TASLP .

StoRM: A Diffusion-based Stochastic Regeneration Model for Speech Enhancement and Dereverberation

Jean-Marie Lemercier, *Student Member, IEEE*, Julius Richter, *Student Member, IEEE*, Simon Welker, *Student Member, IEEE*, Timo Gerkmann, *Senior Member, IEEE*

arxiv open review .

***Abstract*—Diffusion models have shown a great ability at bridging the performance gap between predictive and generative approaches for speech enhancement. We have shown that they may even outperform their predictive counterparts for non-additive corruption types or when they are evaluated on mismatched conditions. However, diffusion models suffer from a high computational burden, mainly as they require to run a neural network for each reverse diffusion step, whereas predictive approaches only require one pass. As diffusion models are generative approaches they may also produce vocalizing and breathing artifacts in adverse conditions. In comparison, in such difficult scenarios, predictive models typically do not produce such artifacts but tend to distort the target speech instead, thereby degrading the speech quality.**

In this work, we present a *stochastic regeneration* approach where an estimate given by a predictive model is provided as a guide for further diffusion. We show that the proposed approach uses the predictive model to remove the vocalizing and breathing artifacts while producing very high quality samples thanks to the diffusion model, even in adverse conditions. We further show that this approach enables to use lighter sampling schemes with fewer diffusion steps without sacrificing quality, thus lifting the computational burden by an order of magnitude.

Source code and audio examples are available online[1].

***Index Terms*—score-based generative models, diffusion models, speech enhancement, speech dereverberation, predictive learning.**

I. Introduction

In real-life scenarios and modern communication devices, clean speech sources are often polluted by background noise, interfering speakers, room acoustics and codec degradation [1], [2]. We refer to this phenomenon as *speech corruption*, and denote by *speech restoration* the art of recovering clean speech from the corrupted signal [3]. On the one hand, traditional speech restoration methods leverage the statistical properties of the target and interference signals in various domains e.g. time, spectrum, cepstrum or spatial distribution [4]. On the other hand, machine learning techniques try to learn these statistical properties and how to exploit them from data [5]. Machine learning algortihms can be categorized into predictive (also called discriminative) approaches and generative approaches. We will choose the term *predictive* over *discriminative* as it fits both classification and regression tasks [6]. The field of speech restoration is dominated by predictive approaches that use supervised learning to learn a single best deterministic mapping between corrupted speech $\mathbf{y}$ and the corresponding clean speech target $\mathbf{x}$ [5]. These methods include for instance time-frequency (T-F) masking [7], time domain methods [8], [9] or direct spectro-temporal mapping [10]. They have contributed to drastically increasing the quality of speech restoration algorithms. However, they candistort target speech and suffer from generalizability issues that we expose hereafter [11], [12].

In contrast, generative models implicitly or explicitly learn the target distribution and allow to generate multiple valid estimates instead of a single best estimate as for predictive approaches [6]. Generative approaches include variational auto-encoders (VAEs) learning explicit density estimations [13]–[16], normalizing flows adding invertible transforms to obtain tractable marginal likelihoods [17], [18], generative adversarial networks (GANs) estimating implicit distributions [19], [20] and diffusion approaches [21]–[23]. We talk of *conditional* generative models when a covariate $\mathbf{c}$ is used to guide the generation, leading to the conditional distribution $p(\mathbf{x}|\mathbf{c})$ [6]. This conditioning can either be another modality describing the data (e.g. $\mathbf{c}$ could be video when $\mathbf{x}$ is speech), or a modified version of the data, an obvious example being corrupted speech $\mathbf{y}$ when the underlying task is speech restoration. By integrating stochasticity in their latent structure, generative models can capture the inherent uncertainty of the data distribution and produce realistic samples belonging to that distribution rather than a mean of optimal candidates [6]. In doing so, they may obtain better perceptual metrics at the cost of higher point-wise distortion [24]. In the imaging domain for instance, it was observed that predictive approaches tend to brush over the fine-grained details of the considered domain [24], [25]. Furthermore, predictive models may result in limited generalization abilities towards unseen corruption types or speakers as compared to generative models, which is demonstrated for diffusion-based generative speech enhancement in Richter et al. [12].

We focus in this work on such diffusion-based generative models, or simply *diffusion models*, which have met great success in generating high-quality samples of natural images [21]–[23], [26]. Diffusion models use a *forward process* to slowly turn data into a tractable prior, usually a standard normal distribution, and train a neural network to solve the *reverse process* to generate clean data from this prior [27]. These diffusion models can also be used for conditional generation in restoration tasks, which has recently been proposed for speech processing tasks such as enhancement and dereverberation [12], [28]–[30] as well as bandwidth extension [11], [31].

先.

增强 去混响

带宽扩展

One limiting aspect of diffusion models is their heavy computational burden. Several steps are needed for reverse diffusion, each

This work has been funded by the Federal Ministry for Economic Affairs and Climate Action, project 01MK20012S, AP380, the German Research Foundation (DFG) in the transregio project Crossmodal Learning (TRR 169) and DASHH (Data Science in Hamburg - HELMHOLTZ Graduate School for the Structure of Matter) with the Grant-No. HIDSS-0002.

Simon Welker is with the Signal Processing Group, Department of Informatics, Universität Hamburg, 22527 Hamburg Germany, and with the Center for Free-Electron Laser Science, DESY, 22607 Hamburg, Germany (e-mail: simon.welker@uni-hamburg.de).

The authors are with the Signal Processing Group, Department of Informatics, Universität Hamburg, 22527 Hamburg Germany (e-mail:{jeanmarie.lemercier; julius.richter; timo.gerkmann}@uni-hamburg.de).

[1] https://uhh.de/inf-sp-storm

指模型计算得分来源

of them calling the neural network used for score estimation. Much effort has been recently put into reducing this number of steps, either by optimization of the reverse noise schedule [32], modifications in the formulation of the diffusion processes [33], [34], or projection into a latent space [35] or a reduced subspace [36]. We also observed in past experiments that our previously proposed diffusion model is prone to confuse phonemes and generate vocalizing artifacts when facing very adverse conditions. This is due to the generative behaviour of the model under high uncertainty over the presence or nature of speech, and this naturally leads to a degradation, e.g. in automatic speech recognition (ASR) experimental results.

当语音处于高度不确定时，模型生成行为会导致退化

结合预测和生成模型的随机再生方案

In this work, we propose a *stochastic regeneration* scheme combining predictive and generative models to produce high quality samples while reducing the computational burden of diffusion models and their tendency to generate unwanted artifacts. We propose to first use a predictive approach to estimate a restored version of the corrupted speech. This estimate is then used as a guide by a diffusion model, which requires only a few diffusion steps to output a final clean speech estimation where the distortions introduced by the predictive stage are corrected while vocalizing artifacts and phonetic confusions are avoided. Both listening experiments and instrumental metrics confirm an impressive state-of-the-art perceptual quality of our proposed approach. Other refinement approaches using diffusion models were recently proposed. The *stochastic refinement* approach [24], [37] subtracts the output of the predictive model from the corrupted speech, and this residual is used for further estimation by a diffusion model. We argue hereafter that learning the residual is however a hard task and demonstrate that our approach outperforms this stochastic refinement in terms of instrumentally measured speech quality. Another refinement approach using diffusion models is *denoising diffusion restoration models* [38]–[40], where the corruption operator is assumed to be known (or at least its singular value decomposition) and is used to modify the reverse diffusion process without retraining the score model.

We evaluate our proposed approach for speech enhancement with low input signal-to-noise ratios (SNRs) and speech dereverberation, using clean speech from the WSJ0 corpus [41]. We also show ASR results on the TIMIT dataset [42], and report results on the standardized Voicebank/DEMAND dataset [43]. Finally, several ablation studies are performed with respect to sampling efficiency, intial predictor mismatch and training strategy.

II. Score-Based Diffusion Models

Diffusion models originally use discrete-time diffusion processes modeled by Markov chains [22]. They have been recently extended to continuous-time diffusion processes formulated by stochastic differential equations (SDEs) in [44], allowing for new training paradigms such as score matching [45], [46]. This class model is subsequently denoted as *score-based diffusion models*.

Score-based diffusion models are defined by three components: a forward diffusion process, a score function estimator, and a sampling method for inference.

A. Forward and reverse processes

随机前向过程

The stochastic forward process $\{\mathbf{x}_\tau\}_{\tau=0}^{T}$ used in score-based diffusion models is defined as an Itô SDE [44], [47]:

$$\mathrm{d}\mathbf{x}_\tau = \mathbf{f}(\mathbf{x}_\tau, \tau)\mathrm{d}\tau + g(\tau)\mathrm{d}\mathbf{w} \qquad (1)$$

where $\mathbf{x}_\tau$ is the current state of the process indexed by a continuous time variable $\tau \in [0,T]$ with the initial condition representing clean speech $\mathbf{x_0} = \mathbf{x}$. The *diffusion time* variable τ relates to the progress of the stochastic process and should not to be mistaken for our usual notion of *signal time*. As our process is defined in the complex spectrogram domain, independently for each T-F bin, the variables in bold are assumed to be vectors in $\mathbb{C}^d$ containing the coefficients of a flattened complex spectrogram— with d being the product of the time and frequency dimensions— whereas variables in regular font represent real scalar values. The set $\{\mathbf{x}_\tau\}_{\tau\in]0,T[}$ can be considered as latent variables used to parameterize the conditional distribution $p(\mathbf{x}_\tau|\mathbf{x_0}, \mathbf{y})$. The stochastic process $\mathbf{w}$ denotes a standard d-dimensional Brownian motion, that is, $\mathrm{d}\mathbf{w}$ is a zero-mean Gaussian random variable with standard deviation $\mathrm{d}\tau$ for each T-F bin.

The *drift* function $\mathbf{f}$ and *diffusion* coefficient g as well as the initial condition $\mathbf{x_0}$ and the final diffusion time T uniquely define the Itô process $\{\mathbf{x}_\tau\}_{\tau=0}^{T}$ [47]. Under some regularity conditions on $\mathbf{f}, g$ allowing a unique and smooth solution to the Kolmogorov equations associated to (1), the reverse process $\{\mathbf{x}_\tau\}_{\tau=T}^{0}$ is another diffusion process defined as the solution of the following SDE [44], [48]:

$$\mathrm{d}\mathbf{x}_\tau = \left[-\mathbf{f}(\mathbf{x}_\tau, \tau) + g(\tau)^2 \nabla_{\mathbf{x}_\tau} \log p_\tau(\mathbf{x}_\tau)\right]\mathrm{d}\tau + g(\tau)\mathrm{d}\bar{\mathbf{w}}, \qquad (2)$$

where $\mathrm{d}\bar{\mathbf{w}}$ is a d-dimensional Brownian motion for the time flowing in reverse and $\nabla_{\mathbf{x}_\tau} \log p_\tau(\mathbf{x}_\tau)$ is the *score function*, i.e. the gradient of the logarithm data distribution for the current process state $\mathbf{x}_\tau$.

Speech restoration tasks can be regarded either as one-to-one mapping tasks between corrupted speech $\mathbf{y}$ and $\mathbf{x_0}$, which leads to predictive modelling; or as conditional generation tasks, i.e. generation of $\mathbf{x_0}$ conditioned on $\mathbf{y}$. Previous diffusion-based approaches proposed to condition the process explicitly within the neural network [49] or through guided classification [26]. In [28], the conditioning is directly incorporated into the diffusion process by defining the forward process as the solution to the following SDE:

$$\mathrm{d}\mathbf{x}_\tau = \underbrace{\gamma(\mathbf{y} - \mathbf{x}_\tau)}_{:=\mathbf{f}(\mathbf{x}_\tau, \mathbf{y})}\mathrm{d}\tau + \underbrace{\left[\sigma_{\min}\left(\frac{\sigma_{\max}}{\sigma_{\min}}\right)^{\tau}\sqrt{2\log\left(\frac{\sigma_{\max}}{\sigma_{\min}}\right)}\right]}_{:=g(\tau)}\mathrm{d}\mathbf{w}. \qquad (3)$$

This equation belongs to the class of Ornstein-Uhlenbeck SDEs [47], a subclass of Itô SDEs in which the drift function $\mathbf{f}$ is affine in $\mathbf{x}_\tau$ and does not depend on τ, and the diffusion coefficient g only depends on τ. The equation introduces a *stiffness* hyperparameter γ controlling the slope of the decay from $\mathbf{y}$ to $\mathbf{x_0}$, and σ_{min} and σ_{max} are two hyperparameters controlling the *noise scheduling*, that is, the amount of Gaussian white noise injected at each timestep of the process.

The interpretation of our forward process in Eq. (3), visualized on Fig. 1, is as follows: at each time step and for each T-F bin independently, an infinitesimal amount of corruption is added to the current process state $\mathbf{x}_\tau$, along with Gaussian noise with standard deviation $g(\tau)\mathrm{d}\tau$. Therefore, the mean of the current process decays exponentially towards $\mathbf{y}$ while the variance increases as in the variance-exploding scheme of Song et al. [44], leading to a final distribution $\mathbf{x}_\tau$ which is the corrupted signal $\mathbf{y}$ with some additional Gaussian noise. Given an initial condition $\mathbf{x}_0$ and the covariate

X_τ实际上是受损信号y加上高斯噪声得到的

T-F
对于每个时频单元：

正向过程

漂移项

扩散系数：决定每一步随机扰动的强度.

$$\mathrm{d}\mathbf{x}_\tau = \mathbf{f}(\mathbf{x}_\tau,\tau)\mathrm{d}\tau + g(\tau)\mathrm{d}\mathbf{w} \tag{1}$$

当前的过程状态

标准d维布朗运动，模拟随机波动
（均值为0方差为dτ）

逆向过程

：漂移项修正

扩散函数

$$\mathrm{d}\mathbf{x}_\tau = \left[-\mathbf{f}(\mathbf{x}_\tau,\tau) + \boxed{g(\tau)^2\nabla_{\mathbf{x}_\tau}\log p_\tau(\mathbf{x}_\tau)}\right]\mathrm{d}\tau + g(\tau)\mathrm{d}\bar{\mathbf{w}}, \tag{2}$$

~~分数函数~~
对数概率分布的梯度

时间反向流动的d维布朗运动

生成任务下，该项是被学习对象，以学习不同τ下数据分布情况

超参数：决定回归速率.

正向解释：

$$\mathrm{d}\mathbf{x}_\tau = \underbrace{\gamma(\mathbf{y}-\mathbf{x}_\tau)}_{:=\mathbf{f}(\mathbf{x}_\tau,\mathbf{y})}\mathrm{d}\tau + \underbrace{\left[\sigma_{\min}\left(\frac{\sigma_{\max}}{\sigma_{\min}}\right)^\tau\sqrt{2\log\left(\frac{\sigma_{\max}}{\sigma_{\min}}\right)}\right]}_{:=g(\tau)}\mathrm{d}\mathbf{w}. \tag{3}$$

标准布朗运动
（随机波动）

漂移项.
推动x_τ向y的方向回归

扩散系数
：控制加噪

x_τ：随机分布（期望分布）
y：添噪后的数据（受污染的观测信号）

前向过程：
x_0随时间步逐步加入（无限小）高斯噪声，该噪声具有 g(t)dt 标准差，

使x_τ均值趋于y，方差逐步增大，最终达到噪声较大状态.

X₀可以以指数速率变为y . y也可以以指数速率变回X₀ .

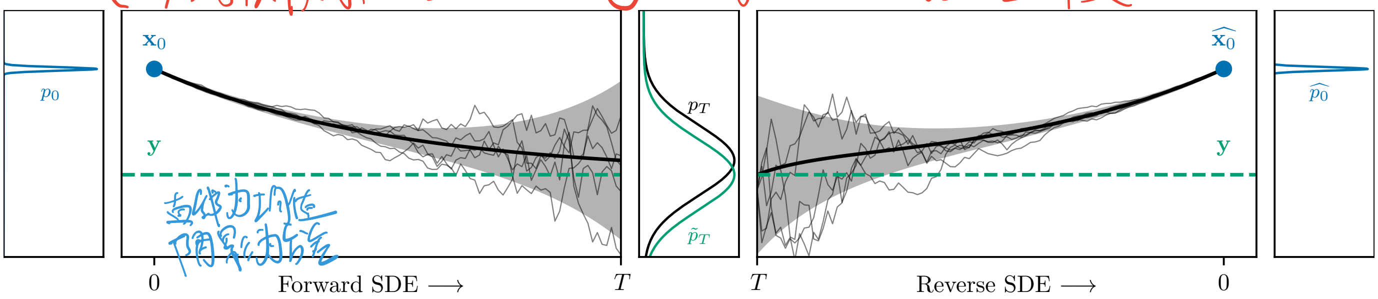


Fig. 1: *Visualization of the forward and backward processes in* (3)*. Mean curve* (5) *is in solid black and variance* (6) *is represented by the greyed area. Several realizations of the diffusion process are represented by thin black lines. The mismatch between p_τ centered on $\mathbf{x}_\tau$ and $\tilde{p}_\tau$ centered on $\mathbf{y}$ comes from the fact that the mean in* (5) *can not reach $\mathbf{y}$ in finite time. This mismatch causes unavoidable bias in the reverse process, even were the score perfectly known.*

$\mathbf{y}$, the solution to (3) admits the following complex Gaussian distribution for the process state $\mathbf{x}_\tau$ called *perturbation kernel*:

扰动核的分布

$$p_{0,\tau}(\mathbf{x}_\tau|\mathbf{x}_0,\mathbf{y}) = \mathcal{N}_\mathbb{C}\big(\mathbf{x}_\tau;\boldsymbol{\mu}(\mathbf{x}_0,\mathbf{y},\tau),\sigma(\tau)^2\mathbf{I}\big), \tag{4}$$

Following [50], we determine closed-form solutions for the mean $\boldsymbol{\mu}$ and variance $\sigma(\tau)^2$:

均值

$$\boldsymbol{\mu}(\mathbf{x}_0,\mathbf{y},\tau) = \mathrm{e}^{-\gamma\tau}\mathbf{x}_0 + (1-\mathrm{e}^{-\gamma\tau})\mathbf{y}, \tag{5}$$

均值以指数速率趋近y

方差

$$\sigma(\tau)^2 = \frac{\sigma_{\min}^2\left(\left(\sigma_{\max}/\sigma_{\min}\right)^{2\tau} - \mathrm{e}^{-2\gamma\tau}\right)\log(\sigma_{\max}/\sigma_{\min})}{\gamma + \log(\sigma_{\max}/\sigma_{\min})}. \tag{6}$$

方差逐步增加.

使用神经网络
可以估计该函数

B. Score function estimator

When performing inference, one tries to solve the reverse SDE in Eq. (2). In the general case, the score function $\nabla_{\mathbf{x}_\tau}\log p_\tau(\mathbf{x}_\tau)$ is not readily available, it can however be estimated by a deep neural network (DNN) $\mathbf{s}_\theta$, called the *score model*. Given the simple Gaussian form of the perturbation kernel $p_{0,\tau}(\mathbf{x}_\tau|\mathbf{x}_0,\mathbf{y})$ (4) and the regularity conditions exhibited by the mean and variance, a *denoising score matching* objective can be used to train the score model $\mathbf{s}_\theta$ [45], [46].

The score function of the perturbation kernel is:

$$\nabla_{\mathbf{x}_\tau}\log p_{0,\tau}(\mathbf{x}_\tau|\mathbf{x}_0,\mathbf{y}) = -\frac{\mathbf{x}_\tau - \boldsymbol{\mu}(\mathbf{x}_0,\mathbf{y},\tau)}{\sigma(\tau)^2}. \tag{7}$$

We can reparameterize the denoising score matching objective as follows [44]:

分数匹配目标:

$$\mathcal{J}^{(\mathrm{DSM})}(\phi) = \mathbb{E}_{t,(\mathbf{x}_0,\mathbf{y}),\mathbf{z},\mathbf{x}_\tau}\left[\left\|\mathbf{s}_\phi(\mathbf{x}_\tau,\mathbf{y},\tau) + \frac{\mathbf{z}}{\sigma(\tau)}\right\|_2^2\right]. \tag{8}$$

Here, we sample τ sampled uniformly in $[\tau_\epsilon,T]$ where τ_ϵ is a minimal diffusion time used to avoid numerical instabilities. The clean $\mathbf{x_0}$ and noisy $\mathbf{y}$ utterances are picked in the training set and the current process state is obtained as $\mathbf{x}_\tau = \boldsymbol{\mu}(\mathbf{x}_0,\mathbf{y},\tau) + \sigma(\tau)\mathbf{z}$, with $\mathbf{z} \sim \mathcal{N}_\mathbb{C}(\mathbf{z};\mathbf{0},\mathbf{I})$. This approach is analogous to the denoising objective used in the discrete-time formulation by [22], where one directly estimates the noise added at each step to learn the reverse process.

类似2020论文，直接估计每一步添加的
噪声来学习反向过程

C. Inference through reverse sampling

At inference time, we first sample an initial condition of the reverse process, corresponding to $\mathbf{x}_\tau$, with:

$$\mathbf{x}_\tau \sim \mathcal{N}_\mathbb{C}(\mathbf{x}_\tau;\mathbf{y},\sigma^2(\tau)\mathbf{I}), \tag{9}$$

This sample only approximates the training condition, as the final process distribution $p(\mathbf{x}_\tau)$ does not perfectly match $p(\mathbf{y})$ (see Fig. 1).

Conditional generation is then performed by solving the so-called *plug-in reverse SDE* from $\tau = T$ to $\tau = 0$, where the score function is replaced by its estimator $\mathbf{s}_\phi$, assuming the latter was trained e.g. according to Section II-B:

$$\mathrm{d}\mathbf{x}_\tau = \left[-\mathbf{f}(\mathbf{x}_\tau,\mathbf{y}) + g(\tau)^2\mathbf{s}_\phi(\mathbf{x}_\tau,\mathbf{y},\tau)\right]\mathrm{d}\tau + g(\tau)\mathrm{d}\bar{\mathbf{w}} \tag{10}$$

We use classical numerical solvers based on a discretization of (10) according to a uniform grid of N points on the interval $[0,T]$ (no minimal diffusion time is needed here). Classical solvers include the Euler-Maruyama method, higher-order single-step methods, and predictor-corrector sampler schemes [44]. In the latter, at each reverse step τ, the predictor uses a single-step method like Euler-Maruyama to generate $\mathbf{x}_\tau$, and the corrector uses the output of the score network to ensure consistency of the resulting sample with the estimated marginal distribution whose score is the score network estimate.

For notational convenience, we will denote by G_ϕ the generative model corresponding to the reverse diffusion process solver parameterized by the plug-in SDE (10) and the score network s_ϕ, such that the final estimate is $\hat{\mathbf{x}} = G_\phi(\mathbf{y})$.

III. Stochastic Regeneration with Diffusion Models

A. Predictive artifacts for images and spectrograms

Predictive models output single best estimates, having seen several optimal or near-optimal versions of clean speech corresponding to the same corrupted speech. This is especially true for ill-posed problems such as deblurring in image processing and dereverberation in speech processing. The consequence of this "regression to the mean" (as mentioned in [24]) in some image processing studies is that predictive approaches are incapable of reproducing fine-grained details like e.g. hair structure in human portraits [24], [25]. We also observed that, when trained to output clean speech spectrograms, predictive models tend to introduce distortions in the target speech when the corruption level is high, leading to *overdenoising* effects [11], [51].

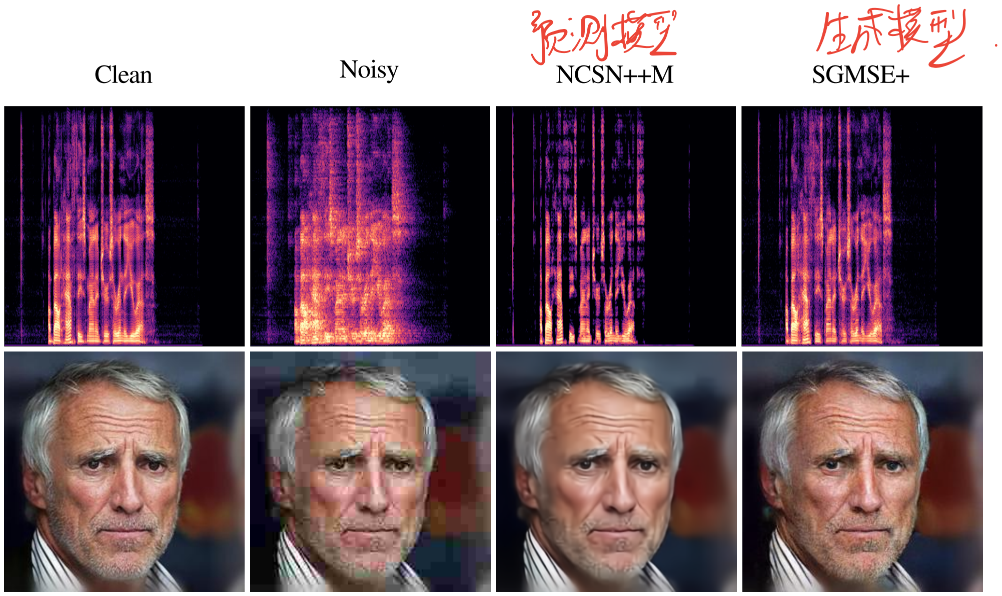
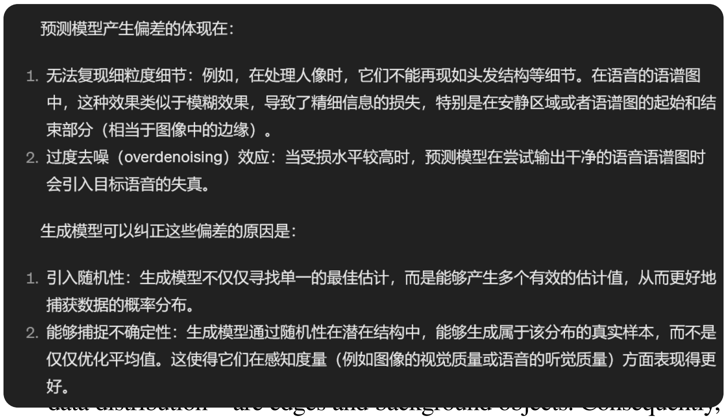


Fig. 2: *Visualization of samples obtained with predictive NCSN++M and generative SGMSE+ for two ill-posed problems, namely speech dereverberation (top, from [11]) and JPEG artifact removal (bottom, from [25]).*



预测模型产生偏差的体现在：

1. 无法复现细粒度细节：例如，在处理人像时，它们不能再现如头发结构等细节。在语音的语谱图中，这种效果类似于模糊效果，导致了精细信息的损失，特别是在安静区域或者语谱图的起始和结束部分（相当于图像中的边缘）。
2. 过度去噪（overdenoising）效应：当受损水平较高时，预测模型在尝试输出干净的语音语谱图时会引入目标语音的失真。

生成模型可以纠正这些偏差的原因是：

1. 引入随机性：生成模型不仅仅寻找单一的最佳估计，而是能够产生多个有效的估计值，从而更好地捕获数据的概率分布。
2. 能够捕捉不确定性：生成模型通过随机性在潜在结构中，能够生成属于该分布的真实样本，而不是仅仅优化平均值。这使得它们在感知度量（例如图像的视觉质量或语音的听觉质量）方面表现得更好。

data distribution – are edges and background objects. Consequently, these are the most difficult to represent with single point estimates. This is visualized in the third row of Fig. 2, where we directly compare spectrograms from our previous study [11] with images in Welker et al. [25, Fig. 1] which we partially reproduce here. w

Several paradigms can be considered using generative modelling to correct this bias of the predictive model without having to resort to a full-fletched computationally heavy diffusion-based generative model. Next, we present two of these approaches, namely *stochastic refinement* by Whang et al. [24] and *stochastic regeneration* which we propose here.

B. Stochastic refinement

Instead of performing the full-fletched reverse diffusion process from noisy speech to clean speech estimate, the *stochastic refinement* approach by Wang et al. uses both a predictive approach and a generative diffusion model for efficient inference.

A predictive model D_θ serves as an *initial predictor* producing an estimate $D_\theta(\mathbf{y})$. This estimate often lacks fine-grained detail and has significant target speech distortions, especially for corruption models like reverberation [11]. Let us write the predictive model output as:

$$\underbrace{D_\theta(\mathbf{y})}_{\text{预测模型估计}} = \mathbf{x} - \overset{\text{预测模型引入的目标失真（由过度去噪引起）}}{\mathbf{x}^{(\mathrm{dis})}} + \overset{\text{残留噪音}}{\tilde{\mathbf{n}}}. \tag{11}$$

The *target distortion* $\mathbf{x}^{(\mathrm{dis})}$ is the artifact introduced by the predictive model: it contains target cues that were mistaken for corruption by the model, and consequently distorted. The residual corruption $\tilde{\mathbf{n}}$ is what remains of the interference (e.g. noise or reverberation) after being processed by the model. There is behind this decomposition an underlying orthogonality assumption between $\mathbf{x}$ and $\mathbf{n}$, which implies orthogonality between $\mathbf{x}^{(\mathrm{dis})}$ and $\tilde{\mathbf{n}}$ [53].

A diffusion-based generative model G_ϕ is then used to learn the distribution of the *ideal residue* $\mathbf{r}_x = \mathbf{x} - D_\theta(\mathbf{y})$, starting from the *noisy residue* $\mathbf{r}_\mathbf{y} = \mathbf{y} - D_\theta(\mathbf{y})$. Finally, the ideal residue estimate is added to the predictor estimate:

$$\hat{\mathbf{x}} = D_\theta(\mathbf{y}) + \overset{\text{残差}}{\hat{\mathbf{r}}_\mathbf{x}} \tag{12}$$

$$= D_\theta(\mathbf{y}) + \overset{\text{生成模型}}{G_\phi}(\mathbf{y} - D_\theta(\mathbf{y})) \tag{13}$$

Results in [24], [37] seem to indicate that this stochastic refinement approach performs as expected, outperforming the initial predictor on perceptual metrics and the pure generative approach with fewer diffusion steps. However, we argue that learning the residual is suboptimal as the residual data distribution $p(\mathbf{r}_x)$ does not have a structure like the target data distribution $p(\mathbf{x})$. Indeed, using (11), one can rewrite $\mathbf{r}_x$ as:

$$\mathbf{r}_\mathbf{x} = \mathbf{x} - D_\theta(\mathbf{y})$$

$$= \mathbf{x}^{(\mathrm{dis})} - \tilde{\mathbf{n}}, \tag{14}$$

残差分布高度依赖模型选择

and notice that the distribution of $\mathbf{r_0}$ highly depends on the choice of the predictive model as well as on the task, which does not assure a structured distribution in the general case. We show examples in Fig. 3 of residuals generated by predictive models for speech enhancement and dereverberation, which confirm this observation. For dereverberation (similarly for deblurring, shown in [24]), the residue has an overall structure somewhat similar to the target, because of the convolutional corruption model. However, the formants structure is severely degraded. For denoising, the residue has no clear structure (compared to e.g. clean speech) and cannot be easily estimated by a generative process. In Whang et al. [24], it is shown that the residual distribution has lower entropy per pixel than the original distribution, which makes learning the residual easier. This is also true for our denoising and dereverberation tasks. However the pointwise entropy of a distribution relates to the quantity of information that needs to

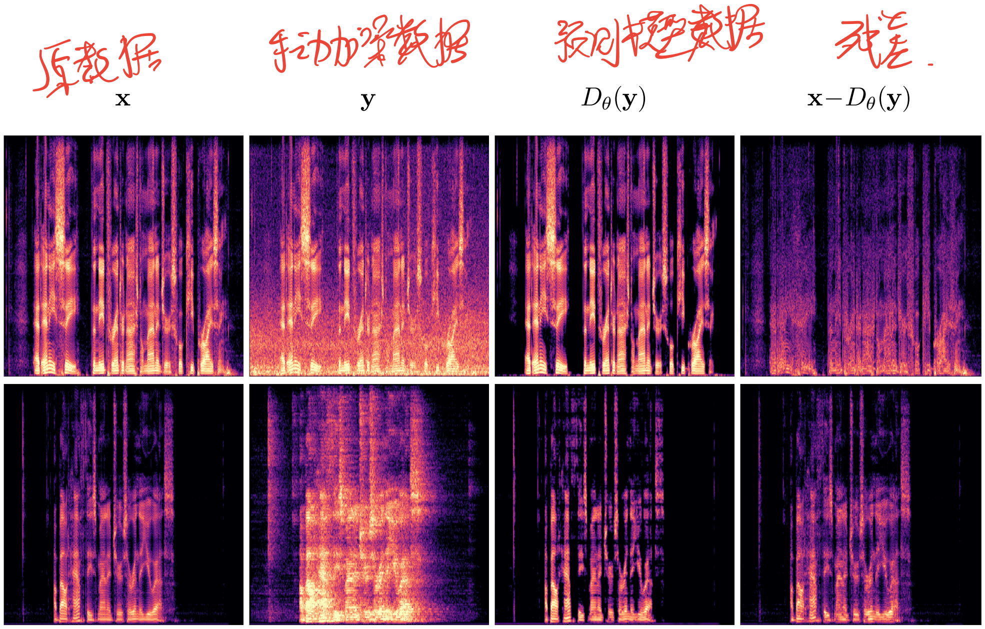


Fig. 3: *Log-energy spectrograms of clean, noisy, processed and residual utterances for denoising (top) and dereverberation (bottom). The predictor used is NCSN++M .*

be learnt by the model and does not capture global structures in the data which can actually help facilitate training.

Most importantly, when rewriting the noisy residue $\mathbf{r_y}$ as:

$$\begin{aligned}\mathbf{r_y} &= \mathbf{y} - D_\theta(\mathbf{y}) \\ &= \mathbf{x}^{(\text{dis})} + \mathbf{n} - \tilde{\mathbf{n}}, \end{aligned} \tag{15}$$

若逆过程开始就学习残差 会导致SNR非常低 由于 x(dis) 幅度 小于ñ

one notices that the resulting *a priori* SNR of the starting point of the reverse process (without accounting for the added Gaussian noise) is very low, as $||\mathbf{x}^{(\text{dis})}||, ||\tilde{\mathbf{n}}|| \ll ||\mathbf{n}||$ for low-enough SNRs and good-enough initial predictor. This makes learning extremely difficult, and we therefore propose to use a different refinement process called *stochastic regeneration*, explained hereafter.

随机再生

C. Stochastic regeneration

For *stochastic regeneration* we propose to cascade the predictive model D_θ and the generative diffusion model G_ϕ. The generative model then learns to *regenerate* the clean speech based on the distorted version provided by the predictive approach. This is conceptually different from the stochastic refinement approach, where the target cues exist in in the residual (but are very hard to access given the amount of noise present in the noisy residue $\mathbf{r}$) and need to be *refined* by the diffusion model.

The task of the diffusion model is then to guide generation of the clean speech $\mathbf{x_0}$ given the first estimate $D_\theta(\mathbf{y})$. If we look at the decomposition in (11), we simply have to remove the residual noise $\tilde{\mathbf{n}}$ and restore the distorted target cues $\mathbf{x}^{(\mathbf{dis})}$. The resulting *a priori* SNR in the starting point (again without considering the added Gaussian noise) is very high, as for a reasonable predictor $||\mathbf{x}^{(\text{dis})}||, ||\tilde{\mathbf{n}}|| \ll ||\mathbf{n}||$. The estimate is then obtained as:

将结果作为输入传入扩散模型.

$$\hat{\mathbf{x}} = G_\phi(D_\theta(\mathbf{y})) \tag{16}$$

A visualization of the inference process is shown in Fig. 4. We name the resulting **Sto**chastic **R**egeneration **M**odel *StoRM*.

For training, we use a criterion $\mathcal{J}^{(\text{StoRM})}$ combining denoising score matching $\mathcal{J}^{(\text{DSM})}$ in (8) and a supervised regularization term $\mathcal{J}^{(\text{Sup})}$—e.g. mean square error—matching the output of the initial predictor to the target speech:

去噪分数匹配　监督正则化项

$$\begin{aligned}\mathcal{J}^{(\text{StoRM})}(\theta,\phi) &= \mathcal{J}^{(\text{DSM})}(\theta) + \alpha \mathcal{J}^{(\text{Sup})}(\phi) \\ &= \mathbb{E}_{\tau,(\mathbf{x_0},\mathbf{y}),\mathbf{z},\mathbf{x}_\tau}\left[\left\|\mathbf{s}_\phi(\mathbf{x}_\tau,\mathbf{y},\tau) + \frac{\mathbf{z}}{\sigma(\tau)}\right\|_2^2\right] \\ &\quad + \alpha \mathbb{E}_{(\mathbf{x_0},\mathbf{y})}\left[||\mathbf{x_0} - D_\theta(\mathbf{y})||_2^2\right], \end{aligned} \tag{17}$$

正则化系数

where α is a balance term that we empirically set to 1.

One may object that the estimate $D_\theta(\mathbf{y})$ is not a sufficient statistic for the model to reconstruct target cues. However, by their very learning principle, generative models are able to create data based on the clean examples seen during training, hence our choice of the terminology *stochastic regeneration*. Stochastic regeneration is still a generative model with respect to the definition given in introduction, as it is able to output realistic samples belonging to a posterior distribution. However this posterior distribution is not $p(\mathbf{x}|\mathbf{y})$ anymore, but $p(\mathbf{x}|D_\theta(\mathbf{y}))$, the posterior distribution of target speech conditioned on the initial prediction.

IV. Experimental Setup

A. Data

a) Speech Enhancement:

- The WSJ0+Chime dataset is generated using clean speech extracts from the Wall Street Journal corpus [41] and noise signals from the CHiME3 dataset [54]. The mixture signal is created by randomly selecting a noise file and adding it to a clean utterance with a SNR sampled uniformly between -6 and 14dB.
- The TIMIT+Chime dataset is similarly generated as WSJ0+Chime, simply replacing the clean speech with clean utterances from the TIMIT corpus [42]. We use this dataset for ASR as oracle annotations are available for word error rate (WER) evaluation.
- The VoiceBank/DEMAND dataset is a classical benchmark dataset for speech enhancement using clean speech from the

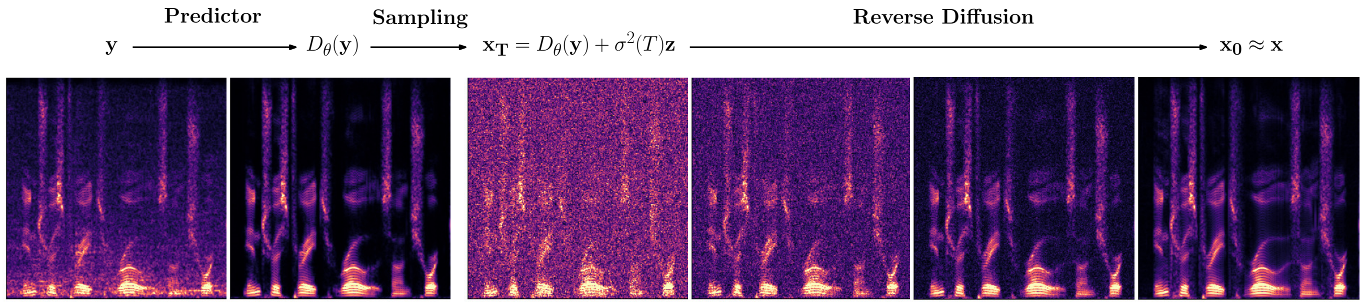


Fig. 4: *Stochastic regeneration inference process. The predictive network is first used to generate a denoised version $D_\theta(\mathbf{y})$. Diffusion-based generation G_ϕ is then performed by adding Gaussian noise $\sigma(T)^2\mathbf{z}$ to obtain the start sample $\mathbf{x_T}$ and solving the reverse diffusion SDE* (10)*, yielding a sample from the estimated posterior $\mathbf{x_0} \sim p(\mathbf{x}|D_\theta(\mathbf{y}))$.*

VCTK corpus [43] excluding two speakers. The utterances are corrupted by recorded noise from the DEMAND database [55] and two artificial noise types (babble and speech shaped) at SNRs of 0, 5, 10, and 15 dB for training and validation. The SNR levels of the test set are 2.5, 7.5, 12.5, and 17.5 dB.

b) Speech Dereverberation: The WSJ0+Reverb dataset is generated using clean speech data from the WSJ0 dataset and convolving each utterance with a simulated room impulse response (RIR). We use the `pyroomacoustics` engine [56] to simulate the RIRs. The reverberant room is modeled by sampling uniformly a target T_{60} between 0.4 and 1.0 seconds and a room length, width and height in [5,15]×[5,15]×[2,6] m. This results in an average direct to reverberant ratio (DRR) of around -9 dB and average *measured* T_{60} of 0.91 s. A dry auralized version of the room is generated as the reference clean speech using the same geometric parameters with a fixed absorption coefficient of 0.99, to generate the corresponding anechoic target.

B. Hyperparameters and training configuration

用短时傅里叶变换表示语音数据.

a) Data representation: Utterances are transformed using a short-time Fourier transform (STFT) with a window size of 510, a hop length of 128 and a square-root Hann window. A square-root magnitude warping is used to compress the dynamical range of the input spectrograms [12]. For training, sequences of 256 STFT frames (i.e. 2s) are randomly extracted from the full-length utterances and normalized before being fed to the network.

b) Forward and reverse diffusion: For all diffusion models, similar values are chosen to parameterize the forward and reverse stochastic processes. The stiffness parameter is fixed to $\gamma = 1.5$, the extremal noise levels to $\sigma_{\min} = 0.05$ and $\sigma_{\max} = 0.5$, and the minimal diffusion time to $\tau_\epsilon = 0.03$ as in [12]. Unless stated otherwise (that is, for all results except those in Figure 5), $N = 50$ time steps are used for reverse diffusion and we adopt the predictor-corrector scheme [44] with one step of annealed Langevin dynamics correction and a step size of $r = 0.5$.

c) Baselines: For comparison on WSJ0-based datasets, we compare our proposed approach StoRM to the purely generative SGMSE+ [12] and purely predictive NCSN++M. We also report results using GaGNet [57], a state-of-the-art predictive approach using parallel magnitude- and complex-domain processing in the T-F domain. We complement the benchmark on the Voicebank/DEMAND dataset with the predictive ConvTasNet [8] and MetricGAN+ [58], as well as the generative unsupervised dynamical VAE (DVAE) [59], conditional time-domain diffusion model CDiffuse [29], stochastic refinement time-domain enhancement scheme SRTNet [37] and original SGMSE [28].

d) Network architecture: The backbone architecture we use is a lighter configuration of the NCSN++ architecture variant proposed in [44], which was used in our previous study [11] and that we denote here as *NCSN++M*. The modifications brought to the network are that the attention layer in the bottleneck is removed, the number of layers in the encoder-decoder structure is decreased from 7 to 4, and only one ResNet block is used per encoder-decoder layer instead of two. This results in a network capacity of roughly 27.8M parameters instead of 65M, without significant degradation of the speech enhancement performance, be it for predictive or generative modelling.

When this NCSN++M configuration is used for score estimation in SGMSE+, the noisy speech spectrogram $\mathbf{y}$ and the current diffusion process estimate $\mathbf{x}_\tau$ real and imaginary channels are stacked and fed to the network as input, and the current noise level $\sigma(\tau)$ is provided as a conditioner. For our proposed approach StoRM, the initial prediction $D_\theta(\mathbf{y})$ is also stacked together with $\mathbf{y}$ and $\mathbf{x}_\tau$: the influence of this double conditioning is examined in an ablation study in Section V-F. For the predictive approach, denoted directly as NCSN++M, the noise-conditioning layers are removed and only the noisy speech spectrogram real and imaginary channels are used. This ablation removes only 1.8% of the original number of parameters, which hardly modifies the network capacity.

We also use ConvTasNet [8] and GaGNet [57] as alternative initial predictors for StoRM. We train using NCSN++M as the initial predictor and swap it during inference with one of the two networks mentioned above, in order to test the robustness of our proposed stochastic regeneration approach towards unseen predictors.

e) Training configuration: We use the Adam optimizer [60] with a learning rate of 10^{-4} and an effective batch size of 16. We track an exponential moving average of the DNN weights with a decay of 0.999 to be used for sampling, as it showed to be very effective [61]. We train DNNs for a maximum of 1000 epochs using early stopping based on the validation loss with a patience of 10 epochs. All models converged before reaching the maximum number of epochs. The generative approach is trained with the denoising score matching criterion (8), and the predictive methods use a simple mean-square error loss on the complex spectrogram. The stochastic regeneration approach uses the combined criterion in (17). The default training strategy is that we pre-train the initial predictor with a simple mean-square error loss, then jointly train the

TABLE I: *Denoising results obtained on WSJ0+Chime. Values indicate mean and standard deviation. All approaches use NCSN++M as backbone architecture. Diffusion models use $N = 50$ steps for reverse diffusion.*

Method	WV-MOS	PESQ	ESTOI	SI-SDR	SI-SIR	SI-SAR
Mixture	1.43 ± 0.66	1.38 ± 0.32	0.65 ± 0.18	4.3 ± 5.8	4.3 ± 5.8	-
SGMSE+	3.63 ± 0.38	2.33 ± 0.61	0.86 ± 0.10	13.3 ± 5.0	27.4 ± 6.3	13.5 ± 4.9
NCSN++M	3.47 ± 0.53	2.21 ± 0.65	$\mathbf{0.89 \pm 0.09}$	$\mathbf{16.4} \pm \mathbf{4.4}$	31.1 ± 5.0	$\mathbf{16.6} \pm \mathbf{4.4}$
GaGNet	3.34 ± 0.54	2.19 ± 0.61	0.87 ± 0.09	15.7 ± 4.3	27.6 ± 4.7	16.0 ± 4.4
StoRM	$\mathbf{3.72 \pm 0.40}$	$\mathbf{2.58 \pm 0.61}$	$\mathbf{0.88} \pm 0.08$	15.1 ± 4.2	$\mathbf{31.6 \pm 5.0}$	15.3 ± 4.2

predictor and score networks with (17). Different training strategies are examined in an ablation study in Section V-F.

C. Evaluation metrics

评价标准：

For instrumental evaluation of the speech enhancement and dereverberation performance with clean test data available, we use intrusive measures such as Perceptual Evaluation of Speech Quality (PESQ) [62] to assess speech quality, extended short-term objective intelligibility (ESTOI) [63] for intelligibility and scale-invariant signal to distortion ratio (SI-SDR), scale-invariant signal to interference ratio (SI-SIR) and scale-invariant signal to artifacts ratio (SI-SAR) [64] for noise removal. As in [11], we complement our metrics benchmark with WV-MOS [65][2], which is a DNN-based mean opinion score (MOS) estimation, and was used by the authors for reference-free assessment of bandwidth extension or speech enhancement performance.

We also evaluate our proposed approach on ASR, using NVidia's temporal convolutional network QuartzNet [3] [66] as the speech recognition model, and classical WER dynamic programming evaluation with the `jiwer` Python library[4]. We use the pretrained `Base-en` 18.9M parameters version of QuartzNet for specialized English speech recognition.

Finally, we organize a medium-scale MUSHRA listening test with 9 participants. We ask the participants to rate 10 samples with a single number representing overall quality, including speech distortion, residual distortions and potential artifacts. We use the `webMUSHRA`[5] tool with `pymushra`[6] server management. The samples are randomly extracted from the WSJ0+Chime and WSJ0+Reverb test sets, ensuring gender and task balance as well as speaker exclusivity (within a given task, a speaker is used once at most). The approaches evaluated are the predictive NCSN++M, score-based generative model SGMSE+ and our proposed approach StoRM. The noisy mixture is given as a low anchor, and a supplementary anchor is created by increasing the input SNR by 10dB in comparison to the noisy mixture.

V. Experimental Results and Discussion

A. Comparison to baselines

a) WSJ0+Chime and WSJ0+Reverb: In tables I and II, we show results of the proposed stochastic regeneration StoRM approach as compared to purely predictive GaGNet and NCSN++M and purely generative SGMSE+, for denoising on WSJ0+Chime and dereverberation on WSJ0+Reverb respectively.

We confirm the results from [11], which is that predictive NCSN++M and GaGNet provide samples with good interference removal (high SI-SDR) and intelligibility (high ESTOI) but lower quality (lower PESQ and WV-MOS) compared to diffusion-based generative SGMSE+. This gap is stronger for dereverberation than for denoising as already observed, since the average input SNR for dereverberation is much lower than for denoising. Also, the reverberation interference being a filtered version of the target speech, the predictive method cannot suppress reverbreation without introducing significant distortion, which is particularly visible in NCSN++M and GaGNet results. The generative SGMSE+, however, is able to extract the speech cues and directly reconstructs it without any trace of reverberation.

We observe that our proposed StoRM associates the best of both the predictive and generative worlds, by producing samples with very high quality like generative SGMSE+, while being approximately as good with interference removal as the predictive NCSN++M. Again, the observed gap is more significant for dereverberation, where the proposed StoRM outperforms both SGMSE+ and NCSN++M on all metrics. Example spectrograms are displayed on Figure 6 and 7, for denoising and dereverberation respectively.

b) VoiceBank/DEMAND: We report in Table III results of our StoRM configuration against various state-of-the-art speech enhancement baselines on the VoiceBank/DEMAND benchmark. The SNRs in Voicebank/DEMAND are always positive and distributed around 10dB, which is not very challenging compared to the conditions in our WSJ0+Chime dataset. Consequently, the gap between SGMSE+ and StoRM on Voicebank/DEMAND is not as large as on WSJ0+Chime, which shows that using the initial predictor is particularly useful in difficult conditions. In easier environments such as that simulated in Voicebank/DEMAND, diffusion-based generative modelling can take the noisy mixture as the initial condition for reverse diffusion without being further guided. Still, our proposed method StoRM still slightly outperforms the purely generative SGMSE+ on ESTOI and SI-SDR, setting a new state-of-the-art record for generative models on this benchmark. As a sidenote, while the best overall PESQ scores are obtained by MetricGAN+, it should be stressed that this approach is designed to maximize PESQ, and that the superior performance indicated by PESQ is not supported by informal and formal listening [12].

B. Efficient sampling

We report in Figure 5 the performance of the SGMSE+ and StoRM schemes as a function of the number of steps used for

[2] https://github.com/AndreevP/wvmos

[3] https://catalog.ngc.nvidia.com/orgs/nvidia/models/quartznet15x5

[4] https://github.com/jitsi/jiwer

[5] https://github.com/audiolabs/webMUSHRA

[6] https://github.com/nils-werner/pymushra

TABLE II: *Dereverberation results obtained on Reverb-WSJ0. Values indicate mean and standard deviation. All approaches use NCSN++M as backbone architecture. Diffusion models use $N = 50$ steps for reverse diffusion.*

Method	WV-MOS	PESQ	ESTOI	SI-SDR	SI-SIR	SI-SAR
Mixture	1.78 ± 0.99	1.36 ± 0.19	0.46 ± 0.12	-7.3 ± 5.5	-7.5 ± 5.4	-
SGMSE+	3.49 ± 0.39	2.66 ± 0.45	0.85 ± 0.06	2.4 ± 7.2	11.6 ± 9.9	2.8 ± 6.8
NCSN++M	2.99 ± 0.38	2.08 ± 0.47	0.85 ± 0.06	6.1 ± 3.8	21.4 ± 7.0	6.1 ± 3.7
GaGNet	2.40 ± 0.52	1.59 ± 0.37	0.68 ± 0.09	-0.5 ± 4.8	7.7 ± 4.0	0.2 ± 5.1
StoRM	**3.73 ± 0.32**	**2.83 ± 0.42**	**0.88 ± 0.04**	**6.5 ± 4.0**	**22.9 ± 8.2**	**6.5 ± 3.9**

步数与评价指标

每秒MAC操作数量

实线为加入退火朗之万动力学校正器，对SGMSE+有明显改善，对StoRM却一般，还会增加计算复杂度

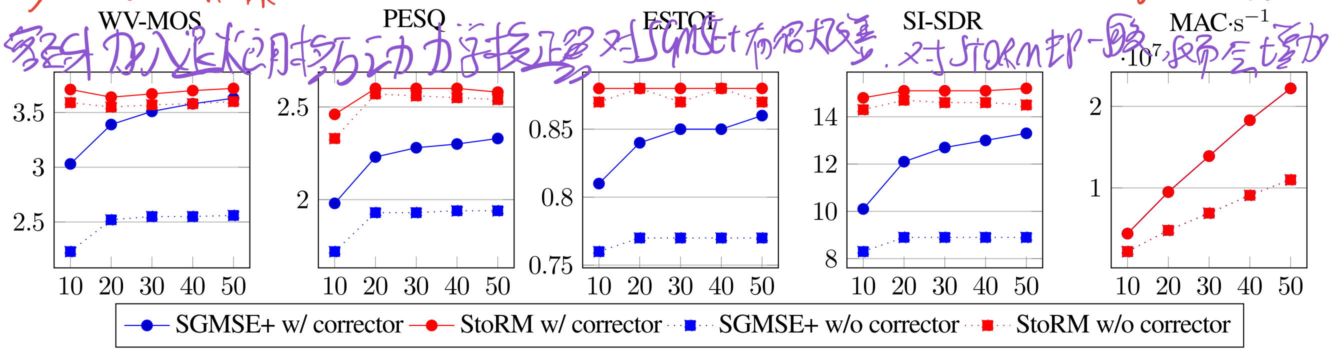


Fig. 5: *Results for speech denoising on WSJ0+Chime as a function of the number of reverse diffusion steps N. All approaches use the same NCSN++M architecture. The corrector uses one step of Annealed Langevin Dynamics with a step size of $r = 0.5$. The number of MAC operations per second is virtually the same for SGMSE+ and StoRM, therefore only one curve is visible here.*

TABLE III: *Speech enhancement results obtained on VoiceBank/DEMAND. Here SGMSE+ uses the large NCSN++L architecture and $N = 30$ steps as in [12].Our StoRM approach uses the lighter NCSN++M architecture and $N = 30$ steps.* P *means predictive and* G *generative.* * *means that figures are directly reported from the associated paper.*

Method	Type	PESQ	ESTOI	SI-SDR
Mixture		1.97	0.79	8.4
NCSN++M	P	2.82	**0.88**	**19.9**
Conv-TasNet [8]	P	2.84	0.85	19.1
MetricGAN+ [58]	P	**3.13**	0.83	8.5
GaGNet [57]	P	2.94	0.86	**19.9**
DVAE [59]	G	2.43	0.79	16.4
CDiffuSE [29]	G	2.46	0.79	12.6
SRTNet* [37]	G	2.69	-	-
SGMSE [28]	G	2.28	0.80	16.2
SGMSE+ [12]	G	**2.93**	0.87	17.3
StoRM (ours)	G	**2.93**	**0.88**	**18.8**

reverse diffusion. We additionally provide an estimation of the number of multiply-accumulate (MAC) operations per second as measured by the `python-papi` package.

We observe that StoRM is able to maintain performance at a near-optimal level even using only 10 steps, using the initial predictive estimate as a reasonable guess for further diffusion. In comparison, SGMSE+ performance degrades rapidly as the number of steps decreases. Furthermore, StoRM is able to produce very high-quality samples without even needing the Annealed Langevin Dynamics corrector during sampling, whereas SGMSE+ performance dramatically degrades without this corrector. Since each corrector step makes an additional call to the score network, avoiding its use further relaxes the computational complexity. This demonstrates the flexibility of StoRM with respect to the compromise between fast inference and high sample quality. In the end, using StoRM with 20 steps and no corrector produces near-optimal sample quality at a cost of 4.5M MAC·s^{-1}, versus 23M MAC·s^{-1} for the optimal SGMSE+ setting (50 steps and Annealed Langevin Dynamics correction). Furthermore, StoRM still outperforms the optimal SGMSE+ setting using 10 steps and no corrector, thus reducing computational complexity by a full order of magnitude.

TABLE IV: *Denoising results on WSJ0+Chime for StoRM using matched and mismatched initial predictors. The predictor architecture used for training is NCSN++M. All approaches use NCSN++M as score network architecture and $N = 50$ steps for reverse diffusion. Values indicate mean and standard deviation.*

Initial Predictor	Matched	PESQ	ESTOI	SI-SDR
Mixture	-	1.38 ± 0.32	0.65 ± 0.18	4.3 ± 5.8
NCSN++M	✓	**2.53 ± 0.63**	**0.88 ± 0.09**	**14.7 ± 4.3**
GaGNet [57]	✗	2.52 ± 0.62	0.87 ± 0.09	14.7 ± 4.1
ConvTasNet [8]	✗	2.36 ± 0.60	0.86 ± 0.09	9.9 ± 1.7

C. Generalization to mismatched predictors

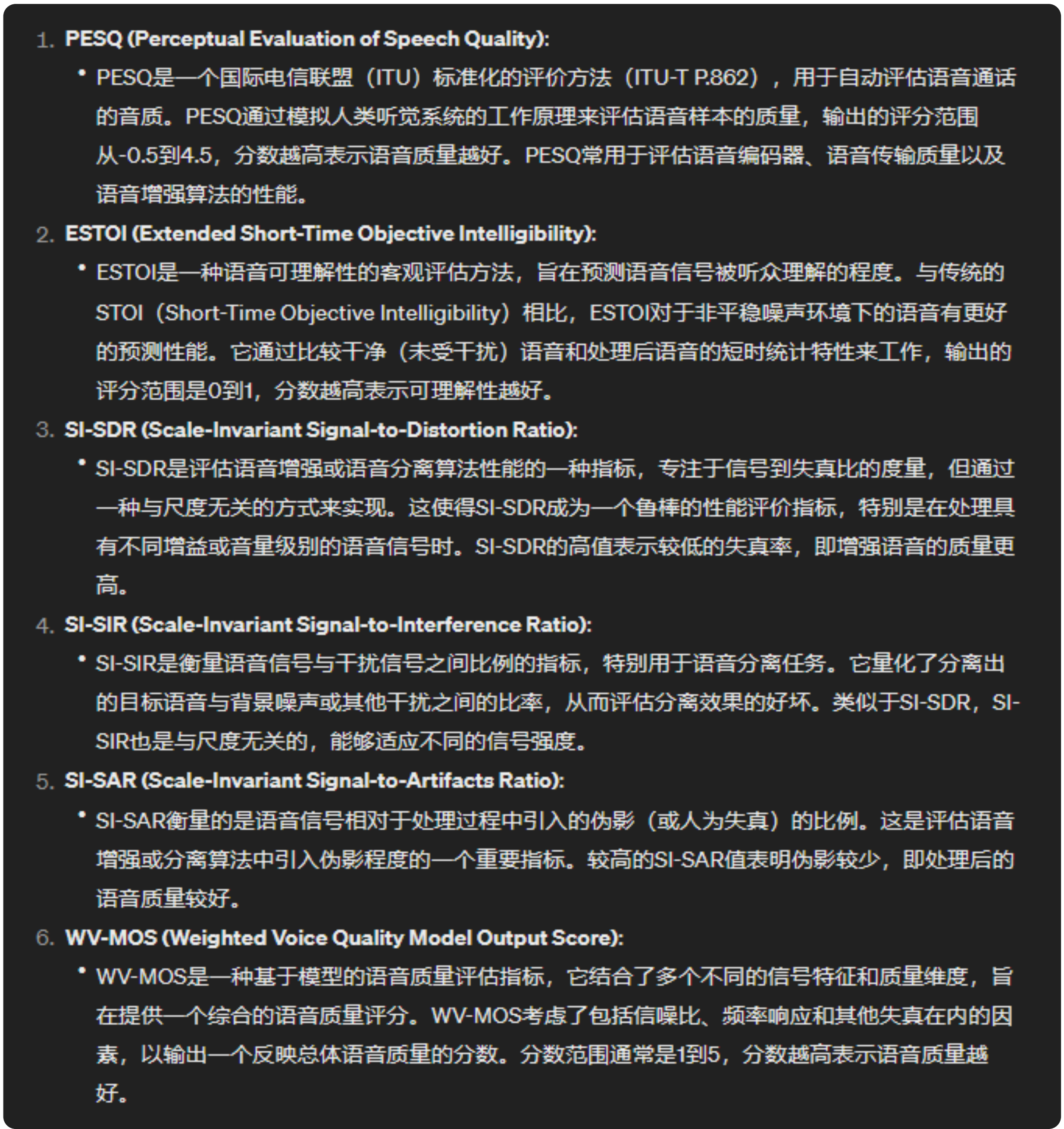
In Table IV, we report results for StoRM using different initial predictors than the one used during training. The approach is trained using the NCSN++M as initial predictor as before, and we test using ConvTasNet [8] and GaGNet [57] as alternative initial predictors by exchanging this predictor during sampling. GaGNet also processes speech in the T-F domain, therefore the artifacts are of a similar nature than those of NCSN++M. StoRM is entirely robust to such a slight mismatch, as indicated by the equivalent performance of using NCSN++M and GaGNet as the initial predictor. ConvTasNet is a time-domain method using a fully learnt encoder: the speech

SGMSE：基于得分的生成模型，将模型扩展到复短时傅里叶变换域（STFT），训练目标采用随机微分方程（SDEs）

NCSN++M：NCSN++. 先生成低质图像，再进行扩展（预测模型）

NCSN++M
使用扩散模型，将应用任务扩展到
降噪、去混响、带宽扩展。

衡量指标：



1. **PESQ (Perceptual Evaluation of Speech Quality):**
   - PESQ是一个国际电信联盟（ITU）标准化的评价方法（ITU-T P.862），用于自动评估语音通话的音质。PESQ通过模拟人类听觉系统的工作原理来评估语音样本的质量，输出的评分范围从-0.5到4.5，分数越高表示语音质量越好。PESQ常用于评估语音编码器、语音传输质量以及语音增强算法的性能。
2. **ESTOI (Extended Short-Time Objective Intelligibility):**
   - ESTOI是一种语音可理解性的客观评估方法，旨在预测语音信号被听众理解的程度。与传统的STOI（Short-Time Objective Intelligibility）相比，ESTOI对于非平稳噪声环境下的语音有更好的预测性能。它通过比较干净（未受干扰）语音和处理后语音的短时统计特性来工作，输出的评分范围是0到1，分数越高表示可理解性越好。
3. **SI-SDR (Scale-Invariant Signal-to-Distortion Ratio):**
   - SI-SDR是评估语音增强或语音分离算法性能的一种指标，专注于信号到失真比的度量，但通过一种与尺度无关的方式来实现。这使得SI-SDR成为一个鲁棒的性能评价指标，特别是在处理具有不同增益或音量级别的语音信号时。SI-SDR的高值表示较低的失真率，即增强语音的质量更高。
4. **SI-SIR (Scale-Invariant Signal-to-Interference Ratio):**
   - SI-SIR是衡量语音信号与干扰信号之间比例的指标，特别用于语音分离任务。它量化了分离出的目标语音与背景噪声或其他干扰之间的比率，从而评估分离效果的好坏。类似于SI-SDR，SI-SIR也是与尺度无关的，能够适应不同的信号强度。
5. **SI-SAR (Scale-Invariant Signal-to-Artifacts Ratio):**
   - SI-SAR衡量的是语音信号相对于处理过程中引入的伪影（或人为失真）的比例。这是评估语音增强或分离算法中引入伪影程度的一个重要指标。较高的SI-SAR值表明伪影较少，即处理后的语音质量较好。
6. **WV-MOS (Weighted Voice Quality Model Output Score):**
   - WV-MOS是一种基于模型的语音质量评估指标，它结合了多个不同的信号特征和质量维度，旨在提供一个综合的语音质量评分。WV-MOS考虑了包括信噪比、频率响应和其他失真在内的因素，以输出一个反映总体语音质量的分数。分数范围通常是1到5，分数越高表示语音质量越好。

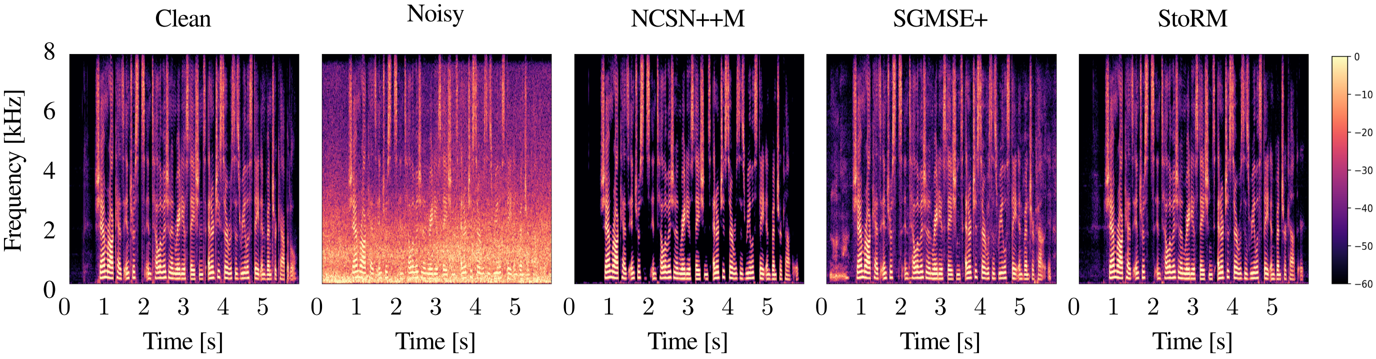


Fig. 6: *Log-energy spectrograms of clean, noisy and processed utterances from the WS0+Chime dataset. Input SNR is -0.9* dB*. Vocalizing artifacts are visible at the beginning of the utterance in the SGMSE+ sample. Severe speech distortions are observed all across the NCSN++M sample. StoRM corrects these distortions without introducing vocalizing artifacts, thus yielding both high quality and intelligibility.*

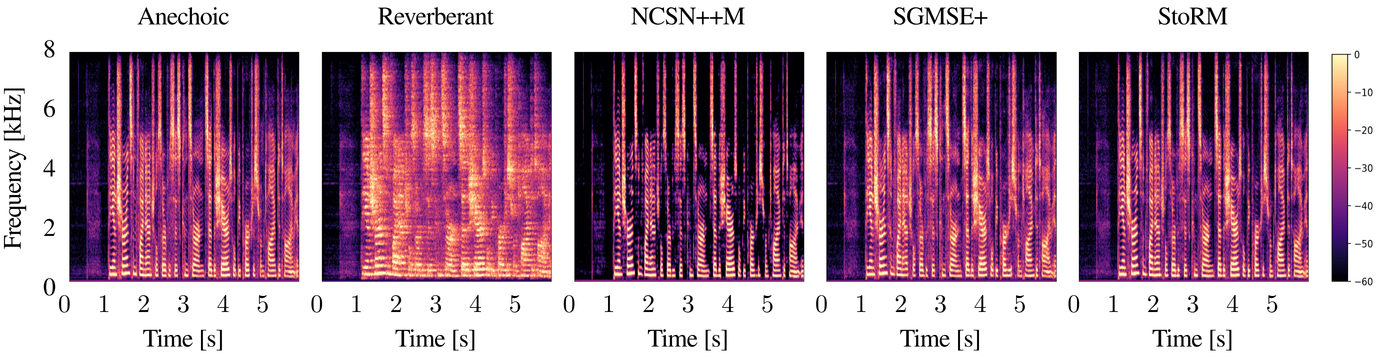


Fig. 7: *Log-energy spectrograms of anechoic, reverberant and processed utterances from the WS0+Reverb dataset. Input T_{60} is 1.06* s*. Formant structure is partly destroyed by SGMSE+ sample around the 3* s *tag, and severe speech distortions are observed all across the NCSN++M sample. StoRM corrects the distortions and reproduces the formant structure without residual reverberation.*

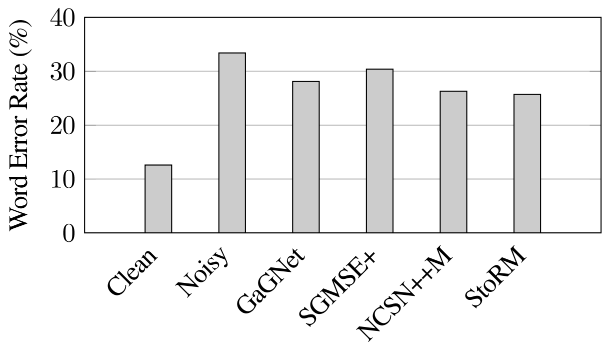


Fig. 8: *ASR results for speech enhancement on TIMIT+Chime. SGMSE+ and StoRM use NCSN++M as the backbone architecture. Diffusion models use $N = 50$ steps for reverse diffusion.*

distortions are thus quite different than those of NCSN++M or GaGNet. Additionally, ConvTasNet's original performance is slightly worse than its two counterparts. Consequently, we observe that the performance of StoRM using ConvTasNet as the initial predictor is poorer but still very relatable to that of using NCSN++M as the predictor. This demonstrates relative robustness to unseen conditions provided by the generative modelling stage.

D. ASR results

In Figure 8, we compare the predictive, generative and stochastic regeneration approaches on speech enhancement for ASR using the TIMIT+Chime dataset. We observe that SGMSE+ results in poorer speech recognition abilities than its predictive GaGNet and NCSN++M counterparts, and hardly improves the ASR performance over the noisy mixture. This can be explained by the previously mentioned undesired vocalizing artifacts and phonetic confusions, which are created by the generative approach under uncertainty over the presence and phonetic nature of speech respectively and are heavily punished by WER evaluation. Using the predictive estimate as a guide for generation, StoRM improves the WER performance by a relative factor of 19% as compared to SGMSE+, even slightly outperforming NCSN++M, which shows that most of the artifacts and confusions are corrected.

E. Listening experiment

We show in the boxplot on Figure 9 the results of our MUSHRA listening test. On average, the participants clearly rated the proposed StoRM higher than the purely predictive NCSN++M and purely generative SGMSE+. This confirms the results provided by the intrusive and non-intrusive metrics provided in tables I and II. Participants rated NCSN++M slightly better than SGMSE+ on average, which is linked to the rating criterion described in Section IV-C. This seems to indicate that participants put more weight on ”residual distortions” and ”potential artifacts” (vocalizing/breathing/confusions) than on ”speech distortion”.

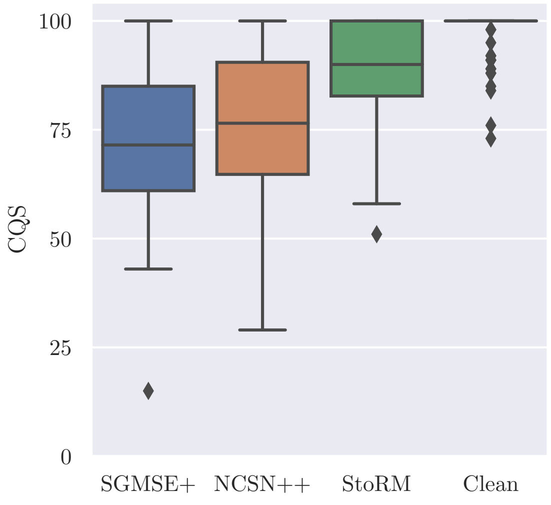


Fig. 9: *Listening test results. CQS is the "continuous quality scale" on which participants are asked to rate. Inner line represents the median. 9 participants rated 10 samples randomly selected from WSJ0+Chime and WSJ0+Reverb.*

TABLE V: *Denoising results on WSJ0+Chime for StoRM using different conditioning inputs for the score network. Values indicate mean and standard devation. All approaches use NCSN++M as backbone architecture and $N = 50$ steps for reverse diffusion.*

Conditioning	PESQ	ESTOI	SI-SDR
Noisy	2.30 ± 0.60	0.84 ± 0.10	11.5 ± 5.2
PostDenoiser	2.50 ± 0.62	0.87 ± 0.09	14.7 ± 4.3
Both	$\mathbf{2.53} \pm \mathbf{0.63}$	$\mathbf{0.88} \pm \mathbf{0.08}$	$\mathbf{15.1} \pm \mathbf{4.2}$

F. Ablation studies

We conduct ablation studies on the WSJ0+Chime dataset, to observe the respective influence of score network conditioning on the one hand and the training strategy on the other hand.

a) Conditioning of the score network: In Table V, we report instrumental results when using different conditioning inputs for the score network used in the proposed StoRM. We input either the noisy speech $\mathbf{y}$ ("Noisy"), the denoised estimate $D_\theta(\mathbf{y})$ ("PostDenoiser"), or both ("Both", which is the default setting for StoRM). Using only the noisy speech ("Noisy") is detrimental to the performance. It seems that the score network does need the information from the original distortions in $D_\theta(\mathbf{y})$ at time step $\tau{=}T$, to properly learn the score at time step $\tau{<}T$. This mismatch at the first denoising steps is detrimental to performance. We also observe that instrumental metrics tend to slightly favor the "Both" conditioning over the "PostDenoiser" conditioning.

b) Training strategies: We show in Table VI the results of StoRM using different training strategies. We see that jointly training the initial predictor and the score network slightly improves results for denoising. However, training the initial predictor from scratch or having it pre-trained first does not seem to make a difference, as long as one regularizes the training criterion with the supervised criterion $\mathcal{J}^{(\text{Sup})}$ which matches the output of the initial predictor to the target. Indeed, as shown in the third line of Table VI, if we use a randomly initialized predictor and train both the predictor and score networks only with the score matching criterion $\mathcal{J}^{(\text{DSM})}$— i.e. setting α in (17) to 0,—the performance dramatically drops. This is to be expected since the learning task then becomes much more complicated given the size of the search space and the lack of regularization. The proposed combination of joint training and regularization with $\mathcal{J}^{(\text{Sup})}$ performs most favorably. This indicates that it is best to train the predictor to output something resembling clean speech rather than arbitrary learned encoder features, while still leaving some room for the predictor to adapt its output to the score model.

TABLE VI: *Denoising results on WSJ0+Chime for StoRM using different training strategies for the score network. All approaches use NCSN++M as backbone architecture and $N = 50$ steps for reverse diffusion. Standard deviation is omitted for easier reading.*

Pre-train D_θ	Fine-tune D_θ	Use $\mathcal{J}^{(\text{Sup})}$	PESQ	ESTOI	SI-SDR
✗	✓	✓	**2.58**	**0.88**	**15.1**
✓	✗	✗	2.53	**0.88**	14.7
✗	✓	✗	1.11	0.62	-0.3
✓	✓	✓	**2.58**	**0.88**	**15.1**

VI. Conclusion

We presented a generative stochastic regeneration scheme combining a predictive model as initial predictor and a diffusion-based generative approach regenerating the target cues distorted by the first stage. On the one hand, the approach improves sample quality compared to pure predictive approaches as it leverages generative modelling to output samples that have high probability on the target posterior distribution manifold, rather than regressing to their mean. On the other hand, it uses predictive power to provide a good initial prediction of the target sample, which avoids typical generative artifacts such as vocalizing and breathing effects, and increases the interference removal performance, especially in difficult environments. Intrusive and reference-free instrumental metrics as well as formal listening tests confirmed the superiority of the stochastic regeneration approach over the baselines. The resulting approach allows efficient sampling, requiring fewer steps and avoiding the use of Annealed Langevin Dynamics correction during reverse diffusion, thus reducing computational complexity by an order of magnitude without sacrificing quality, compared to the original diffusion model.

References

[1] P. A. Naylor and N. D. Gaubitch, *Speech Dereverberation.* Noise Control Engineering Journal, 2011, vol. 59.

[2] S. J. Godsill, P. J. W. Rayner, and O. Cappé, *Digital audio restoration*. Springer, 1998.

[3] R. C. Hendriks, T. Gerkmann, and J. Jensen, *DFT-domain based single-microphone noise reduction for speech enhancement: A survey of the state-of-the-art.* Morgan & Claypool, 2013.

[4] T. Gerkmann and E. Vincent, "Spectral masking and filtering," in *Audio Source Separation and Speech Enhancement*, E. Vincent, Ed. John Wiley & Sons, 2018.

[5] D. Wang and J. Chen, "Supervised speech separation based on deep learning: An overview," *IEEE Trans. Audio, Speech, Language Proc.*, 2018.

[6] K. P. Murphy, *Probabilistic Machine Learning: Advanced Topics.* MIT Press, 2023.

[7] D. S. Williamson and D. Wang, "Time-frequency masking in the complex domain for speech dereverberation and denoising," *IEEE/ACM Trans. Audio, Speech, Language Proc.*, vol. 25, no. 7, pp. 1492–1501, 2017.

[8] Y. Luo and N. Mesgarani, "Real-time single-channel dereverberation and separation with time-domain audio separation network," in *ISCA Interspeech*, 2018.

[9] J. Lin, Y. Wang, K. Kalgaonkar, G. Keren, D. Zhang, and C. Fuegen, "A two-stage approach to speech bandwidth extension," in *ISCA Interspeech*, 2021.

[10] K. Han, Y. Wang, D. Wang, W. S. Woods, I. Merks, and T. Zhang, "Learning spectral mapping for speech dereverberation and denoising," *IEEE/ACM Trans. Audio, Speech, Language Proc.*, vol. 23, no. 6, pp. 982–992, 2015.

[11] J.-M. Lemercier, J. Richter, S. Welker, and T. Gerkmann, "Analysing diffusion-based generative approaches versus discriminative approaches for speech restoration," *arXiv preprint*, 2022.

[12] J. Richter, S. Welker, J.-M. Lemercier, B. Lay, and T. Gerkmann, "Speech enhancement and dereverberation with diffusion-based generative models," *arXiv preprint*, 2022.

[13] D. P. Kingma and M. Welling, "Auto-encoding variational Bayes," *Int. Conf. Learning Repr. (ICLR)*, 2014.

[14] H. Fang, G. Carbajal, S. Wermter, and T. Gerkmann, "Variational autoencoder for speech enhancement with a noise-aware encoder," *IEEE Int. Conf. Acoustics, Speech, Signal Proc. (ICASSP)*, 2021.

[15] S. Leglaive, L. Girin, and R. Horaud, "A variance modeling framework based on variational autoencoders for speech enhancement," *IEEE Int. Workshop on Machine Learning for Signal Proc. (MLSP)*, pp. 1–6, 2018.

[16] J. Richter, G. Carbajal, and T. Gerkmann, "Speech enhancement with stochastic temporal convolutional networks." *ISCA Interspeech*, 2020.

[17] D. J. Rezende and S. Mohamed, "Variational inference with normalizing flows," in *Int. Conf. Machine Learning (ICML)*, 2015.

[18] I. Kobyzev, S. J. Prince, and M. A. Brubaker, "Normalizing flows: An introduction and review of current methods," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 43, no. 11, pp. 3964–3979, nov 2021.

[19] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, "Generative adversarial networks," in *Neural Information Processing Systems (NIPS)*, vol. 27, 2014.

[20] J. Kong, J. Kim, and J. Bae, "Hifi-gan: Generative adversarial networks for efficient and high fidelity speech synthesis," in *Neural Information Processing Systems (NIPS)*, 2020.

[21] J. Sohl-Dickstein, E. Weiss, N. Maheswaranathan, and S. Ganguli, "Deep unsupervised learning using nonequilibrium thermodynamics," *Int. Conf. Machine Learning (ICML)*, 2015.

[22] J. Ho, A. Jain, and P. Abbeel, "Denoising diffusion probabilistic models," *Neural Information Processing Systems (NIPS)*, 2020.

[23] Y. Song and S. Ermon, "Generative modeling by estimating gradients of the data distribution," *Neural Information Processing Systems (NIPS)*, vol. 32, 2019.

[24] J. Whang, M. Delbracio, H. Talebi, C. Saharia, A. G. Dimakis, and P. Milanfar, "Deblurring via stochastic refinement," in *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2021.

[25] S. Welker, H. N. Chapman, and T. Gerkmann, "DriftRec: Adapting diffusion models to blind image restoration tasks," *arXiv preprint*, 2022.

[26] P. Dhariwal and A. Nichol, "Diffusion models beat GANs on image synthesis," in *Neural Information Processing Systems (NIPS)*, vol. 34, 2021.

[27] L. Yang, Z. Zhang, Y. Song, S. Hong, R. Xu, Y. Zhao, Y. Shao, W. Zhang, B. Cui, and M.-H. Yang, "Diffusion models: A comprehensive survey of methods and applications," *arXiv preprint*, 2022.

[28] S. Welker, J. Richter, and T. Gerkmann, "Speech enhancement with score-based generative models in the complex STFT domain," in *ISCA Interspeech*, 2022.

[29] Y.-J. Lu, Z.-Q. Wang, S. Watanabe, A. Richard, C. Yu, and Y. Tsao, "Conditional diffusion probabilistic model for speech enhancement," *IEEE Int. Conf. Acoustics, Speech, Signal Proc. (ICASSP)*, 2022.

[30] J. Serrà, S. Pascual, J. Pons, R. O. Araz, and D. Scaini, "Universal speech enhancement with score-based diffusion," *arXiv preprint*, 2022.

[31] S. Han and J. Lee, "Nu-wave 2: A general neural audio upsampling model for various sampling rates," in *ISCA Interspeech*, 2022.

[32] D. P. Kingma, T. Salimans, B. Poole, and J. Ho, "Variational diffusion models," in *Neural Information Processing Systems (NIPS)*, 2021.

[33] J. Song, C. Meng, and S. Ermon, "Denoising diffusion implicit models," in *Int. Conf. Learning Repr. (ICLR)*, 2021.

[34] M. W. Y. Lam, J. Wang, D. Su, and D. Yu, "BDDM: Bilateral denoising diffusion models for fast and high-quality speech synthesis," in *Int. Conf. Learning Repr. (ICLR)*, 2022.

[35] R. Rombach, A. Blattmann, D. Lorenz, P. Esser, and B. Ommer, "High-resolution image synthesis with latent diffusion models," in *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2022.

[36] B. Jing, G. Corso, R. Berlinghieri, and T. Jaakkola, "Subspace diffusion generative models," in *European Conf. on Computer Vision (ECVA)*, 2022.

[37] Z. Qiu, M. Fu, Y. Yu, L. Yin, F. Sun, and H. Huang, "SRTNet: Time domain speech enhancement via stochastic refinement," *arXiv preprint*, 2022.

[38] B. Kawar, M. Elad, S. Ermon, and J. Song, "Denoising diffusion restoration models," in *Neural Information Processing Systems (NIPS)*, 2022.

[39] K. Saito, N. Murata, T. Uesaka, C.-H. Lai, Y. Takida, T. Fukui, and Y. Mitsufuji, "Unsupervised vocal dereverberation with diffusion-based generative models," *arXiv preprint*, 2022.

[40] R. Sawata, N. Murata, Y. Takida, T. Uesaka, T. Shibuya, S. Takahashi, and Y. Mitsufuji, "A versatile diffusion-based generative refiner for speech enhancement," *arXiv preprint*, 2022.

[41] J. S. Garofolo, D. Graff, D. Paul, and D. Pallett, "CSR-I (WSJ0) Complete." [Online]. Available: https://catalog.ldc.upenn.edu/LDC93S6A

[42] J. Garofolo, L. Lamel, W. Fisher, J. Fiscus, D. Pallett, N. Dahlgren, and V. Zue, "TIMIT acoustic-phonetic continuous speech corpus," *Linguistic Data Consortium*, 11 1992.

[43] C. Valentini-Botinhao, X. Wang, S. Takaki, and J. Yamagishi, "Investigating RNN-based speech enhancement methods for noise-robust text-to-speech," *9th ISCA Speech Synthesis Workshop*, 2016.

[44] Y. Song, J. Sohl-Dickstein, D. P. Kingma, A. Kumar, S. Ermon, and B. Poole, "Score-based generative modeling through stochastic differential equations," *Int. Conf. Learning Repr. (ICLR)*, 2021.

[45] A. Hyvärinen and P. Dayan, "Estimation of non-normalized statistical models by score matching." *Journal of Machine Learning Research*, 2005.

[46] P. Vincent, "A connection between score matching and denoising autoencoders," *Neural Computation*, vol. 23, no. 7, pp. 1661–1674, 2011.

[47] B. Øksendal, *Stochastic Differential Equations: An Introduction with Applications*. Journal of the American Statistical Association, 2000, vol. 82.

[48] B. D. Anderson, "Reverse-time diffusion equation models," *Stochastic Processes and their Applications*, 1982.

[49] N. Chen, Y. Zhang, H. Zen, R. J. Weiss, M. Norouzi, and W. Chan, "WaveGrad: Estimating gradients for waveform generation," *Int. Conf. Learning Repr. (ICLR)*, 2021.

[50] S. Särkkä and A. Solin, *Applied Stochastic Differential Equations*. Cambridge University Press, 2019.

[51] A. Avila, A. Alam, D. O'Shaughnessy, and T. Falk, "Investigating speech enhancement and perceptual quality for speech emotion recognition," in *ISCA Interspeech*, 2018.

[52] A. Kendall and Y. Gal, "What uncertainties do we need in bayesian deep learning for computer vision?" in *Neural Information Processing Systems (NIPS)*, 2017.

[53] E. Vincent, R. Gribonval, and C. Fevotte, "Performance measurement in blind audio source separation," *IEEE/ACM Trans. Audio, Speech, Language Proc.*, vol. 14, no. 4, pp. 1462–1469, 2006.

[54] J. Barker, R. Marxer, E. Vincent, and S. Watanabe, "The third 'CHiME' speech separation and recognition challenge: Dataset, task and baselines," *IEEE Workshop on Automatic Speech Recognition and Understanding (ASRU)*, 2015.

[55] J. Thiemann, N. Ito, and E. Vincent, "The Diverse Environments Multi-channel Acoustic Noise Database (DEMAND): A database of multichannel environmental noise recordings," *The Journal of the Acoustical Society of America*, vol. 133, no. 5, pp. 3591–3591, 2013.

[56] R. Scheibler, E. Bezzam, and I. Dokmanic, "Pyroomacoustics: A python package for audio room simulation and array processing algorithms," in *IEEE Int. Conf. Acoustics, Speech, Signal Proc. (ICASSP)*, apr 2018.

[57] A. Li, C. Zheng, L. Zhang, and X. Li, "Glance and gaze: A collaborative learning framework for single-channel speech enhancement," *Applied Acoustics*, 2022.

[58] S.-W. Fu, C. Yu, T.-A. Hsieh, P. Plantinga, M. Ravanelli, X. Lu, and Y. Tsao, "MetricGAN+: An improved version of metricgan for speech enhancement," in *ISCA Interspeech*, 2022, pp. 7412–7416.

[59] X. Bie, S. Leglaive, X. Alameda-Pineda, and L. Girin, "Unsupervised speech enhancement using dynamical variational autoencoders," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 30, pp. 2993–3007, 2022.

[60] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *Int. Conf. Learning Repr. (ICLR)*, 2015.

[61] Y. Song and S. Ermon, "Improved techniques for training score-based generative models," in *Neural Information Processing Systems (NIPS)*, 2020.

[62] A. Rix, J. Beerends, M. Hollier, and A. Hekstra, "Perceptual evaluation of speech quality (PESQ)-a new method for speech quality assessment of telephone networks and codecs," in *IEEE Int. Conf. Acoustics, Speech, Signal Proc. (ICASSP)*, 2001.

[63] J. Jensen and C. Taal, "An algorithm for predicting the intelligibility of speech masked by modulated noise maskers," *IEEE/ACM Trans. Audio, Speech, Language Proc.*, 2016.

[64] J. L. Roux, S. Wisdom, H. Erdogan, and J. R. Hershey, "Sdr - half-baked or well done?" in *IEEE Int. Conf. Acoustics, Speech, Signal Proc. (ICASSP)*, 2019.

[65] P. Andreev, A. Alanov, O. Ivanov, and D. Vetrov, "Hifi++: a unified framework for bandwidth extension and speech enhancement," *arXiv preprint*, 2022.

[66] O. Kuchaiev, J. Li, H. Nguyen, O. Hrinchuk, R. Leary, B. Ginsburg, S. Kriman, S. Beliaev, V. Lavrukhin, J. Cook *et al.*, "Nemo: a toolkit for building AI applications using neural modules," *arXiv preprint*, 2019.