GAUSSIAN FLOW BRIDGES FOR AUDIO DOMAIN TRANSFER WITH UNPAIRED DATA

Eloi Moliner[*]
Acoustics Lab, DICE
Aalto University, Espoo, Finland
eloi.moliner@aalto.fi

Sebastian Braun *Hannes Gamper*
Microsoft Research Redmond
first.last@microsoft.com

ABSTRACT

Audio domain transfer is the process of modifying audio signals to match characteristics of a different domain, while retaining the original content. Examples include transferring room acoustics or altering audio effects such as distortion. This paper investigates the potential of Gaussian Flow Bridges, an emerging approach in generative modeling, for these problems. The presented framework addresses the transport problem across different distributions of audio signals through the implementation of a series of two deterministic probability flows. The proposed framework facilitates manipulation of the target distribution properties through a continuous control variable, which defines a certain aspect of the target domain. Notably, this approach does not rely on paired examples for training. To address identified challenges on maintaining the speech content consistent, we recommend a training strategy that incorporates chunk-based minibatch Optimal Transport couplings of data samples and noise. Comparing our unsupervised method with established baselines, we find competitive performance in tasks of reverberation and distortion manipulation. Despite encountering limitations, the intriguing results obtained in this study underscore potential for further exploration.

***Index Terms*—** audio processing, probabilistic modeling, machine learning

1. INTRODUCTION

The search for data-driven methods that allow for controlled modification of audio signals has attracted considerable attention and research efforts throughout the past decade. The majority of proposed techniques are dependent on supervised learning, necessitating the availability of paired "input" and "target" samples for effective training. A prominent instance of this is speech enhancement, which has seen significant advancements in performance due to meticulously designed data processing pipelines and optimization strategies [1]. However, the requirement for paired samples introduces substantial constraints, making it impractical in certain scenarios. Obtaining such data can be challenging or impossible due to the high cost and effort involved in producing or collecting it, limited access to specific real-world conditions, potential mismatches when using synthetic data, or difficulties in achieving precise time alignment between pairs. Consequently, unsupervised methods offer a promising research avenue. In the context of audio, there have been several contributions in this direction, employing techniques such as mixture-invariant training [2], or various forms of generative adversarial networks [3, 4, 5].

Recent studies have demonstrated the effectiveness of diffusion models in unsupervised audio restoration and editing. A known approach consists of utilizing the generative priors from diffusion models to sample from posterior distributions [6]. However, this framework requires exact knowledge of a forward degradation and is not readily applicable for general settings. A work closely related to ours is by Popov et al. [7], who employed a bridge to transport mel-spectrograms for voice conversion and instrument timbre transfer. Similarly, Manor and Michaeli explored a technique called "DDPM inversion" for unsupervised editing of mel-spectrograms [8].

In this study, we explore a method we term *Gaussian Flow Bridges* (GFBs), which offers a general way to handle audio domain transfer tasks in an unsupervised manner. GFBs address a transport problem between probability densities, known as the Schrödinger bridge problem [9], by applying two deterministic processes or flows. The first process transforms an audio waveform into a latent vector within a Gaussian distribution, while the second changes this latent vector into a modified waveform. GFBs enable many-to-many mappings within audio domains. This approach aligns with concepts previously explored as "Dual Diffusion Implicit Bridges" [10], or "DDIM inversion"[11]. This paper applies this idea through the Flow Matching framework [12], hence the distinct terminology.

Unlike prior work [7, 8], our research focuses on the development of GFBs in the waveform domain. This approach eliminates the need for a spectrogram inversion model, thereby simplifying the operational framework. A pivotal aspect of our study is addressing the complexities introduced by waveform representation in GFBs, particularly when maintaining content fidelity in speech signals. These complexities often manifest as undesirable artifacts, including abrupt identity shifts or unintelligible speech. Our work adheres to optimal transport principles and underscores the importance of linear transformation paths within the GFB framework, suggesting that maintaining linear trajectories is crucial for preserving content integrity. To enhance the model performance while adhering to this principle, we introduce a training methodology that uses chunk-based minibatch optimal transport (OT) couplings.

The experiments outlined in Sec. 4 delve into two key areas: speech reverberation and distortion. While our methodology showcases promising results in these domains, it is important to emphasize the broader applicability of the discussed approach.

2. BACKGROUND

2.1. Continuous Normalizing Flows

Continuous Normalizing Flows (CNFs) are designed to iteratively transport samples between two probability distributions, q_0 and q_1, across a defined *time* interval $\tau \in [0, 1]$. Let $\mathbf{x}_\tau$ denote the sample at any time τ within this interval, starting with $\mathbf{x}_0 \sim q_0$, and ending with $\mathbf{x}_1 \sim q_1$. The underlying process is formalized with an Ordinary Differential Equation (ODE), characterized by a time-dependent vector field:

$$\mathrm{d}\mathbf{x}_\tau = u(\mathbf{x}_\tau, \tau)\mathrm{d}\tau. \tag{1}$$

[*]Work done while intern at Microsoft Research

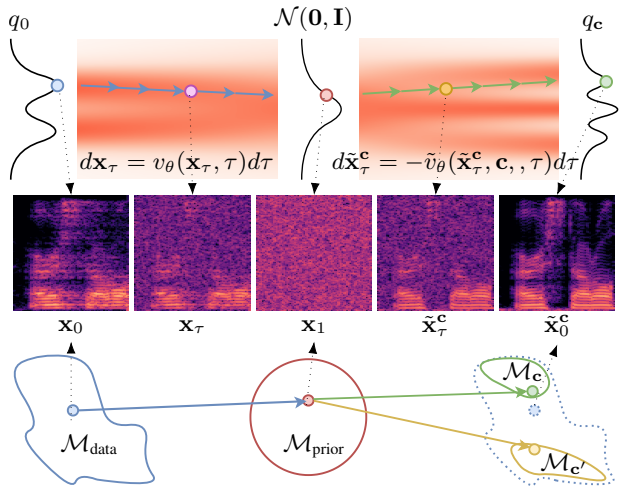

Fig. 1. *(Top) Illustration of a GFB in one-dimensional space. (Middle) A sequential display of spectrograms, showcasing the stages of audio signal transformation. (Bottom) Geometrical interpretation highlighting the mapping of data points through encoding and decoding within a Gaussian space.*

With a specified vector field u, it becomes feasible to transport samples from $\mathbf{x}_0 \sim q_0$ to $\mathbf{x}_1 \sim q_1$ and vice versa by solving the ODE both in the *forward* direction, where τ varies from 0 to 1, and in the *backward* direction, where τ varies from 1 to 0. Particularly, when one of these distributions is a tractable one, such as a Gaussian defined by $p_1 = \mathcal{N}(\mathbf{0}, \mathbf{I})$, CNFs function as generative models and exhibit notable parallels with diffusion models [13].

2.2. Conditional Flow Matching

Several works [12, 14] approximate the vector field $u(\mathbf{x}_\tau, \tau)$ with a deep neural network $v_\theta(\mathbf{x}_\tau, \tau)$ with parameters θ. This approximation is achieved by optimizing the parameters θ to minimize the following Conditional Flow Matching objective:

$$\mathcal{L}_{\text{CFM}} = \mathbb{E}_{\tau, q_0, q_1} \|v_\theta(\mathbf{x}_\tau, \tau) - u(\mathbf{x}_\tau, \tau | \mathbf{x}_0, \mathbf{x}_1)\|^2, \qquad (2)$$

where $\mathbb{E}$ is the expectation operator. As suggested in [14, 12], a valid strategy for designing the probability path is linear interpolation: $\mathbf{x}_\tau = (1-\tau)\mathbf{x}_0 + \tau\mathbf{x}_1$, corresponding to a vector field with constant velocity over time: $u(\mathbf{x}_\tau, \tau | \mathbf{x}_0, \mathbf{x}_1) = \mathbf{x}_1 - \mathbf{x}_0$. Such parameterization leads to linear trajectories which, when one of the distributions is a Gaussian, represent optimal transport paths [12].

Conditional information, if available, can be used to direct the trajectories using a technique known as Classifier-Free Guidance [15], where a conditional vector $\mathbf{c}$ is added as an auxiliary input to the model. This allows the conditional influence to be modulated as

$$\tilde{v}_\theta(\mathbf{x}_\tau, \mathbf{c}, \tau) = \gamma v_\theta(\mathbf{x}_\tau, \mathbf{c}, \tau) + (1-\gamma) v_\theta(\mathbf{x}_\tau, \mathbf{c} = \emptyset, \tau), \qquad (3)$$

where $v_\theta(\mathbf{x}_\tau, \mathbf{c} = \emptyset, \tau)$ implies that the conditioning vector $\mathbf{c}$ is not included, and the hyperparemeter γ weights both model evaluations.

3. METHODS

3.1. Gaussian Flow Bridges

The Gaussian Flow Bridges (GFB) method involves a two-step process using two deterministic flows evaluated in opposite directions, an encoder and a decoder. As represented in Fig. 1, a starting sample $\mathbf{x}_0 \in \mathbb{R}^n$ at $\tau = 0$ is first encoded using an unconditional vector field model $v_\theta(\mathbf{x}_\tau, \tau)$ into a Gaussian distribution $q_1 = \mathcal{N}(\mathbf{0}, \mathbf{I})$ at $\tau = 1$, producing a latent sample $\mathbf{x}_1 \in \mathbb{R}^n$. This latent sample is then decoded with a conditional model $\tilde{v}_\theta(\mathbf{x}_\tau, \mathbf{c}, \tau)$, yielding a modified sample $\tilde{\mathbf{x}}^\mathbf{c}_0 \in \mathbb{R}^n$. The condition $\mathbf{c}$ is a continuous variable, enabling GFB to create a diverse range of outcomes based on $\mathbf{c}$. It is worth noting that although the encoding and decoding processes are designed to be optimal transport paths between different distributions and a Gaussian, this does not guarantee optimal displacement between the endpoints themselves.

The bottom of Fig 1 represents our interpretation of the GFB concept from a geometric perspective. The data points $\mathbf{x}_0$ are described as part of a data manifold $\mathcal{M}_{\text{data}} \subset \mathbb{R}^n$. Through the encoding, these points are mapped to a Gaussian noise space or hypersphere ($\mathcal{M}_{\text{prior}}$). During decoding, the conditional vector field guides these points to a specific subset of the original data manifold $\tilde{\mathbf{x}}^\mathbf{c}_0 \in \mathcal{M}_\mathbf{c}$, which depends on $\mathbf{c}$. We hypothesize that the optimal endpoints $\mathbf{x}^\mathbf{c}_0$, those that reflect the attributes specified by $\mathbf{c}$ while minimally altering unrelated features, should ideally lie in close Euclidean proximity to the initial sample $\mathbf{x}_0$. This conjecture supports the notion that leveraging a Gaussian distribution as a bridge can facilitate a valid approximation.

3.2. Chunk-based minibatch optimal transport couplings

As will be shown in the analysis Sec. 4.3, we observe that GFBs struggle to preserve content that should be orthogonal to the conditioning variable $\mathbf{c}$. We hypothesize that this issue arises because the sampling trajectories deviate from straight linear paths and exhibit curvature. As suggested in [16, 14, 17], such curvature arises from employing data-independent couplings during training. Specifically, when the pairs of data $\mathbf{x}_0$ and noise $\mathbf{x}_1$ are sampled independently, the resulting training trajectories tend to intersect. This intersection causes the model to approximate a suboptimal average of these paths, rather than identifying distinct, optimal paths individually [14]. Such convergence towards an average trajectory deviates from the intended direct paths and can potentially harm the efficacy and reliability of the GFB strategy.

With the goal of minimizing trajectory curvature, some works propose to assign the data/noise pairs during training using a minibatch optimal transport strategy [17, 18] . According to their findings, this approach effectively minimizes the trajectory curvature and reduces the variance of gradients during training [17]. However, we realize that this strategy does by default not scale well for data of very high dimensionality, such as audio waveforms, which are typically sampled at high rates. As the dimensionality increases, the number of possible data configurations grows exponentially, necessitating larger minibatch sizes for effective coverage. This leads to increased computational and memory requirements, potentially causing bottlenecks in our training pipeline.

We observe that speech signals have high information density, with significant data concentration occurring within localized time windows of just a few milliseconds. To adapt the above approach for practical use with audio data, we propose redefining minibatches $\{\mathbf{x}_0^{(i)} \in \mathbb{R}^N\}_{i=0}^B$, which originally comprise B instances of size N, into smaller chunks of size $N_\text{c} \ll N$. This results in minibatches with a larger number of instances $B_\text{c} = B\frac{N}{N_\text{c}}$

The next step involves computing an optimal transport coupling between the chunked minibatch and an equally sized set of noise samples. We start calculating a matrix $\mathbf{C}_{ij}$ of pairwise L2 distances between all the elements i and j in the two minibatches. This matrix $\mathbf{C}_{ij}$ is used by an optimal transport solver to assign the corresponding pairs $(\overline{\mathbf{x}}_0, \overline{\mathbf{x}}_1)$. We use off-the-shelf solvers from the

Algorithm 1 Training with chunk-based minibatch OT

for each training iteration **do**
  Sample minibatches $\{\mathbf{x}_0^{(i)}\}_{i=0}^{B} \sim q_0$ and $\{\mathbf{x}_1^{(j)}\}_{j=0}^{B} \sim q_1$
  Split $(\mathbf{x}_0, \mathbf{x}_1)$ into chunks
  Compute $\mathbf{C}_{ij} = \|\mathbf{x}_0^{(i)} - \mathbf{x}_1^{(j)}\|_2^2$
  Solve OT for $\mathbf{C}_{ij}$, get coupling $(\overline{\mathbf{x}}_0, \overline{\mathbf{x}}_1)$
  Reshape $(\overline{\mathbf{x}}_0, \overline{\mathbf{x}}_1)$ to the original length
  $\mathbf{x}_\tau \leftarrow (1-\tau)\overline{\mathbf{x}}_0 + \tau\overline{\mathbf{x}}_1$, $\tau \sim \mathcal{U}(0,1)$
  $\mathcal{L}_{\text{CFM}}(\theta) \leftarrow \mathbb{E}_{\tau,q_0,q_1}\|v_\theta(\mathbf{x}_\tau, \tau) - (\overline{\mathbf{x}}_1 - \overline{\mathbf{x}}_0)\|_2^2$
  $\theta \leftarrow \text{update}(\theta, \nabla_\theta \mathcal{L}_{\text{CFM}}(\theta))$
end for

Python Optimal Transport library [19], in particular, when using $N_c = 512$, we use ot.emd, an exact optimal transport solver. When experimenting with smaller N_c, we instead opted for *ot.sinkhorn*, an entropy-regularized solver that provides approximate OT solutions at a lower computational cost. After, the coupled pairs are reshaped to their original dimensions and the CFM objective (Eq. 2) is computed. The training loop employing the chunk-based minibatch OT methodology is detailed in Algorithm 1.

4. EXPERIMENTS

We conducted experiments in two areas, namely *speech reverberation* and *distortion*. In *speech reverberation*, we focused on the task of acoustics transfer, which extends beyond the dereverberation task to modifying the characteristics of reverberation. Such a controllable approach holds potential for a variety of applications in augmented and virtual reality [20], where the ability to modify audio signals to match expected acoustics can significantly enhance listener experience. We design a GFB where the initial distribution p_{data} contains speech with undetermined acoustic properties and the terminal distribution $p_{\mathbf{c}}$ comprises speech signals with a specified acoustic condition $\mathbf{c}$. We experiment with two reverberation descriptors: reverberation time (T_{60}) and clarity (C_{50}).

For *distortion*, we explore our method's ability to handle nonlinear effects, with our experiments focusing on *speech clipping*. Here, the GFB is trained with both clipped and clean speech signals, and the goal is to transform initial samples to a specific Signal-to-Distortion Ratio (SDR). Additionally, we provide qualitative insights into *guitar distortion* manipulation in the companion webpage[1].

4.1. Experimental setup

As training data, we used studio quality speech samples from VCTK [21]. For our reverberation experiments, we convolved the speech recordings with single-channel room impulse responses (RIRs), collected by combining several public datasets [22, 23, 24, 25, 26], using RIRs with T_{60} values ranging from 0 to 1s. For the clipping experiment, the training speech samples were clipped at different SDR levels. During the training, we also include, with a probability of 10%, clean speech samples. The reverberation descriptors T_{60} and C_{50}, and the SDR in the case of declipping, are estimated and concatenated into a conditioning vector $\mathbf{c}$. All signals are resampled to 16 kHz and are randomly cropped to a segment size of 4.09 s.

In our experiments, we use a backbone architecture v_θ based on the Short-Time Fourier Transform (STFT). A forward and an inverse STFT are applied wrapping trainable neural network layers in a similar way as in [6]. The complex-valued spectrograms are processed as double-real signals, stacking the real and imaginary parts into the channel dimension. The architecture is a U-Net with roughly 44 M trainable parameters, mainly consisting of 2-Dimensional convolutional layers, The conditioning vector $\mathbf{c}$, alongside with the time variable τ, is fed into the neural network through feature modulations. During training, the conditioning vector $\mathbf{c}$ is randomly dropped with a probability of 20%, to allow unconditional sampling and the use of Classifier-Free Guidance. All models compared in the experiments are trained for 300k iterations using the Adam optimizer, with a learning rate of 10^{-4}, and a batch size of 8.

[1]Code and examples available at microsoft.github.io/GFB-audio-control/dist/

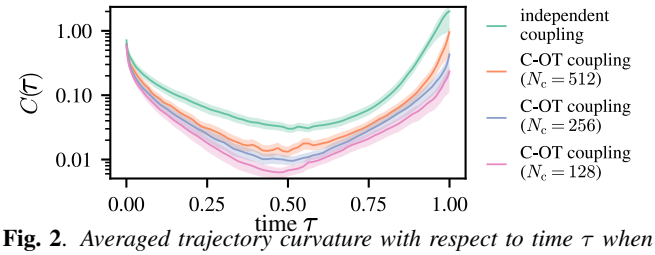

Fig. 2. *Averaged trajectory curvature with respect to time τ when different coupling strategies are used. The shaded area represents the 25% and 75% percentiles.*

4.2. Coupling configurations and trajectory curvature analysis

Our investigation begins by examining the influence of the training couplings on the sampling trajectories. Following the methodologies described in [16, 14], we utilize a surrogate metric to analyze trajectory curvature $C(\tau) = \|(\mathbf{x}_1 - \mathbf{x}_0) - \frac{\partial \mathbf{x}_t}{\partial \tau}\|_2^2$, which compares the local slope at every time τ with the total displacement. Ideally, if the paths were completely straight, this metric should yield a value of 0.

Figure 2 displays the distribution of $C(\tau)$ values for different timesteps during the forward sampling process. These trajectories begin from each example in the test set at $\tau = 0$ and progress toward a Gaussian distribution at $\tau = 1$. We compare the results obtained with a model trained on the reverberant speech dataset with the default training setup (independent couplint) against other models trained with the proposed chunked minibatch OT (C-OT) couplings. For the latter, we study the effect of the chunk length N_c which, assuming a fixed sample length N and batch size B, affects directly the chunked minibatch size $B_c = B\frac{N}{N_c}$. Three different chunk lengths N_c are considered: 512, 256, and 128 samples; which correspond to 32, 16 and 8 ms.

The results reveal that the use of C-OT couplings significantly reduces the observed curvature, with a consistent reduction when the chunk size N_c is decreased, showcasing the effect of the C-OT couplings. It can also be observed that all configurations show lower curvature values around the midpoint of the process ($t \approx 0.5$), but these values notably increase toward the extremes, specially at $t \approx 1$. This observed behavior inspires the adoption of a discretization scheme based on a raised cosine schedule that prioritizes smaller time steps at the extremes: $\tau_{i<T} = 0.5 + 0.5\cos(\pi i/T + \pi))$. We used this schedule, with $T = 25$ steps, in the rest of experiments. Although not the primary focus of this study, the observed lower curvature of C-OT couplings indicate a potential for more efficient sampling compared to conventional flow or diffusion-based models.

4.3. Speech reverberation evaluation

Our investigation focuses on assessing the performance of different models in reverberation control, emphasizing the trade-off between two aspects: *acoustics accuracy* and *speech content consistency*. Acoustics accuracy assesses the model's capability to recreate speech aligned with predetermined acoustic features, particularly

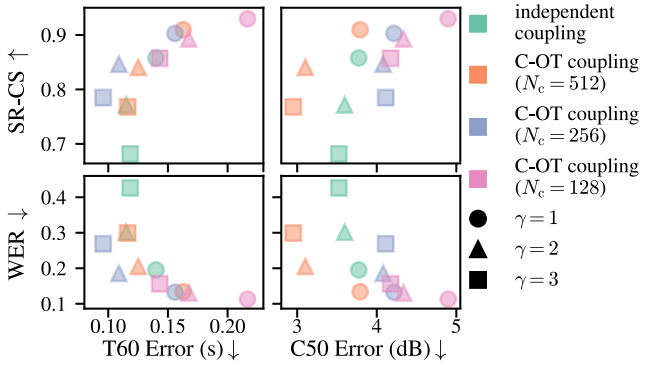

Fig. 3. *Scatter plots illustrating the trade-offs between SR-CS and WER versus T_{60} and C_{50} errors for models conditioned on specific acoustic features. Points represent aggregated test set results, highlighting the effects of chunk length (N_c) and CFG scale (γ).*

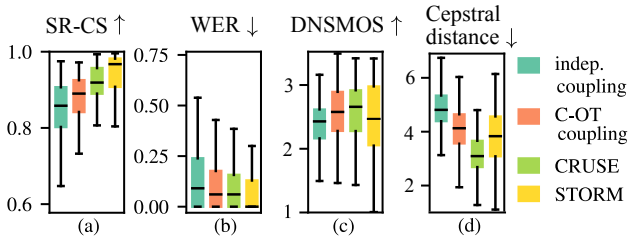

Fig. 4. *Objective evaluation on speech dereverberation.*

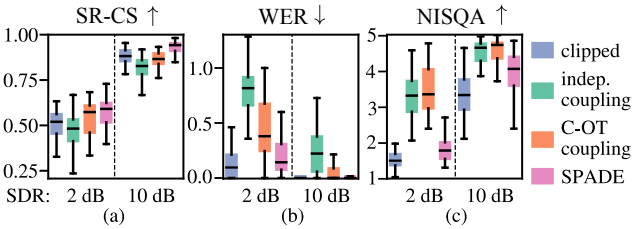

Fig. 5. *Objective evaluation on speech declipping.*

T60 and C50. To quantify the models' fidelity in reproducing the target acoustic characteristics, we utilize a blind acoustic parameter estimator [27]. We calculate the mean absolute error between the model-predicted T60 and C50 values and their actual measurements.

When analyzing speech content consistency, we assess the model's ability to retain the original speech content. This involves addressing two critical issues: alterations in speaker identity and potential loss of intelligibility. To adress the first issue, we measure the cosine similarity between embeddings derived from the speaker recognition model [28], we refer to this metric as Speaker Recognition Cosine Similarity (SR-CS). With respect to the second, we compare Automatic Speech Recognition transcripts before and after the GFB. We use the "small" version of Whisper [29] and report the Word Error Rate (WER) with the original sample's transcription as a benchmark. We assume these two models to be robust and approximately invariant to acoustics.

We use a test set conforming 20 minutes of 4-s length studio quality speech examples from DAPS [30], a different dataset than the one used for training. These examples are convolved with a set real RIRs containing uniformly balanced T60 values ranging from 0 to 1s, and C50 values ranging from 0 to 25 dB. We use 320 RIRs extracted from datasets not used during training [31, 32, 33, 34]. All the examples in the test set are transformed to 8 different endpoints using the proposed GFB, each of them corresponding to a distinct conditioning setting with specific T60 and C50 values.

In Figure 3, we provide a detailed analysis reflecting the average SR-CS and WER in relation to both T_{60} and C_{50} errors. These scatter plots are generated from the averaged results across the test set. The figure illustrates the outcomes for various models trained using different C-OT couplings, wherein the chunk length N_c varies. Additionally, each model's behavior concerning the Classifier-Free Guidance scaling parameter γ is examined. Our observations reveal that, for smaller N_c, the speech consistency gets improved, but usually at the cost of reduced acoustic accuracy. In addition, the parameter γ reflects a notable trade-off, as increasing this value allows for more precise adjustments in acoustics characteristics at the expense of introducing artifacts that compromise speech consistency.

Additionally, we assess the proposed method performance for dereverberation. We utilize the speech consistency metrics *SR-CS* and *WER*, the MOS prediction metric *DNSMOS* [35], and the cepstral distance [36]. In our evaluation, two state-of-the-art baselines are included, CRUSE [1] and STORM [37]. Unlike the proposed method, these baselines were trained using paired data. Results are shown on a subset of the test set, specifically 160 utterances with T60 values falling within the range of 0.5 to 1. The conditioning parameters for the diffusion bridge are set to T60=0.1s and C50=20dB, a dry but not anechoic specification. We conduct a comparative analysis between the model trained with independent coupling and the one trained with C-OT coupling, employing $N_c = 512$ and $\gamma = 1$. The results are presented in Figure 4. Although none of the compared versions of the proposed method surpass the baseline performance, the results of the C-OT method notably converge closely. It should be noted that the proposed method and the baselines were trained using different data, thus the results depend on the models' generalization capabilities.

4.4. Declipping evaluation

The performance of both versions of our method in the task of speech declipping is compared against the clipped speech and SPADE [38], a popular sparsity-based declipping baseline. In Figure 5, we report three objective metrics: SR-CS and WER, as introduced in Section 4.3, and NISQA [39], a MOS prediction model that is known to correlate well with declipping performance. The results show a strong improvement of the proposed method against SPADE in terms of NISQA. However, in terms of WER and SR-CS, GFB does not reach the same consistency scores of SPADE, and neither of the clipped speech. We also notice that the usage of C-OT couplings is critical at reducing the WER in this setting.

5. CONCLUSION

This paper studied the application of GFBs for unsupervised audio domain transfer, with experiments on reverberation and distortion control. The experiments show that, in the majority of cases, GFBs effectively manage to alter an audio effect characteristic while preserving the content integrity, a notable achievement considering it was not specifically trained for this task. Furthermore, the method exhibits the ability to generalize to unseen speakers and acoustic conditions. Qualitative assessments indicate that GFBs yield results free from typical artifacts seen in speech reverberation and declipping. However, occasional inconsistencies in speech content and speaker identity are observed, posing a significant challenge for the method's potential applications. Nonetheless, the performance of GFBs shows promising progress, paving the way for further enhancements and applications in diverse tasks and domains.

6. REFERENCES

[1] S. Braun and H. Gamper, "Effect of noise suppression losses on speech distortion and ASR performance," in *Proc. ICASSP*, 2022.

[2] V. A. Trinh and S. Braun, "Unsupervised speech enhancement with speech recognition embedding and disentanglement losses," in *Proc. ICASSP*, 2022.

[3] T. Kaneko and H. Kameoka, "CycleGAN-VC: Non-parallel voice conversion using cycle-consistent adversarial networks," in *Proc. EUSIPCO*, 2018.

[4] S-W. Fu, C. Yu, K-H. Hung, et al., "MetricGAN-U: Unsupervised speech enhancement/dereverberation based only on noisy/reverberated speech," in *Proc. ICASSP*, 2022.

[5] A. Wright, V. Välimäki, and L. Juvela, "Adversarial guitar amplifier modelling with unpaired data," in *Proc. ICASSP*, 2023.

[6] E. Moliner, J. Lehtinen, and V. Välimäki, "Solving audio inverse problems with a diffusion model," in *Proc. ICASSP*, 2023.

[7] V. Popov, A. Amatov, M. Kudinov, et al., "Optimal transport in diffusion modeling for conversion tasks in audio domain," in *Proc. ICASSP*, 2023.

[8] H. Manor and T. Michaeli, "Zero-shot unsupervised and text-based audio editing using ddpm inversion," *arXiv preprint arXiv:2402.10009*, 2024.

[9] V. De Bortoli, J. Thornton, J. Heng, et al., "Diffusion schrödinger bridge with applications to score-based generative modeling," *NeurIPS*, vol. 34, 2021.

[10] X. Su, J. Song, C. Meng, and S. Ermon, "Dual diffusion implicit bridges for image-to-image translation," in *Proc. ICLR*, 2023.

[11] R. Mokady, A. Hertz, K. Aberman, et al., "Null-text inversion for editing real images using guided diffusion models," in *Proc. CVPR*, 2023.

[12] Y. Lipman, R. TQ Chen, H. Ben-Hamu, et al., "Flow matching for generative modeling," in *Proc. ICLR*, 2022.

[13] J. Ho, A. Jain, and P. Abbeel, "Denoising diffusion probabilistic models," *NeurIPS*, vol. 33, 2020.

[14] X. Liu, C. Gong, and Q. Liu, "Flow straight and fast: Learning to generate and transfer data with rectified flow," in *Proc. ICLR*, 2022.

[15] J. Ho and T. Salimans, "Classifier-free diffusion guidance," *arXiv preprint arXiv:2207.12598*, 2022.

[16] S. Lee, B. Kim, and J. C. Ye, "Minimizing trajectory curvature of ode-based generative models," in *Proc. ICML*, 2023.

[17] A.-A. Pooladian, H. Ben-Hamu, C. Domingo-Enrich, et al., "Multisample flow matching: Straightening flows with minibatch couplings," in *Proc. ICML*, 2023.

[18] A. Tong, N. Malkin, G. Huguet, et al., "Improving and generalizing flow-based generative models with minibatch optimal transport," in *ICML Workshop on New Frontiers in Learning, Control, and Dynamical Systems*, 2023.

[19] R. Flamary, N. Courty, A. Gramfort, et al., "POT: Python optimal transport," *JMLR*, vol. 22, no. 1, 2021.

[20] Changan Chen, Ruohan Gao, Paul Calamia, and Kristen Grauman, "Visual acoustic matching," in *Proc. CVPR*, 2022.

[21] J. Yamagishi, C. Veaux, K. MacDonald, et al., "CSTR VCTK corpus: English multi-speaker corpus for cstr voice cloning toolkit (version 0.92)," *University of Edinburgh. CSTR*, 2019.

[22] K. Prawda, S. J Schlecht, and V. Välimäki, "Calibrating the Sabine and Eyring formulas," *J. Acoust. Soc. Am.*, vol. 152, no. 2, 2022.

[23] I. Szöke, M. Skácel, L. Mošner, et al., "Building and evaluation of a real room impulse response dataset," *IEEE Journal of Selected Topics in Signal Processing*, vol. 13, no. 4, 2019.

[24] R. Stewart and M. Sandler, "Database of omnidirectional and b-format room impulse responses," in *Proc. ICASSP*, 2010.

[25] D. T Murphy and S. Shelley, "OpenAir: An interactive auralization web resource and database," in *Proc. Conv. Audio Eng. Soc.*, 2010.

[26] J. Eaton, N. D. Gaubitch, A. H. Moore, and P. A. Naylor, "The ACE challenge — corpus description and performance evaluation," in *Proc. WASPAA*, 2015.

[27] H. Gamper and I. J. Tashev, "Blind reverberation time estimation using a convolutional neural network," in *Proc. IWAENC*, 2018.

[28] R. Wang, Z. Wei, H. Duan, et al., "EfficientTDNN: efficient architecture search for speaker recognition," *IEEE/ACM Trans. Audio, Speech, Language Processing*, vol. 30, 2022.

[29] A. Radford, J. W. Kim, T. Xu, et al., "Robust speech recognition via large-scale weak supervision," in *Proc. ICML*, 2023.

[30] G. J Mysore, "Can we automatically transform speech recorded on common consumer devices in real-world environments into professional production quality speech?—a dataset, insights, and challenges," *IEEE Signal Processing Letters*, vol. 22, no. 8, 2014.

[31] J. Traer and J. H McDermott, "Statistics of natural reverberation enable perceptual separation of sound and space," *Proc. National Academy of Sciences*, vol. 113, no. 48, 2016.

[32] D. Di Carlo, P. Tandeitnik, C. Foy, et al., "dechorate: a calibrated room impulse response dataset for echo-aware signal processing," *EURASIP J. Audio, Speech, and Music Processing*, vol. 2021, 2021.

[33] K. Kinoshita, M. Delcroix, T. Yoshioka, et al., "The reverb challenge: A common evaluation framework for dereverberation and recognition of reverberant speech," in *Proc. WASPAA*, 2013.

[34] J. K. Nielsen, J. R. Jensen, S. H. Jensen, and M. G. Christensen, "The single- and multichannel audio recordings database (SMARD)," in *Proc. IWAENC*, 2014.

[35] C. KA Reddy, V. Gopal, and R. Cutler, "DNSMOS: A non-intrusive perceptual objective speech quality metric to evaluate noise suppressors," in *Proc. ICASSP*, 2021.

[36] N. Kitawaki, H. Nagabuchi, and K. Itoh, "Objective quality evaluation for low-bit-rate speech coding systems," *IEEE Journal on Selected Areas in Communications*, vol. 6, no. 2, 1988.

[37] J-M. Lemercier, J. Richter, S. Welker, and T. Gerkmann, "StoRM: A diffusion-based stochastic regeneration model for speech enhancement and dereverberation," *IEEE/ACM Trans. Audio, Speech, Language Processing*, 2023.

[38] S. Kitić, N. Bertin, and R. Gribonval, "Sparsity and cosparsity for audio declipping: a flexible non-convex approach," in *Proc. Latent Variable Analysis and Signal Separation*, 2015.

[39] G. Mittag, B. Naderi, A. Chehadi, and S. Möller, "NISQA: A deep cnn-self-attention model for multidimensional speech quality prediction with crowdsourced datasets," in *INTERSPEECH*, 2021.