CONSISTENT ITERATIVE HARD THRESHOLDING FOR SIGNAL DECLIPPING

S. Kitic[1], L. Jacques[1], N. Madhu[2], M. P. Hopwood[2], A. Spriet[2] and C. De Vleeschouwer[1].

[1]ELEN Departement, ICTEAM, Université catholique de Louvain, Belgium.

[2]NXP Software, Leuven, Belgium.

ABSTRACT

Clipping or saturation in audio signals is a very common problem in signal processing, for which, in the severe case, there is still no satisfactory solution. In such case, there is a tremendous loss of information, and traditional methods fail to appropriately recover the signal. We propose a novel approach for this signal restoration problem based on the framework of Iterative Hard Thresholding. This approach, which enforces the consistency of the reconstructed signal with the clipped observations, shows superior performance in comparison to the state-of-the-art declipping algorithms. This is confirmed on synthetic and on actual high-dimensional audio data processing, both on SNR and on subjective user listening evaluations.

Index Terms— Signal Clipping, Sparse Recovery, Inverse Problems, Greedy Methods, Audio Processing.

1. INTRODUCTION

Signal *clipping* is the corruption of the dynamic range of a signal, manifested as a corruption of the signal magnitude at some boundary level. This phenomenon usually occurs during the very first stages of signal recording, typically when the input range of a device is not sufficiently large as in A/D converters or when the response of a system is not linear beyond a certain level. Quite naturally, the task of "declipping" such a corrupted signal has therefore attracted a lot of research recently in the signal processing community, and in particular, in the audio restoration field – bearing in mind the sensitivity of the human ear to unnatural sound artifacts.

There exist actually two main kinds of clipping models: hard and soft clipping. The first, as illustrated in Fig. 1, simply replaces the saturated signal amplitudes by some constant saturation level, while the second, which is not treated in this paper, corresponds to a reduction of the amplitude gain beyond this level. Once a clipped audio signal is replayed, humans perceive it as an unnatural and unpleasantly distorted sound. In music, for instance, all notes of a clipped sound seem loud, because both soft and loud notes are clipped to the same level, which reduces the auditive contrast.

This paper presents a novel iterative method for signal *declipping*. As explained in Sec. 2, this is based on the premise that the underlying signal has a *sparse* structure in a convenient representation. After introducing a model for the clipping scenario in Sec. 3, Sec. 4 defines the declipping operation as an inverse problem regularized by the sparsity assumption and which stays consistent with the whole clipping process. The main interest of this formulation is to provide guidelines for developing a *consistent* variant of the Iterative Hard Thresholding [1] adapted to the non-linear clipping alteration. After a brief review of the literature in the field (Sec. 5), the efficiency of this approach is finally demonstrated on synthetic and actual audio signal restoration in comparison with state-of-the-art methods (Sec. 6).

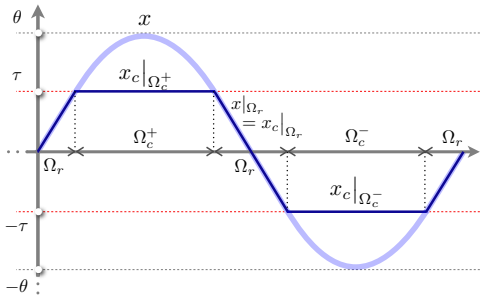

Fig. 1: Hard-clipping example: $x = x(t)$ is the original signal (light curve) and $x_c(t)$ is its clipped version (dark curve). The notations are explained in Sec. 3.

Conventions: Most of domain dimensions (*e.g.*, M, N) are denoted by capital roman letters. Vectors and matrices are associated to bold symbols while lowercase light letters are associated to scalar values. The i^{th} component of a vector $\boldsymbol{u}$ is u_i or $(\boldsymbol{u})_i$. The identity matrix is $\mathbb{I}$. Vectors of zeros and ones are denoted by $\mathbf{0}$ and $\mathbf{1}$ respectively. The set of indices in $\mathbb{R}^D$ is $[D] = \{1, \cdots, D\}$. Scalar product between two vectors $\boldsymbol{u}, \boldsymbol{v} \in \mathbb{R}^D$ reads $\boldsymbol{u}^T\boldsymbol{v} = \langle \boldsymbol{u}, \boldsymbol{v} \rangle$ (using the transposition $(\cdot)^T$). For any $p \geqslant 1$, $\|\cdot\|_p$ represents the ℓ_p-norm such that $\|\boldsymbol{u}\|_p^p = \sum_i |u_i|^p$ with $\|\boldsymbol{u}\| = \|\boldsymbol{u}\|_2$ and $\|\boldsymbol{u}\|_\infty = \max_i |u_i|$. The ℓ_0 "norm" is $\|\boldsymbol{u}\|_0 = \#\operatorname{supp}\boldsymbol{u}$, where $\#$ is the cardinality operator and $\operatorname{supp}\boldsymbol{u} = \{i : u_i \neq 0\} \subset [D]$. For $\mathcal{S} \subset [D]$, $\boldsymbol{u}|_\mathcal{S} \in \mathbb{R}^{\#\mathcal{S}}$ (or $\boldsymbol{\Phi}|_\mathcal{S}$) denotes the vector (resp. the matrix) obtained by retaining the components (resp. columns) of $\boldsymbol{u} \in \mathbb{R}^D$ (resp. $\boldsymbol{\Phi} \in \mathbb{R}^{D' \times D}$) belonging to $\mathcal{S} \subset [D]$. Alternatively, $\boldsymbol{u}|_\mathcal{S} = \boldsymbol{R}_\mathcal{S}\boldsymbol{u}$ or $\boldsymbol{\Phi}|_\mathcal{S} = \boldsymbol{\Phi}\boldsymbol{R}_\mathcal{S}^T$ where $\boldsymbol{R}_\mathcal{S} := (\mathbb{I}|_\mathcal{S})^T \in \{0,1\}^{\#\mathcal{S} \times D}$ is the *restriction* operator. We denote by $(\boldsymbol{x})_+$ the positive thresholding $(x_i)_+ = (x_i + |x_i|)/2$, while the negative counterpart reads $(\boldsymbol{x})_- = -(-\boldsymbol{x})_+$.

LJ and CDV are supported by the Belgian FRS-FNRS fund. NM, MPH and AS are supported by NXP software, Leuven. Part of this work has been funded by the SPORTIC project (WIST3), Walloon Region, Belgium.

2. SPARSE SIGNAL REPRESENTATION

The vast majority of real-life audio signals have compressible structures, meaning that these signals may be represented or approximated as the linear combination of few elements taken in a set of elementary wave forms (*e.g.*, DCT, Wavelets [2, 3]).

Mathematically, this *sparsity* concept is applied to 1-D temporal signals (*e.g.*, audio) as follows. We assume that, within a certain time window $T \subset \mathbb{R}$, a continuous signal $x(t)$ has been sampled with N regular samples gathered in a column vector $\boldsymbol{x} \in \mathbb{R}^N$. We consider then that for an appropriate sparsity basis $\boldsymbol{\Psi} \in \mathbb{R}^{N \times D}$ with $N \leqslant D$, $\boldsymbol{x}$ can be described as

$$\boldsymbol{x} \approx \boldsymbol{\Psi}\boldsymbol{\alpha}, \text{with } \|\boldsymbol{\alpha}\|_0 \ll N \text{ and } \|\boldsymbol{x} - \boldsymbol{\Psi}\boldsymbol{\alpha}\| \ll \|\boldsymbol{x}\|. \quad (1)$$

When $\boldsymbol{\Psi}$ is an orthonormal basis with $D = N$, there exists only one vector $\boldsymbol{\alpha}^*$ satisfying $\boldsymbol{x} = \boldsymbol{\Psi}\boldsymbol{\alpha}^*$. A sparse coefficient vector answering the problem (1) is found by taking the best K-term approximation of $\boldsymbol{\alpha}^*$ given a fixed sparsity level $K \ll N$. In other words, $\boldsymbol{\alpha} = \boldsymbol{\alpha}^*_K = \mathcal{H}_K(\boldsymbol{\alpha}^*)$ where $\mathcal{H}_K$ is the K-term thresholding operator setting to zero all but the K highest-magnitude coefficients of $\boldsymbol{\alpha}^*$. If $D > N$, there are many coefficient vectors whose re-synthesis with $\boldsymbol{\Psi}$ approximates $\boldsymbol{x}$. This redundancy is often useful to select sparser coefficient vector. Despite the NP-hardness of (1) [4], "relaxed" optimization methods and greedy algorithms exist in order to find such sparse vectors under additional requirement on $\boldsymbol{\Psi}$ [5, 6].

Amongst them, the Iterative Hard Thresholding (IHT) offers interesting advantages like fast convergence and provable sparse decomposition guarantees [1]. If a signal $\boldsymbol{x}$ is assumed K-sparse in $\boldsymbol{\Psi} \in \mathbb{R}^{N \times D}$ (with $K \ll N$), this algorithm is designed to find one minimizer of a Lasso-type [7] restatement of (1):

$$\hat{\boldsymbol{\alpha}} = \underset{\widetilde{\boldsymbol{\alpha}} \in \mathbb{R}^D}{\arg\min} \tfrac{1}{2}\|\boldsymbol{x} - \boldsymbol{\Psi}\widetilde{\boldsymbol{\alpha}}\|^2 \text{ s.t. } \|\widetilde{\boldsymbol{\alpha}}\|_0 \leqslant K. \quad (2)$$

IHT approximates the solution of (2) by performing the following iterative evaluation:

$$\boldsymbol{\alpha}^{(n+1)} = \mathcal{H}_K[\boldsymbol{\alpha}^{(n)} + \boldsymbol{\Psi}^T(\boldsymbol{x} - \boldsymbol{\Psi}\boldsymbol{\alpha}^{(n)})],\ \boldsymbol{\alpha}^{(0)} = \boldsymbol{0}. \quad (3)$$

In words, at each iteration, this algorithm hard-thresholds the previous solution updated by a gradient descent on the fidelity cost of (2).

The IHT procedure is very general and is also applicable to the recovery of signals indirectly observed by a sensing matrix $\boldsymbol{\Phi} \in \mathbb{R}^{M \times N}$, as in the Compressed Sensing (CS) framework [8, 9]. In such case, the algorithm above simply undergoes the replacement $\boldsymbol{\Psi} \to \boldsymbol{\Phi}\boldsymbol{\Psi} \in \mathbb{R}^{M \times D}$ for integrating this sensing.

3. CLIPPING MODEL

Assuming a symmetric clipping (as in Fig. 1) associated to a clipping threshold $\tau > 0$, the (hard) clipping operation C_τ is mathematically defined as:

$$\boldsymbol{x}_c = C_\tau(\boldsymbol{x}) := \min(|\boldsymbol{x}|, \tau)\,\mathrm{sign}(\boldsymbol{x}), \quad (4)$$

where all the operations are applied component wise on $\boldsymbol{x}$.

From this clipping operation, we can actually define different sets of samples in $\boldsymbol{x}$. The set of reliable data, those which are not subject to clipping, is $\Omega_r = \{i \in [N] : |x_i| < \tau\}$, while the clipped index set $\Omega_c = \{i \in [N] : |x_i| \geqslant \tau\}$ can be split into two disjoint subsets $\Omega_c^\pm = \{i \in [N] : \pm x_i \geqslant \tau\}$, with $\Omega_c = \Omega_c^+ \cup \Omega_c^-$ and $\Omega_r \cup \Omega_c = [N]$.

In this work, we assume that these sets are known. This happens for instance if the clipping process is hard and not corrupted by a strong noise, in which case all the previous sets can be deduced from the observation of $\boldsymbol{x}_c$.

The knowledge of the sets Ω_r and $\Omega_c^\pm$ simplifies the corresponding forward model (4), *i.e.*,

$$\boldsymbol{x}_c = \boldsymbol{M}_{\Omega_r}\boldsymbol{x} + \tau\boldsymbol{M}_{\Omega_c^+}\boldsymbol{1} - \tau\boldsymbol{M}_{\Omega_c^-}^T\boldsymbol{1}, \quad (5)$$

with $\boldsymbol{M}_{\mathcal{S}} = \boldsymbol{R}_{\mathcal{S}}^T\boldsymbol{R}_{\mathcal{S}} \in \mathbb{R}^{N \times N}$ is a diagonal masking matrix, *i.e.*, $(\boldsymbol{M}_{\mathcal{S}}\boldsymbol{u})_i = u_i$ if $i \in \mathcal{S} \subset [N]$ and 0 otherwise.

4. DECLIPPING INVERSE PROBLEM

A naive approach to the declipping problem is to use directly the IHT algorithm (3) on the partial observation of the unclipped signal samples, *i.e.*, considering $\boldsymbol{x}_c|_{\Omega_r} = \boldsymbol{R}_{\Omega_r}\boldsymbol{x}$ as the partial observations of $\boldsymbol{x}$ realized by the sensing matrix $\boldsymbol{\Phi} = \boldsymbol{R}_{\Omega_r}\boldsymbol{\Psi}$. As explained in [10, 11], the downfall of this method is that it does not take into account the information contained in the clipped samples, namely the clipping threshold (τ) and (possibly) the uppermost absolute magnitude (θ).

Therefore, in this work, inspired by the ideal objective (2) targeted by IHT, we address the following inverse problem

$$\hat{\boldsymbol{\alpha}} = \underset{\widetilde{\boldsymbol{\alpha}} \in \mathbb{R}^D}{\arg\min} \tfrac{1}{2}\|\mathcal{B}(\boldsymbol{x}_c - \boldsymbol{\Psi}\widetilde{\boldsymbol{\alpha}})\|^2 \text{ s.t. } \|\widetilde{\boldsymbol{\alpha}}\|_0 \leqslant K, \quad (6)$$

where the corresponding reconstructed signal is $\hat{\boldsymbol{x}} = \boldsymbol{\Psi}\hat{\boldsymbol{\alpha}}$.

The key function $\mathcal{B} : \mathbb{R}^N \to \mathbb{R}^N$ involved in (6) reads

$$\mathcal{B}(\boldsymbol{u}) = \boldsymbol{M}_{\Omega_r}\boldsymbol{u} + (\boldsymbol{M}_{\Omega_c^+}\boldsymbol{u})_+ + (\boldsymbol{M}_{\Omega_c^-}\boldsymbol{u})_-. \quad (7)$$

Minimizing $\mathcal{E}(\widetilde{\boldsymbol{x}}) := \frac{1}{2}\|\mathcal{B}(\boldsymbol{x}_c - \widetilde{\boldsymbol{x}})\|^2$ forces a signal candidate $\widetilde{\boldsymbol{x}} = \boldsymbol{\Psi}\widetilde{\boldsymbol{\alpha}}$ to be *consistent* with the observed clipped signal $\boldsymbol{x}_c$. Indeed, from (7), this cost can be split in three parts, *i.e.*,

$$\begin{aligned}\mathcal{E}(\widetilde{\boldsymbol{x}}) &:= \tfrac{1}{2}\|\boldsymbol{M}_{\Omega_r}(\boldsymbol{x}_c - \widetilde{\boldsymbol{x}})\|^2 \\ &+ \tfrac{1}{2}\|\boldsymbol{M}_{\Omega_c^+}(\tau\boldsymbol{1} - \widetilde{\boldsymbol{x}})_+\|^2 + \tfrac{1}{2}\|\boldsymbol{M}_{\Omega_c^-}(-\tau\boldsymbol{1} - \widetilde{\boldsymbol{x}})_-\|^2. \quad (8)\end{aligned}$$

A small first term promotes the candidate signal to match the observed clipped signal in the unclipped domain, while, thanks to the $+$ or $-$ thresholding functions, having a minimal second (or third) term enforces $\widetilde{\boldsymbol{x}}$ to be bigger (resp. smaller) than τ (resp. $-\tau$) on the set Ω_c^+ (resp. Ω_c^-).

A few remarks can be made on (6). First, this declipping program corresponds intuitively to picking from the set of all signals consistent with the clipped observation $\boldsymbol{x}_c$, one which has a sparsity smaller than K. Provided the original signal $\boldsymbol{x}$ (approximately) respects this sparsity requirement, the declipping program will have a solution.

Second, we impose actually a strict sparsity model on $\widetilde{\boldsymbol{\alpha}}$ parametrized by a sparsity order $K \ll N$ assumed optimal. We will see later how we can estimate this value. Notice also that additional constraints can be added, such as the knowledge that the original signal has a bound amplitude (see Fig. 1), *i.e.*, we can additionally impose $\|\boldsymbol{\Psi}\widetilde{\boldsymbol{\alpha}}\|_\infty \leqslant \theta$ (*e.g.*, as done in [10, 11]).

Of course, directly solving (6) is as hard as recovering a sparse signal in (2). However, a novel iterative hard thresholding adjusted to (6) can be obtained by following the same guidelines than those used to derive IHT from (2) [1]. Since $(\cdot)_+^2$ is differentiable, the cost $\mathcal{E}(\boldsymbol{\Psi\beta})$ is actually a smooth convex function of $\boldsymbol{\beta} \in \mathbb{R}^D$ whose gradient reads:

$$\boldsymbol{\nabla}_{\boldsymbol{\beta}}\, \mathcal{E}(\boldsymbol{\Psi\beta}) = -\boldsymbol{\Psi}^T \mathcal{B}(\boldsymbol{x}_c - \boldsymbol{\Psi\beta}).$$

Consequently, remembering that the internal part of the thresholding in (3) is a gradient descent, we propose to solve a version of Iterative Hard Thresholding adjusted to DeClipping that we call IHT-DC:

$$\boldsymbol{\alpha}^{(n+1)} = \mathcal{H}_K[\boldsymbol{\alpha}^{(n)} + \mu^{(n)}\, \boldsymbol{\Psi}^T \mathcal{B}(\boldsymbol{x}_c - \boldsymbol{\Psi\alpha}^{(n)})], \qquad (9)$$

where $\boldsymbol{\alpha}^{(0)} = \mathbf{0}$. The value $\mu^{(n)}$ is simply selected by a fast 1-D convex minimization (*e.g.*, using golden section [18]) of

$$g(\mu) = \mathcal{E}\big(\boldsymbol{\alpha}^{(n)} + \mu\, \boldsymbol{\Psi}^T \mathcal{B}(\boldsymbol{x}_c - \boldsymbol{\Psi\alpha}^{(n)})\big).$$

5. PRIOR WORKS IN THE FIELD

Different strategies have been developed so far in the literature to address the declipping problem. One of the oldest is the Autoregressive (AR) method [14]. AR assumes that the underlying signal can be modeled as an autoregressive process [15], and based on that premise it estimates the AR coefficients and interpolates the missing samples. AR relies thus on one specific generative signal model which is well adapted to speech signals only. It suffers a lack of flexibility for modeling and restoring other kind of signals (as music); an issue solved by sparsity-driven methods.

A more recent approach is the Constrained Orthogonal Matching Pursuit (cOMP) audio inpainting algorithm [10]. This one is fundamentally based on Orthogonal Matching Pursuit [16] and constrained optimization. In the first stage, cOMP discards non-reliable samples from the data and attempts to detect the optimal basis vectors using only reliable samples. In the second stage, clipping constraints are imposed to the chosen basis using some external optimization toolbox. Despite reported improvements relatively to the AR declipping results, one drawback of this algorithm is that the first stage does not take into account the information stored in the clipped samples. This may yield to an incorrect sparse support estimation impacting the whole method. A second drawback is that the number of iterations is directly related to the length of the estimated sparsity support, due to the fact that OMP gradually increases the support length at each iteration.

The work in [11] solves a variant of (6) where the sparsity inducing ℓ_0 "norm" is minimized and the clipping consistency is a constraint. This technique uses a reweighted ℓ_1 minimization for approaching ℓ_0 [12] and it addresses the shortfall of the cOMP. Unfortunately, we were unable to extensively run the corresponding toolbox in our high dimensional audio setting (see Sec. 6). There exist also some commercial declipping softwares like the Adobe® Audition DeClipper™. However, their black box nature make difficult any theoretical comparison with other methods.

Finally, but not directly related, clipping is associated to saturation in compressive data quantization [19] or to time-frequency signal corruption [20], while in the extreme case where $\tau \to 0^+$, the cost $\mathcal{E}$ in (8) is reminiscent of the energy implicitly minimized in the Binary Iterative Hard Thresholding (BIHT) in the context of 1-bit CS [13].

6. EXPERIMENTS

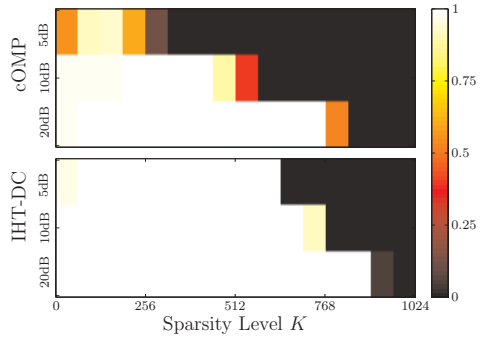

Fig. 2: Probability of success for accurate recovery under severe (5 dB), moderate (10 dB) and mild (20 dB) clipping.

Two sets of the benchmark tests have been conducted, one with synthetic signals and the other with actual audio (music) data. For both data types, signals are degraded by hard-clipping (4). The performance of the algorithms are measured in dB through input and output (restored signal) SNRs. These metrics are defined as $\text{iSNR} = 10\log_{10}(\|\boldsymbol{x}\|^2/\|\boldsymbol{x} - \boldsymbol{x}_c\|^2)$ and $\text{oSNR} = 10\log_{10}(\|\boldsymbol{x}\|^2/\|\boldsymbol{x} - \hat{\boldsymbol{x}}\|^2)$ respectively, where $\boldsymbol{x}$, $\boldsymbol{x}_c$ and $\hat{\boldsymbol{x}}$ are the original, the clipped and the declipped signals, respectively.

Benchmark on synthetic data: This type of benchmark is suitable for sparsity-based declipping algorithms such as cOMP and IHT-DC. Synthetic signals have been generated in $\mathbb{R}^{N=1024}$ as random K-sparse signals in a DCT basis $\boldsymbol{\Psi} \in \mathbb{R}^{N\times N}$ for various values of $1 \leqslant K \leqslant N$. These sparse signal were defined by first selecting uniformly at random a K-length support T in $[D]$ and by drawing identically and independently the dictionary coefficients of this support according to a normal distribution, the coefficients outside of T being set to 0.

For both reconstruction methods, the "success" of a declipped reconstruction has been arbitrarily defined as $\text{oSNR} > 80\,\text{dB}$. For different values of K and for different clipping scenarios, we estimated the probability of the

accurate recovery by the frequency of success over 100 trials. The results for 3 levels of clipping are presented in Fig. 2. These are defined as "mild" ($\mathrm{iSNR} = 20\,\mathrm{dB}$), "moderate" ($\mathrm{iSNR} = 10\,\mathrm{dB}$) and "severe" ($\mathrm{iSNR} = 5\,\mathrm{dB}$). It appears that the proposed method outperforms cOMP in all cases. The advantage of using IHT-DC is especially pronounced for the severe and moderate clipping, while it becomes smaller for the mild clipping level. This is in accordance with the fact that, at this level, more reliable observations are available and cOMP manages to detect the correct basis vectors more often. Interestingly, the global appearance of Fig. 2 is similar to the transition recovery diagram existing in sparse signal recovery in Compressed Sensing [17], the y-axis in this figure measuring implicitly the amount of reliable observations.

About the processing time, our programming of IHT-DC in Matlab© is fast when the sparsity level is smaller than the success/failure transition point reported in Fig. 2. For instance, under moderate clipping and for $K \leqslant 512$ (the transition occurring in $K \in [576, 768]$), the recovery is perfect and IHT-DC stops after less than 100 iterations, *i.e.*, less than 2s on a 2.93GHz laptop.

Benchmark on audio data: In this benchmark, we test the efficiency of our approach on two actual samples of music track, *i.e.*, "You Are My Kind" by *Seal & Santana* (15s) and "Sultans of Swing" by *Dire Straits* (10s), each one having a very different speech and tonal content. Each track is sampled at 16kHz (32bits per sample), and the declipping operation was realized by individual processing of 75%-overlapping temporal slices of length $N = 1024$, before to re-synthesize the full length music signal with a symmetric sinusoidal window [10].

Since the audio signals are not truly sparse, IHT-DC cannot be readily applied to restore them in a clipping scenario. In order to avoid us to arbitrarily set one sparsity level K, *e.g.*, according to the class of audio signals considered, we prefer to make the IHT-DC adaptive by enforcing it to "learn" an optimal sparsity level K. Selecting $\mathbf{\Psi}$ as a two-times redundant DCT dictionary for efficient audio signal representations [2], this adaptivity is achieved by gradually increasing the sparsity requirement by 1 at each iteration, starting from a low value, until the residual between the clipped observations and the reclipped reconstructed signal has a sufficiently small energy.

The *SNR gain* of this adaptive IHT-DC is then measured by the difference between oSNR and iSNR; obviously, it should be as high as possible. The values in Table 1(top) show the SNR gains for different clipping levels illustrating severe ($5\,\mathrm{dB}$), moderate ($10\,\mathrm{dB}$) and mild ($15\,\mathrm{dB}$) signal corruption, in comparison with other methods. Once again, IHT-DC outperforms all others in this benchmark. Furthermore, only Adobe software (apart from the proposed algorithm) is somewhat successful in improving the clipped signal, whereas other methods usually degrade signals even more. Overall, IHT-DC was about 5 times faster than cOMP, with ratio between processing time and track duration close to 60 and 240 for moderate and hard clipping, respectively.

SNR gain	Music 1			Music 2		
iSNR =	5 dB	10 dB	15 dB	5 dB	10 dB	15 dB
Adobe DC	2.9	4.6	5.1	0.5	1.2	2.0
AR	−1.6	−2.2	1.6	−1.2	−0.5	−0.5
IHT-DC	**4.4**	**6.5**	**7.2**	**5.4**	**5.9**	**5.4**
cOMP	−2.4	−4.5	−2.3	−2.6	−1.7	−1.2
Evaluation	5 dB	10 dB	15 dB	5 dB	10 dB	15 dB
Adobe DC	1.5	3.0	4.0	2.0	3.0	4.0
AR	1.0	3.0	4.0	1.0	3.0	4.0
Clipped	1.0	2.0	4.0	1.5	2.0	4.0
IHT-DC	**3.0**	**4.0**	4.0	**3.0**	**4.0**	4.0
cOMP	1.0	2.0	4.0	1.0	3.0	4.0

Table 1: (top) The SNR gain (in dB) *vs* clipping for the two audio tracks. (bottom) Listening evaluation (16 pers.) for the two music samples. Evaluation score are between 1 "very poor" and 5 "excellent quality".

In addition to the previous SNR gain comparisons, we also wish to depict how end-users perceive audio data processed by the different algorithms. Therefore, we have designed a randomized subjective test as follows. We have considered again the 6 clipped audio contents generated by applying our 3 clipping scenarios to the 2 music tracks. Then, 5 resulting versions for each clipped track have been collected: the reconstructions obtained for each of the 4 declipping algorithms, plus the raw clipped version. A blind test has then been run independently with 16 subjects. The 6 clipped tracks have been considered in a random order (to prevent the "training effect"). For each clipped track, the 5 versions were evaluated by the listeners in a random order (preserving objectivity). The subjects were asked to score the result between 1 (very poor) and 5 (excellent).

The results are presented at the Table 1(bottom). For the two music signals the proposed algorithm clearly outperforms the competitors. Again, it seems that only Adobe DeClipper (Adobe DC) provides some improvement in perceived quality, while cOMP and AR methods are in some cases graded lower than actual clipped signal. In case of a mild clipping, users were in most cases unable to hear the difference between audio clips.

7. CONCLUSION

A new iterative method for declipping signals has been presented. This extends the IHT algorithm by integrating a clipping consistency during the iterations. Experimental results on synthetic and actual audio data have demonstrated the efficiency of this approach compared to other known techniques. In the future, we plan to justify the theoretical conditions for guaranteeing the convergence of the IHT-DC. In particular, we will analyze how the "phase" transition diagram obtained in Fig. 2 can be predicted according to both the clipping and the signal sparsity levels. On the practical perspective, an important gain could also be obtained by exploiting different dictionaries. The redundant DCT used for our tests is a very generic dictionary that is not especially well suited for purely speech signals, for instance.

References

[1] T. Blumensath and M. E. Davies, "Iterative thresholding for sparse approximations," *Journal of Fourier Analysis and Applications*, vol. 14(5), pp. 629-654, 2008.

[2] M. D. Plumbley, T. Blumensath, L. Daudet, R. Gribonval, and M. E. Davies, "Sparse representations in audio and music: from coding to source separation," *Proceedings of the IEEE*, vol. 98(6), pp. 995–1005, 2010.

[3] S. Mallat, *A Wavelet Tour of Signal Processing, Third Edition: The Sparse Way*, Academic Press, 3 edition, Dec. 2008.

[4] B. K. Natarajan, "Sparse approximate solutions to linear systems," *SIAM Journal on Scientific Computing*, vol. 24(2), pp. 227–234, 1995.

[5] S. S. Chen, D. L. Donoho, and M. A. Saunders, "Atomic decomposition by basis pursuit," *SIAM Journal on Scientific Computing*, vol. 20(1), pp. 31–61, 1998.

[6] J. A. Tropp, "Greed is good: Algorithmic results for sparse approximation," *IEEE Transactions on Information theory*, vol. 50(10), pp. 2231–2242,2004.

[7] R. Tibshirani, "Regression shrinkage and selection via the lasso," *Journal of the Royal Statistical Society. Series B (Methodological)*, pp. 267–288, 1996.

[8] E. J. Candès and M. B. Wakin, "An introduction to compressive sampling," *IEEE Signal Processing Magazine*, vol. 25(2), pp. 21–30, 2008.

[9] T. Blumensath and M. E. Davies, "Iterative hard thresholding for compressed sensing," *Applied and Computational Harmonic Analysis*, vol. 27(3), pp. 265–274, 2009.

[10] A. Adler, V. Emiya, M. G. Jafari, M. Elad, R. Gribonval, and M.D. Plumbley, "Audio inpainting.", in *Audio, Speech, and Language Processing, IEEE Transactions on*, vol 20(3), pp. 922–932, 2012.

[11] A. J. Weinstein and M. B. Wakin, "Recovering a clipped signal in sparseland," *preprint, arXiv:1110.5063*, 2011. Toolbox available at `https://github.com/aweinstein/declipping`.

[12] E. J. Candès, M. B. Wakin, and S. Boyd "Enhancing Sparsity by Reweighted l1 Minimization" Journal of Fourier Analysis and Applications, vol. 14(5), pp. 877-905, 2008.

[13] L. Jacques, J. N. Laska, P. Boufounos and R. Baraniuk "Robust 1-bit compressive sensing via binary stable embeddings of sparse vectors" arXiv preprint arXiv:1104.3160, 2011

[14] A. Janssen, R. Veldhuis, and L. Vries, "Adaptive interpolation of discrete-time signals that can be modeled as autoregressive processes," *IEEE Transactions on Acoustics, Speech and Signal Processing*, vol. 34(2), pp. 317-330, 1986.

[15] M. H. Hayes, *Statistical Digital Signal Processing and Modeling*, Wiley, 1996.

[16] Y. C Pati, R. Rezaiifar, and P. S. Krishnaprasad, "Orthogonal matching pursuit: recursive function approximation with applications to wavelet decomposition," in *Conference Record of The Twenty-Seventh Asilomar Conference on Signals, Systems and Computers*, pp. 40–44, 1993.

[17] A. Maleki and D. L. Donoho, "Optimally tuned iterative reconstruction algorithms for compressed sensing," in *Selected Topics in Signal Processing, IEEE Journal of*, vol. 4(2), pp. 330–341, 2010.

[18] J. Kiefer, "Sequential minimax search for a maximum", in *Proceedings of the American Mathematical Society* vol. 4(3), pp. 502-506, doi:10.2307/2032161, 1953.

[19] J. N. Laska, P. T. Boufounos, M. A. Davenport, R. G. Baraniuk, "Democracy in action: Quantization, saturation, and compressive sensing", in Applied and Computational Harmonic Analysis, vol. 31(3), pp. 429–443, 2011.

[20] P. Smaragdis, R. Bhiksha, S. Madhusudana, "Missing data imputation for time-frequency representations of audio signals." in Journal of signal processing systems, vol. 65(3), pp. 361–370, 2011.