PLUG-AND-PLAY AUDIO RESTORATION WITH DIFFUSION DENOISER

Michal Švento *Pavel Rajmic* *Ondřej Mokrý*

Dept. of Telecommunications, FEEC
Brno University of Technology, Brno, Czech Republic
michal.svento@vut.cz

ABSTRACT

There have been a plethora of methods developed to tackle diverse audio reconstruction problems. Recently, deep generative models have affected this field strongly, some of them allowing to solve multiple problems with only a minimal need for adaptation. However, long inference times still represent a barrier to their real-world deployment. We propose a plug-and-play approach to audio reconstruction enabling a shorter duration of signal generation. We present our approach on a number of inverse problems, all evaluated on a piano sound dataset. Subjectively, the proposed strategy performs competitively with recent methods, however, this is rarely reflected by objective metrics.

***Index Terms*—** diffusion model, plug-and-play, audio restoration, reconstruction, declipping, denoising, inpainting

1. INTRODUCTION

Audio restoration covers diverse problems whose common focus is reconstructing a corrupted or degraded signal. Depending on the particular need, the reconstruction quality can be regarded as high if either the reconstruction is close to the original or if the human/machine listener cannot spot a manipulation with the signal. The restoration problems can be divided into two classes, based on whether a linear operator has been involved (e.g. inpainting, bandwidth extension) or it has been nonlinear (e.g. phase retrieval, declipping). Before the rise of the generative deep neural models, the audio restoration problems were treated mostly by optimization-based approaches, where more or less, every type of restoration problem followed its own branch of the development. There are a few exceptions, such as the framework [1] covering multiple degradations, even at the same time, or the approaches [2, 3], where combined problems are solved in a sequential way. In the era of generative networks it becomes more and more common to encounter frameworks able to solve various problems with minimal amount of adaptation [4, 5].

The work was supported by the Czech Science Foundation (GAČR) Project No. 23-07294S. The authors are grateful to NVIDIA for their donation of the Titan XP graphic card, which has been used in this research. The authors are also grateful to Eloi Moliner for valuable advices.

The generative diffusion models have proven good ability to produce meaningful data in image [6], video [7] and audio applications [4]. Nonetheless, in contrast to pure content generation, conditioning of the generative process is necessary if a restoration problem has to be solved, i.e., the generated data must be consistent with the observed/degraded part of the signal being restored. The most successful diffusion-based algorithms (CQT-Diff [5], VRDMG [8]) for audio reconstruction follow the trends from the image restoration field (DDRM [9], RePaint [10], REDDiff [11], diffusion posterior sampling – DPS [12]). CQT-Diff and VRDMG adapt the DPS [12] which has shown the ability to solve linear and nonlinear inverse problems by involving the gradient of the log likelihood of the measurement in each step of the diffusion sampling. The biggest disadvantage of such an approach is a very long inference time.

In this paper, we utilize the plug-and-play methodology. We solve several audio inverse problems with the help of the half-quadratic splitting algorithm (HQS) [13]. We propose to use an unconditional diffusion model in the denoising step. Our denoiser is trained on sounds of a piano. The paper should answer the question whether a better perceptual quality and/or less computational cost can be achieved, compared with CQT-Diff.

This paper is organized as follows: in Sec. 2 a brief introduction is provided to audio inverse problems, diffusion models and the HQS splitting algorithm. Sec. 3 proposes an adapted plug-and-play algorithm for solving multiple tasks. Sec. 4 evaluates the proposed approach in denoising, declipping and compressive sensing on piano recordings.

2. BACKGROUND

2.1. Audio inverse problems

Denote $\mathbf{x}$ the uncorrupted single-channel audio signal. Naturally, vector $\mathbf{x}$ is not accessible in typical practical applications. The degraded observation of the signal is obtained as

$$\mathbf{y} = \mathcal{A}(\mathbf{x}) + \mathbf{n}, \tag{1}$$

where $\mathcal{A}(\cdot)$ is the degradation operator and $\mathbf{n}$ represents a noise vector. The operator $\mathcal{A}$ may either be linear or

nonlinear; moreover, it can operate directly on the waveform or it can operate in the time-frequency domain, whereto we get using a linear signal transform such as the usual short-time Fourier transform (STFT). In all discussed applications in this paper, $\mathcal{A}$ will be a time-domain, sample-wise operator.

Audio inverse problems aiming at the estimation of $\mathbf{x}$ from the observation given by (1) are inherently ill-posed. Typically, there are infinitely many solutions satisfying the observation model. From the Bayesian perspective, the solution can be characterized by the maximum a posteriori (MAP) estimation, i.e., a solution to the problem

$$\hat{\mathbf{x}} = \arg\max_{\mathbf{x}} \ \log p(\mathbf{y}|\mathbf{x}) + \log p(\mathbf{x}), \tag{2}$$

where $\log p(\mathbf{y}|\mathbf{x})$ is the log-likelihood of the observation $\mathbf{y}$ given $\mathbf{x}$, and $\log p(\mathbf{x})$ is the log-prior of the clean audio signal. Problem (2) can be equivalently rewritten as

$$\hat{\mathbf{x}} = \arg\min_{\mathbf{x}} \frac{1}{2\sigma^2}\|\mathbf{y} - \mathcal{A}(\mathbf{x})\|^2 + \lambda\mathcal{R}(\mathbf{x}). \tag{3}$$

The solution thus minimizes a cost function consisting of the data-fidelity term $\frac{1}{2}\|\mathbf{y} - \mathcal{A}(\mathbf{x})\|^2$ and the regularization term $\mathcal{R}(\mathbf{x})$, weighted by the noise variance σ^2 and a balancing hyperparameter $\lambda > 0$.

2.2. Diffusion model

The idea of gradually adding noise to the data until the mixture reaches a prior distribution (such as Gaussian) and then learning the backward step toward the data distribution stands behind the generative diffusion models. Pioneer diffusion models suffered from a high inference time. Recently, diffusion models are formulated using differential equations (stochastic or ordinary flow) and available numerical solvers of the respective discretized equations significantly reduced the inference time [15, 16].

In our method, we adopt the CQT-diff learned model [17] solving the following reverse ordinary flow differential equation (ODE) [15]:

$$\mathrm{d}\mathbf{x}_\tau = -\tau\, \nabla_{\mathbf{x}_\tau} \log p_\tau(\mathbf{x}_\tau)\, \mathrm{d}\tau. \tag{4}$$

The score function $\log p_\tau(\mathbf{x}_\tau)$ serves as a guide for an ODE solver and it is approximated by a trained deep denoiser $D_\theta(\mathbf{x}_\tau, \tau)$ such that

$$\nabla_{\mathbf{x}_\tau} \log p_\tau(\mathbf{x}_\tau) \approx (D_\theta(\mathbf{x}_\tau, \tau) - \mathbf{x}_\tau)/\sigma_\tau^2, \tag{5}$$

where σ_τ^2 is the noise variance at timestep τ. The schedule of a total of K variances is prescribed for $k = 1, \dots K-1$ as

$$\sigma_k = \left(\sigma_{\max}^{\frac{1}{\rho}} + \tfrac{k}{K-1}\left(\sigma_{\min}^{\frac{1}{\rho}} - \sigma_{\max}^{\frac{1}{\rho}}\right)\right)^{\rho}, \tag{6}$$

and $\sigma_K = 0$. Parameters $\sigma_{\min}, \sigma_{\max}$ are boundaries for the sampler and $\rho > 0$ is affecting the warping of the schedule.

2.3. HQS algorithm

One of the efficient ways how to numerically solve problem (3) is to use the half quadratic splitting (HQS) algorithm. This approach introduces a slack variable, in effect leading to iterative optimization of the first and the second term in (3) separately. Therefore, each HQS iteration contains two distinct principal steps, one of which can be interpreted as a denoiser [14], which is crucial for our considerations. Despite the trivial use of the algorithm, in practice only a proper setting of the HQS hyperparameters leads to a good performance. The choice of hyperparameters will be discussed in Section 3. We refer the reader to [13, 14] for further details about the HQS.

3. PROPOSED METHOD

The proposed plug-and-play method for audio restoration is presented as algorithm 1. The loop gradually solves the problem by applying two steps arising from the HQS idea:

The step at line 4 corresponds to the data-fidelity part of (3). For some specific inverse problems, this step has a closed-form solution; for instance, if $\mathcal{A}(\mathbf{x}) = \mathbf{m} \odot \mathbf{x}$, i.e., $\mathcal{A}$ is a (linear) subsampling operator determined by the binary mask $\mathbf{m}$, we have

$$\mathbf{x}_{k-1} = \frac{\mathbf{m} \odot \mathbf{y} + \alpha_k \mathbf{z}_k}{\mathbf{m} + \alpha_k}. \tag{7}$$

All operations in (7) are element-wise.

The denoiser in step 5 corresponds to the prior part of (3). The denoising consists of one reverse diffusion step with the scheduled σ_k, involving $D_\theta(\mathbf{x}_\tau, \tau)$ from (5). We use the sampler as a black-box denoiser and suppose that it serves well with predefined values from [17] in an unconditional manner. The utilization of the HQS in a plug-and-play scheme allows to use step 5 fixed across various reconstruction tasks, where only the data-fidelity part changes.

Algorithm 1 Plug-and-Play Audio Restoration

Require: θ, K, $\mathbf{y}$, σ_{audio}, $\{\sigma_k\}_{k=1}^K$, λ
1: Initialize $\mathbf{z}_K \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$,
2: pre-calculate $\alpha_k = \lambda\sigma_{\text{audio}}^2/\sigma_k^2$ for $k = K, \dots, 1$
3: **for** $k = K$ **to** 1 **do**
4: $\quad \mathbf{x}_{k-1} = \arg\min_{\mathbf{x}} \|\mathbf{y} - \mathcal{A}(\mathbf{x})\|^2 + \alpha_k\|\mathbf{x} - \mathbf{z}_k\|^2$
5: $\quad \mathbf{z}_{k-1} = Denoise_\theta(\mathbf{x}_{k-1}, \sigma_k)$
6: **end for**
7: **return** $\mathbf{z}_0$

It should be emphasized that the parameter α_k is the most crucial coefficient for the success of the algorithm. We have predefined σ_k by schedule (6). Therefore, it remains to look only for two parameters σ_{audio} and λ. Parameter σ_{audio} represents the noise variance in audio. It is defined easily when we use Gaussian noise, but with natural noise it is hard to

estimate. The latter parameter $\lambda > 0$ serves as a guidance parameter and affects whether the restoration ends with a reasonable result. In Fig. 1 we see the different behaviour for the mixture coefficients of (7), if we separate $\mathbf{y}$ (fidelity part) and $\mathbf{z}_k$ (generative part) with respect to changing λ.

In our ablation study we found the best setting of λ to reside in the range from 3 to 7. For the test, the variance σ_{audio} is set in the range 10^{-4} to 10^{-2} and λ from 1 to 10. Parameter λ achieving the highest signal-to-noise ratio (SNR) between the restored and clean signals was chosen for the experiments. This decision was also confirmed by the informal listening test. A higher λ leads to inconsistent solutions and smaller λ has not reached the solution.

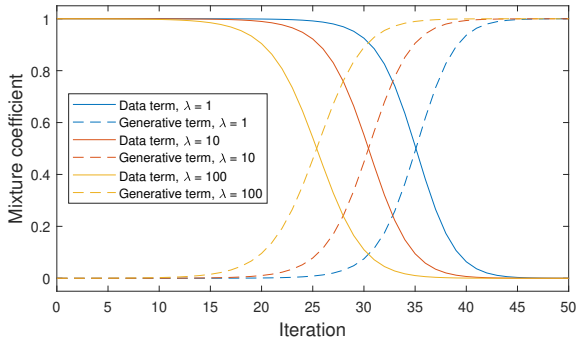

Fig. 1. Characteristics of mixtures depending on the choice of λ. We fixed $\sigma_{\text{audio}} = 0.01$ for the sake of this plot.

4. EXPERIMENTS

We describe the setup for denoising, declipping and compressive sensing in the following subsections. We use the sampler as in [17] which has been trained on the MAESTRO dataset [18] that was resampled to 22.05 kHz. We test the algorithm for fixed number of stochastic diffential equation (SDEs) solver iterations $K = 50$, $\sigma_{\max} = 10$, $\sigma_{\min} = 10^{-6}$, $\rho = 13$. All experiments are available on `https://michalsvento.github.io/PAR/`. Computational demands of the algorithm are discussed in Sec. 4.5.

4.1. Denoising

Noise reduction is probably the most widespread problem in signal processing. Neural networks have already been proposed for this task, see for example [19], [20]. Returning to (1), the signal degradation is due to noise only, i.e., $\mathcal{A}$ is the identity operator, and therefore $\mathbf{m}$ is a vector of all ones.

We test the denoising performance on a single signal from the MAESTRO dataset. We add Gaussian noise to this signal with variance σ_{audio}, where the variance is calculated so as to reach desired SNRs: -20, -10, -5, 0, 1, 3, 5, 7, 10, 15, 20 dB. The respective value of σ_{audio} is then used for the determination of α_k. We show the results of the objective metrics

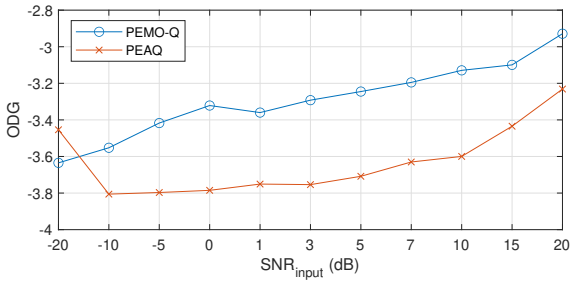

Fig. 2. Denoising experiment: the plot shows the objective difference grade (ODG) of PEMO-Q and PEAQ for a single input signal across the input SNRs.

in Fig. 2. The results show that objective metrics rise with descending level of noise as expected.

4.2. Declipping

The nonlinear, symmetric hard-clipping operator is defined as $\mathcal{A}(\mathbf{x}) = (|\mathbf{x}+c| - |\mathbf{x}-c|)/2$, where c is the clipping threshold. In this case, the inverse problem (coined declipping) pursues to reconstruct the degraded samples outside the range $[-c, c]$. We use the vector of ones as the mask $\mathbf{m}$ in (3).

We compare our method with results from paper [5] on the identical 149 excerpts from the MAESTRO dataset [18]. Our method is also compared with popular declipping algorithms A/S-SPADE [21] and SS-PEW [22]. We test the dataset in two settings: high degradation (where the undamaged data is clipped to satisfy an input signal-to-distortion ratio (SDR) of 1 dB) and mild degradation (10 dB).

First, we approach declipping as simple denoising, i.e., (7) is used with $\mathbf{m}$ being a vector of all ones. Our second approach properly finds the solution of step 4 with $\mathcal{A}$ defined above: Since clipping is an elementwise operation, it is sufficient to solve the optimization problem

$$\arg\min_x (y - \mathcal{A}(x))^2 + \alpha_k (x - z_k)^2 \tag{8}$$

for scalars x, y and z_k. It can be shown that the minimizer of (8) is one of the following candidates:

$$\begin{aligned} x_1 &= \min\{z_k, -c\} \\ x_2 &= \max\{-c, \min\{\tfrac{y+\alpha_k z_k}{1+\alpha_k}, c\}\} \\ x_3 &= \max\{c, z_k\}. \end{aligned}$$

The minimizer can be decided by simply testing the value of (8) for the three candidates.

The results are presented in Table 1, utilizing the PEMO-Q [23] and the perceptual evaluation of audio quality (PEAQ) [24] metrics. Table 1 shows that our plug-and-play method in both variants is competitive with the state-of-the-art methods in the low input SDR regime, but does not perform as good

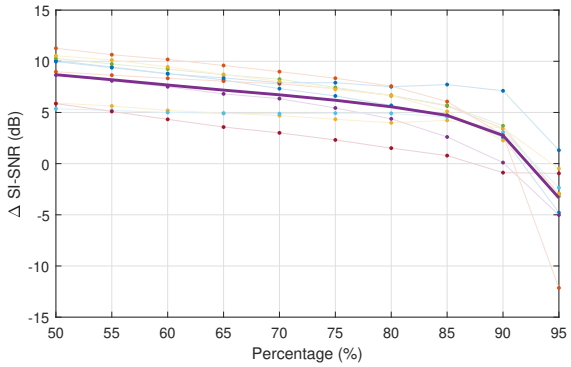

Fig. 3. The mean (bold line) and ten realizations of compressive sensing experiment showing ΔSI-SNR metric.

with mild declipping setting. The best result is marked bold and the second best is underlined.

The simplified version PAR1, pretending that $\mathcal{A}$ is linear, performed better in the objective tests. On the other hand, PAR2, leading to the exact solution to problem (8), often resulted in samples exceeding the digital levels of ± 1, causing new clipping. This negatively affected the objective results.

Table 1. Objective metrics for audio declipping tests.

	Input SDR $= 1$ dB		Input SDR $= 10$ dB	
	PEAQ ↑	PEMO-Q ↑	PEAQ ↑	PEMO-Q ↑
Clipped	-3.88	-3.82	-3.87	-3.78
CQT-Diff (RG) [5]	$\mathbf{-3.53}$	$\mathbf{-3.36}$	-2.99	$\underline{-2.56}$
A-SPADE [25]	-3.87	-3.58	$\underline{-2.97}$	-2.60
S-SPADE [21, 26]	-3.89	-3.64	-3.53	-3.34
SS-PEW [22]	-3.78	-3.62	$\mathbf{-2.05}$	$\mathbf{-0.93}$
PAR1 (Ours)	$\underline{-3.55}$	$\underline{-3.40}$	-3.32	-3.09
PAR2 (Ours)	-3.90	-3.66	-3.89	-3.76

4.3. Compressive sensing

In a narrowed meaning, the compressive sensing (CS) task is a variation on the audio inpainting problem; in CS, the missing samples are at random positions, or in other words, the missing samples do not form compact gaps. The mask $\mathbf{m}$ here is defined naturally as follows: it has 0 value in place where samples are missing and 1 at the reliable positions.

We test 10 audio files randomly chosen from the MAESTRO test set [18]. The ratios of missing samples under test ranged from 50 to 95 % with a predefined seed to corrupt the same sample positions for all signals. The results are in Fig. 3, where we show the differences in terms of the scale-invariant SNR (SI-SNR), which is a quality metric suitable for compression tasks [27]. We observe that the algorithm starts to generate content which is not consistent with the context from about 85 % of missing samples.

4.4. Source separator and vocoder

The greatest disadvantage of the proposed method is the limitation that the denoiser has been trained solely on piano recordings. On the other hand, the experiments has lead us to exciting results when we test our algorithm on partially or fully non-piano recording.

In case of reducing noise in a multiinstrument recording, the algorithm works as piano separator (i.e., other instruments are considered noise). With non-piano instruments, the method works as a kind of piano vocoder. The model is trained to generate piano recordings, so if we want to reconstruct a non-piano recording, the model attempts to generate it as if it were a piano piece.

4.5. Computational demands

Regarding the computational time of the iterative HQS scheme in Alg. 1, the vast majority of time in taken by step 5 which is the evaluation of the neural network. Thanks to no use of guidance in the diffusion model, our proposed algorithm has significantly shorter inference time. On a 8 second long audio recording, 50 iterations of HQS lasts approximately 45 seconds, which is significantly less than 170 seconds required by CQT-diff.

5. CONCLUSION

In this paper we adapt the concept of plug-and-play to audio restoration. Our framework shows good perceptual results for declipping, denoising and compressive sensing. We used an unconditional sampler as the denoiser, which leads to a faster inference with good subjective results. However, the objective metrics did not reach the expected values mainly due to no conditioning of the diffusion model which has the strongest effect in last iterations and create new samples, which reasonably do not outperform the competitors in terms of the objective metrics.

6. REFERENCES

[1] Ondřej Mokrý, Pavel Rajmic, and Pavel Záviška, "Flexible framework for audio reconstruction," in *Proceedings of the 23rd International Conference on Digital Audio Effects (DAFx2020)*, 2020–21, vol. 1.

[2] Arun A. Nair and Kazuhito Koishida, "Cascaded time + time-frequency UNet for speech enhancement: Jointly addressing clipping, codec distortions, and gaps," in *2021 IEEE ICASSP*, 2021, pp. 7153–7157.

[3] Aditya Raikar, Sourya Basu, and Rajesh M. Hegde, "Single channel joint speech dereverberation and denoising using deep priors," in *2018 IEEE GlobalSIP*, pp. 216–220.

[4] Jean-Marie Lemercier, Julius Richter, Simon Welker, Eloi Moliner, Vesa Välimäki, and Timo Gerkmann, "Diffusion models for audio restoration," 2024. https://arxiv.org/abs/2402.09821

[5] Eloi Moliner, Jaakko Lehtinen, and Vesa Välimäki, "Solving audio inverse problems with a diffusion model," in *2023 IEEE Intl. Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2023.

[6] Robin Rombach et al., "High-resolution image synthesis with latent diffusion models," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2022.

[7] Yixin Liu et al. "Sora: A review on background, technology, limitations, and opportunities of large vision models," 2024. https://arxiv.org/abs/2402.17177

[8] Carlos Hernandez-Olivan, Koichi Saito, Naoki Murata, Chieh-Hsin Lai, Marco A. Martínez-Ramírez, Wei-Hsiang Liao, and Yuki Mitsufuji, "VRDMG: Vocal restoration via diffusion posterior sampling with multiple guidance," in *IEEE ICASSP*, 2024, pp. 596–600.

[9] Bahjat Kawar, Michael Elad, Stefano Ermon, and Jiaming Song, "Denoising diffusion restoration models," in *Advances in Neural Inf. Proc. Systems*, 2022.

[10] Andreas Lugmayr et al. "Repaint: Inpainting using denoising diffusion probabilistic models," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2022.

[11] Morteza Mardani, Jiaming Song, Jan Kautz, and Arash Vahdat, "A variational perspective on solving inverse problems with diffusion models," *arXiv preprint arXiv:2305.04391*, 2023.

[12] Hyungjin Chung et al., "Diffusion posterior sampling for general noisy inverse problems," in *The 11th Intl. Conference on Learning Representations*, 2023.

[13] D. Geman and Chengda Yang, "Nonlinear image recovery with half-quadratic regularization," *IEEE Transactions on Image Processing*, vol. 4, no. 7, 1995.

[14] Kai Zhang, Yawei Li, Wangmeng Zuo, Lei Zhang, Luc Van Gool, and Radu Timofte, "Plug-and-Play Image Restoration with Deep Denoiser Prior," 2021.

[15] Tero Karras, Miika Aittala, Timo Aila, and Samuli Laine, "Elucidating the design space of diffusion-based generative models," in *Proc. NeurIPS*, 2022.

[16] Yang Song et al., "Score-based generative modeling through stochastic differential equations," in *International Conference on Learning Representations*, 2021.

[17] Eloi Moliner, Maija Turunen, Filip Elvander, and Vesa Välimäki, "A diffusion-based generative equalizer for music restoration," 2024.

[18] Curtis Hawthorne et al. "Enabling factorized piano music modeling and generation with the MAESTRO dataset," in *International Conference on Learning Representations*, 2019.

[19] Eloi Moliner and Vesa Välimäki, "A Two-Stage U-Net for High-Fidelity Denoising of Historical Recordings," in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2022.

[20] Christian J. Steinmetz, Thomas Walther, and Joshua D. Reiss, "High-Fidelity Noise Reduction with Differentiable Signal Processing," Tech. Rep., Oct. 2023, arXiv:2310.11364.

[21] Pavel Záviška, Pavel Rajmic, Ondřej Mokrý, and Zdeněk Průša, "A proper version of synthesis-based sparse audio declipper," in *2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, Brighton, United Kingdom, 2019.

[22] M. Kowalski, K. Siedenburg, and M. Dörfler, "Social sparsity! Neighborhood systems enrich structured shrinkage operators," *Signal Processing, IEEE Transactions on*, vol. 61, no. 10, pp. 2498–2511, 2013.

[23] R. Huber and B. Kollmeier, "PEMO-Q—A new method for objective audio quality assessment using a model of auditory perception," *IEEE Trans. Audio Speech Language Proc.*, vol. 14, no. 6, pp. 1902–1911, Nov. 2006.

[24] P. Kabal, "An examination and interpretation of ITU-R BS.1387: Perceptual evaluation of audio quality," Tech. Rep., MMSP Lab Technical Report, Dept. Electrical & Computer Engineering, McGill University, May 2002.

[25] Srđan Kitić, Nancy Bertin, and Rémi Gribonval, "Sparsity and cosparsity for audio declipping: a flexible non-convex approach," in *LVA/ICA 2015 – The 12th International Conference on Latent Variable Analysis and Signal Separation*, Czech Republic, 2015, pp. 243–250.

[26] Pavel Záviška, Ondřej Mokrý, and Pavel Rajmic, "S-SPADE Done Right: Detailed Study of the Sparse Audio Declipper Algorithms," Tech. Rep., Brno University of Technology, Sept. 2018. https://arxiv.org/pdf/1809.09847

[27] Jonathan Le Roux, Scott Wisdom, Hakan Erdogan, and John R. Hershey, "SDR – half-baked or well done?," in *2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2019, pp. 626–630.