

A DETAILED PROOF

In this section, we give detailed proof of Theorems.

A.1 ROBUSTNESS OF SELF-GENERATION

We assume the generative structure

$$p(\mathbf{x}^{\text{hist}}, \mathbf{x}^{\text{pred}}, \tilde{\mathbf{x}}^{\text{hist}}) = p(\mathbf{x}^{\text{hist}}) p(\mathbf{x}^{\text{pred}} | \mathbf{x}^{\text{hist}}) p(\tilde{\mathbf{x}}^{\text{hist}} | \mathbf{x}^{\text{hist}})$$

and apply independent VP forward kernels to $(\mathbf{x}^{\text{hist}}, \mathbf{x}^{\text{pred}}, \tilde{\mathbf{x}}^{\text{hist}})$ to obtain $(\mathbf{x}_t^{\text{hist}}, \mathbf{x}_t^{\text{pred}}, \mathbf{c}_t)$ with $\mathbf{c}_t \equiv \tilde{\mathbf{x}}_t^{\text{hist}}$. Throughout we abbreviate $p_t(\cdot) = p(\cdot \text{ at time } t)$.

Assumption A.1 (Smoothness, log-concavity). (A1) (*Lipschitz-in-history score for the future*) Define $\phi_{\mathbf{h}}(\mathbf{p}) := \nabla_{\mathbf{p}} \log p_t(\mathbf{p} | \mathbf{h})$. There exists $L > 0$ such that $\|\phi_{\mathbf{h}}(\mathbf{p}) - \phi_{\mathbf{h}'}(\mathbf{p})\| \leq L \|\mathbf{h} - \mathbf{h}'\|$ uniformly in $(\mathbf{h}, \mathbf{h}', \mathbf{p}, t)$.

(A2) (*Strong log-concavity in \mathbf{x}^{hist}*) $-\log p_t(\mathbf{x}^{\text{hist}} | \mathbf{c})$ is m_x -strongly convex in \mathbf{x}^{hist} , uniformly over (\mathbf{c}, t) ; i.e.,

$$(\nabla_{\mathbf{x}^{\text{hist}}} \log p_t(\mathbf{x}^{\text{hist}} | \mathbf{c}) - \nabla_{\mathbf{x}^{\text{hist}}} \log p_t(\mathbf{x}^{\text{hist}'} | \mathbf{c}))^\top (\mathbf{x}^{\text{hist}} - \mathbf{x}^{\text{hist}'}) \leq -m_x \|\mathbf{x}^{\text{hist}} - \mathbf{x}^{\text{hist}'}\|^2.$$

(A3) (*Coercivity in \mathbf{x}^{pred}*) $-\log p_t(\mathbf{x}^{\text{pred}} | \mathbf{x}^{\text{hist}})$ is m_y -strongly convex in \mathbf{x}^{pred} uniformly in $(\mathbf{x}^{\text{hist}}, t)$.

Assumption (A1) covers conditional exponentials (e.g., conditionally Gaussian models), and (A2)–(A3) are standard to ensure contractive reverse flows⁴.

Lemma A.2 (Conditional factorization along diffusion). *For any $t \in (0, 1]$,*

$$p_t(\mathbf{x}_t^{\text{hist}}, \mathbf{x}_t^{\text{pred}} | \mathbf{c}_t) = p_t(\mathbf{x}_t^{\text{hist}} | \mathbf{c}_t) \cdot p_t(\mathbf{x}_t^{\text{pred}} | \mathbf{x}_t^{\text{hist}}),$$

hence $\mathbf{x}_t^{\text{pred}} \perp \mathbf{c}_t | \mathbf{x}_t^{\text{hist}}$.

Proof. By construction $p(\mathbf{x}^{\text{hist}}, \mathbf{x}^{\text{pred}}, \tilde{\mathbf{x}}^{\text{hist}}) = p(\mathbf{x}^{\text{hist}}) p(\mathbf{x}^{\text{pred}} | \mathbf{x}^{\text{hist}}) p(\tilde{\mathbf{x}}^{\text{hist}} | \mathbf{x}^{\text{hist}})$, and the VP forward kernels act independently on (X, Y, \tilde{X}) . The Markov property yields $C_t \perp Y_t | X_t$ and thus the factorization $p_t(\mathbf{x}_t^{\text{hist}}, \mathbf{x}_t^{\text{pred}} | \mathbf{c}_t) = p_t(\mathbf{x}_t^{\text{hist}} | \mathbf{c}_t) \cdot p_t(\mathbf{x}_t^{\text{pred}} | \mathbf{x}_t^{\text{hist}})$. \square

From Lemma A.2

$$\nabla_{\mathbf{x}_t^{\text{pred}}} \log p_t(\mathbf{x}_t^{\text{hist}}, \mathbf{x}_t^{\text{pred}} | \mathbf{c}_t) = \nabla_{\mathbf{x}_t^{\text{pred}}} \log p_t(\mathbf{x}_t^{\text{pred}} | \mathbf{x}_t^{\text{hist}}),$$

$$\nabla_{\mathbf{x}_t^{\text{hist}}} \log p_t(\mathbf{x}_t^{\text{hist}}, \mathbf{x}_t^{\text{pred}} | \mathbf{c}_t) = \nabla_{\mathbf{x}_t^{\text{hist}}} \log p_t(\mathbf{x}_t^{\text{hist}} | \mathbf{c}_t) + \nabla_{\mathbf{x}_t^{\text{hist}}} \log p_t(\mathbf{x}_t^{\text{pred}} | \mathbf{x}_t^{\text{hist}}).$$

Thus the \mathbf{x}^{pred} -component of the *total* score depends on the (being-denoised) $\mathbf{x}_t^{\text{hist}}$, whereas the prediction-only score uses the marginal $p_t(\mathbf{x}_t^{\text{pred}} | \mathbf{c}_t)$.

Lemma A.3 (Fisher / mixture identity for the prediction-only score). *For any $(\mathbf{x}_t^{\text{pred}}, \mathbf{c}_t)$,*

$$\nabla_{\mathbf{x}_t^{\text{pred}}} \log p_t(\mathbf{x}_t^{\text{pred}} | \mathbf{c}_t) = \mathbb{E} \left[\nabla_{\mathbf{x}_t^{\text{pred}}} \log p_t(\mathbf{x}_t^{\text{pred}} | \mathbf{x}_t^{\text{hist}}) \mid \mathbf{x}_t^{\text{pred}}, \mathbf{c}_t \right]. \quad (1)$$

Proof. Differentiating $\log p_t(\mathbf{x}_t^{\text{pred}}, \mathbf{c}_t) = \log \int p_t(\mathbf{x}_t^{\text{pred}} | \mathbf{x}) p_t(\mathbf{x} | \mathbf{c}_t) d\mathbf{x}$ under the integral and applying Bayes' rule yields equation 1. \square

We now compare the reverse SDE drifts (omitting the common diffusion term $g(t) d\bar{\mathbf{w}}_t$).

First, **Prediction-only** reverse process in \mathbf{x}^{pred} takes drift in

$$\dot{\mathbf{x}}_t^{\text{pred}} = \mathbf{f}_{\text{pred}}(t, \mathbf{x}_t^{\text{pred}}) - g(t)^2 \nabla_{\mathbf{x}_t^{\text{pred}}} \log p_t(\mathbf{x}_t^{\text{pred}} | \mathbf{c}_t), \quad (2)$$

⁴SFdiff trains the total-sequence *conditional* score via DSM and samples with a PC sampler; the DSM equivalence for conditionals is given in Theorem 3.2

with $\mathbf{x}_t^{\text{hist}} \equiv \mathbf{c}_t$ fixed as a (noisy) condition.

On the other hand, **Self-generation reverse drift** (SFdiff) takes drift in

$$\dot{\mathbf{x}}_t^{\text{hist}} = \mathbf{f}_{\text{hist}}(t, \mathbf{x}_t^{\text{hist}}) - g(t)^2 \left(\nabla_{\mathbf{x}_t^{\text{hist}}} \log p_t(\mathbf{x}_t^{\text{hist}} | \mathbf{c}_t) + \nabla_{\mathbf{x}_t^{\text{hist}}} \log p_t(\mathbf{x}_t^{\text{pred}} | \mathbf{x}_t^{\text{hist}}) \right), \quad (3)$$

$$\dot{\mathbf{x}}_t^{\text{pred}} = \mathbf{f}_{\text{pred}}(t, \mathbf{x}_t^{\text{pred}}) - g(t)^2 \nabla_{\mathbf{x}_t^{\text{pred}}} \log p_t(\mathbf{x}_t^{\text{pred}} | \mathbf{x}_t^{\text{hist}}). \quad (4)$$

Hence, $\mathbf{x}_t^{\text{hist}}$ is *purified on-the-fly* by $\nabla_{\mathbf{x}_t^{\text{hist}}} \log p_t(\mathbf{x}_t^{\text{hist}} | \mathbf{c}_t)$, and $\mathbf{x}_t^{\text{pred}}$ uses the increasingly denoised $\mathbf{x}_t^{\text{hist}}$.

Proposition A.4 (Contractivity of the \mathbf{x}^{hist} -flow (purification)). *Under (A2) and the VP drift $\mathbf{f}_{\text{hist}}(t, \mathbf{x}) = -\frac{1}{2}\beta(t)\mathbf{x}$, consider two runs of equation 3 with the same $\{\mathbf{x}_t^{\text{pred}}\}$ and \mathbf{c}_t , but different $\mathbf{x}_t^{\text{hist}}$ and $\mathbf{x}_t^{\text{hist}'}$. Then*

$$\frac{d}{dt} \|\mathbf{x}_t^{\text{hist}} - \mathbf{x}_t^{\text{hist}'}\|^2 \leq -2m_x g(t)^2 \|\mathbf{x}_t^{\text{hist}} - \mathbf{x}_t^{\text{hist}'}\|^2, \quad \Rightarrow \quad \|\mathbf{x}_t^{\text{hist}} - \mathbf{x}_t^{\text{hist}'}\| \leq e^{-m_x \int_t^1 g(s)^2 ds} \|\mathbf{x}_1^{\text{hist}} - \mathbf{x}_1^{\text{hist}'}\|.$$

Thus $\mathbf{x}_t^{\text{hist}}$ contracts exponentially toward the mode/mean of $p_t(\mathbf{x}^{\text{hist}} | \mathbf{c}_t)$ along the reverse flow.

Proof. Monotonicity of $\nabla_{\mathbf{x}} \log p_t(\mathbf{x} | \mathbf{c})$ with parameter m_x controls the symmetric part of the Jacobian of the drift in equation 3 the contribution of \mathbf{f}_{hist} further aids dissipation. The cross-term $\nabla \log p_t(\mathbf{x}_t^{\text{pred}} | \mathbf{x}_t^{\text{hist}})$ is 1-Lipschitz in $\mathbf{x}_t^{\text{hist}}$ under (A1) at $\mathbf{x}_t^{\text{pred}}$ and is dominated by m_x ; the standard Grönwall argument gives the stated inequality. \square

We measure robustness by a Lipschitz-type sensitivity of the *output* $\mathbf{x}_0^{\text{pred}}$ with respect to the *condition path* \mathbf{c} . Fix two noisy conditions \mathbf{c} and \mathbf{c}' , and couple the reverse noise so differences stem only from the drifts.

Theorem A.5 (Self-generation yields smaller sensitivity). *Assume (A1)–(A3). Let $H(t) := \int_t^1 g(s)^2 ds$ and $G := H(0) = \int_0^1 g(s)^2 ds > 0$. Then*

$$\text{(prediction-only)} \quad \|\mathbf{x}_0^{\text{pred}} - \mathbf{x}_0^{\text{pred}'}\| \leq L \int_0^1 g(s)^2 \|\mathbf{c}_s - \mathbf{c}'_s\| ds \leq L G \sup_{s \in [0,1]} \|\mathbf{c}_s - \mathbf{c}'_s\|, \quad (5)$$

$$\text{(total-sequence)} \quad \|\mathbf{x}_0^{\text{pred}} - \mathbf{x}_0^{\text{pred}'}\| \leq L \int_0^1 g(s)^2 e^{-m_x H(s)} \|\mathbf{c}_s - \mathbf{c}'_s\| ds = \frac{L}{m_x} (1 - e^{-m_x G}) \sup_s \|\mathbf{c}_s - \mathbf{c}'_s\|. \quad (6)$$

Consequently,

$$\frac{L}{m_x} (1 - e^{-m_x G}) < L G \quad \Rightarrow \quad \|\mathbf{x}_0^{\text{pred}} - \mathbf{x}_0^{\text{pred}'}\| < \|\mathbf{x}_0^{\text{pred}} - \mathbf{x}_0^{\text{pred}'}\|_{\text{pred-only}}. \quad (7)$$

Thus the total-sequence conditional score produces forecasts with strictly smaller sensitivity to condition perturbations than the prediction-only score.

Proof. Let $\Delta(t) := \mathbf{x}_t^{\text{pred}} - \mathbf{x}_t^{\text{pred}'}$. We couple the two reverse processes (prediction-only or total-sequence) with the *same* reverse Gaussian noise so that pathwise differences arise only from the drift terms

Write the two reverse drifts (prediction-only vs. total) as

$$\dot{\mathbf{x}}_t^{\text{pred}} = \mathbf{f}_{\text{pred}}(t, \mathbf{x}_t^{\text{pred}}) - g(t)^2 S_t, \quad \dot{\mathbf{x}}_t^{\text{pred}'} = \mathbf{f}_{\text{pred}}(t, \mathbf{x}_t^{\text{pred}'} - g(t)^2 S'_t,$$

where S_t and S'_t are, respectively,

$$S_t = \begin{cases} \nabla_{\mathbf{x}_t^{\text{pred}}} \log p_t(\mathbf{x}_t^{\text{pred}} | \mathbf{c}_t) & \text{(prediction-only),} \\ \nabla_{\mathbf{x}_t^{\text{pred}}} \log p_t(\mathbf{x}_t^{\text{pred}} | \mathbf{x}_t^{\text{hist}}) & \text{(total),} \end{cases} \quad S'_t = \begin{cases} \nabla_{\mathbf{x}_t^{\text{pred}'}} \log p_t(\mathbf{x}_t^{\text{pred}'} | \mathbf{c}'_t) & \text{(prediction-only),} \\ \nabla_{\mathbf{x}_t^{\text{pred}'}} \log p_t(\mathbf{x}_t^{\text{pred}'} | \mathbf{x}_t^{\text{hist}'}) & \text{(total).} \end{cases}$$

Subtracting and taking the inner product with the unit vector $\mathbf{u}(t) := \Delta(t)/\|\Delta(t)\|$ (for $\Delta \neq 0$) yields

$$\begin{aligned} \frac{d}{dt} \|\Delta(t)\| &= \mathbf{u}(t)^\top (\dot{\mathbf{x}}_t^{\text{pred}} - \dot{\mathbf{x}}_t^{\text{pred}'}) \\ &= \mathbf{u}^\top (\mathbf{f}_{\text{pred}}(t, \mathbf{x}_t^{\text{pred}}) - \mathbf{f}_{\text{pred}}(t, \mathbf{x}_t^{\text{pred}'})) - g(t)^2 \mathbf{u}^\top (S_t - S'_t). \end{aligned}$$

Under VP, $\mathbf{f}_{\text{pred}}(t, \mathbf{y}) = -\frac{1}{2}\beta(t)\mathbf{y}$ contributes a *contractive* term $-\frac{1}{2}\beta(t)\|\Delta(t)\|$. Moreover, by Assumption (A3) (m_y -strong convexity in \mathbf{x}^{pred}),

$$(\nabla_{\mathbf{x}^{\text{pred}}} \log p_t(\mathbf{x}^{\text{pred}} | \cdot) - \nabla_{\mathbf{x}^{\text{pred}}} \log p_t(\mathbf{x}^{\text{pred}'} | \cdot))^\top (\mathbf{x}^{\text{pred}} - \mathbf{x}^{\text{pred}'}) \leq -m_y \|\Delta(t)\|^2,$$

which implies

$$-\mathbf{u}^\top (S_t - S'_t) \leq -m_y \|\Delta(t)\| + \Gamma_t,$$

with the *forcing* term

$$\Gamma_t := \begin{cases} \|\nabla_{\mathbf{x}_t^{\text{pred}}} \log p_t(\mathbf{x}_t^{\text{pred}} | \mathbf{c}_t) - \nabla_{\mathbf{x}_t^{\text{pred}}} \log p_t(\mathbf{x}_t^{\text{pred}} | \mathbf{c}'_t)\|, & \text{(prediction-only),} \\ \|\nabla_{\mathbf{x}_t^{\text{pred}}} \log p_t(\mathbf{x}_t^{\text{pred}} | \mathbf{x}_t^{\text{hist}}) - \nabla_{\mathbf{x}_t^{\text{pred}}} \log p_t(\mathbf{x}_t^{\text{pred}} | \mathbf{x}_t^{\text{hist}'})\|, & \text{(total).} \end{cases}$$

Absorbing the extra $-\frac{1}{2}\beta(t)\|\Delta\|$ (beneficial contraction) into m_y and combining the above,

$$\frac{d}{dt} \|\Delta(t)\| \leq -m_y g(t)^2 \|\Delta(t)\| + g(t)^2 \Gamma_t. \quad (8)$$

Case 1: prediction-only. By the Fisher/mixture identity (Lemma A.3),

$$\nabla_{\mathbf{x}_t^{\text{pred}}} \log p_t(\mathbf{x}_t^{\text{pred}} | \mathbf{c}_t) = \mathbb{E}[\nabla_{\mathbf{x}_t^{\text{pred}}} \log p_t(\mathbf{x}_t^{\text{pred}} | \mathbf{x}_t^{\text{hist}}) | \mathbf{x}_t^{\text{pred}}, \mathbf{c}_t].$$

Hence, using Jensen and (A1) (Lipschitz in the history argument),

$$\begin{aligned} \Gamma_t &= \left\| \mathbb{E}[\phi_{\mathbf{x}_t^{\text{hist}}}(\mathbf{x}_t^{\text{pred}}) | \mathbf{x}_t^{\text{pred}}, \mathbf{c}_t] - \mathbb{E}[\phi_{\mathbf{x}_t^{\text{hist}}}(\mathbf{x}_t^{\text{pred}}) | \mathbf{x}_t^{\text{pred}}, \mathbf{c}'_t] \right\| \\ &\leq L \cdot W_1(p_t(\mathbf{x}_t^{\text{hist}} | \mathbf{c}_t), p_t(\mathbf{x}_t^{\text{hist}} | \mathbf{c}'_t)), \end{aligned}$$

where W_1 is the 1-Wasserstein distance. Under the VP channel and strong log-concavity (A2), the posterior map $\mathbf{c}_t \mapsto p_t(\mathbf{x}_t^{\text{hist}} | \mathbf{c}_t)$ is 1-Lipschitz in W_1 (e.g., by contraction of Gaussian channels / Brascamp-Lieb), yielding

$$\Gamma_t \leq L \|\mathbf{c}_t - \mathbf{c}'_t\|.$$

Plugging into Step 2 and using $e^{-m_y(H(0)-H(s))} \leq 1$,

$$\|\Delta(0)\| \leq \int_0^1 g(s)^2 L \|\mathbf{c}_s - \mathbf{c}'_s\| ds \leq L G \sup_{s \in [0,1]} \|\mathbf{c}_s - \mathbf{c}'_s\|,$$

which is exactly equation 5.

Case 2: total-sequence. Here

$$\Gamma_t = \|\phi_{\mathbf{x}_t^{\text{hist}}}(\mathbf{x}_t^{\text{pred}}) - \phi_{\mathbf{x}_t^{\text{hist}'}}(\mathbf{x}_t^{\text{pred}})\| \leq L \|\mathbf{x}_t^{\text{hist}} - \mathbf{x}_t^{\text{hist}'}\| \quad \text{by (A1).}$$

By Proposition A.4 (contractivity of the \mathbf{x}^{hist} -flow under (A2)),

$$\|\mathbf{x}_t^{\text{hist}} - \mathbf{x}_t^{\text{hist}'}\| \leq e^{-m_x H(t)} \|\mathbf{x}_1^{\text{hist}} - \mathbf{x}_1^{\text{hist}'}\|.$$

With the standard synchronous terminal coupling of the reverse SDE, $\mathbf{x}_1^{\text{hist}}$ and $\mathbf{x}_1^{\text{hist}'}$ share the same Gaussian noise, so their difference is controlled by the difference of the (forward) conditions; in particular,

$$\|\mathbf{x}_1^{\text{hist}} - \mathbf{x}_1^{\text{hist}'}\| \leq \sup_{u \in [0,1]} \|\mathbf{c}_u - \mathbf{c}'_u\|.$$

Therefore

$$\Gamma_t \leq L e^{-m_x H(t)} \sup_u \|\mathbf{c}_u - \mathbf{c}'_u\|.$$

Plugging this bound into Step 2, we obtain

$$\|\Delta(0)\| \leq L \sup_u \|\mathbf{c}_u - \mathbf{c}'_u\| \int_0^1 e^{-m_y(H(0)-H(s))} g(s)^2 e^{-m_x H(s)} ds.$$

Since $e^{-m_y(H(0)-H(s))} \leq 1$, it suffices to compute

$$\int_0^1 g(s)^2 e^{-m_x H(s)} ds.$$

Using $H'(s) = -g(s)^2$, the change of variables $u = H(s)$ gives

$$\int_0^1 g(s)^2 e^{-m_x H(s)} ds = \int_{u=G}^0 (-e^{-m_x u}) du = \int_0^G e^{-m_x u} du = \frac{1 - e^{-m_x G}}{m_x}.$$

Hence

$$\|\Delta(0)\| \leq \frac{L}{m_x} (1 - e^{-m_x G}) \sup_s \|\mathbf{c}_s - \mathbf{c}'_s\|,$$

which is equation [6](#)

Finally, since $1 - e^{-a} < a$ for all $a > 0$, we have

$$\frac{L}{m_x} (1 - e^{-m_x G}) < L G,$$

which combined with the two bounds proves the strict inequality equation [7](#) and completes the proof. \square

A.2 CONVERGENCE JUSTIFICATION OF SELF-GENERATION

Theorem A.6. For each $L_{SM}^{pred}(\theta)$ and $L_{SM}^{total}(\theta)$, its denoising score matching are represented as follows:

$$\begin{aligned} L_{DSM}^{pred}(\theta) &= \mathbb{E}_{t, \mathbf{x}^{total}, \mathbf{x}_t^{total}} [\lambda(t) \|s_\theta(t, \mathbf{x}_t^{pred}, \mathbf{x}^{hist}) - \nabla_{\mathbf{x}_t^{pred}} \log p(\mathbf{x}_t^{total} | \mathbf{x}^{total})\|_2^2] \\ L_{DSM}^{total}(\theta) &= \mathbb{E}_{t, \mathbf{x}^{total}, \mathbf{x}_t^{total}} [\lambda(t) \|s_\theta(t, \mathbf{x}_t^{total}, \mathbf{x}^{hist}) - \nabla_{\mathbf{x}_t^{total}} \log p(\mathbf{x}_t^{total} | \mathbf{x}^{total})\|_2^2] \end{aligned}$$

Therefore, these models aim same conditional score function since $\nabla_{\mathbf{x}_t^{total}} \log p(\mathbf{x}_t^{total} | \mathbf{x}^{total}) = \nabla_{[\mathbf{x}_t^{hist}, \mathbf{x}_t^{pred}]} \log p(\mathbf{x}_t^{total} | \mathbf{x}^{total})$.

Remark. Our total-generation loss *recovers*—rather than merely matches—the same conditional score function. Concretely: (i) although SFdiff is trained on the full-sequence score, it still preserves the conditional distribution $p(\mathbf{x}^{pred} | \mathbf{x}^{hist})$; (ii) by generating the entire sequence, the model simultaneously denoises the historical part, improving robustness when the input history contains anomalies.

Proof. We prove denoising score matching loss of prediction, $L_{DSM}^{total}(\theta)$. The result of $L_{DSM}^{pred}(\theta)$ can be derived similarly. We start from decomposing it:

$$\begin{aligned} L_{SM}^{total}(\theta) &= -2 \cdot \mathbb{E}_t \mathbb{E}_{\mathbf{x}^{hist}} \mathbb{E}_{\mathbf{x}_t^{total}} \langle s_\theta(t, \mathbf{x}_t^{total}, \mathbf{x}^{hist}), \nabla_{\mathbf{x}_t^{total}} \log p(\mathbf{x}_t^{total} | \mathbf{x}^{hist}) \rangle \\ &\quad + \mathbb{E}_t \mathbb{E}_{\mathbf{x}^{hist}} \mathbb{E}_{\mathbf{x}_t^{total}} \left[\|s_\theta(t, \mathbf{x}_t^{total}, \mathbf{x}^{hist})\|_2^2 \right] + C_1 \end{aligned}$$

Here, C_1 is a constant that does not depend on the parameter θ , and $\langle \cdot, \cdot \rangle$ means the inner product. Then, the first part's expectation of the right-hand side can be expressed as follows:

$$\begin{aligned}
& \mathbb{E}_{\mathbf{x}_t^{total}} \langle s_\theta(t, \mathbf{x}_t^{total}, \mathbf{x}^{hist}), \nabla_{\mathbf{x}_t^{total}} \log p(\mathbf{x}_t^{total} | \mathbf{x}^{hist}) \rangle \\
&= \int_{\mathbf{x}_t^{total}} \langle s_\theta(t, \mathbf{x}_t^{total}, \mathbf{x}^{hist}), \nabla_{\mathbf{x}_t^{total}} \log p(\mathbf{x}_t^{total} | \mathbf{x}^{hist}) \rangle p(\mathbf{x}_t^{total} | \mathbf{x}^{hist}) d\mathbf{x}_t^{total} \\
&= \int_{\mathbf{x}_t^{total}} \langle s_\theta(t, \mathbf{x}_t^{total}, \mathbf{x}^{hist}), \frac{1}{p(\mathbf{x}^{hist})} \frac{\partial p(\mathbf{x}_t^{total}, \mathbf{x}^{hist})}{\partial \mathbf{x}_t^{total}} \rangle d\mathbf{x}_t^{total} \\
&= \int_{\mathbf{x}^{total}} \int_{\mathbf{x}_t^{total}} \langle s_\theta(t, \mathbf{x}_t^{total}, \mathbf{x}^{hist}), \frac{1}{p(\mathbf{x}^{hist})} \frac{\partial p(\mathbf{x}_t^{total}, \mathbf{x}^{hist}, \mathbf{x}^{total})}{\partial \mathbf{x}_t^{total}} \rangle d\mathbf{x}_t^{total} d\mathbf{x}^{total} \\
&= \int_{\mathbf{x}^{total}} \int_{\mathbf{x}_t^{total}} \langle s_\theta(t, \mathbf{x}_t^{total}, \mathbf{x}^{hist}), \frac{\partial p(\mathbf{x}_t^{total} | \mathbf{x}^{total})}{\partial \mathbf{x}_t^{total}} \rangle \frac{p(\mathbf{x}^{hist}, \mathbf{x}^{total})}{\mathbf{x}^{hist}} d\mathbf{x}_t^{total} d\mathbf{x}^{total} \\
&= \int_{\mathbf{x}^{total}} \int_{\mathbf{x}_t^{total}} \langle s_\theta(t, \mathbf{x}_t^{total}, \mathbf{x}^{hist}), \frac{\partial p(\mathbf{x}_t^{total} | \mathbf{x}^{total})}{\partial \mathbf{x}_t^{total}} \rangle p(\mathbf{x}^{total} | \mathbf{x}^{hist}) d\mathbf{x}_t^{total} d\mathbf{x}^{total} \\
&= \mathbb{E}_{\mathbf{x}^{total}} \left[\int_{\mathbf{x}_t^{total}} \langle s_\theta(t, \mathbf{x}_t^{total}, \mathbf{x}^{hist}), \frac{\partial p(\mathbf{x}_t^{total} | \mathbf{x}^{total})}{\partial \mathbf{x}_t^{total}} \rangle d\mathbf{x}_t^{total} \right] \\
&= \mathbb{E}_{\mathbf{x}^{total}} \left[\int_{\mathbf{x}_t^{total}} \langle s_\theta(t, \mathbf{x}_t^{total}, \mathbf{x}^{hist}), \nabla_{\mathbf{x}_t^{total}} \log p(\mathbf{x}_t^{total} | \mathbf{x}^{total}) \rangle p(\mathbf{x}_t^{total} | \mathbf{x}^{total}) d\mathbf{x}_t^{total} \right] \\
&= \mathbb{E}_{\mathbf{x}^{total}} \mathbb{E}_{\mathbf{x}_t^{total}} [\langle s_\theta(t, \mathbf{x}_t^{total}, \mathbf{x}^{hist}), \nabla_{\mathbf{x}_t^{total}} \log p(\mathbf{x}_t^{total} | \mathbf{x}^{total}) \rangle] \\
&= \mathbb{E}_{\mathbf{x}^{total}} \mathbb{E}_{\mathbf{x}_t^{total}} [\langle s_\theta(t, \mathbf{x}_t^{total}, \mathbf{x}^{hist}), \nabla_{\mathbf{x}_t^{total}} \log p(\mathbf{x}_t^{total} | \mathbf{x}^{total}) \rangle]
\end{aligned}$$

The second part's expectation of the right-hand side can be rewritten similarly, therefore we can derive following result:

$$\begin{aligned}
L_{SM}^{total}(\theta) &= -2 \cdot \mathbb{E}_t \mathbb{E}_{\mathbf{x}^{total}} \mathbb{E}_{\mathbf{x}_t^{total}} \langle s_\theta(t, \mathbf{x}_t^{total}, \mathbf{x}^{hist}), \nabla_{\mathbf{x}_t^{total}} \log p(\mathbf{x}_t^{total} | \mathbf{x}^{hist}) \rangle \\
&\quad + \mathbb{E}_t \mathbb{E}_{\mathbf{x}^{hist}} \mathbb{E}_{\mathbf{x}^{total}} \mathbb{E}_{\mathbf{x}_t^{total}} \left[\|s_\theta(t, \mathbf{x}_t^{total}, \mathbf{x}^{hist})\|_2^2 \right] + C_1 \\
&= L_{DSM}^{total}(\theta) + C_1
\end{aligned}$$

C is a constant that does not depend on the parameter θ .

Similarly, we compute $L_{DSM}^{pred}(\theta)$. We give proof on inner product part and the other are deduced directly from the case of $L_{DSM}^{total}(\theta)$.

$$\begin{aligned}
& \mathbb{E}_{\mathbf{x}_t^{pred}} \langle s_\theta(t, \mathbf{x}_t^{pred}, \mathbf{x}^{hist}), \nabla_{\mathbf{x}_t^{pred}} \log p(\mathbf{x}_t^{pred} | \mathbf{x}^{hist}) \rangle \\
&= \int_{\mathbf{x}_t^{pred}} \langle s_\theta(t, \mathbf{x}_t^{pred}, \mathbf{x}^{hist}), \nabla_{\mathbf{x}_t^{pred}} \log p(\mathbf{x}_t^{pred} | \mathbf{x}^{hist}) \rangle p(\mathbf{x}_t^{pred} | \mathbf{x}^{hist}) d\mathbf{x}_t^{pred} \\
&= \int_{\mathbf{x}_t^{pred}} \langle s_\theta(t, \mathbf{x}_t^{pred}, \mathbf{x}^{hist}), \frac{1}{p(\mathbf{x}^{hist})} \frac{\partial p(\mathbf{x}_t^{pred}, \mathbf{x}^{hist})}{\partial \mathbf{x}_t^{pred}} \rangle d\mathbf{x}_t^{pred} \\
&= \int_{\mathbf{x}^{pred}} \int_{\mathbf{x}_t^{total}} \langle s_\theta(t, \mathbf{x}_t^{pred}, \mathbf{x}^{hist}), \frac{1}{p(\mathbf{x}^{hist})} \frac{\partial p(\mathbf{x}_t^{total}, \mathbf{x}^{hist}, \mathbf{x}^{pred})}{\partial \mathbf{x}_t^{pred}} \rangle d\mathbf{x}_t^{total} d\mathbf{x}^{pred} \\
&= \int_{\mathbf{x}^{pred}} \int_{\mathbf{x}_t^{total}} \langle s_\theta(t, \mathbf{x}_t^{pred}, \mathbf{x}^{hist}), \frac{\partial p(\mathbf{x}_t^{total} | \mathbf{x}^{total})}{\partial \mathbf{x}_t^{total}} \rangle \frac{p(\mathbf{x}^{hist}, \mathbf{x}^{pred})}{\mathbf{x}^{hist}} d\mathbf{x}_t^{total} d\mathbf{x}^{pred} \\
&= \int_{\mathbf{x}^{pred}} \int_{\mathbf{x}_t^{total}} \langle s_\theta(t, \mathbf{x}_t^{pred}, \mathbf{x}^{hist}), \frac{\partial p(\mathbf{x}_t^{total} | \mathbf{x}^{total})}{\partial \mathbf{x}_t^{total}} \rangle p(\mathbf{x}^{pred} | \mathbf{x}^{hist}) d\mathbf{x}_t^{total} d\mathbf{x}^{pred} \\
&= \mathbb{E}_{\mathbf{x}^{pred}} \left[\int_{\mathbf{x}_t^{total}} \langle s_\theta(t, \mathbf{x}_t^{pred}, \mathbf{x}^{hist}), \frac{\partial p(\mathbf{x}_t^{total} | \mathbf{x}^{total})}{\partial \mathbf{x}_t^{total}} \rangle d\mathbf{x}_t^{total} \right] \\
&= \mathbb{E}_{\mathbf{x}^{pred}} \left[\int_{\mathbf{x}_t^{total}} \langle s_\theta(t, \mathbf{x}_t^{pred}, \mathbf{x}^{hist}), \nabla_{\mathbf{x}_t^{total}} \log p(\mathbf{x}_t^{total} | \mathbf{x}^{total}) \rangle p(\mathbf{x}_t^{total} | \mathbf{x}^{total}) d\mathbf{x}_t^{total} \right] \\
&= \mathbb{E}_{\mathbf{x}^{pred}} \mathbb{E}_{\mathbf{x}_t^{total}} [\langle s_\theta(t, \mathbf{x}_t^{pred}, \mathbf{x}^{hist}), \nabla_{\mathbf{x}_t^{total}} \log p(\mathbf{x}_t^{total} | \mathbf{x}^{total}) \rangle]
\end{aligned}$$

□

B DESCRIPTIONS OF DATASETS, HYPERPARAMETERS AND MISCELLANEOUS ENVIRONMENTS

In this section, we describe model architecture, datasets and hyperparameters.

We first describe the diffusion architecture used in SFdiff. To effectively capture the conditional score function along the temporal axis, we adapt DiffWave (Kong et al., 2021) to our settings. Since SFdiff is based on DiffWave, we highlight the key differences. As derived in Theorem 3.2, the input consists of the diffusion timestep, the diffused target data, and historical data, i.e. $t, \mathbf{x}_t^{\text{mst}}, \mathbf{x}_t^{\text{total}}$. Consistent with previous works (Ho et al., 2020; Kong et al., 2021), the timestep t is embedded into a continuous domain using sinusoidal embedding:

$$\text{embedding}(t) = [\sin(t/N^{0/d}), \dots, \sin(t/N^{d-1/d}), \cos(t/N^{0/d}), \dots, \cos(t/N^{d-1/d})]$$

, where d is embedding dimension and N is hyperparameterset to 128 and 10,000, respectively. Furthermore, since our main diffusion-based forecasting baselines are DDPM methods, we use VP SDE and an Euler-Maruyama sampling predictor without corrector, which are a generalized formulation of DDPM (c.f. Section 2.1) and a default setting of VP SDE in Song et al. (2020), respectively. All experiments are conducted by using the following software and hardware environments: UBUNTU 18.04 LTS, PYTHON 3.9.12, CUDA 9.1, NVIDIA Driver 470.141, i9 CPU, and GEFORCE RTX 2080 Ti.

Table 5: Description of datasets and hyperparameters.

	Dimension	Timesteps	Domain	γ	L_{hist}	L_{pred}	N_{step}	N_{iter}	w	$N_{\text{iter}}^{\text{CFG}}$	# test sample
Exchange	8	6071	\mathbb{R}^+	0.1	90	30	100	72	0.01	36	7
Solar	137	7009	\mathbb{R}^+	0.1	72	24	200	61	0.01	71	7
Electricity	370	5833	\mathbb{R}^+	0.5	72	24	50	44	0.01	66	7
Taxi	1214	1488	\mathbb{N}	0.1	48	24	50	14	0.1	24	56
Wiki	2000	792	\mathbb{N}	0.1	90	30	250	5	0.01	6	5

C CLASSIFIER-FREE GUIDANCE (CFG)

While CFG is a well-known technique in discrete-time DDPMs, we adapt it to the continuous score-SDE framework. We thus provide additional details on how our continuous-time interpretation modifies the standard CFG approach for time-series.

To incorporate an auxiliary classifier in naïve conditional generation, Dhariwal & Nichol (2021) introduced classifier guidance, modifying the standard denoising process by adjusting the estimated noise. Originally, $\epsilon(\mathbf{x}_t|\mathbf{c}) \sim -\sigma_t \nabla_{\mathbf{x}_t} \log p(\mathbf{x}_t|\mathbf{c})$ is replaced with $\tilde{\epsilon}(\mathbf{x}_t|\mathbf{c}) = \epsilon(\mathbf{x}_t|\mathbf{c}) - w\sigma_t \nabla_{\mathbf{x}_t} \log p(\mathbf{c}|\mathbf{x}_t)$, where w is a weighting term, and an additional classifier is trained to calculate $p(\mathbf{c}|\mathbf{x}_t)$. From the perspective of score-based SDEs, this approach can be interpreted as altering the score function $\nabla_{\mathbf{x}_t} \log p(\mathbf{x}_t|\mathbf{c})$ to $\nabla_{\mathbf{x}_t} \log \tilde{p}(\mathbf{x}_t|\mathbf{c}) = \nabla_{\mathbf{x}_t} \log p(\mathbf{x}_t|\mathbf{c}) + w \nabla_{\mathbf{x}_t} \log p(\mathbf{c}|\mathbf{x}_t) = \nabla_{\mathbf{x}_t} \log p(\mathbf{x}_t|\mathbf{c}) p(\mathbf{c}|\mathbf{x}_t)^w$, which means $\tilde{p}(\mathbf{x}_t|\mathbf{c}) \sim p(\mathbf{x}_t|\mathbf{c}) p(\mathbf{c}|\mathbf{x}_t)^w$ and effectively incorporating the classifier into the generative process.

To address the dependency on an additional classifier, Ho & Salimans (2022) proposed classifier-free guidance (CFG), allowing the generation process to be guided without the need for a separately trained classifier. In CFG, the model learns the modified noise estimate $\tilde{\epsilon}(\mathbf{x}_t|\mathbf{c}) = (1+w)\epsilon(\mathbf{x}_t|\mathbf{c}) - w\epsilon(\mathbf{x}_t)$ by training a single model that handles both conditional and unconditional generations. This is achieved by training with zero-padding for the unconditional case, resulting in $\tilde{\epsilon}_\theta(\mathbf{x}_t, \mathbf{c}) = (1+w)\epsilon_\theta(\mathbf{x}_t, \mathbf{c}) - w\epsilon_\theta(\mathbf{x}_t, \mathbf{0})$.

Generally, applying Classifier-Free Guidance (CFG) to historical data can be seen as a subset of conditional generation, which is a natural concept. However, in many existing conditional diffusion approaches (e.g., text-to-image or time-series forecasting), noisy conditioning inputs can degrade performance, as highlighted by (Na et al., 2024) (see Appendix C of their paper). This is why CFG

has rarely been adopted in time-series forecasting methods so far—if the historical data is noisy, naive CFG can amplify that noise and harm predictions, as highlighted in Table 1 of our paper.

In our framework, though, self-generation cleans the historical data during the diffusion process, alleviating overdependency on noised historical sequence. Hence, rather than reinforcing noise, CFG strengthens the useful conditional signal. Put differently, we first reduce the noise in, while applying CFG to guide the prediction more effectively. This synergy between self-generation and CFG is described in Section 3 where we show that while naive CFG can degrade performance under noisy conditions, our approach remains robust precisely because of the joint denoising mechanism.

D EXPLANATIONS ABOUT BASELINES

Classical Methods.

- **VAR/VAR-Lasso** (Lütkepohl, 2005): Vector AutoRegression (VAR) estimates linear dependencies across multiple time-series. VAR-Lasso adds an ℓ_1 penalty to mitigate overfitting in high-dimensional data.
- **GARCH** (van der Weide, 2002): A model that captures time-varying volatility (conditional variance), particularly popular in financial contexts.
- **VES** (Hyndman et al., 2008): A vectorized exponential smoothing method using state-space formulations to handle multivariate trends.

VAE/State-Space Method.

- **KVAE** (Fraccaro et al., 2017): Combines a Kalman Filter with a Variational Autoencoder to learn complex latent dynamics for forecasting.

Deep Learning-Based Methods.

- **Vec-LSTM (ind-scaling and low-copula)** (Salinas et al., 2019): LSTM-based multivariate forecasting; the “ind-scaling” version assumes independent output distributions, while the “low-copula” version models their joint distribution via copulas.
- **GP scaling/copula** (Salinas et al., 2019): Similar to Vec-LSTM but leverages Gaussian Processes for uncertainty estimation, offering more flexible distributional modeling at higher computational cost.
- **Transformer MAF** (Rasul et al., 2020): A Transformer architecture combined with a Masked Autoregressive Flow for modeling complex, long-range dependencies in multivariate time series.

Diffusion-Based Methods.

- **TimeGrad** (Rasul et al., 2021): A DDPM-based approach that generates forecasts autoregressively, one step at a time.
- **CSDI** (Tashiro et al., 2021): Primarily designed for time-series imputation but can also perform one-shot forecasting of the future horizon by treating it as a masked region.

Unlike these baselines, our **SFdiff** method denoises the entire time-series (both past and future) during generation. By jointly modeling historical and future observations, SFdiff can more effectively handle noisy inputs, leading to improved probabilistic forecasts.

D.1 COMPARISON WITH LTSF BASELINES

Throughout the main paper we used the standard predictor-corrector (PC) sampler (which we denote “Naive”) with optional classifier-free guidance (CFG). Because LTSF methods such as DLinear, FEDformer and PatchTST are trained via deterministic regression losses, comparing them against our method gives a fairer picture of pure point forecasting quality.

To evaluate, we follow the “moderate-horizon” configuration used 72 historical steps \rightarrow 24-step forecast for SOLAR, ELECTRICITY, TAXI, and WIKI. Metrics are RMSE and MAE (lower is better), which is typical setting of deterministic sampling methods. SFdiff results are averaged over five seeds; \pm indicates one-standard-deviation.

Model	Dataset	Approach	RMSE	MAE
DLinear	Solar	–	30.12	18.60
	Electricity	–	409.82	49.88
	Taxi	–	5.07	3.34
	Wiki	–	6887.80	1471.84
FEDformer	Solar	–	30.08	19.36
	Electricity	–	602.59	80.57
PatchTST	Solar	–	31.00	18.86
	Electricity	–	348.36	45.28
SFdiff	Solar	CFG	30.34 ± 0.12	14.02 ± 0.04
		Naive	28.53 ± 0.43	12.46 ± 0.13
	Electricity	CFG	321.85 ± 36.44	40.39 ± 0.10
		Naive	326.00 ± 52.76	40.35 ± 0.12
	Taxi	CFG	3.95 ± 0.00	2.61 ± 0.00
		Naive	4.15 ± 0.00	2.74 ± 0.00
	Wiki	CFG	5910.28 ± 12.03	698.78 ± 0.64
		Naive	5905.92 ± 70.48	718.16 ± 17.88

Table 6: Moderate-horizon forecasting: Naive PC sampler versus applying CFG, plus LTSF regressors. Bold indicates the best among SFdiff variants.

Table 6 underscores that SFdiff remains strong even when evaluated with a fully deterministic sampler, while also highlighting its stability compared to popular regression-style LTSF architectures.

Notably, Robustness in higher dimensions is worth to be mentioned. As dimensionality increases (Electricity \rightarrow Wiki), LTSF models’ errors grow quickly, whereas SFdiff’s degradation is moderate—confirming our full-sequence denoising advantage. Moreover, Resource constraints became matter. FEDformer required custom frequency hacks on Electricity and failed on Wiki; PatchTST ran out of GPU memory on several settings. SFdiff, while generative, scales gracefully once trained—it needs no architectural changes for different frequencies or horizons.

E WHY GENERATE THE TOTAL SEQUENCE CONDITIONED ON HISTORICAL DATA?

In this section, we empirically justify the necessity of conditioning the generation of the total sequence on historical observations. To clearly demonstrate this, we conduct two additional comparisons: 1) total generation without conditioning (unconditional total generation) using replacement methods, and 2) unconditional generation guided by Observation Self-Guidance (OSG) (Kollovich et al. 2023a)⁵. Both comparisons highlight the benefits and practical advantages of our proposed Self-generation approach.

Replacement Method. Previous approaches (Song et al. 2020; Ho et al. 2022) suggest continuously replacing the historical portion during diffusion with corresponding diffused historical values from the forward SDE. Specifically, at denoising step t , the history portion of $\mathbf{x}_t^{\text{total}}$ is replaced with forward-diffused historical values. This approach solely utilizes denoised historical information without leveraging original historical data.

⁵We omit OSG on our main paper, since SFdiff aims to multivariate generation while OSG targets univariate generation.

Observation Self-Guidance (OSG). Another method, OSG, guides unconditional generation using score-SDE by computing gradients of the score network (Song et al., 2020). Although OSG improves performance notably (Solar dataset achieved 0.2490 ± 0.0094), it requires repeated gradient computations of the score network, resulting in significantly increased computational overhead and limiting practical applicability in high-dimensional settings (Electricity).

Table 7: Performance comparison of total sequence generation methods: conditioned SFdiff (ours), unconditional generation methods (Replacement and OSG), and prediction-only generation.

Dataset	SFdiff (ours)	Replacement	OSG	Prediction-only
Exchange	.0054\pm.0002	.0057 \pm .0001	.0055 \pm .0003	.0063 \pm .0002
Solar	.2501 \pm .0080	.2774 \pm .0083	.2490\pm.0094	.2871 \pm .0202
Electricity	.0153\pm.0003	.0173 \pm .0012	-	.0210 \pm .0013

Table 7 summarizes the performance of the unconditional total-generation methods compared to SFdiff (conditioned total generation). Additionally, results of prediction-only generation (presented previously in Table 1 in the main paper) are included for reference.

The results highlight two key observations:

- **Conditioning Improves Forecasting Quality.** Conditioned SFdiff consistently outperforms both unconditional total-generation (Replacement) and prediction-only generation methods across most datasets, while achieving comparable performance to OSG on Solar. This indicates that conditioning on historical data is beneficial, likely because it preserves residual information from the original conditions.
- **Practical Efficiency and Scalability.** Although OSG achieves slightly better performance on the Solar dataset, its requirement for repeated gradient computations makes it computationally expensive and restricts scalability. In contrast, SFdiff directly integrates denoising into the reverse diffusion process, achieving comparable or superior performance across datasets with significantly lower computational overhead.

Overall, these empirical comparisons provide strong justification for conditioning on historical data when generating the total sequence, as our Self-generation approach effectively balances performance and computational efficiency.

F ADDITIONAL QUANTITATIVE EVALUATION ON TOY DATASETS

Model	Dataset	γ	MSE	MAE
SFdiff	2D	0.01	.2953 \pm .0000	.3601 \pm .0000
		0.1	.0006 \pm .0000	.0147 \pm .0000
	3D	0.01	.0210 \pm .0000	.0956 \pm .0000
		0.1	.0002 \pm .0000	.0076 \pm .0000

Table 8: Quantitative results on toy datasets. Lower MSE and MAE indicate better forecasting accuracy.

In this section, we provide additional quantitative evaluation of SFdiff using the toy datasets described in Section 4.1. Specifically, we report the Mean Squared Error (MSE) and Mean Absolute Error (MAE) metrics computed on the prediction segments of the test sets. As shown in Table 8, both MSE and MAE significantly improve with increased γ , aligning with the visual evidence of purified historical conditions demonstrated in Figure 3. These results further support our claim that total sequence generation is effective in enhancing forecasting robustness under noisy historical conditions.