M,\boldsymbol{x}_0}\end{bmatrix}$. The associated Stochastic Differential Equation (SDE) is expressed as:

$$dx = [f(t)x + \phi(t,x_0)]\,dt + g(t)\sum_{m=1}^{M}h_{m,\boldsymbol{x}_0}dw_t^{(m)}, \tag{23}$$

where $dw_t^{(m)}$ are independent Wiener processes satisfying $\mathbb{E}[dw_t^{(m)}dw_t^{(n)}] = \delta_{mn}dt$. The coefficient functions are derived as follows:

$$f(t) = \frac{s'(t)}{s(t)} \tag{24}$$

$$\phi_{x_0}(t) = \frac{\eta s(t)\sigma'(t)}{\eta+1}\sum_{m=1}^{M}h_{m,\boldsymbol{x}_0} \tag{25}$$

$$g(t) = \frac{s(t)}{\eta+1}\sqrt{\frac{d\sigma^2(t)}{dt}} \tag{26}$$

Derivation: The derivation is performed in two parts: Mean and Variance.

1. Derivation of the Mean Component

The mean of the forward diffusion process $P(x_t|x_0)$ is given by:

$$\mu(t) = s(t)x_0 + \frac{\eta s(t)\sigma(t)}{\eta+1}\sum_{m=1}^{M}h_{m,\boldsymbol{x}_0} \tag{27}$$

For the SDE:

$$\mathrm{d}x = [f(t)x + \phi_{x_0}(t)]\,\mathrm{d}t + g(t)\sum_{m=1}^{M}h_{m,\boldsymbol{x}_0}\mathrm{d}w_t^{(m)} \tag{28}$$

the differential equation governing the mean $\mu(t)$ is:

$$\frac{\mathrm{d}\mu(t)}{\mathrm{d}t} = f(t)\mu(t) + \phi_{x_0}(t), \quad \mu(0) = x_0 \tag{29}$$

Solving this first order linear non-homogeneous differential equation by the integrating factor method with $\lambda(t) = \exp\left(-\int_0^t f(\tau)\mathrm{d}\tau\right)$ yields:

$$\mu(t) = \exp\Big(\int_0^t f(\tau)\,\mathrm{d}\tau\Big)x_0 \\ + \exp\Big(\int_0^t f(\tau)\,\mathrm{d}\tau\Big)\int_0^t \phi(\tau)\exp\Big(-\int_0^\tau f(s)\,\mathrm{d}s\Big)\,\mathrm{d}\tau.$$

Equating this to the mean of $P(x_t|x_0)$, we identify $s(t) = \exp\left(\int_0^t f(\tau)\mathrm{d}\tau\right)$. So we obtain:

$$f(t) = \frac{s'(t)}{s(t)} \qquad \square$$

And for the second term:

$$s(t)\int_0^t \frac{\phi_{x_0}(\tau)}{s(\tau)}\mathrm{d}\tau = \frac{\eta s(t)\sigma(t)}{\eta+1}\sum_{m=1}^{M}h_{m,\boldsymbol{x}_0} \tag{30}$$

Canceling $s(t)$ from both sides and differentiating with respect to t:

$$\frac{\phi_{x_0}(t)}{s(t)} = \frac{\mathrm{d}}{\mathrm{d}t}\left[\frac{\eta\sigma(t)}{\eta+1}\sum_{m=1}^{M}h_{m,\boldsymbol{x}_0}\right] \tag{31}$$

Because $\sum_{m=1}^{M}h_{m,\boldsymbol{x}_0}$ is time-independent, we finally obtain:

$$\phi_{x_0}(t) = \frac{\eta s(t)\sigma'(t)}{\eta+1}\sum_{m=1}^{M}h_{m,\boldsymbol{x}_0} \qquad \square$$

2. Derivation of the Variance Component

The covariance matrix of the forward diffusion process is given by:

$$\Sigma(t) = \frac{s^2(t)\sigma^2(t)}{(\eta+1)^2}\Sigma_{x_0}, \tag{32}$$

where $\Sigma_{x_0} = \sum_{m=1}^{M}h_{m,\boldsymbol{x}_0}h_{m,\boldsymbol{x}_0}^\top$.

Applying Itô's lemma to the covariance matrix $\Sigma_S(t)$ of the SDE, the differential equation is satisfied:

$$\frac{\mathrm{d}\Sigma_S(t)}{\mathrm{d}t} = 2f(t)\Sigma_S(t) + g^2(t)\sum_{m=1}^{M}h_{m,\boldsymbol{x}_0}h_{m,\boldsymbol{x}_0}^\top, \quad \Sigma(0) = 0 \tag{33}$$

Solving this equation, and we obtain:

$$\Sigma_S(t) = s^2(t) \int_0^t \frac{g^2(\tau)}{s^2(\tau)} \mathrm{d}\tau \cdot \sum_{m=1}^{M} h_{m,\boldsymbol{x}_0} h_{m,\boldsymbol{x}_0}^\top \tag{34}$$

Equating $\Sigma_S(t)$ to the forward process covariance $\Sigma(t)$:

$$\frac{s^2(t)\sigma^2(t)}{(\eta+1)^2} \Sigma_{x_0} = s^2(t) \int_0^t \frac{g^2(\tau)}{s^2(\tau)} \mathrm{d}\tau \cdot \Sigma_{x_0} \tag{35}$$

Canceling $s^2(t)\Sigma_{x_0}$ from both sides yields:

$$\int_0^t \frac{g^2(\tau)}{s^2(\tau)} \mathrm{d}\tau = \frac{\sigma^2(t)}{(\eta+1)^2} \tag{36}$$

Differentiating both sides with respect to t:

$$\frac{g^2(t)}{s^2(t)} = \frac{1}{(\eta+1)^2} \frac{\mathrm{d}\sigma^2(t)}{\mathrm{d}t} \tag{37}$$

Taking the square root and solving for $g(t)$, we obtain:

$$g(t) = \frac{s(t)}{(\eta+1)} \sqrt{\frac{\mathrm{d}\sigma^2(t)}{\mathrm{d}t}} \qquad \square$$

B. Derivation of Probability Flow ODE for EDA

Problem Setup

Consider the forward process SDE for EDA:

$$\mathrm{d}x = [f(t)x + \phi_{x_0}(t)]\,\mathrm{d}t + g(t) \sum_{m=1}^{M} h_{m,\boldsymbol{x}_0} \mathrm{d}w_t^{(m)} \tag{38}$$

where:
- $f(t)$: Scalar drift coefficient
- $\phi_{x_0}(t)$: Deterministic offset term
- $g(t)$: Diffusion coefficient
- $h_{m,\boldsymbol{x}_0}$: Basis functions (only dependent on initial condition x_0)
- $\mathrm{d}w_t^{(m)}$: Independent Wiener process increments

Our goal is to find the corresponding Probability Flow ODE (PFODE):

$$\frac{\mathrm{d}x}{\mathrm{d}t} = v_{x_0}(x,t) \tag{39}$$

whose solution preserves the probability distribution $p(x|x_0)$ of the original SDE.

Derivation via Fokker-Planck Framework

For a general SDE $\mathrm{d}x_t = \mu(x_t,t)\mathrm{d}t + \bar{\Sigma}(x_t,t)\mathrm{d}w_t$, the Fokker-Planck equation (FPE) is:

$$\frac{\partial p}{\partial t} = -\nabla\cdot\left[\mu p\right] + \frac{1}{2}\nabla\nabla : \left[\bar{\Sigma}\bar{\Sigma}^\top p\right] \tag{40}$$

where $\nabla\cdot$ denotes divergence and $\nabla\nabla :$ denotes the double divergence operator.

Substituting our SDE coefficients:

- Drift term: $\mu(x,t) = f(t)x + \phi_{x_0}(t)$
- Diffusion term: $\bar{\Sigma}(x,t) = g(t)\cdot[h_{1,\boldsymbol{x}_0},\ldots,h_{M,\boldsymbol{x}_0}]$

The covariance matrix becomes:

$$\bar{\Sigma}\bar{\Sigma}^\top = g^2(t)\sum_{m=1}^{M} h_{m,\boldsymbol{x}_0} h_{m,\boldsymbol{x}_0}^\top \triangleq g^2(t)\Sigma_{x_0} \tag{41}$$

The FPE specializes to:

$$\frac{\partial p}{\partial t} = -\nabla\cdot\left[(f(t)x + \phi_{x_0}(t))p\right] + \frac{1}{2}g^2(t)\nabla\nabla : \left[\Sigma_{x_0}p\right] \tag{42}$$

Constructing the PFODE

The probability flow ODE requires the continuity equation:

$$\frac{\partial p}{\partial t} + \nabla\cdot\left[v_{x_0}(x,t)p\right] = 0 \tag{43}$$

Equating with the FPE, we obtain:

$$\nabla\cdot\left[v_{x_0}p\right] = \nabla\cdot\left[(f(t)x + \phi_{x_0}(t))p\right] - \frac{1}{2}g^2(t)\nabla\nabla : \left[\Sigma p\right] \tag{44}$$

Solving for the Velocity Field

Introduce the **score function**:

$$\bar{s}(x,t) = \nabla_x \ln p(x|x_0) = \frac{\nabla_x p}{p} \tag{45}$$

For the diffusion term:

$$\begin{aligned}\nabla\nabla : \left[\Sigma_{x_0}p\right] &= \sum_{i,j}\frac{\partial^2}{\partial x_i \partial x_j}\left[(\Sigma_{x_0})_{ij}p\right] \\ &= \nabla\cdot[\Sigma_{x_0}\nabla p] \\ &= \nabla\cdot[\Sigma_{x_0}p\nabla\ln p] \quad (\text{using } \nabla p = p\nabla\ln p)\end{aligned}$$

This establishes the critical connection between the diffusion term and the score function. Then we transform the diffusion term:

$$\frac{1}{2}\nabla\nabla : \left[\Sigma_{x_0}p\right] = \nabla\cdot\left[\frac{1}{2}\Sigma_{x_0}\nabla p\right] \tag{46}$$

$$= \nabla\cdot\left[\frac{1}{2}\Sigma_{x_0}ps\right] \tag{47}$$

Substituting back, we obtain:

$$\nabla\cdot\left[v_{x_0}p\right] = \nabla\cdot\left[(f(t)x + \phi_{x_0}(t))p - \frac{1}{2}g^2(t)\Sigma_{x_0}p\bar{s}\right] \tag{48}$$

This identifies the velocity field:

$$v_{x_0}(x,t) = f(t)x + \phi_{x_0}(t) - \frac{1}{2}g^2(t)\Sigma_{x_0}\bar{s}(x,t) \tag{49}$$

Final PFODE Form

Substituting $s(x,t) = \nabla_x \ln p(x,t)$, we obtain the complete PFODE:

$$\frac{\mathrm{d}x}{\mathrm{d}t} = f(t)x + \phi_{x_0}(t) - \frac{1}{2}g^2(t)\Sigma_{x_0}\nabla_x \ln p(x|x_0) \qquad \square$$

where $\Sigma_{x_0} = \sum_{m=1}^{M} h_{m,\boldsymbol{x}_0} h_{m,\boldsymbol{x}_0}^\top$.

C. Derivation of Deterministic Sampling Formula from PFODE

PFODE Simplification

Given the Probability Flow ODE (PFODE):

$$\frac{\mathrm{d}x}{\mathrm{d}t} = f(t)x + \phi_{x_0}(t) - \frac{1}{2}g^2(t)\Sigma_{x_0}\nabla_x \ln p(x|x_0) \tag{50}$$

Substitute the parameter relationships derived in Appendix A:

$$\begin{aligned} f(t) &= \frac{s'(t)}{s(t)}, \quad \phi_{x_0}(t) = \frac{\eta\, s(t)\, \sigma'(t)}{\eta+1}\sum_{m=1}^{M} h_{m,\boldsymbol{x}_0}, \\ g^2(t) &= \frac{2\, s^2(t)\, \sigma(t)\, \sigma'(t)}{(\eta+1)^2}. \end{aligned} \tag{51}$$

This transforms the PFODE into:

$$\begin{aligned} \frac{\mathrm{d}x}{\mathrm{d}t} &= \frac{s'(t)}{s(t)}\,x + \frac{\eta\, s(t)\, \sigma'(t)}{\eta+1}\sum_{m=1}^{M} h_{m,\boldsymbol{x}_0} \\ &\quad - \frac{s^2(t)\, \sigma(t)\, \sigma'(t)}{(\eta+1)^2}\Sigma_{x_0}\, \nabla_{x_t} \log p(x \mid x_0). \end{aligned} \tag{52}$$

Denoiser Learning

Define the loss function $\mathcal{L}_{x_0}(D;\sigma(t))$:

$$\mathcal{L}_{x_0}(D;\sigma(t)) = \mathbb{E}_{x\sim\mathcal{N}(\mu(x_0,t),\bar{\Sigma}(t))}\|D(x;\sigma) - x_0\|^2 \tag{53}$$

where:

$$\begin{aligned} \mu(x_0,t) &= s(t)x_0 + \frac{s(t)\sigma(t)}{\eta+1}\sum_{m=1}^{M}\eta h_{m,\boldsymbol{x}_0} \\ \bar{\Sigma}(t) &= \frac{s^2(t)\sigma^2(t)}{(\eta+1)^2}\Sigma_{x_0} \end{aligned}$$

Expanding the $\mathcal{L}_{x_0}(D;\sigma(t))$ yields:

$$\mathcal{L}_{x_0}(D;\sigma(t)) = \mathbb{E}_{x\sim\mathcal{N}(\mu(x_0,t),\bar{\Sigma}(t))}\|D(x;\sigma) - x_0\|^2, \tag{54}$$

$$\begin{aligned} &= \int_{\mathbb{R}^d} \mathcal{N}\Big(s(t)x_0 + \tfrac{\eta\, s(t)\, \sigma(t)}{\eta+1}\sum_{m=1}^{M} h_{m,\boldsymbol{x}_0},\ \tfrac{s^2(t)\, \sigma^2(t)}{(\eta+1)^2}\,\Sigma_{x_0}\Big) \\ &\qquad \times \|D_{x_0}(x;\sigma) - x_0\|^2\, \mathrm{d}x, \end{aligned} \tag{55}$$

$$\triangleq \int_{\mathbb{R}^d} \mathcal{L}_{x_0}(D;\sigma(t),x)\, \mathrm{d}x. \tag{56}$$

Then minimize $\mathcal{L}_{x_0}(D;\sigma(t),x)$. This convex optimization problem is solved by setting the gradient to zero:

$$0 = \nabla_{D_{x_0}(x,\sigma)}\mathcal{L}_{x_0}(D;\sigma(t),x), \tag{57}$$

$$\begin{aligned} 0 = \nabla_{D_{x_0}(x,\sigma)}\Big[\mathcal{N}\Big(s(t)x_0 + \tfrac{s(t)\sigma(t)}{\eta+1}\sum_{m=1}^{M}\eta h_{m,\boldsymbol{x}_0}, \\ \tfrac{s^2(t)\sigma^2(t)}{(\eta+1)^2}\Sigma_{x_0}\Big) \qquad \times \|D_{x_0}(x;\sigma) - x_0\|^2\Big] \end{aligned} \tag{58}$$

$$0 = \nabla_{D_{x_0}(x,\sigma)}\|D_{x_0}(x;\sigma) - x_0\|^2. \tag{59}$$

This leads to the optimal solution:

$$D_{x_0}(x;\sigma) = x_0. \tag{60}$$

Score Function Derivation

For the conditional distribution, we define the mean $\boldsymbol{\mu}_c$ and covariance $\boldsymbol{\Sigma}_c$ as:

$$\boldsymbol{\mu}_c = s(t)x_0 + \frac{\eta s(t)\sigma(t)}{\eta+1}\sum_{m=1}^{M} h_{m,\boldsymbol{x}_0}, \quad \boldsymbol{\Sigma}_c = \frac{s^2(t)\sigma^2(t)}{(\eta+1)^2}\Sigma_{x_0}. \tag{61}$$

The distribution is then expressed compactly as $p(x_t|x_0) = \mathcal{N}(x_t;\boldsymbol{\mu}_c,\boldsymbol{\Sigma}_c)$. The score function is derived as:

$$\nabla_{x_t}\log p(x_t|x_0) = \frac{\nabla_{x_t}\mathcal{N}(x_t;\boldsymbol{\mu}_c,\boldsymbol{\Sigma}_c)}{\mathcal{N}(x_t;\boldsymbol{\mu}_c,\boldsymbol{\Sigma}_c)} \tag{62}$$

$$= \frac{\mathcal{N}(x_t;\boldsymbol{\mu}_c,\boldsymbol{\Sigma}_c)\,\left[\boldsymbol{\Sigma}_c^{-1}(\boldsymbol{\mu}_c - x_t)\right]}{\mathcal{N}(x_t;\boldsymbol{\mu}_c,\boldsymbol{\Sigma}_c)} \tag{63}$$

$$= \boldsymbol{\Sigma}_c^{-1}(\boldsymbol{\mu}_c - x_t). \tag{64}$$

Substituting the definitions of $\boldsymbol{\mu}_c$ and $\boldsymbol{\Sigma}_c$ back yields the final analytical form.

Substituting into the score function yields:

$$\begin{aligned} \nabla_{x_t}\log p(x_t|x_0) &= \frac{(\eta+1)^2}{s^2(t)\sigma^2(t)}\Sigma_{x_0}^{-1}\Big(s(t)x_0 \\ &\quad + \frac{\eta\, s(t)\, \sigma(t)}{\eta+1}\sum_{m=1}^{M} h_{m,\boldsymbol{x}_0} - x_t\Big). \end{aligned} \tag{65}$$

ODE Simplification

Substituting the score function into the PFODE:

$$\begin{aligned}\frac{\mathrm{d}x}{\mathrm{d}t} &= \frac{s'(t)}{s(t)}x + \frac{\eta s(t)\sigma'(t)}{\eta+1}\sum_{m=1}^{M} h_{m,\boldsymbol{x}_0} \\ &\quad - \frac{s^2(t)\sigma(t)\sigma'(t)}{(\eta+1)^2}\Sigma_{x_0}\left(\frac{(\eta+1)^2}{s^2(t)\sigma^2(t)}\Sigma_{x_0}^{-1}\right. \\ &\quad \left.\left(s(t)x_0 + \frac{\eta s(t)\sigma(t)}{\eta+1}\sum_{m=1}^{M} h_{m,\boldsymbol{x}_0} - x\right)\right) \quad (66) \\ &= \frac{s'(t)}{s(t)}x + \frac{\eta s(t)\sigma'(t)}{\eta+1}\sum_{m=1}^{M} h_{m,\boldsymbol{x}_0} \\ &\quad - \frac{\sigma'(t)}{\sigma(t)}\left(s(t)x_0 + \frac{\eta s(t)\sigma(t)}{\eta+1}\sum_{m=1}^{M} h_{m,\boldsymbol{x}_0} - x\right) \quad (67) \\ &= \left(\frac{s'(t)}{s(t)} + \frac{\sigma'(t)}{\sigma(t)}\right)x_t - \frac{\sigma'(t)s(t)}{\sigma(t)}x_0 \quad (68) \\ &= \left(\frac{s'(t)}{s(t)} + \frac{\sigma'(t)}{\sigma(t)}\right)x_t - \frac{\sigma'(t)s(t)}{\sigma(t)}D_{x_0}(x;\sigma) \quad (69) \\ & \quad (70)\end{aligned}$$

Use a uniform network $D(x;\sigma)$ to learn all $D_{x_0}(x;\sigma)$, the loss function is:

$$\begin{aligned}&\mathcal{L}(D;\sigma(t)) \\ &= \mathbb{E}_{y\sim P_{\text{data}}}\mathbb{E}_{x\sim\mathcal{N}(\mu(y,t),\bar{\Sigma}(t))}\|D(x;\sigma) - D_y(x;\sigma)\|^2 \quad (71) \\ &= \mathbb{E}_{y\sim P_{\text{data}}}\mathbb{E}_{x\sim\mathcal{N}(\mu(y,t),\bar{\Sigma}(t))}\|D(x;\sigma) - y\|^2 \quad (72) \\ & \quad (73)\end{aligned}$$

when the network is fitted:

$$D(x;\sigma) \approx D_y(x;\sigma) = y, \quad \forall y \sim P_{\text{data}} \quad (74)$$

Note that the $\approx$ here has an error and is not the theoretical optimal point of a convex problem. When the error is small enough during training, substituting the $D(x;\sigma)$ into the simplified PFODE:

$$\frac{\mathrm{d}x}{\mathrm{d}t} = \left(\frac{s'(t)}{s(t)} + \frac{\sigma'(t)}{\sigma(t)}\right)x - \frac{\sigma'(t)s(t)}{\sigma(t)}D(x;\sigma) \qquad \square$$

Notably, all terms in the derivation of Eq. 66 related to the data-dependent basis $h_{m,\boldsymbol{x}_0}$ simplify entirely. This results in the deterministic sampling procedure for our any-noise-based EDA becoming identical to that of the EDM framework Eq. 7.

D. Derivation of Optimal Case for EDA

Proposition 4. *The approximation in Eq. 74 of Appendix C introduces errors by substituting the denoiser output $D(x;\sigma)$ with the ground-truth training target y upon training convergence. However, these errors vanish when the basis set is independent of data sample x_0, $H = [h_1, \ldots, h_M]$, in contrast to the sample-dependent basis $H_{x_0} = \left[h_{1,\boldsymbol{x}_0}, ..., h_{M,\boldsymbol{x}_0}\right]$ in Eq. 22. This represents the optimal case for EDA.*

Proof. When the basis set H is independent of data sample x_0, the PFODE Eq. 50 could use the marginal distribution $p(x)$ to substitute the conditional $p(x|x_0)$, which more accurately represents the distribution of x_t. The covariance matrix becomes:

$$\boldsymbol{\Sigma} = \sum_{m=1}^{M} h_m h_m^\top \quad (75)$$

yielding the modified PFODE:

$$\frac{\mathrm{d}x}{\mathrm{d}t} = f(t)x + \phi(t) - \frac{1}{2}g^2(t)\boldsymbol{\Sigma}\nabla_x \log p(x) \quad (76)$$

with coefficients:

$$\begin{aligned}f(t) = \frac{s'(t)}{s(t)}, \quad \phi(t) = \frac{\eta s(t)\sigma'(t)}{\eta+1}\sum_m h_m, \\ g(t) = \frac{s(t)}{\eta+1}\sqrt{\frac{\mathrm{d}\sigma^2(t)}{\mathrm{d}t}} \quad (77)\end{aligned}$$

Suppose our training set consists of a finite samples $[y_1, \ldots, y_Y]$. Thus $p_{\text{data}}(x)$ is represented by a mixture of Dirac delta distributions:

$$p_{\text{data}}(x) = \frac{1}{Y}\sum_{i=1}^{Y}\delta(x - y_i) \quad (78)$$

The marginal distribution $p(x)$ is:

$$\begin{aligned}p(x) &= \int_{\mathbb{R}^d} p_{\text{data}}(x_0)\,\mathcal{N}\big(x;\, \mu(x_0,t),\, \Sigma_t\big)\,\mathrm{d}x_0 \quad (79) \\ &= \int_{\mathbb{R}^d} \tfrac{1}{Y}\sum_{i=1}^{Y}\delta(x_0 - y_i)\,\mathcal{N}\big(x;\, \mu(x_0,t),\, \Sigma_t\big)\,\mathrm{d}x_0 \quad (80) \\ &= \tfrac{1}{Y}\sum_{i=1}^{Y}\int_{\mathbb{R}^d}\delta(x_0 - y_i)\,\mathcal{N}\big(x;\, \mu(x_0,t),\, \Sigma_t\big)\,\mathrm{d}x_0 \quad (81) \\ &= \tfrac{1}{Y}\sum_{i=1}^{Y}\mathcal{N}\big(x;\, \mu(y_i,t),\, \Sigma_t\big), \quad (82)\end{aligned}$$

where

$$\begin{aligned}\mu(x_0,t) = s(t)x_0 + \tfrac{\eta\, s(t)\, \sigma(t)}{\eta+1}\sum_m h_m, \\ \Sigma_t = \tfrac{s^2(t)\,\sigma^2(t)}{(\eta+1)^2}\boldsymbol{\Sigma}. \quad (83)\end{aligned}$$

Now consider the training objective:

$$\mathcal{L}(D;\sigma) = \mathbb{E}_{y\sim p_{\text{data}}}\mathbb{E}_{x\sim p(x|y)}\|D(x;\sigma)-y\|^2, \tag{84}$$

$$= \mathbb{E}_{y\sim p_{\text{data}}}\int_{\mathbb{R}^d}\mathcal{N}\big(x;\mu_y(t),\bar{\Sigma}(t)\big)\|D(x;\sigma)-y\|^2\,\mathrm{d}x, \tag{85}$$

$$= \tfrac{1}{Y}\sum_{i=1}^{Y}\int_{\mathbb{R}^d}\mathcal{N}\big(x;\mu_{y_i}(t),\bar{\Sigma}(t)\big)\|D(x;\sigma)-y_i\|^2\,\mathrm{d}x, \tag{86}$$

$$= \int_{\mathbb{R}^d}\tfrac{1}{Y}\sum_{i=1}^{Y}\mathcal{N}\big(x;\mu_{y_i}(t),\bar{\Sigma}(t)\big)\|D(x;\sigma)-y_i\|^2\,\mathrm{d}x \tag{87}$$

$$\triangleq \int_{\mathbb{R}^d}\mathcal{L}(D;x,\sigma)\,\mathrm{d}x. \tag{88}$$

where

$$\mu_y(t) = s(t)y + \frac{\eta\, s(t)\,\sigma(t)}{\eta+1}\sum_m h_m,$$
$$\bar{\Sigma}(t) = \frac{s^2(t)\sigma^2(t)}{(\eta+1)^2}\boldsymbol{\Sigma}.$$

We minimize $\mathcal{L}(D;\sigma)$ by pointwise minimization: for each fixed x, the objective reduces to a weighted least-squares problem

$$0 = \nabla_{D(x;\sigma)}\mathcal{L}(D;x,\sigma) \tag{89}$$

$$= \nabla_{D(x;\sigma)}\tfrac{1}{Y}\sum_{i=1}^{Y}w_i(x)\,\|D(x;\sigma)-y_i\|^2 \tag{90}$$

$$= \tfrac{2}{Y}\sum_{i=1}^{Y}w_i(x)\,\big(D(x;\sigma)-y_i\big), \tag{91}$$

$$\Rightarrow \quad D(x;\sigma) = \frac{\sum_{i=1}^{Y}w_i(x)\,y_i}{\sum_{i=1}^{Y}w_i(x)}. \tag{92}$$

where

$$w_i(x) := \mathcal{N}\big(x;\mu_{y_i}(t),\bar{\Sigma}(t)\big),$$
$$\mu_{y_i}(t) = s(t)y_i + \tfrac{\eta\, s(t)\,\sigma(t)}{\eta+1}\sum_m h_m,$$
$$\bar{\Sigma}(t) = \tfrac{s^2(t)\sigma^2(t)}{(\eta+1)^2}\boldsymbol{\Sigma}.$$

and $D(x;\sigma)$ denotes the pointwise minimizer (denoiser output) at location x and noise level σ.

Substituting the marginal distribution $p(x)$ in Eq. 82, the score function becomes:

$$\nabla_{x_t}\log p(x_t) = \frac{\nabla_{x_t}p(x_t)}{p(x_t)} \tag{93}$$

$$= \frac{\nabla_{x_t}\frac{1}{Y}\sum_{i=1}^{Y}\mathcal{N}\big(x_t;\mu_i(t),\bar{\Sigma}(t)\big)}{\frac{1}{Y}\sum_{i=1}^{Y}\mathcal{N}\big(x_t;\mu_i(t),\bar{\Sigma}(t)\big)} \tag{94}$$

$$= \frac{\sum_{i=1}^{Y}\nabla_{x_t}\mathcal{N}\big(x_t;\mu_i(t),\bar{\Sigma}(t)\big)}{\sum_{i=1}^{Y}\mathcal{N}\big(x_t;\mu_i(t),\bar{\Sigma}(t)\big)}. \tag{95}$$

where the term $\nabla_{x_t}\mathcal{N}(\cdot)$ is given by:

$$\nabla_{x_t}\mathcal{N}\big(x_t;\mu_i(t),\bar{\Sigma}(t)\big) = \mathcal{N}\big(x_t;\mu_i(t),\bar{\Sigma}(t)\big)$$
$$\nabla_{x_t}\Big[-\tfrac{1}{2}(x_t-\mu_i(t))^{\top}\bar{\Sigma}^{-1}(x_t-\mu_i(t))\Big] \tag{96}$$

$$= \mathcal{N}\big(x_t;\mu_i(t),\bar{\Sigma}(t)\big)\,\bar{\Sigma}^{-1}\big(\mu_i(t)-x_t\big)$$
$$\cdot\,\tfrac{(\eta+1)^2}{s^2(t)\sigma^2(t)}. \tag{97}$$

Substituting $\nabla_{x_t}\mathcal{N}(\cdot)$ into Eq. 95 yields:

$$\nabla_{x_t}\log p(x_t) = A(t)\frac{\sum_i\mathcal{N}_i\,[s(t)y_i+B(t)-x_t]}{\sum_i\mathcal{N}_i}$$
$$= A(t)\left[s(t)\frac{\sum_i\mathcal{N}_i y_i}{\sum_i\mathcal{N}_i}+B(t)-x_t\right]. \tag{98}$$

Substituting $D(x_t;\sigma)$ gives:

$$\nabla_{x_t}\log p(x_t) = A(t)\,[s(t)D(x_t;\sigma)+B(t)-x_t]. \tag{99}$$

Finally, substituting Eq. 99 into the PFODE yields the deterministic sampling form:

$$\frac{\mathrm{d}x}{\mathrm{d}t} = \frac{s'(t)}{s(t)}x + \frac{\eta s(t)\sigma'(t)}{\eta+1}\sum_m h_m$$
$$- s^2(t)\sigma(t)\sigma'(t)\left[\frac{A(t)}{(\eta+1)^2}(s(t)D(x;\sigma)+B(t)-x)\right]$$
$$= \left(\frac{s'(t)}{s(t)}+\frac{\sigma'(t)}{\sigma(t)}\right)x - \frac{\sigma'(t)s(t)}{\sigma(t)}D(x;\sigma). \qquad \square$$

where for compactness we define:

$$\mathcal{N}_i := \mathcal{N}\Big(x_t;\ s(t)y_i + \tfrac{\eta s(t)\sigma(t)}{\eta+1}\sum_m h_m,\ \tfrac{s^2(t)\sigma^2(t)}{(\eta+1)^2}\boldsymbol{\Sigma}\Big),$$

$$A(t) := \tfrac{(\eta+1)^2}{s^2(t)\sigma^2(t)}\boldsymbol{\Sigma}^{-1}, \qquad B(t) := \tfrac{\eta s(t)\sigma(t)}{\eta+1}\sum_{m=1}^{M}h_m.$$

To summarize, there are two distinctions exist between this framework and Appendix C: (1) The marginal probability density Eq. 76 replaces the conditional density Eq. 50 in PFODE, for more accurate representation of x_t distribution; and (2) As shown in Eq. 92, the derivation of the optimal output of denoiser $D(x;\sigma)$ employs a rigorous convex optimization, rather than the approximation with errors as in Eq. 74. Consequently, when the basis set H is independent of the data sampling x_0, the theoretical framework achieves greater completeness and represents the optimal case for EDM.

E. Implementation Details

E.1. Specific Experimental Setup

For BFC, the mediator $\eta = 0$, the basis set H was set as **Case 1** in Prop. 1. The input MRIs were 256×256 and trained for 500 epochs with a batch size of 1. For MAR and SR, the $\eta = 10$ and the basis set H followed **Case 2** in Prop. 1. The input CTs were 416×416 and trained 1000 epochs with a batch size of 1. The Shadow Images were $256 \times 256 \times 3$, trained 2000 epochs with a batch size of 4.

E.2. Bias Field correction in MRI

We use the Adam with the momentum as $(0.9, 0.999)$. The initial learning rate is set to 2×10^{-5}, decay to $1e^{-8}$ with factor is 0.6 and the patient is 25 epochs. We use 0.9999 Exponential Moving Average (EMA). GDMP is trained for 800 epochs and the batch size is 1. The total diffusion steps T are set to 100. The noise schedule is defined as $\bar{\alpha} = \prod_{s=1}^{t} \alpha_s$ as in [9, 28], where $\alpha_t = 1 - \beta_t$ and β_t is linearly increasing from 0.0001 to 0.02.

Since the relationship between bias fields and MRI is usually modeled as a multiplicative, we first apply logarithmic transformation to the image, converting the multiplicative relationship between the image and noise into an additive one, as $log(A \times B) = log(A) + log(B)$. The η is set to zero. If the diffused noise is set to be bias field, its smoothness properties must to be satisfied, so we set the basis set H in Eq. 9 to include low-order Legendre polynomials and slowly varying trigonometric functions. We denote the i-th Legendre polynomial by $P_i(x)$. Then the two-dimensional Legendre polynomial is $P_{m,n}(x, y) = P_m(x)P_n(y)$. The two-dimensional Legendre polynomials with the highest degree less than or equal to N are used as basis functions, $L_N(x, y) \triangleq \{P_{m,n}(x, y) | m + n \leq N\}$, where N is a hyperparameter. We use a rotation function $f(x, y, \theta) = xcos(\theta) + ysin(\theta)$ and define $S_n(x, y) = \{cos(n \cdot f(x, y, \theta)), sin(n \cdot f(x, y, \theta)) | mod(\theta, 10) = 0, \theta \leq 180\}$. Trigonometric basis set is $T_N(x, y) \triangleq \{S_n(x, y) | 2 \leq n \leq N\}$, where N is a hyperparameter. The functions with n less than 2 are removed because they change too slowly. In summary, the basis set is:

$$H_{N_1,N_2}(x, y) \triangleq \{L_{N_1}(x, y), T_{N_2}(x, y)\} \tag{100}$$

where the range of both x and y coordinates is $[-1, 1]$. Each basis function in $H_{N_1,N_2}(x, y)$ is mapped linearly to the range centered at $[0.9, 1.1]$ for even fairness. We use $H_{3,5}(x, y)$ as basis set. The neural network θ predicts the diffused noise in x_t referred to [9, 28], and the loss function is:

$$L_{BFC}(\theta) = \mathbb{E}\|N_\theta\left(x_t, t\right) - N\|_2^2 \tag{101}$$

E.3. Metal Artifact Reduction in CT

We use the Adam with the momentum as $(0.9, 0.999)$. The initial learning rate is set to 1×10^{-5}, decays to $1e^{-8}$ with factor is 0.6 and a patience of 25000 steps. We use 0.9999 Exponential Moving Average (EMA). GDMP is trained for 500 epochs and the batch size is 1. The total diffusion steps T are set to 100. The noise schedule is defined as $\bar{\alpha} = \prod_{s=1}^{t} \alpha_s$ as in [9, 28], where $\alpha_t = 1 - \beta_t$ and β_t is linearly increasing from 0.0001 to 0.02.

The metal-affected CT is X_{ma} and the metal-free CT x_0. The diffused noise is image difference between X_{ma} and x_0, the η is 10, denoted by $N = (10+\epsilon)(X_{ma}-x_0)/11, \epsilon \sim \mathcal{N}(0, 1)$. In metal artifact reduction, the metal domain has far more severe artefacts than the non-metal domain, but the restoration of the metal domains is non-essential and it reduces the prediction accuracy of the artefacts in the non-metal domains. Hence, we optimised the loss function using a weighted MSE that balances the artefact intensity between metal and non-metal domains:

$$\begin{aligned} L_{MAR}(\theta) =& \mathbb{E}\left(1 + (1-m)\frac{max(m \cdot N)}{max((1-m) \cdot N)}\right) \\ & \times \|N_\theta(x_t, t) - N\|_2^2 \end{aligned}$$

where the m is the metal mask.

E.4. Shadow Removal in Natural Image

For shadow removal, the training hyperparameters are consistent with the official implementations in ShadowFormer. The total diffusion steps T are set to 100. The noise schedule is defined as $\bar{\alpha} = \prod_{s=1}^{t} \alpha_s$ as in [9, 28], where $\alpha_t = 1 - \beta_t$ and β_t is linearly increasing from 0.0001 to 0.02. Since the GDMP adds shadow maps of different intensities during diffusion, keep the coefficients of the shadow mask in the ShadowFormer consistent with the coefficients of the shadow maps, which is $\sqrt{1 - \alpha_t}$. The η is 10.

The diffused noise N as $(10 + \epsilon)(X_S - x_0)/11$ similar to MAR, where the X_S is shadow image and x_0 is non-shadow image. Our learning objective should be consistent with the baseline model ShadowFormer that directly learns the non-shadow image x_0. Hence the loss function is:

$$L_{SR}(\theta) = \mathbb{E}\|x_\theta(x_t, t) - x_0\|_2^2 \tag{102}$$