A Non Intrusive Audio Clarity Index (NIAC) and its Application to Blind Source Separation

Gaël Mahé[a,*], Giulio G. R. Suzumura[b], Lionel Moisan[c], Ricardo Suyama[b]

[a] *Université de Paris, LIPADE, F-75006 Paris, France*
[b] *Universidade Federal do ABC (UFABC), CECS, Santo André,Brazil*
[c] *Université de Paris, MAP5, CNRS UMR8145, F-75006 Paris, France*

Abstract

We propose a Non-Intrusive (or reference-free) Audio Clarity index (NIAC), inspired from previous works on image sharpness and defined as the sensitivity of the spectrogram sparsity to a convolution of the audio signal with a white noise. A closed-form formula is provided, which only involves the signal itself and very little parameter setting. Tested in various noise and reverberation conditions, the NIAC exhibits a high correlation with the well-established Speech Transmission Index, both for speech and music. It can also be used as a clarity criterion to drive sound enhancement algorithms. We propose a NIAC-based source separation algorithm, and show that its performance is comparable to that of a state-of-the-art algorithm, FastICA.

Keywords: audio clarity; sparsity; blind source separation

1. Introduction

Audio clarity can be defined as the easiness to spot individual phonemes in speech or individual notes in music [1]. Many objective measures have been proposed to predict the perceived clarity, generally specifically dedicated to music or to speech. In the latter case, clarity is generally equated to intelligibility.

For speech, a first class of methods are intrusive or full-reference methods, based on a comparison between the distorted signal and a clean signal.

*Corresponding author
Email address: gael.mahe@u-paris.fr (Gaël Mahé)

Important examples are the Speech Intelligibility Index (SII [2]) and Speech Transmission Index (STI [3]), and the recent Short-Time Objective Intelligibility (STOI [4]) and intelligibility predictor based on mutual information (SIMI [5]).

When the clean signal is not available, non-intrusive (or reference-free) measures are required. Most of them are based on machine-learning techniques and derive indicators from a large set of signal parameters by maximizing the correlation with reference indicators on a training corpus [6, 7, 8]. The drawback of this approach is that the indicators depend on the training conditions and that they are blind to the physical grounds of intelligibility. Another approach, the speech to reverberation modulation energy (SRMR) proposed by [9], stemmed from the idea that the modulation energy tends to spread towards high modulation frequencies in case of reverberation.

Less work is dedicated to music clarity, which assessment often relies on room acoustic parameters [10], especially the clarity index C_{80}, defined as the ratio between the energy within the first 80ms and the energy of the rest of the room impulse response (RIR). Recent works replace the sound pressure energies involved in this ratio by perceptually relevant quantities: nerve firing [11] or perceived loudness [12]. A content-specific measure was proposed in [1], based on the perceived loudnesses of direct and reverberated components of a given signal.

Two important remarks can be made here. Firstly, works on speech assimilate clarity and intelligibility, although the latter does not only rely on the easiness to spot individual phonemes (clarity), but also on one's cultural background which allows to "fill the blanks". Secondly, all measures consider clarity as the non-alteration of the sound by noise, reverberation, and other impairments, or, equivalently, as the quality of the transmission channel. This highlights the need for a measure of intrinsic sound clarity that would be independent from its high-level content (text or music notes).

In the present paper, we propose a new measure of sound clarity inspired by previous works on image quality assessment [13, 14]. Transposed to the field of digital images, audio clarity could be compared to image sharpness. Several

reference-free objective measures of sharpness are based on the importance of Fourier (or wavelet) phase in the perception of blur [15]. In particular, the global phase coherence (GPC [13]) measures how the regularity of an image – defined by its total variation (TV) – is affected by the destruction of the phase information. The Sharpness Index (SI [14]) measures the sensitivity of the TV to the convolution of the image with a white noise. It behaves similarly to the GPC but is computationally simpler, and was successfully used as a criterion for blind image deblurring.

How could GPC or SI be transposed to audio signals? A sharp image has a sparse gradient and this sparsity is reduced by phase randomization (GPC) or white noise convolution (SI), which increases the TV. On the contrary, the TV of a blurred or noisy image is much less sensitive to those operations. A similar behavior is found in audio signals: a clear sound has a sparse spectrogram, unlike a reverberated or noisy sound. Convolving the sound with a white noise should reduce the spectrogram sparsity for a clear sound, while leaving it almost unchanged for a reverberated or noisy sound. This leads us to propose a Non-Intrusive Audio Clarity index (NIAC), defined as the sensitivity of the spectrogram sparsity to a convolution of the signal with a white noise.

Our goal is not to formally assess the NIAC as an objective measure of clarity that would outperform state of the art indices in terms of correlation with the perceived clarity, but to show that this approach provides a relevant indicator of clarity, which can be used as an efficient criterion to drive audio enhancement algorithms. We shall illustrate this on Blind Source Separation (BSS). The objective in BSS is to recover a set of source signals from a set of observed signals (which are supposed to be mixtures of the sources), relying on a minimum amount of prior information about the sources and the mixing process [16, 17]. In the simplest scenario, the mixing process is modeled as a non-degenerate instantaneous linear system with the same number of inputs and outputs, so that the sources can be recovered by a linear combination of the mixtures.

A popular solution for source separation in this scenario is provided by In-

dependent Component Analysis (ICA) [17], assuming that the sources are mutually independent and that at most one source has a Gaussian distribution. In this case, since the distribution of the mixtures are closer to a Gaussian one than the sources alone, signals can be recovered using a deflation approach [18], estimating the sources one after another by finding linear combinations of the mixtures that maximize a non-Gaussianity measure, as implemented in the FastICA algorithm [19]. Here, following the idea that a source alone is clearer than a mixture, we propose to extract a source by finding the combination of mixtures that maximizes the NIAC.

The article is structured as follows. In Section 2 we define the NIAC through the analogy with the image sharpness index. We assess its ability to measure audio clarity in Section 3. In Section 4, we propose a NIAC-based source separation algorithm, which performances are evaluated in Section 5.

2. The Non-Intrusive Audio Clarity index (NIAC)

2.1. Spectrogram

Considering the time-frequency analysis of a finite-length discrete-time signal s, with analysis windows of length N and an overlap of $(1-\lambda)N$ samples between consecutive windows ($0 < \lambda < 1$, $\lambda N \in \mathbb{N}$), we define the spectrogram of s as

$$S(f,t) = \sum_{n=0}^{N-1} s(t+n)h(n)C(f,n), \quad f \in \{0,1,\ldots,N_f-1\}, \quad t \in \lambda N\mathbb{Z}, \quad (1)$$

where the apodization function h, the base functions C, and the value of N_f (N or $N/2$) depend on the real-valued transform used, denoted by $\mathfrak{T}$ in the following. For instance, for the Modified Discrete Cosine Transform (MDCT), $N_f = N/2$ and

$$C(f,n) = \frac{2}{\sqrt{N}} \cos\left(\frac{2\pi}{N}\Big(n+\frac{1}{2}+\frac{N}{4}\Big)\Big(f+\frac{1}{2}\Big)\right). \quad (2)$$

The sparsity of the spectrogram will be measured by

$$\|S\|_1 = \sum_{f,t} |S(f,t)|. \quad (3)$$

2.2. Definition of the Non-Intrusive Audio Clarity index

Inspired by the image Sharpness Index [14], we propose to measure the audio clarity by the sensitivity of the spectrogram sparsity of a signal s to the degradation caused by the convolution of s with a Gaussian white noise.

Let $s' = s * w$, where $*$ denotes the discrete convolution product and $w : \mathbb{Z} \to \mathbb{R}$ is a white Gaussian noise with zero mean and variance $\sigma_w^2 = 1/N_s$, N_s being the number of samples of s. The expectation of $\|s'\|_2^2$ (written $\mathbb{E}\big[\|s'\|_2^2\big]$) is equal to $\|s\|_2^2$. Let S and S' be the spectrograms of s and s' respectively, as defined by Eq. (1), the support of S' being truncated to N_t samples.

The aforementioned sensitivity can be expressed through the probability that the convolution of s with a white noise does not increase the sparsity of its spectrogram, that is,

$$p = \mathrm{Prob}\big[\|S'\|_1] \leq \|S\|_1\big]. \tag{4}$$

This probability p is expected to be very small for a clean (and informative) audio signal, and not so small for a noisy and/or reverberated signal. Assuming that $\|S'\|_1$ is nearly Gaussian (which is observed in practice), we can approximate the quantity $-\log p$ (more adapted than p to a computer scale since values like $p = 10^{-10000}$ could be easily observed) by

$$-\log\left(\mathrm{Prob}\left[X \leq \|S\|_1 \;\middle|\; X \sim \mathcal{N}\big(\mathbb{E}[\|S'\|_1], \mathrm{Var}[\|S'\|_1]\big)\right]\right). \tag{5}$$

We define this quantity as the *Non-Intrusive Audio Clarity index* (NIAC)

$$\mathcal{C}(s) \triangleq -\log\left(\Phi\left(\frac{\mathbb{E}[\|S'\|_1] - \|S\|_1}{\sqrt{\mathrm{Var}[\|S'\|_1]}}\right)\right), \tag{6}$$

where $\mathrm{Var}[X]$ denotes the variance of a random variable X, and

$$\Phi(t) = \frac{1}{\sqrt{2\pi}} \int_t^{+\infty} e^{-x^2/2}\, dx \tag{7}$$

is the tail of the standard normal distribution.

Note that the NIAC is invariant by scaling: $\forall\, \lambda \in \mathbb{R}, \quad \mathcal{C}(\lambda s) = \mathcal{C}(s)$.

2.3. Computation

Theorem 1. *The expectation and the variance of* $\|S'\|_1$ *are*

$$\mathbb{E}[\|S'\|_1] = \sqrt{\frac{2}{\pi}} N_t \sum_{f=0}^{N_f-1} \sigma_{S'}(f) \tag{8}$$

$$\mathrm{Var}[\|S'\|_1] = \frac{2}{\pi} \sum_{\substack{0\le f,f'\le N_f-1 \\ 1-N_t\le\Delta\le N_t-1}} (N_t - |\Delta|)\sigma_{S'}(f)\sigma_{S'}(f')\omega\left(\frac{\Gamma_{S'}(f,f',\Delta\lambda N)}{\sigma_{S'}(f)\sigma_{S'}(f')}\right), \tag{9}$$

respectively, where

- N_t *and* N_f *are the numbers of columns and lines of* S*;*
- $\Gamma_{S'}(f,f',\tau) \triangleq \sigma_w^2 \mathfrak{T}[\tilde{R}_{s,\tau}(n,n')]$*;*
- $\tilde{R}_{s,\tau}(n,n') \triangleq R_s(\tau+n-n')h(n)h(n')$*, where* R_s *stands for the auto-correlation of* s *(finite and deterministic);*
- $\sigma^2_{S'}(f) \triangleq \Gamma_{S'}(f,f,0)$*;*
- $\forall\, x \in [-1,1], \quad \omega(x) \triangleq x \arcsin x + \sqrt{1-x^2} - 1.$

According to Lemma 1 of [14], the function ω can be approximated by $\omega(x) \simeq x^2/2$, leading to the following approximation of Eq. (9):

$$\mathrm{Var}[\|S'\|_1] \simeq \frac{1}{\pi} \sum_{\substack{0\le f,f'\le N_f-1 \\ 1-N_t\le\Delta\le N_t-1}} (N_t - |\Delta|)\frac{\Gamma^2_{S'}(f,f',\Delta\lambda N)}{\sigma_{S'}(f)\sigma_{S'}(f')} \tag{10}$$

Proof. Convolving the deterministic finite-length signal s with the white noise w produces s' stationary, Gaussian with zero mean. Hence, $S'(f,t)$ is stationary too, and

$$\begin{aligned}
\mathbb{E}[S'(f,t)] &= 0 && (11)\\
\mathrm{Var}[S'(f,t)] &= \sum_{m,n=0}^{N-1} \mathbb{E}[s'(t+m)s'(t+n)]h(m)h(n)C(f,m)C(f,n) \\
&= \sigma_w^2 \sum_{m,n=0}^{N-1} R_s(m-n)h(m)h(n)C(f,m)C(f,n) \\
&\triangleq \tilde{\sigma}^2_{S'}(f) \quad \text{(independent of } t\text{)} && (12)
\end{aligned}$$

Since $S'(f,t)$ is Gaussian and using Lemma 4 of [14],

$$\mathbb{E}[|S'(f,t)|] = \tilde{\sigma}_{S'}(f)\sqrt{\frac{2}{\pi}}, \tag{13}$$

so that

$$\mathbb{E}[\|S'\|_1] = \sum_{f,t} \mathbb{E}[|S'(f,t)|] = \sqrt{\frac{2}{\pi}} N_t \sum_{f=0}^{N_f-1} \tilde{\sigma}_{S'}(f). \tag{14}$$

To obtain $\mathbb{E}[\|S'\|_1^2]$, we first compute

$$\begin{aligned}
\mathbb{E}[S'(f,t)S'(f',t')] &= \sum_{n,n'=0}^{N-1} \mathbb{E}[s'(t+n)s'(t'+n')]h(n)h(n')C(f,n)C(f',n') \\
&= \sigma_w^2 \sum_{n,n'=0}^{N-1} R_s(t-t'+n-n')h(n)h(n')C(f,n)C(f',n') \\
&= \sigma_w^2 \mathfrak{T}[\tilde{R}_{s,t-t'}(n,n')] \\
&= \Gamma_{S'}(f,f',t-t'). \qquad (15)
\end{aligned}$$

Note that $\tilde{\sigma_{S'}}^2(f) = \Gamma_{S'}(f,f,0) = \sigma_{S'}^2(f)$, so that Eq. (14) is equivalent to Eq. (8). Moreover, using Lemma 5 of [14] with $Z = [S'(f,t), S'(f',t')]^\top$, we obtain

$$\mathbb{E}[|S'(f,t)S'(f',t')|] = \frac{2}{\pi}\sigma_{S'}(f)\sigma_{S'}(f')\omega\left(\frac{\Gamma_{S'}(f,f',t-t')}{\sigma_{S'}(f)\sigma_{S'}(f')}\right) + \frac{2}{\pi}\sigma_{S'}(f)\sigma_{S'}(f'). \tag{16}$$

$$\begin{aligned}
\mathbb{E}[\|S'\|_1^2] &= \sum_{\substack{0\le f,f'\le N_f-1 \\ 0\le k,k'\le N_t-1}} \mathbb{E}\big[|S'(f,k\lambda N)S'(f',k'\lambda N)|\big] \\
&= \frac{2}{\pi} \sum_{\substack{0\le f,f'\le N_f-1 \\ 0\le k,k'\le N_t-1}} \sigma_{S'}(f)\sigma_{S'}(f')\omega\left(\frac{\Gamma_{S'}(f,f',(k-k')\lambda N)}{\sigma_{S'}(f)\sigma_{S'}(f')}\right) + N_t^2 \frac{2}{\pi}\left(\sum_{0\le f\le N_f-1} \sigma_{S'}(f)\right)^2 \qquad (17)
\end{aligned}$$

Since the second term of Eq. (17) is equal to $\mathbb{E}[\|S'\|_1]^2$, we deduce (9). □

2.4. NIAC of a mixture

Theorem 2. *Let y be linear combination of p signals $x_1 \dots x_p$, that is,*

$$y = \sum_{i=1}^{p} \alpha_i x_i. \tag{18}$$

The NIAC of y can be computed using Eg. (6) and Theorem 1, with

$$\Gamma_{Y'}(f,f',\tau) = \sum_{1\le i,j\le p} \alpha_i\alpha_j \Gamma_{X'_iX'_j}(f,f',\tau), \tag{19}$$

where

- $\Gamma_{X'_iX'_j}(f,f',\tau) \triangleq \sigma_w^2 \mathfrak{T}[\tilde{R}_{x_ix_j,\tau}(n,n')]$*;*
- $\tilde{R}_{x_ix_j,\tau}(n,n') \triangleq R_{x_ix_j}(\tau+n-n')h(n)h(n')$*, where $R_{x_ix_j}$ stands for the inter-correlation between x_i and x_j (finite and deterministic).*

Proof. The base of $\Gamma_{Y'}(f,f',\tau)$ calculation is the deterministic auto-correlation of y,

$$R_y(\tau+n-n') = \sum_{1\le i,j\le p} \alpha_i\alpha_j R_{x_ix_j}(\tau+n-n'). \tag{20}$$

Similarly,

$$\tilde{R}_{y,\tau}(n,n') = \sum_{1\le i,j\le p} \alpha_i\alpha_j \tilde{R}_{x_ix_j,\tau}(n,n'), \tag{21}$$

and from $\Gamma_{Y'}(f,f',\tau) \triangleq \sigma_w^2\mathfrak{T}[\tilde{R}_{y,\tau}(n,n')]$ the linearity of the transform $\mathfrak{T}$ yields Eq. (19). □

2.5. Complexity of NIAC computation

We measure the complexity as the number of multiplications. The computation of the autocorrelation R_s requires $\Theta(N_f \log_2 N_f)$ multiplications. The construction of each matrix $\Gamma_{S'}(\cdot,\cdot,\tau)$ requires $\Theta(N_f^2 \log_2 N_f)$ multiplications, so that the computation of $\Gamma_{S'}$ requires globally $\Theta(N_t N_f^2 \log_2 N_f)$ multiplications. The variance computation (9) performs $\Theta(N_t N_f^2)$ additional multiplications. Consequently, the NIAC has a computational cost of $\Theta(N_t N_f^2 \log_2 N_f)$ multiplications.

For the NIAC of a mixture, we consider asymptotic equivalents, for further use in Section 4. We suppose that the $\Gamma_{X'_iX'_j}(f,f',\tau)$ values are already available. Each $\Gamma_{Y'}(f,f',\tau)$ requires $O(p^2)$ multiplications, so that $O(p^2 N_t N_f^2)$ multiplications are necessary for $\Gamma_{Y'}$. The variance computation needs $O(N_t N_f^2)$ additional multiplications. The global computational cost of the NIAC of a mixture, given $(\Gamma_{X'_iX'_j})_{i,j}$, is $O(p^2 N_t N_f^2)$.

2.6. Parameter setting

For the spectrogram, we used the MDCT with 50% frame-overlapping (that is, $\lambda = \frac{1}{2}$) and a Kaiser-Bessel apodization function h. This choice is motivated by (i) the fact that the complexity is a quadratic function of the number of frequency bins N_f, which is only half of the window length N in the case of the MDCT; (ii) the practicality of the MDCT for block-processing audio signals in the frequency domain, with a view to using the NIAC as a criterion to drive audio-enhancement algorithms.

The window length N is set to around 20 ms times the sampling frequency, as commonly used in audio processing to ensure a satisfactory trade-off between time and frequency resolutions. Keeping in mind that we aim to measure the spectrogram sensitivity, longer windows make the spectrogram less sensitive to smearing in the time dimension in case of reverberation and increase the complexity, while shorter windows decrease the frequency resolution, especially in the case of harmonic signals, so that the spectrogram is less sensitive to noise.

The choice of the analysis duration T must be driven by the criterion of NIAC stability across time, in the sense that it should not vary much across time in constant conditions of noise, reverberation, etc. In addition to that, the relevant value of T depends on the rhythm of the signal, as illustrated by Fig. 1. The spectrogram is much more modified by the convolution with a white noise if a strong non-stationarity occurs during the period T, like a syllable or a note change, leading to a higher NIAC. If T is lower than the average period corresponding to the rhythm (syllables or notes per second), some T-blocks contain a change, others not, leading to a low mean NIAC and a strong variance. On the contrary, for higher T, each block is very likely to contain a change, which makes the NIAC higher and more stable across time.

Nonetheless, the NIAC can be averaged on long blocks of the same duration. For example, averaging 2048ms blocks yields a stable indicator that does not depend much on the choice of T, as illustrated in Fig. 2 for the same signals as in Fig. 1. This is of great interest, since computational and storage costs increase linearly with T. Additionally, T should be as small as possible in the foresight

of using the NIAC as a criterion for non-stationary enhancement algorithms.

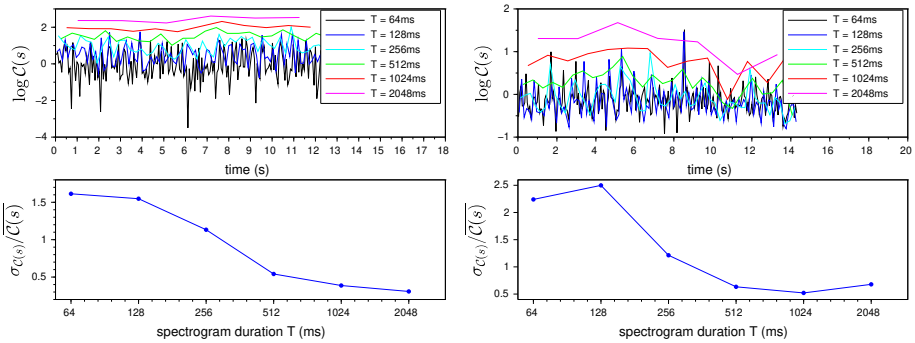


Figure 1: *Top row:* For speech (left) and piano (right), variations of NIAC across time for different values of the spectrogram duration T. *Bottom row:* relative standard deviation of the NIAC as a function of T. The average syllable duration is about 200 ms in the speech signal; the average note duration is about 250 ms in the piano signal. The NIAC is higher and more stable for $T > 256$ ms, that is, for T higher than the average syllable/note duration.

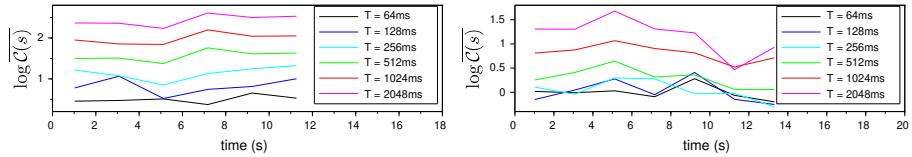


Figure 2: For speech (left) and piano (right), temporal variations of the NIAC averaged on blocks of 2048 ms, for different values of spectrogram duration T. The variability of the log averaged NIAC is similar for all values of T. Consequently, if we average the NIAC over a long duration, T can be set to any value convenient for the constraints given by the context.

3. How well does the NIAC measures audio clarity?

3.1. Sound material

We used a speech corpus and a music corpus. The speech corpus was created from the TIMIT database [20], sampled at 16 kHz. We chose 16 speakers, one male and one female from each of the 8 dialect regions of the USA defined in the documentation. For each speaker, the analyzed signal consists of the five "SX" sentences concatenated and lasts 9 s to 18 s.

The music corpus is a set of 5 extracts from the QUASI database [21, 22], with a duration between 9 s and 16 s, totalling 33 mono-instrument tracks, re-sampled at 32 kHz, from which we processed each track independently.

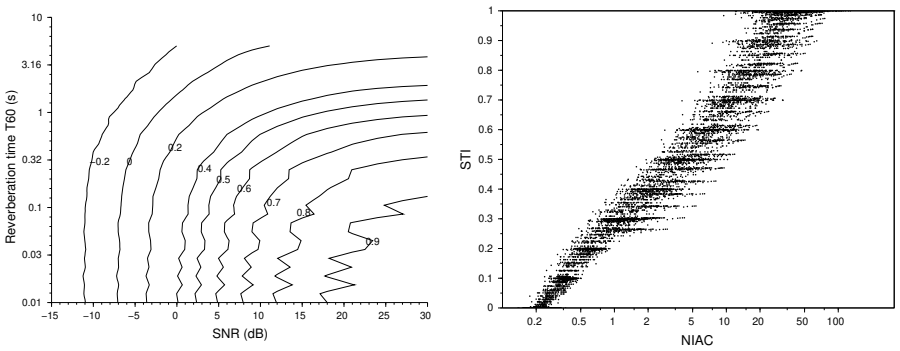
3.2. Experiment

We computed the NIAC for each signal for various noise and reverberation levels. For the reverberation, we considered a purely reverberant room impulse characterized by its reverberation time T60 (the time it takes for the sound level to reduce by 60 dB). For each T60 value, we synthesized an impulse response by multiplying a white Gaussian noise by an exponential envelope matching T60.

For the speech corpus, we tested 30 values of T60, logarithmically distributed between 10 ms and 5 s, and 21 SNR values, linearly distributed between -30 and +30 dB, producing $30 \times 21 = 630$ (T60, SNR) conditions. Before NIAC computation, silence was suppressed in the signals. We computed the NIAC on disjoint blocks of 512 ms and, for each speaker, the mean NIAC on the whole signal. The spectrograms used in NIAC were based on 32 ms analysis windows.

For the music corpus, we tested 10 T60 values, logarithmically distributed between 10 ms and 5 s, and 13 SNR values, linearly distributed between -30 and +30 dB, producing $10 \times 13 = 130$ (T60, SNR) conditions. Before NIAC computations, we suppressed the beginning and ending silences, but we kept the small silences that are part of the signal. Again, the spectrograms were based on 32 ms analysis windows. We considered 4 conditions in the foresight of using the NIAC as a criterion for BSS: averaging time of 1 s and 4 s, with $T = 256$ ms and 1024 ms (to check the independence on T indicated by Fig. 2).

In both experiments, for each (T60, SNR) condition we compared the average NIAC to the STI, computed from the T60 and SNR parameters according to [23]. Although the STI is intended for speech intelligibility assessment, its principles (measuring how the acoustic channel reduces the modulation index for various modulation frequencies in various frequency bands) make it appropriate for any audio signal, provided that the range of modulation frequencies is within 0.63-12.5 Hz, which holds for music instruments [24].



(a) Iso-log(NIAC) lines in the SNR-T60 plane

(b) Relation between NIAC and STI

Figure 3: (a) Iso-log(NIAC) lines in the SNR-T60 plane: for each (SNR,T60) condition, the NIAC is averaged over the 16 speakers. These level lines are similar to the iso-STI lines, suggesting that the NIAC could be used as an alternative to the STI; (b) Relation between NIAC and STI: each point represents one (speaker, SNR, T60) condition, where speaker = 1 to 16, T60 takes 30 logarithmically distributed values between 10 ms and 5 s, and SNR takes 21 linearly distributed values between -30 and +30 dB. The high correlation (0.99) shows that the NIAC (which is reference-free) can be used to predict the full-reference STI.

3.3. Results

For the speech corpus, for each (SNR, T60) condition, we computed the average NIAC over the 16 speakers. Fig. 3a represents the iso-log(NIAC) lines in the SNR-T60 plane, which are very similar to the iso-STI lines (see [23]). To explore this similarity further, we represented in Fig. 3b each triplet (speaker, SNR, T60) as a point in the (log(NIAC), STI) plane. The log of the mean NIAC is linearly correlated with the STI: the global correlation coefficient is 0.99, and the individual correlation coefficients of the speakers are between 0.98 and 0.99. This shows that the NIAC can be considered as a reliable predictor of the STI, and thus used as an intelligibility measure.

For each instrument of the music corpus, the equivalent figure also exhibits a correlation between the log of the mean NIAC and the STI, but the value of the correlation coefficient depends on the instrument, on the averaging time, on the averaging block, and on the spectrogram duration T. Fig. 4 shows how the choice of these parameters influences the correlation. Choosing $T = 256$ ms and

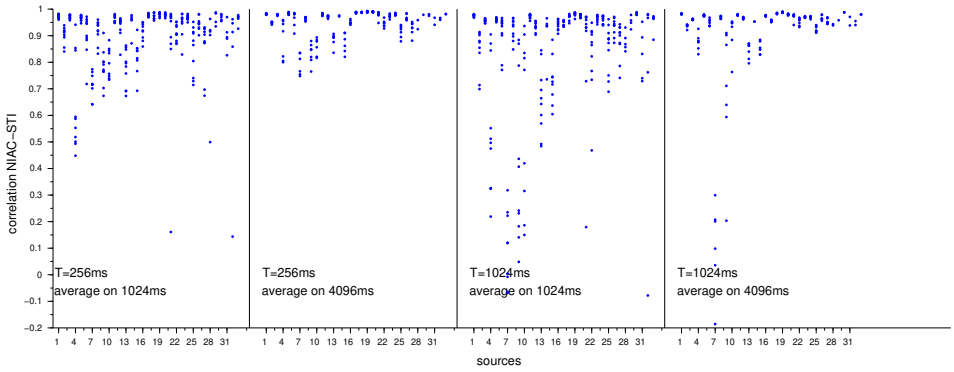


Figure 4: Dispersion of the correlations between log(NIAC) and STI. Each point represents one averaging block. We considered successive disjoint blocks when averaging on 1024 ms, and 75% overlapping blocks when averaging on 4096 ms. We can see that the second setting ($T = 256$ ms, averaging on 4096 s) leads to a more systematically good correlation.

averaging the NIAC on 4096 ms ensures the best correlations and the lowest dependence on the choice of the averaging block.

4. NIAC-based blind source separation

4.1. Problem setting

We consider an instantaneous determined mixture of p signals. Denoting by s the vector of p source signals, x the vector of p mixtures, and A the non-singular $p \times p$ mixing matrix, the mixture can be written $x = As$. The goal of blind source separation (BSS) is to estimate s from x with A unknown.

The initial idea proposed in [25] is that a separated source is clearer than a mixture, so that, under the assumption that the NIAC measures clarity, a source separation algorithm could be driven by NIAC maximization. The experimental results presented in [25] showed that this is only correct when all source signals have a NIAC with the same order of magnitude. If this is not the case, the source signals with the lower NIAC can end up with a higher NIAC when corrupted by a signal with a much higher NIAC, so that their extraction actually corresponds to NIAC minimization (instead of maximization). Thus, extracting one of the

source signals means finding

$$\hat{\alpha} \in \arg\max_{\alpha} \overline{\mathcal{C}}(y_\alpha) \cup \arg\min_{\alpha} \overline{\mathcal{C}}(y_\alpha), \quad \text{with} \quad y_\alpha = \sum_{i=1}^{p} \alpha_i x_i, \; \alpha = [\alpha_1 \dots \alpha_p]^\top \quad (22)$$

and $\overline{\mathcal{C}}$ denotes the average of $\mathcal{C}$ over several blocks of duration T.

Since the NIAC is invariant under scaling, the solutions of Eq. (22) are defined up to a scaling factor. We remove this degree of freedom by imposing $\mathbb{E}[y_\alpha^2] = 1$, that is, $\alpha^\top C_x \alpha = 1$, where $C_x = \mathbb{E}[xx^\top]$ denotes the correlation matrix of x. Since C_x is symmetric and non-negative, we can find a (non-negative symmetric) matrix $\sqrt{C_x}$ such that $C_x = \sqrt{C_x}\sqrt{C_x}^\top$. If we set $\beta = \sqrt{C_x}^\top \alpha$, the constraint $\mathbb{E}[y_\alpha^2] = 1$ becomes

$$\sum_{i=1}^{p} \beta_i^2 = 1. \quad (23)$$

In addition to Eq. (23), p other constraints must be considered: the contributions of y_α to each component of x must have the same sign. Let $\hat{a}$ be the vector of the estimated contributions:

$$\hat{a} = \arg\min_{a} \mathbb{E}[\|x - y_\alpha a\|^2] = \mathbb{E}[y_\alpha x]/\mathbb{E}[y_\alpha^2]. \quad (24)$$

The sign constraints are

$$\pm\mathbb{E}[y_\alpha x] \;\geq\; 0, \quad (25)$$

which means that all components of $\mathbb{E}[y_\alpha x]$ have the same sign. Using Eq. (22) leads to

$$\pm C_x \alpha \;\geq\; 0, \qquad \text{that is} \quad \pm\sqrt{C_x}\beta \;\geq\; 0. \quad (26)$$

Hence, the optimization problem can be summarized as follows:

$$\hat{\beta} \in \arg\max_{\beta} \overline{\mathcal{C}}(y_\beta) \cup \arg\min_{\beta} \overline{\mathcal{C}}(y_\beta) \quad \text{with} \quad y_\beta = x^\top \left(\sqrt{C_x}^\top\right)^{-1} \beta \quad (27)$$

$$\text{under the constraints} \begin{cases} \|\beta\| = 1 \\ \text{and } \pm\sqrt{C_x}\beta \;\geq\; 0. \end{cases} \quad (28)$$

We have to optimize a function on a subregion of $(p-1)$-dimensional sphere of radius 1 defined by linear inequality constraints. Since each point β of the sphere

is equivalent to its symmetric $-\beta$, only one hemisphere has to be explored. Note that for each evaluation of $\overline{\mathcal{C}}(y_\beta)$, most of the calculations are avoided thanks to Theorem 2 (all $\Gamma_{X'_i X'_j}$ are computed once at the beginning).

Since we know that the expected optimums respect the sign constraint $\pm\sqrt{C_x}\beta \geq 0$, we can just check it a posteriori, once the algorithm converged, which simplifies the optimization process. In practice, this sign constraint was satisfied in all the numerical experiments we performed, so we never had to reset the search with different initialization parameters.

Note that in the case of an iterative extraction/deflation process (see Section 4.2), Eq. (26) is applicable only for the first extracted source signal. For the ones that follow, since the extraction coefficients apply to a deflated version of x, another condition on β has to be derived from Eq. (25). For an a posteriori checking, it is simpler to use directly Eq. (25) in all cases. In addition to that, the constraint $\|\beta\| = 1$ may be satisfied by letting $\|\beta\|$ free during the optimization and normalizing the solution at the end, or when needed to control the optimization algorithm (see Subsection 4.3).

4.2. Optimization and separation scheme

A first idea could be to search for all local optima and to extract the source corresponding to each of them. This search can be performed in parallel, using, for example, the Multi-Optima Particle Swarm Optimization (MOPSO) [26], or sequentially, with an inhibition of the successively found optima. The drawback of this solution is its computational cost, especially as the extrema of Eq. (27) may not all correspond to an extraction (see the voice+voice example in [25]).

Another approach is to perform an iterative extraction-deflation process [17]. At each iteration, we first extract one source signal by maximizing or minimizing $\overline{\mathcal{C}}(y_\beta)$, then we estimate its contribution to the mixture in order to subtract it, and finally we reduce the mixture dimension (see Algorithm 1). The successive dimension reductions decrease the computational cost all the more as the complexity of the NIAC computation for a mixture is a quadratic function of the number of sources (see Subsection 2.5).

As a drawback, the separation quality may decrease across iterations and one bad extraction can jeopardize all the ones that follow. To limit this risk, we take advantage of the possibility of maximizing or minimizing the NIAC to extract a source. Consequently at each iteration, we keep the same optimization direction as for the previous iteration to extract a signal, and we assess this extraction through its independence from the residual signal. If the independence is sufficient and the solution fulfills the sign constraint (25), we keep this extraction and go further, otherwise we try the optimization in the other direction. In this case, we keep the one that fulfills the sign constraint and yields the best independence (see Algorithm 2).

The independence between two signals y and x can be evaluated through $|\mathbb{E}[y\phi(x)]|$, which measures the nonlinear correlation between the signals, where ϕ denotes a nonlinear function, e.g., cubic or hyperbolic [17]. In practice, the lower the score, the better the independence. Hence, to measure the independence of an extracted signal y relatively to the residual multi-channel signal $x = [x_1 \dots x_p]^\top$, we use the independence score

$$\mathcal{I}(y, x) \triangleq \max_{1 \leq i \leq p} |\mathbb{E}[y\phi(x_i)]|. \tag{29}$$

A classical continuous optimization method, such as Newton's method, can perform fast and accurately if the gradient and the Hessian of the function to be optimized can be calculated or estimated. But this type of optimization is prone to being trapped in a local optimum. On the other hand, Particle Swarm Optimization (PSO) [27] allows to find the global optimum thanks to its ability to explore large domains but the particles converge slowly to the accurate optimal position. Consequently, we take advantage of both approaches through a two-step optimization scheme: PSO allows to roughly search for the global optimum, then its solution initializes a Newton-type algorithm close to the optimum, which accelerates the convergence of this second optimization and avoids the risk of being trapped in a local optimum. Both algorithms are described in more details in the next subsection.

Algorithm 1 Iterative extraction/deflation process.

$\tilde{x} \leftarrow x$

repeat

Extract y_{max} through NIAC maximization

Estimate the contribution $\hat{a}^{max}$ of y_{max} to $\tilde{x}$ (see Eq. (24))

Deflation: $\tilde{x} \leftarrow \tilde{x} - \hat{a}^{max} y_{max}$

Dimension reduction: write

$$\tilde{A} = \left(\begin{array}{c|c} \hat{a}^{max} & \begin{array}{c} I_{\tilde{p}-1} \\ 0_{1,\tilde{p}-1} \end{array} \end{array} \right), \text{ where } \tilde{p} = \dim(\tilde{x})$$

Decompose $\tilde{A}$ under the form $\tilde{A} = QR$, with Q orthogonal and R upper triangular

$\tilde{Q} \leftarrow Q$ without its first column

$\tilde{x} \leftarrow \tilde{Q}^{\top} \tilde{x}$

until $\dim(\tilde{x}) = 1$

Algorithm 2 Iterative extraction/deflation based on NIAC minimization/maximization, with boolean sign constraint checking S_{opt} and signals independence score I_{opt}. $\overline{X}$ stands for the opposite (negation or inverse optimization direction) of X.

opt $\leftarrow$ max *(that is, we look for a maximum)*

repeat

opt$(NIAC) \longrightarrow$ extraction and deflation $\longrightarrow I_{\text{opt}}, S_{\text{opt}}$

if $I_{\text{opt}} > threshold$ **or** $\overline{S_{\text{opt}}}$ **then**

$\overline{\text{opt}}(NIAC) \longrightarrow$ extraction and deflation $\longrightarrow I_{\overline{\text{opt}}}, S_{\overline{\text{opt}}}$

if $\overline{S_{\text{opt}}}$ **and** $\overline{S_{\overline{\text{opt}}}}$ **then**

error

else if $(\overline{S_{\text{opt}}}$ **and** $S_{\overline{\text{opt}}})$ **or** $(S_{\overline{\text{opt}}}$ **and** $I_{\text{opt}} < I_{\overline{\text{opt}}})$ **then**

keep result of $\overline{\text{opt}}(NIAC)$

opt $\leftarrow \overline{\text{opt}}$

else

keep result of opt$(NIAC)$

end if

end if

until $\dim(\tilde{x}) = 1$

4.3. Optimization algorithms

First step: PSO algorithm.

PSO is a metaheuristic that has been successfully used in a wide range of optimization problems. The basic idea is that a collection of particles, representing solutions to the optimization problem, is scattered in the domain of the function and, from simple update rules for each particle position and velocity, the swarm is able to explore the search space and find the global optimum [27].

The velocity v_t of each particle at the instant t of a PSO algorithm is influenced by the best solution (position of the particle) found so far by the particle itself (*pbest*) and the best solution found by the whole swarm (*gbest*), following a simple update rule given by:

$$v_t = w\, v_{t-1} + c_1\, r_1\, (pbest - x_{t-1}) + c_2\, r_2\, (gbest - x_{t-1}) \qquad , \tag{30}$$

where the inertia weight w determines the contribution rate of previous velocity, r_1 and r_2 are random factors (generated from a uniform distribution), and c_1 and c_2 are acceleration coefficients. The position of each particle is updated from its previous position with

$$x_t = x_{t-1} + v_t \quad . \tag{31}$$

Even though PSO is frequently able to obtain the global solution, its convergence speed can be very low. Nevertheless, for suitable parameter values, the algorithm is able to perform a fast, but rough, exploration of the search space. In this case, an interesting stopping criterion for PSO is based on the *swarm inertia*, defined as the mean squared distance between particles and the swarm barycenter. In other words, if the particles become close to the barycenter, it indicates that the swarm may be converging to a minimum, and provides a good initialization for more accurate search algorithms.

Second step: quasi-Newton.

We propose to use a continuous optimization method, of Newton type. To simplify the calculations, we can notice that since the function $t \mapsto -\log \Phi(t)$ is

increasing, the optimization of $\mathcal{C}(s)$ can be replaced by the optimization of the operand of Φ in Eq. (6), that is, the *pseudo-NIAC*

$$p\mathcal{C}(s) \triangleq \frac{\mathbb{E}[\|S'\|_1] - \|S\|_1}{\sqrt{\mathrm{Var}[\|S'\|_1]}} \tag{32}$$

with the same notations as in Section 2.

The gradient calculation is presented in Appendix. Due to the L^1-norm of the spectrogram involved in the NIAC, the gradient is not defined everywhere, but almost everywherere in the Lebesgue measure sense. In practice, convergence was observed despite these exceptional points.

To avoid the computation of the Hessian of $p\mathcal{C}$, we decided to use a quasi-Newton algorithm with the BFGS approach. The only parameter is the stop criterion, set to

$$\|\beta^{(k)} - \beta^{(k-1)}\| < \varepsilon, \tag{33}$$

where $\beta^{(k)}$ and $\beta^{(k-1)}$ denote the values of β at iterations k and $k-1$, respectively, and ε is a small value. This threshold ε can be directly related to the quality of the separation provided by the solution, as follows. Let $\hat{\beta}$ be the solution, $\hat{\alpha} = (\sqrt{C_x}^\top)^{-1}\hat{\beta}$, and α^{ref} the closest optimal extraction coefficients (corresponding to a line of A^{-1} for the first extraction). We denote by $\hat{y}$ and y^{ref} the corresponding respective extracted signals. Then

$$\hat{y} - y^{ref} = (\hat{\alpha} - \alpha^{ref})^\top x = (\hat{\beta} - \beta^{ref})^\top \big(\sqrt{C_x}\big)^{-1} x, \tag{34}$$

and since $\mathbb{E}[xx^\top] = C_x = \sqrt{C_x}\sqrt{C_x}^\top$, the mean squared error is given by

$$\begin{aligned} \mathbb{E}[(\hat{y} - y^{ref})^2] &= \mathbb{E}\Big[(\hat{\beta} - \beta^{ref})^\top \big(\sqrt{C_x}\big)^{-1} x x^\top \big(\sqrt{C_x}\big)^{-\top} (\hat{\beta} - \beta^{ref})\Big] \\ &= \|\hat{\beta} - \beta^{ref}\|^2. \end{aligned} \tag{35}$$

Since we have the constraint $\mathbb{E}[y^2] = 1$, the signal-to-error ratio is

$$SER \triangleq \frac{\mathbb{E}[y^2]}{\mathbb{E}[(\hat{y} - y^{ref})^2]} = \frac{1}{\|\hat{\beta} - \beta^{ref}\|^2}. \tag{36}$$

Hence the threshold ε can be set according to the desired signal-to-error ratio.

This SER corresponds to the Signal-to-Interference Ratio (SIR) for the first extraction. In an iterative extraction/deflation process, Eq. (36) still holds but

may under-estimate the SIR, since the best extraction coefficients α^{ref} do not avoid the residual interference resulting from the imperfect previous extractions.

Complexity comparison.

The Quasi-Newton algorithm requires the computation of the gradient of $p\mathcal{C}(s)$, which has the same complexity $O(p^2 N_t N_f^2)$ as the NIAC itself (see Subsection 2.5). Indeed, for each α_i,

- $\frac{\partial}{\partial \alpha_i}\Gamma_{Y'}(f, f', \Delta\lambda N)$ requires p multiplications, so that the cost of $\frac{\partial}{\partial \alpha_i}\Gamma_{Y'}$ is $O(pN_t N_f^2)$;
- the cost of $\frac{\partial \|Y\|_1}{\partial \alpha_i}$, $\frac{\partial \mu}{\partial \alpha_i}$, and $\frac{\partial \sigma^2}{\partial \alpha_i}$ are $O(N_t N_f)$, $O(N_f)$, and $O(N_t N_f^2)$, respectively.

In practice, the cost of one iteration of PSO or Quasi-Newton is similar, and the overall optimization time is shared equally among the two steps.

5. Experimental results and discussion

5.1. Sound material, parameters setting, and tools

Following the discussion in Section 3, we used the same music corpus composed of 5 multi-tracks extracts from the QUASI database [21, 22], with duration 9 to 16 s, resampled at 32 kHz. We set the spectrogram duration to $T = 256$ ms, and we averaged the NIAC on 4096 ms.

PSO inertia and acceleration parameters are problem dependent, so choosing the parameters of this type of algorithm is an optimization problem itself [28, 29]. We empirically chose from preliminary simulations the acceleration coefficients $c_1 = 0.5$ and $c_2 = 0.8$, and the inertia weight $w = 0.4$. The swarm generally converges to the global optimal position even with slightly different coefficient values. PSO was initialized with 10 particles in all simulations. The swarm inertia threshold (stop criterion) was set to 0.05. In the QN-BFGS algorithm, the stop criterion was $\varepsilon = 10^{-4}$, which corresponds to a target SER of 80 dB.

We evaluated the separation performance through the signal-to-distortion ratio (SDR), the signal-to-interference ratio (SIR) and the signal-to-artifact ratio

NIAC / ICA	SDR	SIR	SAR
guitar	28 / 38	28 / 38	73 / 71
voice	49 / 27	49 / 27	73 / 71
piano	30 / 44	30 / 44	74 / 71

Table 1: Signal-to-distortion ratio (SDR), signal-to-interference ratio (SIR) and signal-to-artifact ratio (SAR) of a NIAC-based source separation example, compared to FastICA . For each metric (SDR, SIR, SAR), the minimum and maximum values are approximately similar for NIAC and ICA, but not necessarily obtained with the same instruments because the order of extraction is different.

(SAR) [30]. We compared them to the values obtained with a state-of-the-art ready-to-use algorithm, FastICA [19], also using the deflation approach. We fed the algorithm with the same data as the NIAC analysis, that is, the spectrogram on 4096 ms with the same time-frequeny analysis. When running FastICA , one has to chose a non-linear function used for an independence score. As time-frequency samples of audio-signal have generally super-Gaussian distributions [17], the most convenient choice is the Gaussian non-linearity.

5.2. An example of NIAC-based BSS

We consider a mixture of three sources – acoustic guitar, voice, and piano. The PSO approached the maximum in 4 iterations, and the result served as initialization for QN-BFGS optimization, which converged in 10 iterations (16 calls), with $\|\hat{\beta} - \beta^{ref}\| = 3.5 \times 10^{-3}$ and $I_{\max} = 4.1 \times 10^{-2}$, where β^{ref} corresponds to voice extraction. Then, after extraction and deflation, the PSO converged around the maximum in 3 iterations. Finally, the QN-BFGS algorithm initialized by the maximum found by PSO converged in 4 iterations (10 calls), with $\|\hat{\beta} - \beta^{ref}\| = 4.1 \times 10^{-2}$ and $I_{\max} = 2 \times 10^{-2}$, where β^{ref} corresponds to guitar extraction. The results in Table 1 show that the NIAC-based separation performs as well as FastICA on this example.

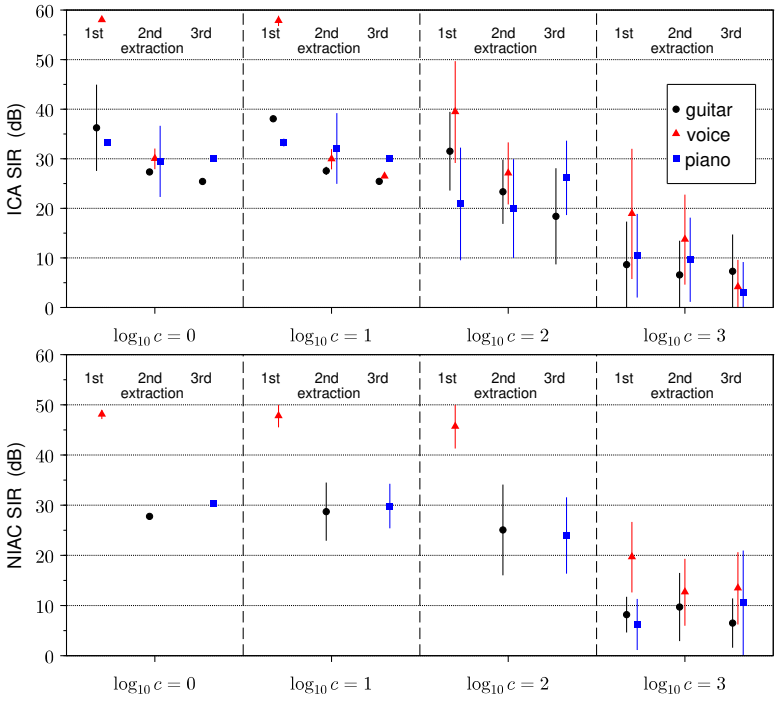


Figure 5: Means and standard deviations of the SIR from FastICA and NIAC-based BSS, for 4 condition numbers. The results are presented by source and by extraction range. The absence of point for a source at an extraction range r means that the source is never extracted at range r, or in less than 10% of the cases. When the condition number increases, the values decrease less and are less scattered with NIAC-BSS than with FastICA .

5.3. Robustness to ill-conditioned mixture matrix

For p sources, we explore the space of mixture matrices A with conditionning number c as follows. We write A as in a singular value decomposition, that is $A = PSQ$ with S a diagonal matrix and $P, Q \in \mathrm{SO}(p)$ (the special orthogonal group). The diagonal of S is filled with the values 1, c, and $p-2$ others values drawn uniformly between 1 and c. P and Q are drawn uniformly in $\mathrm{SO}(p)$ using the algorithm described in [31].

For $p = 3$, we evaluated the performance (measured by the SIR) of FastICA and NIAC-based BSS for $c = 1$, 10, 100 and 1000. For each value of c, we processed FastICA and NIAC-based BSS with the previous sources for 100 and

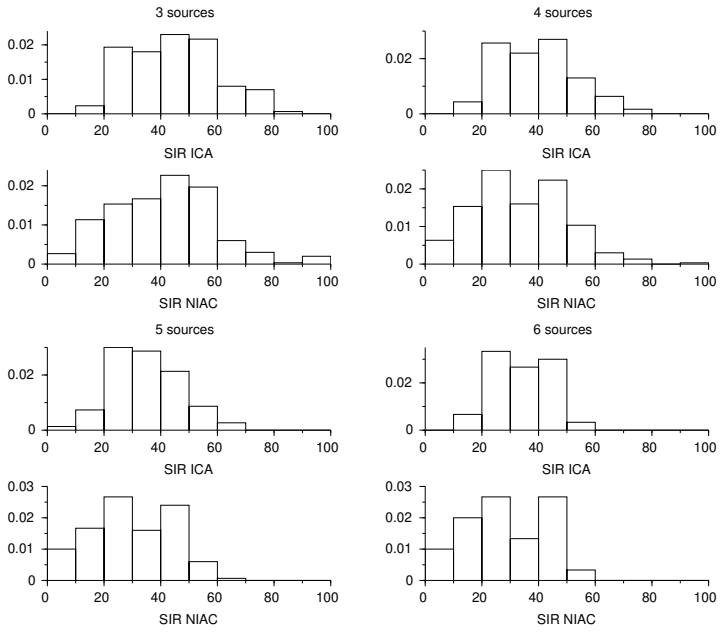


Figure 6: Histograms of SIRs resulting from FastICA and NIAC-based BSS, for mixtures of 3 to 6 sources.

25 mixture matrices, respectively[1], randomly set as specified above. We challenged the robustness to ill-conditioning with a small perturbation, consisting in adding on each mixture channel a noise with SNR of 50 dB. As indicated by Fig. 5, our method appears to be slightly more robust to an ill-conditioned mixture matrix than FastICA .

5.4. Performance evaluation for various sets of sources

For each sources number p =3 to 6 and for each of the 5 extracts, we selected 6 sources that were active on nearly all the extract duration and we ran the NIAC-based BSS and FastICA for each combination of p sources among 6, using the same mixture matrix as in Subsection 5.2. As illustrated by Fig. 6,

[1]The different number of trials is motivated by the fact that the results are analyzed by source and by range of extraction. Whereas the extraction range is generally constant for NIAC-based BSS, it depends on the random initialization for FastICA .

the performance of both methods is similar, but the proportion of SIRs below 20dB is higher for the NIAC-based BSS. The detailed optimization results for these cases show three types of explanation: (i) the choice between maximization and minimization is misled by the independence criterion; (ii) the topography of the NIAC function is difficult (*eg.* maximum on a crest with local irrelevant maxima); (iii) the optimum in the sense of NIAC is slightly different from the optimum in the sense of separation.

5.5. Discussion

Comparing the simulation results, we can observe that NIAC-based BSS and FastICA have similar overall performances. In fact, for the determined instantaneous mixture case, other popular methods such as Infomax or JADE lead to similar separation results. Nevertheless, the choice for FastICA as a basis for comparison is not only justified by its popularity but also by the existence of theoretical studies in the literature [32, 33] providing a very good understanding about its features and limitations.

For example, according to [32], source extraction using FastICA with a gaussian nonlinearity from a three-source mixture is able to achieve an SIR of approximately 51.4 dB (for a mixture of Laplacian sources, considering $N = 500000$ samples, which is roughly the same amount used in our simulations for music signals). For each additional source in the mixture, the performance is reduced by 3 dB. A similar behavior is observed in Fig. 6, and, as mentioned before, is followed closely by the NIAC-based algorithm.

Another point mentioned before is related to the robustness to ill-conditioned mixing matrices, illustrated in Fig. 5. The order in which the sources are extracted by FastICA heavily depends on the initialization of the algorithm, and, as discussed in [33], has an important impact on the quality of the subsequent extracted sources. On the other hand, the NIAC-based algorithm seem to be more robust to initialization, extracting the sources in a same order – which may explain the lower variability of the SIR results.

In addition to that, it is important to highlight some interesting features of

the NIAC-based method. Firstly, it does not rely on the independence of the sources. The independence is used as a secondary criterion to choose between maximization and minimization, but it is not a requirement for the method, which means that even correlated sources could be extracted from the mixture.

Another important point is that it does not assume that the sources are non-Gaussian, an existing limitation in ICA-based methods. As an illustration of this, we ran both methods on the previous corpus with 2-sources mixtures, where all sources were gaussianized according to [34]. While FastICA failed or reached an SIR below 10dB in 77% of the cases, the NIAC-based BSS yielded a mean SIR of 47dB, with 8% of the SIRs below 10dB.

In this sense, one could compare NIAC-based method to other alternative BSS algorithms exploring distinct characteristics of the sources, such as those based on the time structure or the assumption that the sources have a sparse representation [17]. Nevertheless, since the NIAC-based methods explores a criterion closely related to perceptual measures, we consider that it may be a more interesting choice when dealing with audio or speech signal extraction.

6. Conclusion

We have designed the NIAC as a clarity measure that assesses the intrinsic clarity of any audio signal (not specifically speech or music). While highly correlated with STI, it has the advantage of being non-intrusive. Unlike machine-learning-based non-intrusive measures, it does not require any learning and relies on very few parameter settings, without need of fine tuning. It can be used as a criterion to drive audio enhancement algorithms. In the case of blind source separation (BSS) of an instantaneous determined mixture, the NIAC-based BSS exhibits performances similar to those of FastICA, with many advantages: it does not rely on source-independence and non-Gaussianity hypotheses, and it is robust to algorithm initialization and ill-conditioned mixture matrices. The low amount of iterations needed to make the algorithm converge compensates for the complexity of NIAC computation. We have limited the study to a simple

scenario, but the theoretical framework is easily extendable to convolutive mixture separation or dereverberation of a single source recorded by one or several sensors. Since the NIAC needs to be averaged on one to a few seconds, it may however not be appropriate for the correction of non-stationary impairments.

Note that the NIAC design does not restrict it to audio signals: any signal, the cleanness of which is characterized by its time-frequency sparsity, may benefit from this approach, both for quality assessment and enhancement purposes.

Scilab source code for NIAC and NIAC-BSS is freely available at

`https://git.mi.parisdescartes.fr/mahe/niac`

Appendix: calculation of grad$p\mathcal{C}$

Let y be a signal extracted from the p-mixture x with the extraction coefficients α. In the following calculation, considering the notations introduced in Section 2, we represent $\mathbb{E}[\|Y'\|_1]$ and $\sqrt{\mathrm{Var}[\|Y'\|_1]}$ by μ and σ, respectively. We use the approximation (10) of $\mathrm{Var}[\|Y'\|_1]$. For $1 \leq i \leq p$,

$$\frac{\partial p\mathcal{C}(y)}{\partial \alpha_i} = \frac{1}{\sigma^2}\left[\sigma\left(\frac{\partial \mu}{\partial \alpha_i} - \frac{\partial \|Y\|_1}{\partial \alpha_i}\right) - \left(\frac{\mu - \|Y\|_1}{2\sigma}\right)\frac{\partial \sigma^2}{\partial \alpha_i}\right] \tag{37}$$

In this formula, the following elements have to be further calculated: $\partial\|Y\|_1/\partial\alpha_i$, $\partial\mu/\partial\alpha_i$, and $\partial\sigma^2/\partial\alpha_i$.

$$\frac{\partial \|Y\|_1}{\partial \alpha_i} = \sum_{f,t} \frac{\partial}{\partial \alpha_i}\Big|\sum_{j=1}^{p} \alpha_j X_j(f,t)\Big| = \sum_{f,t} \mathrm{sign}\big(Y(f,t)\big) X_i(f,t) \tag{38}$$

$$\frac{\partial \mu}{\partial \alpha_i} = \sqrt{\frac{2}{\pi}} N_t \sum_f \frac{1}{2\sigma_{Y'}(f)} \frac{\partial \sigma^2_{Y'}(f)}{\partial \alpha_i} \tag{39}$$

$$\begin{aligned}\frac{\partial \sigma^2}{\partial \alpha_i} = \frac{1}{\pi} \sum_{f,f',\Delta} (N_t - |\Delta|)\Big\{ & 2\rho_{Y'}(f,f',\Delta\lambda N) \frac{\partial}{\partial \alpha_i}\Gamma_{Y'}(f,f',\Delta\lambda N) \\ & -\frac{1}{2}\rho^2_{Y'}(f,f',\Delta\lambda N)\left(\frac{\sigma_{Y'}(f)}{\sigma_{Y'}(f')}\frac{\partial \sigma^2_{Y'}}{\partial \alpha_i}(f') + \frac{\sigma_{Y'}(f')}{\sigma_{Y'}(f)}\frac{\partial \sigma^2_{Y'}}{\partial \alpha_i}(f)\right)\Big\}\end{aligned} \tag{40}$$

$$\text{with} \qquad \rho_{Y'}(f,f',\Delta\lambda N) = \frac{\Gamma_{Y'}(f,f',\frac{N}{2}\Delta)}{\sigma_{Y'}(f)\sigma_{Y'}(f')}. \tag{41}$$

To conclude, we compute

$$\frac{\partial}{\partial \alpha_i}\Gamma_{Y'}(f,f',\Delta\lambda N) = \sum_{j=1}^{p} \alpha_j \big(\Gamma_{X_i'X_j'}(f,f',\Delta\lambda N) + \Gamma_{X_j'X_i'}(f,f',\Delta\lambda N)\big) \quad (42)$$

from Theorem 2, and similarly,

$$\frac{\partial \sigma^2_{Y'}(f)}{\partial \alpha_i} = \sum_{j=1}^{p} \alpha_j \big(\sigma^2_{X_i'X_j'}(f) + \sigma^2_{X_j'X_i'}(f)\big). \quad (43)$$

Note that the gradient of the pseudo-NIAC relatively to $\beta = \sqrt{C_x}^{\top}\alpha$ is

$$\nabla_\beta p\mathcal{C} = \left(\sqrt{C_x}\right)^{-1} \nabla_\alpha p\mathcal{C}. \quad (44)$$

Acknowledgments

This work is part of the I'CityForAll project, which was funded by a grant from the European program Ambient Assisted Living (AAL-2011-4-056) from 2011 to 2015. See `http://www.icityforall.eu`. The second and fourth authors thank the Brazilian National Council for Scientific and Technological Development (CNPq, grant number 310582/2018-0), and CAPES for the support.

References

[1] J. van Dorp Schuitman, D. de Vries, A. Lindau, Deriving content-specific measures of room acoustic perception using a binaural, nonlinear auditory model, J. Acoust. Soc. of America 133 (3) (2013) 1572–1585.

[2] ANSI, Methods for calculation of the speech intelligibility index, S3.5-1997.

[3] H. J. M. Steeneken, T. Houtgast, A physical method for measuring speech-transmission quality, J. Acoust. Soc. of America 67 (1) (1980) 318–326.

[4] C. H. Taal, R. C. Hendriks, R. Heusdens, J. Jensen, An algorithm for intelligibility prediction of time-frequency weighted noisy speech, IEEE Trans. on Audio, Speech, and Language Processing 19 (7) (2011) 2125–2136.

[5] J. Jensen, C. H. Taal, Speech intelligibility prediction based on mutual information, IEEE/ACM Trans. on Audio, Speech, and Language Processing 22 (2) (2014) 430–440.

[6] P. P. Parada, D. Sharma, J. Lainez, D. Barreda, T. v. Waterschoot, P. A. Naylor, A single-channel non-intrusive C50 estimator correlated with speech recognition performance, IEEE/ACM Trans. on Audio, Speech, and Language Processing 24 (4) (2016) 719–732.

[7] D. Sharma, Y. Wang, P. A. Naylor, M. Brookes, A data-driven non-intrusive measure of speech quality and intelligibility, Speech Communication 80 (2016) 84–94.

[8] A. H. Andersen, J. M. de Haan, Z. Tan, J. Jensen, Nonintrusive speech intelligibility prediction using convolutional neural networks, IEEE Trans. on Audio, Speech, and Language Processing 26 (10) (2018) 1925–1939.

[9] T. H. Falk, C. Zheng, W. Y. Chan, A non-intrusive quality and intelligibility measure of reverberant and dereverberated speech, IEEE Trans. on Audio, Speech, and Language Processing 18 (7) (2010) 1766–1774.

[10] International Organization for Standardization, Acoustics - Measurement of room acoustic parameters - Part 1: Performance spaces, ISO 3382-1:2009.

[11] D. H. Griesinger, What is "clarity", and how it can be measured?, Proc. of Meetings on Acoustics 19 (1) (2013) 015003.

[12] D. Lee, J. van Dorp Schuitman, X. Qiu, I. Burnett, Development of a clarity parameter using a time-varying loudness model, J. Acoust. Soc. of America 143 (6) (2018) 3455–3459.

[13] G. Blanchet, L. Moisan, B. Rouge, Measuring the global phase coherence of an image, in: IEEE Int. Conf. on Image Processing, 2008, pp. 1176–1179.

[14] A. Leclaire, L. Moisan, No-reference image quality assessment and blind deblurring with sharpness metrics exploiting Fourier phase information, J. of Mathematical Imaging and Vision 52 (1) (2015) 145–172.

[15] A. V. Oppenheim, J. S. Lim, The importance of phase in signals, Proc. of the IEEE 69 (5) (1981) 529–541.

[16] J. M. T. Romano, R. Attux, C. C. Cavalcante, R. Suyama, Unsupervised Signal Processing: Channel Equalization and Source Separation, CRC Press, 2010.

[17] P. Comon, P. Jutten, Handbook of Blind Source Separation, Academic Press, 2010.

[18] N. Delfosse, P. Loubaton, Adaptive blind separation of independent sources: A deflation approach, Signal Processing 45 (1) (1995) 59–83.

[19] A. Hyvärinen, Fast and robust fixed-point algorithms for independent component analysis, IEEE Trans. on Neural Networks 10 (3) (1999) 626–634.

[20] J. Garofolo, L. Lamel, W. Fisher, J. Fiscus, D. Pallett, N. Dahlgren, V. Zue, TIMIT acoustic-phonetic continuous speech corpus (1993).

[21] E. Vincent, S. Araki, F. J. Theis, G. Nolte, P. Bofill, H. Sawada, A. Ozerov, B. V. Gowreesunker, D. Lutter, N. Q. Duong, The signal separation evaluation campaign (2007-2010): Achievements and remaining challenges, Signal Processing 92 (2012) 1928–1936.

[22] A. Liutkus, R. Badeau, G. Richard, Gaussian processes for underdetermined source separation, IEEE Trans. on Sig. Proc. 59 (2011) 3155–3167.

[23] T. Houtgast, H. J. M. Steeneken, A review of the MTF concept in room acoustics and its use for estimating speech intelligibility in auditoria, J. Acoust. Soc. of America 77 (3) (1985) 1069–1077.

[24] N. Ding, A. D. Patel, L. Chen, H. Butler, C. Luo, D. Poeppel, Temporal modulations in speech and music, Neuroscience & Biobehavioral Reviews 81 (2017) 181 – 187, the Biology of Language.

[25] G. Mahé, L. Moisan, M. Mitrea, An image-inspired audio sharpness index, in: Proc. European Signal Processing Conf., 2017, pp. 683 – 687.

[26] S. Cheng, Q. Qin, Z. Wu, Y. Shi, Q. Zhang, Multimodal optimization using particle swarm optimization algorithms: CEC 2015 competition on single objective multi-niche optimization, in: IEEE Congress on Evolutionary Computation (CEC), 2015, pp. 1075–1082.

[27] J. Kennedy, Particle swarm optimization, Encyclopedia of machine learning (2010) 760–766.

[28] Y. Shi, R. C. Eberhart, Parameter selection in particle swarm optimization, in: Int. Conf on evolutionary programming, Springer, 1998, pp. 591–600.

[29] M. Jiang, Y. P. Luo, S. Y. Yang, Particle swarm optimization-stochastic trajectory analysis and parameter selection, in: Swarm intelligence, Focus on ant and particle swarm optimization, IntechOpen, 2007.

[30] E. Vincent, R. Gribonval, C. Fevotte, Performance measurement in blind audio source separation, IEEE Trans. on Audio, Speech, and Language Processing 14 (4) (2006) 1462–1469.

[31] P. Diaconis, M. Shahshahani, The subgroup algorithm for generating uniform random variables, in: Probability in the Engineering and Informational Sciences, 1987, pp. 1–15.

[32] P. Tichavsky, Z. Koldovsky, E. Oja, Performance analysis of the FastICA algorithm and Cramér-Rao bounds for linear independent component analysis, IEEE Trans. on Signal Processing 54 (4) (2006) 1189–1203.

[33] E. Ollila, The deflation-based FastICA Estimator: Statistical analysis revisited, IEEE Trans. on Signal Processing 58 (3) (2010) 1527–1541.

[34] I. Mezghani-Marrakchi, G. Mahé, S. Djaziri-Larbi, M. Jaïdane, M. Turki-Hadj Alouane, Nonlinear audio systems identification through audio input Gaussianization, IEEE/ACM Trans. on Audio, Speech, and Language Processing 22 (1) (2014) 41–53.