Introducing SPAIN (SParse Audio INpainter)

Ondřej Mokrý[*,†]   *Pavel Záviška*[*]   *Pavel Rajmic*[*]   *Vítězslav Veselý*[†]

[*] Signal Processing Laboratory, Brno University of Technology, Brno, Czech Republic
[†] Faculty of Mechanical Engineering, Brno University of Technology, Brno, Czech Republic
Email: {170583, xzavis01, rajmic, vitezslav.vesely}@vutbr.cz

Abstract—A novel sparsity-based algorithm for audio inpainting is proposed. It is an adaptation of the SPADE algorithm by Kitić et al., originally developed for audio declipping, to the task of audio inpainting. The new SPAIN (SParse Audio INpainter) comes in synthesis and analysis variants. Experiments show that both A-SPAIN and S-SPAIN outperform other sparsity-based inpainting algorithms. Moreover, A-SPAIN performs on a par with the state-of-the-art method based on linear prediction in terms of the SNR, and, for larger gaps, SPAIN is even slightly better in terms of the PEMO-Q psychoacoustic criterion.

Index Terms—Inpainting, Sparse, Cosparse, Synthesis, Analysis

I. INTRODUCTION

The term "inpainting" has spread to the audio processing community from the image processing field, when a paper by Adler et al. was published, entitled simply "Audio inpainting" [1]. The goal of so termed restoration task is to fill degraded or missing parts of data, based on its reliable part, and preferably in an unobtrusive way. Although the term "inpainting" is rather new in this context, the problem itself is much older and different approaches have been proposed to deal with it in past decades.

One of the first successful approaches was the time-domain interpolation of the missing audio samples. An adaptive interpolation method was presented by Janssen et al. in [2], which was built on modeling the signal as an autoregressive (AR) process. The method can be roughly described as the minimization of a suitable functional, which includes as the variables both the estimates of the missing samples and the AR coefficients of the restored signal. Although the algorithm is designed as iterative, only a single iteration was considered in [2]. On today's hardware, nonetheless, hundreds of iterations can be computed in a reasonably short time. Doing this improves the results substantially and keeps the Janssen algorithm the state-of-the-art in audio inpainting in terms of the SNR. Note, however, that when applied to compact gaps, the AR-modeling suggests that this method is successful for sounds that contain harmonics which do not evolve within the gap. As such, this approach is beneficial for gaps of up to ca 45 milliseconds, which can be observed also from the experiments described below. Various prediction methods were also presented in [3].

A different approach is the sparsity-based audio inpainting, reported more recently in [1]. It benefits from the observation that the energy of an audio signal is often concentrated in a relatively small number of coefficients, with respect to a suitable, redundant time-frequency transform. The goal is to find a signal that has sparse representation under a transform such as the STFT (Short-Time Fourier Transform, aka the Discrete Gabor Transform, DGT), while it belongs to the set of feasible solutions; for audio inpainting, a solution is feasible if it is identical to the reliable parts of the signal.

The above requirements can be formulated as an optimization task. Its solution cannot be attained in practice due to the NP-hardness, but the solution can be reasonably approximated, for example using the Orthogonal Matching Pursuit (OMP) as in [1]. Another kind of approximation uses the ℓ_1 norm, which leads to a convenient convex minimization [4]. One of the modern approaches is also to search for sparsity in a non-local way [5], or using statistical prior information about the sparse representation [6].

Further methods have been developed to deal with long missing segments of audio signal. Stationary signals can be restored using sinusoidal modeling [7] or via model-based interpolation schemes [8]. In repetitive signals such as rock and pop music, there is usually an intact part sufficiently similar to the distorted part; this fact is utilized for filling the missing data in [9]. A different method, still based on self-similarity, was introduced in [10], aiming originally at the packet loss concealment. One of the latest methods employs a deep neural network for the task of audio inpainting [11].

The algorithm presented in the paper is inspired by the state-of-the-art among sparsity-based approaches to audio declipping, the so-called SParse Audio DEclipper (SPADE) [12], [13] and it is in line with the universal framework presented in [14]. We benefit from the close relation between the sparsity-based inpainting and declipping problems. We formulate the two variants of the inpainting problem as the optimization tasks

$$\min_{\mathbf{x},\mathbf{z}} \|\mathbf{z}\|_0 \quad \text{s.\,t.} \quad \mathbf{x} \in \Gamma \ \text{and} \ \|A\mathbf{x} - \mathbf{z}\|_2 \leq \varepsilon, \qquad (1a)$$

$$\min_{\mathbf{x},\mathbf{z}} \|\mathbf{z}\|_0 \quad \text{s.\,t.} \quad \mathbf{x} \in \Gamma \ \text{and} \ \|\mathbf{x} - D\mathbf{z}\|_2 \leq \varepsilon, \qquad (1b)$$

where (1a) and (1b) present the formulation referred to as the analysis and the synthesis variant, respectively. In both formulations, $\Gamma = \Gamma(\mathbf{y}) \subset \mathbb{C}^N$ is the set of feasible solutions (i.e. the set of time-domain signals that are equal to the

observed signal $\mathbf{y}$ in its reliable parts), $A : \mathbb{C}^N \rightarrow \mathbb{C}^P$ is the analysis operator of a Parseval tight frame and $D = A^*$ is its synthesis counterpart, with $P \geq N$ [15]. The formulation (1) reflects the goal to find a signal from Γ with the highest-sparsity representation, as was informally stated above.

The intention is to make use of the Alternating Direction Method of Multipliers (ADMM) for the optimization of the above non-convex problems, as was the case in the original SPADE paper [12]. For this purpose, a more appropriate formulation is needed. We introduce the (unknown) sparsity k, which is to be minimized, and indicator functions[1] of two sets: the set of feasible solutions Γ and the set of k-sparse vectors, the latter denoted $\ell_0 \leq k$ for short. This leads to

$$\min_{\mathbf{x},\mathbf{z},k}\{\iota_\Gamma(\mathbf{x}) + \iota_{\ell_0\leq k}(\mathbf{z})\} \quad \text{s.t.} \quad \begin{cases} A\mathbf{x} - \mathbf{z} = \mathbf{0}, \\ \mathbf{x} - D\mathbf{z} = \mathbf{0}. \end{cases} \quad (2)$$

Note that in (2), a more conservative constraint binding the variables $\mathbf{x}$ and $\mathbf{z}$ is used compared to (1). The reason is that such a constraint falls within the general ADMM scheme [17]. Such an alteration is nevertheless legitimate, since in the resulting iterative algorithm presented below, the stopping criterion will in effect relax the constraint into the form of (1).

II. SPAIN (SPARSE AUDIO INPAINTER)

In [12], the SPADE algorithm was introduced to tackle (2). It is built on the idea that for a fixed k, problem (2) can be approximately solved by the ADMM. The SPADE increases the value of k during the iterations of ADMM (starting from a sufficiently small value), until the condition $\|A\mathbf{x} - \mathbf{z}\|_2 \leq \varepsilon$ (analysis approach, A-SPADE) or $\|\mathbf{x} - D\mathbf{z}\|_2 \leq \varepsilon$ (synthesis approach, S-SPADE) is met for the chosen tolerance $\varepsilon > 0$. This ensures k-sparsity of the signal coefficients during iterations and the algorithm thus provides an approximation of the solution to (1).

Section II-A shows that A-SPAIN can be derived from A-SPADE. In Subsection II-B, on the other hand, S-SPAIN is introduced as a brand new inpainting algorithm. The derivation is in line with the new variant of S-SPADE for declipping, proposed in [18].

A. A-SPAIN derived from A-SPADE

The A-SPADE algorithm (Alg. 1) was originally designed for the declipping task, where the set of feasible solutions Γ is the (convex) set of time-domain signals whose samples are identical to the observed ones in the reliable part, while the restored samples are required to lie above or below the upper or the lower clipping thresholds, respectively; see [12] or [13] for more details.

The key observation leading to SPAIN is that the formulations of inpainting and declipping differ *only by the definition of the set of feasible solutions* Γ, which is actually less restrictive for inpainting, since it does not involve any requirements on the samples at unreliable positions. Alg. 1 is

---

[1] The indicator function of a set Ω, denoted $\iota_\Omega(\mathbf{x})$, attains the value 0 for $\mathbf{x} \in \Omega$ and ∞ otherwise [16].

formulated general enough to cover both the A-SPADE and the new A-SPAIN. In effect, the actual task being solved is a matter of the projection on line 3 of the algorithm.

The operator $\mathcal{H}_k$ used in the algorithm denotes hard thresholding, which sets all but k largest elements of the argument to zero. Note that in our implementation, the structures of D and A and the fact that a real signal is being processed are taken into account, leading to thresholding complex-conjugate pairs of coefficients at a time instead of thresholding of a single vector element.

Although the analysis version of the SPADE algorithm was derived from ADMM in [12] based on the formulation (2), the synthesis variant was surprisingly built on a different basis (see the technical report [19] for further explanation). A more consistent approach is to derive the synthesis variant in analogy to the analysis variant. Thus, in the following subsection, the novel derivation of S-SPAIN from ADMM is briefly described.

B. S-SPAIN derived from ADMM

The ADMM is a scheme to solve optimization problems of the form

$$\min_{\mathbf{z}} f(\mathbf{z}) + g(D\mathbf{z}), \quad (3)$$

where D is a linear operator. The functions f, g are assumed to be real and convex. Problem (3) can be reformulated by introducing a slack variable $\mathbf{x}$ such that $\mathbf{x} = D\mathbf{z}$, leading to

$$\min_{\mathbf{z},\mathbf{x}} f(\mathbf{z}) + g(\mathbf{x}) \quad \text{s.t.} \quad \mathbf{x} - D\mathbf{z} = \mathbf{0}, \quad (4)$$

which corresponds to the synthesis variant of (2). Next, the Augmented Lagrangian is formed as

$$L_\rho(\mathbf{x},\mathbf{z},\boldsymbol{\lambda}) = f(\mathbf{x}) + g(\mathbf{z}) + \boldsymbol{\lambda}^\top(\mathbf{x} - D\mathbf{z}) + \frac{\rho}{2}\|\mathbf{x} - D\mathbf{z}\|_2^2, \quad (5)$$

where $\rho > 0$ is called the penalty parameter and $\boldsymbol{\lambda} \in \mathbb{R}^N$ is the dual variable. The general scheme of ADMM then consists of three steps—minimization of L_ρ over $\mathbf{z}$, minimization of L_ρ over $\mathbf{x}$ and the update of the dual variable $\boldsymbol{\lambda}$.

For the purpose of audio inpainting, we set $f(\mathbf{z}) = \iota_{\ell_0\leq k}(\mathbf{z})$ and $g(\mathbf{x}) = \iota_\Gamma(\mathbf{x})$. Note that such a function f is not convex, therefore the conditions of ADMM are not met. Nevertheless, ADMM may converge even in such a case and the experiments show that it provides reasonable results for audio inpainting.

It is convenient to introduce the so-called scaled dual variable $\mathbf{u} = \boldsymbol{\lambda}/\rho$ at this moment. After this, it is straightforward to arrive at the ADMM steps for S-SPAIN in the following form:

$$\mathbf{z}^{(i+1)} = \arg\min_{\mathbf{z}} \|D\mathbf{z} - \mathbf{x}^{(i)} + \mathbf{u}^{(i)}\|_2^2 \text{ s.t. } \|\mathbf{z}\|_0 \leq k, \quad (6a)$$

$$\mathbf{x}^{(i+1)} = \arg\min_{\mathbf{x}} \|D\mathbf{z}^{(i+1)} - \mathbf{x} + \mathbf{u}^{(i)}\|_2^2 \text{ s.t. } \mathbf{x} \in \Gamma, \quad (6b)$$

$$\mathbf{u}^{(i+1)} = \mathbf{u}^{(i)} + D\mathbf{z}^{(i+1)} - \mathbf{x}^{(i+1)}. \quad (6c)$$

In the convex case, the minimizations over $\mathbf{x}$ and over $\mathbf{z}$ can be switched without affecting the convergence of ADMM [17]. We assume that the convergence will not be violated in the non-convex case either.

Algorithm 1: A-SPADE / A-SPAIN, depending on the particular choice of Γ

Require: $A, \mathbf{y}, \Gamma, s, r, \varepsilon$

1   $\hat{\mathbf{x}}^{(0)} = \mathbf{y}, \mathbf{u}^{(0)} = \mathbf{0}, i = 0, k = s$
2   $\bar{\mathbf{z}}^{(i+1)} = \mathcal{H}_k\big(A\hat{\mathbf{x}}^{(i)} + \mathbf{u}^{(i)}\big)$
3   $\hat{\mathbf{x}}^{(i+1)} = \arg\min_{\mathbf{x}} \|A\mathbf{x} - \bar{\mathbf{z}}^{(i+1)} + \mathbf{u}^{(i)}\|_2^2$ s.t. $\mathbf{x} \in \Gamma$
4   **if** $\|A\hat{\mathbf{x}}^{(i+1)} - \bar{\mathbf{z}}^{(i+1)}\|_2 \leq \varepsilon$ **then**
5   terminate
6   **else**
7     $\mathbf{u}^{(i+1)} = \mathbf{u}^{(i)} + A\hat{\mathbf{x}}^{(i+1)} - \bar{\mathbf{z}}^{(i+1)}$
8     $i \leftarrow i + 1$
9   **if** $i \bmod r = 0$ **then**
10      $k \leftarrow k + s$
11   **end**
12   go to 2
13 **end**
14 **return** $\hat{\mathbf{x}} = \hat{\mathbf{x}}^{(i+1)}$

Algorithm 2: S-SPAIN, task in step 2 is to be approximated either by the hard thresholding or OMP

Require: $D, \mathbf{y}, \Gamma, s, r, \varepsilon$

1   $\hat{\mathbf{x}}^{(0)} = D^*\mathbf{y}, \mathbf{u}^{(0)} = \mathbf{0}, i = 0, k = s$
2   $\bar{\mathbf{z}}^{(i+1)} = \arg\min_{\mathbf{z}} \|D\mathbf{z} - \hat{\mathbf{x}}^{(i)} + \mathbf{u}^{(i)}\|_2^2$ s.t. $\|\mathbf{z}\|_0 \leq k$
3   $\hat{\mathbf{x}}^{(i+1)} = \arg\min_{\mathbf{x}} \|D\bar{\mathbf{z}}^{(i+1)} - \mathbf{x} + \mathbf{u}^{(i)}\|_2^2$ s.t. $\mathbf{x} \in \Gamma$
4   **if** $\|D\bar{\mathbf{z}}^{(i+1)} - \hat{\mathbf{x}}^{(i+1)}\|_2 \leq \varepsilon$ **then**
5   terminate
6   **else**
7     $\mathbf{u}^{(i+1)} = \mathbf{u}^{(i)} + D\bar{\mathbf{z}}^{(i+1)} - \hat{\mathbf{x}}^{(i+1)}$
8     $i \leftarrow i + 1$
9   **if** $i \bmod r = 0$ **then**
10      $k \leftarrow k + s$
11   **end**
12   go to 2
13 **end**
14 **return** $\hat{\mathbf{x}} = \hat{\mathbf{x}}^{(i+1)}$

The solution to the subproblem (6b) is the projection of $(D\mathbf{z}^{(i+1)} + \mathbf{u}^{(i)})$ onto Γ, which is easy to compute in the time domain. The subproblem (6a) is more challenging—it corresponds to the sparse synthesis approximation, where the goal is to get as close as possible to the signal $(\mathbf{x}^{(i)} - \mathbf{u}^{(i)})$ by synthesis using D in such a way that the expansion coefficients are k-sparse. Such a problem is NP-hard due to the non-orthogonality of D in practice; therefore approximate solutions are enforced. One possibility is to utilize hard thresholding and compute

$$\mathbf{z}^{(i+1)} \approx \mathcal{H}_k\Big(D^*(\mathbf{x}^{(i)} - \mathbf{u}^{(i)})\Big). \tag{7}$$

In [19] we show that such an approximation is reasonably close to the solution of (6a). Note that in A-SPAIN, on the contrary, the thresholding step provides an *exact* solution of the corresponding ADMM subproblem. Alternatively, $\mathbf{z}^{(i+1)}$ can be approximated using k iterations of OMP [20], which was not considered neither in the original SPADE [12] nor the novel synthesis version [18]. Such an approach is, however, computationally much more demanding than the thresholding (since each iteration of OMP employs one synthesis and one analysis) and the experiments show that although it provides a better approximation for the solution of (6a), it surprisingly does not lead to a better result of the whole S-SPAIN algorithm.

The S-SPAIN algorithm is presented in Alg. 2. In the following section, the S-SPAIN variants using hard thresholding and OMP as an approximation of step 2 of the algorithm will be denoted as S-SPAIN H and S-SPAIN OMP, respectively.

III. EXPERIMENTS AND RESULTS

A. *Evaluation based on signal-to-noise ratio (SNR)*

For the experiment, ten music recordings sampled at 16 kHz or 44.1 kHz were used, covering different degrees of sparsity of the STFT coefficients. The main source was the signals that were examined in the papers [1], [12], [21].

In each test instance, the objective was to recover a signal containing six gaps, starting at random points in the signal such that the gaps do not overlap and are not too close to affect the restoration [22]. The gap length was chosen from the set of 10 available lengths, distributed between 5 and 50 ms.

As the competitors of SPAIN, we used the Janssen algorithm, the OMP and both the synthesis and the analysis approaches of the ℓ_1 relaxation.[2] In OMP and SPAIN, we used the overcomplete DFT with redundancy of the transform set to 2 (meaning that the number of frequency channels is twice the number of signal samples). All algorithms were applied frame-wise: the signal was windowed using the Hann window 64 ms long with a shift of 16 ms (i.e. 75% overlap), and the restored blocks were combined using the overlap-add scheme. For the ℓ_1 relaxation, the DGT was used with the same window parameters and with number of frequency channels corresponding to length of the transform window in samples. Besides this, weighted ℓ_1 relaxation was also employed.[3] As the performance measure, we used the signal-to-noise ratio (SNR), defined

$$\text{SNR}\,(\mathbf{x}, \hat{\mathbf{x}}) = 10\log_{10}\frac{\|\mathbf{x}\|_2^2}{\|\mathbf{x} - \hat{\mathbf{x}}\|_2^2} \quad [\text{dB}], \tag{8}$$

where $\hat{\mathbf{x}}$ stands for the recovered gap and $\mathbf{x}$ denotes the corresponding segment of the uncorrupted signal [1].

Fig. 1 shows the overall results; for each gap length and each algorithm, the average value of SNR was computed from all the signals and all positions of the gap. It is clear that for

---

[2]Implementation of the Janssen algorithm was taken from the Audio Inpainting Toolbox [1]. OMP was implemented using the Sparsify Toolbox [23]. For the synthesis and the analysis ℓ_1 relaxations, our own implementations of the Douglas-Rachford and the Chambolle-Pock algorithms were used, respectively.

[3]In weighted ℓ_1 relaxation, the objective function is $\|\mathbf{W} \cdot \|_1$. The diagonal matrix $\mathbf{W}$ allows us to favor chosen coefficients of the STFT when searching for the restored signal. The same weighting, i.e. according to ℓ_2 norm of truncated atoms, was proposed in [1] for OMP.
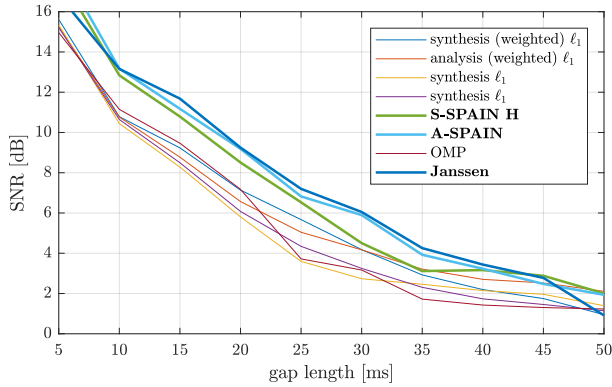
Fig. 1: Inpainting results in terms of the SNR.

Fig. 2: Comparison of S-SPAIN H and S-SPAIN OMP.

gaps of up to 40 ms, Janssen and A-SPAIN outperform all other methods and that these two algorithms behave almost identically in terms of the SNR (the biggest difference is for the longest gaps greater than 45 ms, where the performance of Janssen drops). Regarding S-SPAIN, the performance is comparable with A-SPAIN and Janssen for shorter gaps (up to 25 ms), and it outperforms the OMP and the ℓ_1-relaxation methods even for the longer gaps. A possible explanation of the superiority of A-SPAIN over S-SPAIN is that the ADMM subproblems in the analysis variant are solved exactly, whereas in the synthesis one, the thresholding operator provides just an approximation of (6a). Note that in Fig. 1, results of S-SPAIN OMP are not presented. The reason is the computational complexity of this approach (a single test instance takes up to a few days!).

The scatter plot in Fig. 2 shows detailed comparison of S-SPAIN H with S-SPAIN OMP. Due to the computational time consumed by the S-SPAIN OMP, a shortened experiment was performed with both the gap lengths and the window length shortened to their quarter compared to the first experiment (i.e. window length 16 ms, overlap 4 ms, gap lengths ranging from 1.25 to 12.5 ms). Each cross in Fig. 2 corresponds to one test instance and its coordinates are SNR for S-SPAIN H and S-SPAIN OMP. The diagonal line in the plot divides the areas in which S-SPAIN H (under the line) or S-SPAIN OMP (above the line) perform better. It is clear that except for a small area corresponding to a very low SNR, S-SPAIN H outperforms S-SPAIN OMP in majority (approx. 58 %) of cases. Also the single-tailed Wilcoxon signed rank test[4] suggests that the median of results of S-SPAIN H is greater than in case of S-SPAIN OMP with significance level 0.05. This result is quite surprising since the OMP is supposed to solve the optimization subproblem (6a) better than the simpler hard thresholding does.

For a more thorough analysis of the results, Fig. 3 provides comparison of S-SPAIN H and A-SPAIN with corresponding convex approaches, showing the bootstrap 95% confidence intervals [24] for the mean value of SNR. It can be seen from the width of the confidence intervals that the comparison with
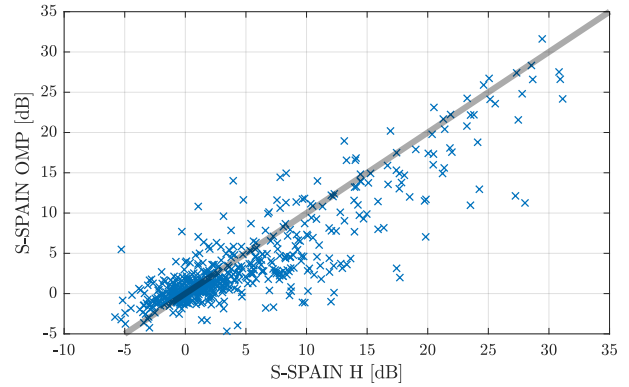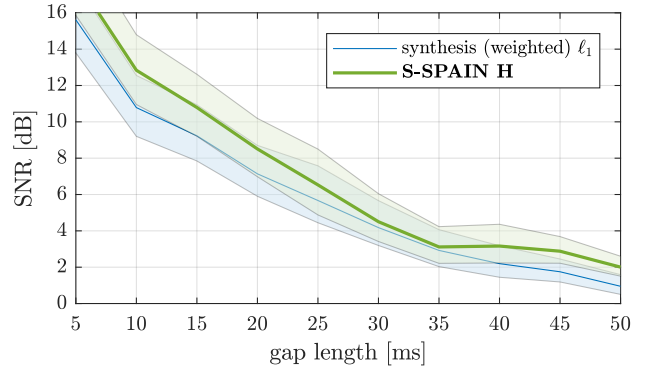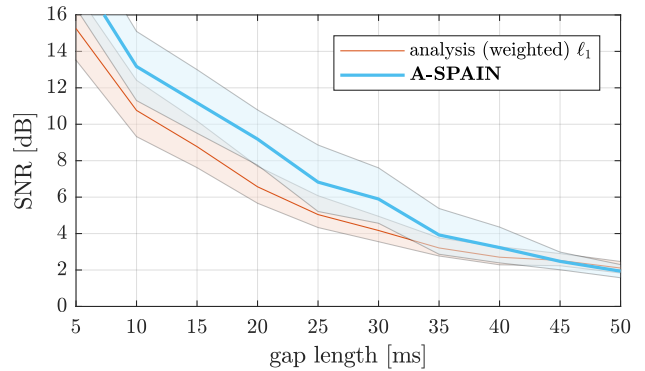
[4]Performed using function `signrank` in MATLAB.

(a) synthesis model

(b) analysis model

Fig. 3: Bootstrap 95% confidence intervals for the mean value of SNR for chosen curves from Fig. 1. The estimates with significance level 0.05 was computed using bootstrapping [24] with 10 000 random draws from the population for each combination of algorithm and gap length.

Janssen would not be statistically significant, while, on the other hand, SPAIN clearly outperforms the ℓ_1 methods in most cases.

B. Evaluation based on PEMO-Q

For further comparison of SPAIN with Janssen, we performed a second experiment, aiming at a more profound evaluation using PEMO-Q [25]. In order to correctly evaluate the restored signals with PEMO-Q, only sound excerpts sampled
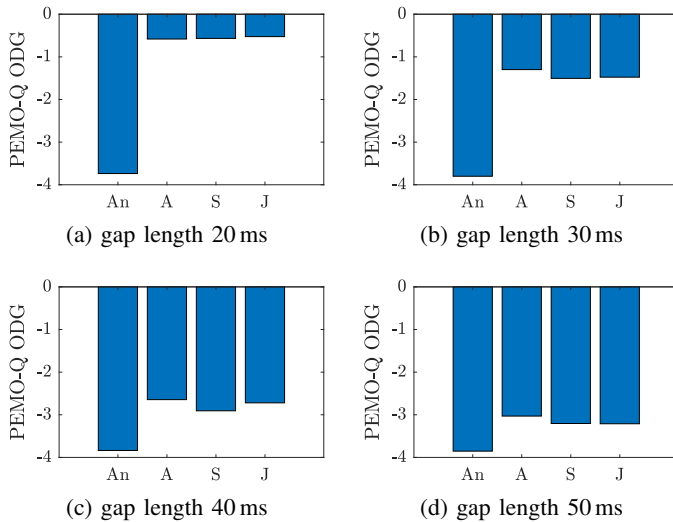
(a) gap length 20 ms

(b) gap length 30 ms

(c) gap length 40 ms

(d) gap length 50 ms

Fig. 4: Average ODG values. Showing results for the observed, degraded signal (anchor, An) and the reconstruction by A-SPAIN (A), S-SPAIN H (S) and Janssen (J).

at 44.1 kHz from the previous experiment had to be used (which makes six test signals in total). We also considered only the gaps with lengths from the set of $\{20, 30, 40, 50\}$ ms. The measured quantity was the *objective difference grade* (ODG) which simulates the human perception, comparing the restored signal to the reference (original, not degraded signal). The ODG ranges from 0 to -4 and rates the audio degradation as:

0.0	Imperceptible
−1.0	Perceptible, but not annoying
−2.0	Slightly annoying
−3.0	Annoying
−4.0	Very annoying

The results in terms of ODG are presented in Fig. 4. Each plot shows average values over the six audio samples, given the gap length. As expected, the plots indicate that the longer the gap, the worse the reconstruction. Nevertheless, the remarkable result here is that besides the gap length of 20 ms, A-SPAIN slightly outperforms both S-SPAIN H and Janssen.

IV. SOFTWARE AND DATA

The MATLAB codes for SPAIN and the sound excerpts are available at https://bit.ly/2zdhbpp.

V. CONCLUSION

The paper presented a novel inpainting algorithm (SPAIN) developed by an adaptation of successful declipping method, SPADE, to the context of inpainting. It was shown that the analysis variant of SPAIN performs the best in terms of SNR among sparsity-based methods. Furthermore, A-SPAIN was demonstrated to reach results on a par with the state-of-the-art Janssen algorithm for audio inpainting in terms of SNR. Finally, the objective test using PEMO-Q, which takes into account the human perception of sound, showed that A-SPAIN even slightly outperforms Janssen for larger gap sizes.

REFERENCES

[1] A. Adler, V. Emiya, M. Jafari, M. Elad, R. Gribonval, and M. Plumbley, "Audio Inpainting," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 20, no. 3, pp. 922–932, 2012.

[2] A. J. E. M. Janssen, R. N. J. Veldhuis, and L. B. Vries, "Adaptive interpolation of discrete-time signals that can be modeled as autoregressive processes," *IEEE Trans. Acoustics, Speech and Signal Processing*, vol. 34, no. 2, pp. 317–330, 1986.

[3] M. Fink, M. Holters, and U. Zölzer, "Comparison of various predictors for audio extrapolation," in *Proc. of the 16th Int. Conference on Digital Audio Effects (DAFx-13)*, Maynooth, 2013, pp. 1–7.

[4] F. Lieb and H.-G. Stark, "Audio inpainting: Evaluation of time-frequency representations and structured sparsity approaches," *Signal Processing*, vol. 153, pp. 291–299, 2018.

[5] I. Toumi and V. Emiya, "Sparse non-local similarity modeling for audio inpainting," in *2018 ICASSP*. IEEE, 2018.

[6] A. Adler, "Covariance-assisted matching pursuit," *IEEE Signal Processing Letters*, vol. 23, no. 1, pp. 149–153, 2016.

[7] M. Lagrange, S. Marchand, and J.-B. Rault, "Long interpolation of audio signals using linear prediction in sinusoidal modeling," *J. Audio Eng. Soc*, vol. 53, no. 10, pp. 891–905, 2005.

[8] P. A. A. Esquef, V. Välimäki, K. Roth, and I. Kauppinen, "Interpolation of long gaps in audio signals using the warped Burg's method," in *Proceedings of the 6th Int. Conference on Digital Audio Effects* (DAFx-03), 2003.

[9] N. Perraudin, N. Holighaus, P. Majdak, and P. Balazs, "Inpainting of long audio segments with similarity graphs," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, pp. 1–1, 2018.

[10] Y. Bahat, Y. Y. Schechner, and M. Elad, "Self-content-based audio inpainting," *Signal Processing*, vol. 111, no. 0, pp. 61–72, 2015.

[11] A. Marafioti, N. Perraudin, N. Holighaus, and P. Majdak, "A context encoder for audio inpainting," 2018. [Online]. Available: https://arxiv.org/pdf/1810.12138.pdf

[12] S. Kitić, N. Bertin, and R. Gribonval, "Sparsity and cosparsity for audio declipping: A flexible non-convex approach," in *LVA/ICA 2015*, Liberec, 2015.

[13] P. Záviška, P. Rajmic, Z. Průša, and V. Veselý, "Revisiting synthesis model in sparse audio declipper," in *Latent Variable Analysis and Signal Separation*. Cham: Springer International Publishing, 2018, pp. 429–445.

[14] C. Gaultier, N. Bertin, S. Kitić, and R. Gribonval, "A modeling and algorithmic framework for (non)social (co)sparse audio restoration," 2017. [Online]. Available: https://arxiv.org/abs/1711.11259

[15] O. Christensen, *Frames and Bases, An Introductory Course*, Birkhäuser: Boston, 2008.

[16] P. Combettes and J. Pesquet, "Proximal splitting methods in signal processing," *Fixed-Point Algorithms for Inverse Problems in Science and Engineering*, pp. 185–212, 2011.

[17] S. P. Boyd, N. Parikh, E. Chu, B. Peleato, and J. Eckstein, "Distributed optimization and statistical learning via the alternating direction method of multipliers." *Foundations and Trends in Machine Learning*, vol. 3, no. 1, pp. 1–122, 2011.

[18] P. Záviška, P. Rajmic, O. Mokrý, and Z. Průša, "A proper version of synthesis-based sparse audio declipper," in *2019 ICASSP*, Brighton, 2019, pp. 591–595.

[19] P. Záviška, O. Mokrý, and P. Rajmic, "S-SPADE done right: Detailed study of the Sparse Audio Declipper algorithms," Brno University of Technology, techreport, 2018. [Online]. Available: https://arxiv.org/pdf/1809.09847.pdf

[20] M. Elad, *Sparse and redundant representations: From theory to applications in signal and image processing*. Springer, 2010.

[21] K. Siedenburg and M. Dörfler, "Structured sparsity for audio signals," in *DAFx-11*, 2011, pp. 23–26.

[22] P. Rajmic, H. Bartlová, Z. Průša, and N. Holighaus, "Acceleration of audio inpainting by support restriction," in *2015 ICUMT*, 2015.

[23] T. Blumensath, "Sparsify toolbox". [Online]. Available: https://www.southampton.ac.uk/engineering/about/staff/tb1m08.page#software

[24] B. Efron and R. J. Tibshirani, *An introduction to the bootstrap*. New York: Chapmann & Hall, 1994.

[25] 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, 2006.