Dissonance Reduction In Polyphonic Audio Using Harmonic Reorganization

Emilio Molina, Ana M. Barbancho, Lorenzo J. Tardón, and Isabel Barbancho, *Senior Member, IEEE*

***Abstract*—In this paper, a method for automatic reduction of dissonance in recorded isolated chords is proposed. Previous approaches address this problem using source separation and note-level processing. In our approach, we manipulate the harmonic structure as a whole in order to avoid beating partials which, according to prior research on dissonance perception, typically produce an unpleasant sound. The proposed system firstly performs a sinusoidal plus residual modeling of the input and analyses the various fundamental frequencies present in the chord. This information is used to create a symbolic representation of the in-tune version of the input according to some musical rules. Then, the partials of the signals are shifted in order to fit the in-tune harmonic structure of the input chord. The input is assumed to contain one isolated chord, with relatively stable fundamental frequencies belonging to the Western chromatic scale. The evaluation has been performed by 31 expert musicians, which have quantified the perceived consonance of six varied, out-of-tune chords in three variants: unprocessed, processed with our system and processed by a state-of-the-art commercial tool (Melodyne Editor). The proposed approach attains an important reduction of the perceived dissonance, showing better performance than Melodyne Editor for most of the cases evaluated.**

***Index Terms*—Audio analysis and synthesis, audio for multimedia, content-based music processing, music processing systems.**

I. Introduction

MUSICAL Tuning has been an important object of study along history. A *tuning system* defines which tones, or pitches, are used when playing music. The first written evidence related to the tuning of instruments belongs to the old Babylon (around 1500 BC), where a detailed description of the Babylonian harp tuning is described in cuneiform script [1]. Such tuning system was based on the frequency ratio 3:2 (perfect fifth). Later, Pythagoras (sixth century BC) developed such tuning system based on the perfect fifth in order to define the well known Pythagorean tuning [2]. Such strong relationship between mathematics and harmony has been studied by many later scientists and musicologists, among which Zarlino was especially important during the Renaissance.

During the process of music recording in modern studios, the tuning of instruments and vocals is an important aspect to take into account [3]. Depending on the style, the presence of out-of-tune sounds can be a reason to repeat a take. However this is not always possible due to technical limitations of the musicians or, simply, due to a lack of time or economic resources. In the case of monophonic instruments or voice, there are many software tools that can be used during a post-production process for tuning adjustment. For instance, *Melodyne Studio* (Celemony[1] 2003) or *Auto-Tune* (Antares Technology[2] 1997) have been widely used for the tuning of vocals in recording studios during the last years.

This task becomes much harder when dealing with polyphonic recordings. Indeed, despite the fact that polyphonic transcription and source separation are trending topics within the research community, current solutions are not fully practical for professional post-processing purposes. The best commercial solution for such problems is *Melodyne Editor* (Celemony 2009). This software addresses the polyphonic tuning problem through multiple-f_0 estimation, source separation and pitch-shifting [4]. However, in the case of out-of-tune chords Melodyne does not perform source separation accurately, and beating partials are still present in the apparently corrected chord [5]. The presence of beating partials is related to the perceived *dissonance* of a sound [6].

The concept of dissonance can be interpreted differently depending on the context. On the one hand, the musical dissonance is defined as the interval that, according to the classical harmony rules, is unpleasant to the ear [7]. Typically the intervals of minor second (1 semitone), major seventh (11 semitones) and tritone (6 semitones) are considered dissonant. On the other hand, sensory dissonance is defined in perceptual terms as the 'roughness' of a sound, and it can be applied either to musical or non musical sounds. This kind of dissonance has been addressed by many authors [8], but the most important study about dissonance perception was carried out by Plomp & Leveltz in 1965 [6]. The main contribution of this work was to relate the perception of dissonance to the concept of critical bands proposed by

Manuscript received April 16, 2013; revised July 23, 2013; accepted October 04, 2013. Date of publication October 23, 2013; date of current version December 31, 2013. This work was supported by the Ministerio de Economía y Competitividad of the Spanish Government under Project No. TIN2010-21089-C03-02 and Project No. IPT-2011-0885-430000, by the Junta de Andalucía under Project No. P11-TIC-7154 and by the Ministerio de Educación, Cultura y Deporte through the "Programa Nacional de Movilidad de Recursos Humanos del Plan Nacional de I-D+i 2008-2011, prorrogado por Acuerdo de Consejo de Ministros de 7 de octubre de 2011." The associate editor coordinating the review of this manuscript and approving it for publication was Prof. Søren Holdt Jensen.

The authors are with the Department of Ingeniería de Comunicaciones, E.T.S.I. Telecomunicación, University of Málaga, 29010 Malaga, Spain (e-mail: emm@ic.uma.es; abp@ic.uma.es; lorenzo@ic.uma.es; ibp@ic.uma.es).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TASLP.2013.2287056

[1]http://www.celemony.com

[2]http://www.antarestech.com

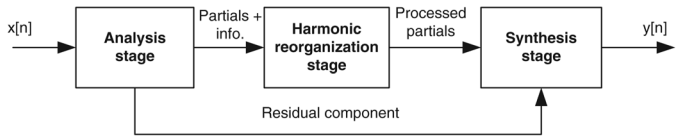

Fig. 1. General scheme of the selected approach for polyphonic dissonance reduction.

Harvey Fletcher in the 1940's [9]. According to Plomp & Leveltz, two tones are perceived as dissonant when they fall within the same critical band.

In this paper, we propose a novel approach for dissonance reduction in polyphonic audio, processing harmony as *a whole* instead of performing source separation. We address both the reduction of *musical dissonance* and *sensory dissonance*. The proposed method assumes that the input chord is stable (f_0s are relatively constant along time) and it is composed of harmonic sounds (i.e. the overtones are placed at integer multiples of each f_0). Some subjective factors related to harmony have been predefined to achieve a good compromise for a practical use in common recording studios. Specifically, we assume that Western music is analyzed.

The selected approach is based on an *analysis-resynthesis* scheme (Fig. 1). This approach is based on the sinusoidal plus residual modeling scheme proposed in [10]. The input to the system is a mono audio signal $x[n]$ containing the original dissonant sound, and the output is a processed mono audio signal $y[n]$. The developed system can be divided into three main blocks:

- **Analysis stage:** The parameters of the sinusoidal component of the signal are extracted and, also, separated. This block is mainly based on the techniques described in [10].
- **Harmonic reorganization stage:** This is the core of the system in which the most interesting techniques presented in this paper are implemented. In this stage, the sinusoidal parametrization previously obtained is manipulated in order to reduce the dissonance of the original sound.
- **Synthesis stage:** This block synthesizes the audio signal making use of the sinusoidal parametrization developed after the Harmonic reorganization stage. It is mainly based on the overlap-add technique [11].

This paper is organized according to the scheme described. The three main blocks are explained in Sections II, III and IV. The evaluation methodology is detailed in Section V, and the results obtained are discussed in Section VI. Finally, in Section VII, a summary of conclusions and contributions about the present work has been included.

II. Analysis Stage

In this section, the *Analysis stage* is described. Some main aspects of the sinusoidal plus residual modeling are presented, namely: sinusoidal estimation (Section II-A), partials tracking (Section II-B) and extraction of the residual component (Section II-C). Also, the method selected for multiple f_0 extraction is explained in Section II-D. A diagram block of the analysis stage is shown in Fig. 2.

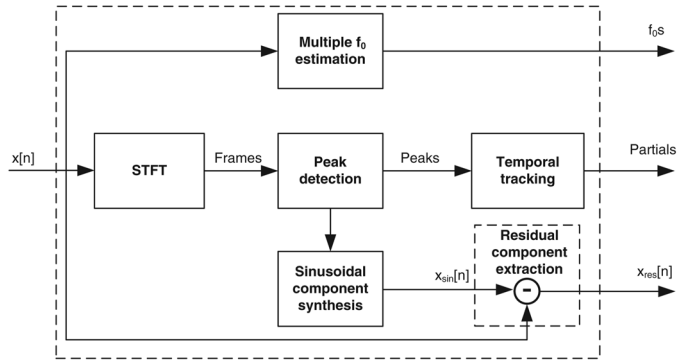

Fig. 2. Block diagram of the analysis stage.

A. Sinusoidal Plus Residual Modeling

The input chords are analyzed using a *sinusoidal plus residual* approach. This model separates the signal $x[n]$ into a sum of stable sinusoids $x_{sin}[n]$ and a residual component $x_{res}[n]$: $x[n] = x_{sin}[n] + x_{res}[n]$ [10]. This model is especially useful for the analysis of sustained musical sounds.

In our approach, the parameters of the model are estimated frame by frame using the STFT. Specifically, for each frame $l \in [1, L]$, three values are estimated: amplitude $\hat{A}_r^l$, frequency $\hat{\omega}_r^l$ and phase $\hat{\phi}_r^l$ corresponding to each partial $r \in [1, R]$. With these three parameters, the sinusoidal component can be synthesized according to:

$$x_{sin}^l[n] = \sum_{r=1}^{R^l} \hat{A}_r^l \cos(\hat{\omega}_r^l \cdot n + \hat{\phi}_r^l) \tag{1}$$

The parameters $(\hat{A}_r^l, \hat{\omega}_r^l, \hat{\phi}_r^l)$ are estimated from the spectrogram $\tilde{X}[l, k]$ of the windowed signal, where l is the frame index and k denotes the frequency value in bins. The STFT computes the N-point spectrum of consecutive, windowed excerpts of length M with a certain hop-size H. Depending on the parameters (H, N, M and window function $w[n]$), the computed spectrogram can vary in terms of resolution and presence of secondary lobes [12]. In the proposed method, low presence of secondary lobes and good frequency resolution, rather than temporal resolution, are desired. Recall that the input sounds are expected to be stable along time, therefore temporal resolution is not considered a critical point. The chosen window $w[n]$ is Blackman-Harris 92 dB with parameters $M = 8001$, $N = 8192$ and $H = 2048$ for a sample rate $f_s = 44100$ Hz. The attenuation of the secondary lobes with respect to the main lobe in this window is 92 dB, and the width of the main lobe is $W_m = 8$ bins. The frequency resolution achieved with this window is:

$$\Delta f = W_m \frac{f_s}{M} \approx 44 \text{ Hz}. \tag{2}$$

Note that in low registers, this frequency resolution could be insufficient for one-semitone distances. This situation has been considered in the *Beating reduction* block of the *Harmonic reorganization* stage (see details in Section III-C).

B. Estimation and Tracking of Sinusoids

The sinusoidal component is estimated frame by frame by analyzing the existing peaks within the spectrum, since a stable sharp peak in frequency corresponds to a stable sinusoid. According to the chosen approach [10], a local maximum above a certain threshold t is considered a peak. A peak is detected at $k = k_r$ if $|X_w[k_r]|$ is larger than its neighboring values in the magnitude spectrum and larger than t. Note that the precise frequency value k_r^* of the r sinusoid could correspond to a non-integer bin value. So, we estimate k_r^* by finding the maximum of the parabola that fits $|X_w[k_r - 1]|$, $|X_w[k_r]|$ and $|X_w[k_r + 1]|$, which is computed through parabolic interpolation as explained in [13]. Additionally, only the largest thirty peaks are selected to avoid noisy regions to be estimated as sinusoids.

Once the peaks have been detected in a frame of the signal, a temporal tracking is performed to group them into the same partial. In this way, we can process each partial independently to maintain the naturalness of the sound. The chosen method for time tracking, proposed in [10], is simple but effective. It connects sinusoids that are close in time, frequency and magnitude. Note that we have assumed that the input sounds are static chords with partials that remain relatively stable along time, so the time tracking approach chosen works well. With our set of parameters, the algorithm considers two spectral peaks to be grouped into the same partial if they fall within a three dimensional mask defined as follows:

(a) they are close in time (< 70 ms)
(b) they are close in frequency (< 0.2 semitones)
(c) they are close in magnitude (< 20 dB)

The mask is wide to allow the tracking of beating partials, which could oscillate in magnitude and frequency. If more than two consecutive peaks fall into the same mask, the Euclidean distance in the three dimensions, time-frequency-magnitude, is considered to decide the correct track of each partial.

The short partials are discarded because we consider that they do not contribute to the perceived dissonance. According to Moore [14], 200 ms is an acceptable duration to correctly perceive the pitch of a sound, therefore we discard all the partials shorter than this value. The discarded short sinusoids are not lost, but kept in the residual component of the signal.

C. Residual Component Extraction

The usual procedure to extract the residual part is to subtract a synthesized version of the sinusoidal component from the original signal: $x_{res}[n] = x[n] - x_{sin}[n]$. The quality of the residual component is directly dependent on the sinusoidal estimation. If the sinusoidal component is properly estimated, the residual component should contain just transients and noise. Observe that phase coherence in the subtraction $x[n] - x_{sin}[n]$ is very important, otherwise a clean residual part would not be obtained.

Since the residual component is not processed at all, the transients and noisy aspects of the signals remain unaltered in the output signal.

D. Multiple-f_0 Estimation

The *Multiple-f_0 estimation* stage analyses the input chord in order to compute a vector of estimated fundamental frequencies $\widehat{\mathbf{f_0}} = [\widehat{f}_{01}, \widehat{f}_{02} \dots \widehat{f}_{0n}]$. Ideally, $\widehat{\mathbf{f_0}}$ would equal the vector $\mathbf{f_0} = [f_{01}, f_{02} \dots f_{0n}]$, which contains the actual fundamental frequencies of the input chord. We assume that these frequencies do not change along time, i.e. we are dealing with an isolated chord. If the input signal consists of a sequence of chords, it must be segmented by the user in order to process each chord separately.

Many f_0 estimation methods have been proposed in the literature [15]–[18]. In our approach, we have used the multiple-f_0 estimation algorithm proposed by Klapuri in 2005 [16] because it is relatively straightforward to implement and it outperforms other reference methods (such as [17] and [18]). This method consists of a computational model of the human auditory periphery, followed by a periodicity analysis mechanism. Estimation of multiple fundamental frequencies is achieved by cancelling each detected sound from the mixture and by repeating the estimation process with the residual. Therefore, three steps can be distinguished:

1) *Auditory filter bank and neural transduction:* The acoustic signal $x(n)$ is filtered by a set of auditory filters uniformly distributed in the critical-band scale [19]. The auditory nerve signal for each channel c is then modeled by a cascade of (i) compression, (ii), half-wave rectification and (iii) low pass filtering.
2) *Periodicity analysis:* In this stage, each channel is analyzed through several operations based on the Fourier Transform. The periodicity information of all the channels is combined to generate a summary magnitude spectrum (SMS). This information leads to the computation of the salience function $\lambda(\tau)$, which represents the strength of each period candidate τ. Finally, the function $\lambda(\tau)$ is normalized in order not to favour either high or low f_0s in order to generate the final salience function $\tilde{\lambda}(\tau)$.
3) *Iterative estimation and cancellation:* The global maximum of $\tilde{\lambda}(\tau)$ is a robust indicator of one of the correct f_0s in polyphonic signals. However, the next-highest weight was often assigned to half or twice of the firstly detected f_0. So, an iterative procedure has been developed in which the cumulative spectrum of the detected f_0s is synthesized and, then, subtracted from the original signal in order to iteratively remove the detected f_0s from the mixture.

We apply this algorithm frame by frame with the following parameters: sampling rate $= 44100$ Hz, window size $= 4096$ samples, hop-size $= 2048$ samples, minimum $f_0 = 50$ Hz, maximum $f_0 = 3$ kHz, 60 auditory filters from 60 Hz to 5 kHz, and compression factor $\nu = 0.33$ for the neural transduction modeling. The degree of polyphony is set to $n = 5$ by default, but this parameter that can be modified by the user.

III. Harmonic Reorganization Stage

The *Harmonic reorganization stage* (see Fig. 1 and 3) is the main contribution of this paper, since the methods employed have been specially designed for the goal described. This section is organized according to the block diagram shown in Fig. 3. There are two parallel paths in this block: the scale fitting of $\widehat{\mathbf{f_0}}$ and the processing of the sinusoidal component. The upper path computes the target fundamental frequencies vector $\widehat{\mathbf{f_0^*}}$ (explained in Section III-A) and then generates a grid of overtones

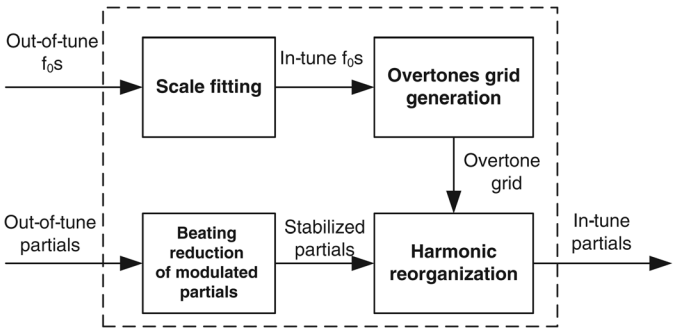

Fig. 3. Block diagram of the Harmonic reorganization stage.

corresponding to the corrected chord (Section III-B). The lower path processes the partials in two steps: first, a beating reduction stage removes the modulation of partials that are merged (due to low frequency resolution, as explained in Section III-C), and then the processed partials are shifted to conform to the in-tune output chord (Section III-D).

A. Scale Fitting

Let $\mathbf{f_S} = [f_{S1}, f_{S2} \dots f_{SN_S}]$ be a vector of frequencies corresponding to N_S notes of a given scale along several octaves. The *Scale fitting* block substitutes each value of the vector $\widehat{\mathbf{f_0}}$ by its closest value in $\mathbf{f_S}$ in order to generate $\widehat{\mathbf{f_0^*}} = [\widehat{f^*_{0\,1}}, \widehat{f^*_{0\,2}} \dots \widehat{f^*_{0\,n}}]$ (corrected f_0s). The distance between notes is measured in cents, because all the frequencies have been converted to MIDI numbers using:

$$MIDI = 69 + 12\ \log_2\left(\frac{f}{440}\right) \tag{3}$$

Therefore, the vector $\widehat{\mathbf{f_0^*}}$ contains the target frequency of each note in the 'in tune' version of the input chord. Note that this stage just handles symbolic information, and it does not apply any processing to the input chord.

We assume that the vector $\mathbf{f_S}$ is made of notes from of the Western tempered chromatic scale. The following three cases have been considered to cover a wide range of practical situations:

1) $\mathbf{f_S}$ *is the whole chromatic scale:* The first approach makes use of the tempered chromatic scale. We consider that the notes of the input chord are out-of-tune if they deviate from the MIDI scale [20] (i.e. the tempered chromatic scale). During the tuning adjustment, every element in $\widehat{\mathbf{f_0}}$ is simply rounded to the closest integer. In this case, no musical assumptions about the input data have been made and deviations larger than one semitone would imply rounding to an incorrect note. This approach is useful for cases in which there is not additional information about the input material.
2) $\mathbf{f_S}$ *depends on* $\widehat{f}_{01}$*:* The *Scale fitting* process can be improved if the user provides the system with some musical knowledge about the input chord. Different presets can be selected by the user: *Pentatonic scale*, *Major scale*, *Minor scale*, *Major chord (root position)*, *Major chord (any inversion)*, etc. The scale $\mathbf{f_S}$ is built upon the lowest note of the chord, $\widehat{f}_{01}$, which is taken as the root note. The resulting scale contains musically meaningful notes related to the input chord. In Fig. 4, an example in which the use of the major scale has been assumed is shown. The lowest note of the input chord is C, and the used scale is C major. The note $F\#4 + 9$ cents has been moved to $G4$ because $F\#4$ is not admitted by the C major scale. The system implemented allows the user to define customized musical constraints. This is useful if the user knows which type of chords must be obtained.
3) $\mathbf{f_S}$ *is customized by the user:* Sometimes, the user knows exactly all the notes of the input chord. In such case, the user provides the system with all the notes that comprise the input chord. The user imposes $\mathbf{f_S} = \mathbf{f_0}$, and so each element in $\widehat{\mathbf{f_0}}$ is rounded to the desired frequency. The customization of $\mathbf{f_S}$ by the user allows the system to achieve good results with uncommon chords.

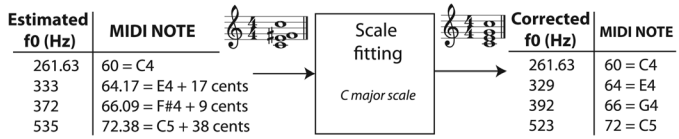

Fig. 4. Adjustment with musical restrictions of a largely out-of-tune C major chord.

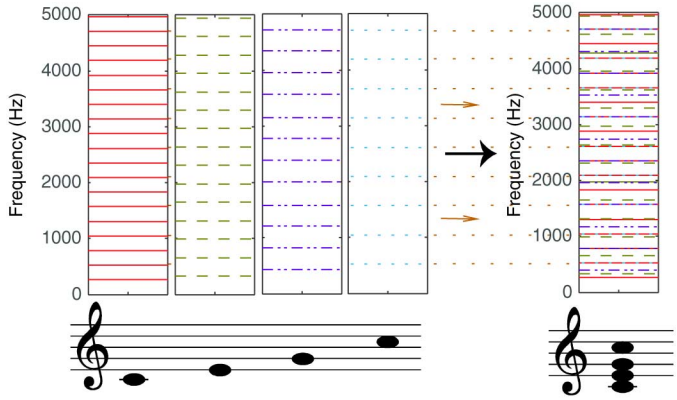

Fig. 5. Generation of overtone grid. The overtones of every note (left) are combined into a single grid for the complete chord (right).

B. Overtone Grid Generation

At this stage, the target frequencies of the notes comprising the output chord, $\widehat{\mathbf{f_0^*}} = [\widehat{f^*_{0\,1}}, \widehat{f^*_{0\,2}} \dots \widehat{f^*_{0\,n}}]$, are known. Then, we compute the partials structure $\widehat{\mathbf{f^*_{H\,whole}}}$ of the whole chord to define the so called *overtone grid*. The overtone grid is made of a combination of the harmonics of every f_0^* within the corrected chord. We denote the harmonic structure of each note as follows:

$$\widehat{\mathbf{f^*_{H\,1}}} = [\widehat{f^*_{0\,1}}, 2\widehat{f^*_{0\,1}} \dots R\widehat{f^*_{0\,1}}] \tag{4}$$
$$\widehat{\mathbf{f^*_{H\,2}}} = [\widehat{f^*_{0\,2}}, 2\widehat{f^*_{0\,2}} \dots R\widehat{f^*_{0\,2}}] \tag{5}$$
$$\vdots \tag{6}$$
$$\widehat{\mathbf{f^*_{H\,n}}} = [\widehat{f^*_{0\,n}}, 2\widehat{f^*_{0\,n}} \dots R\widehat{f^*_{0\,n}}] \tag{7}$$

We have defined the maximum number of harmonics to be $R = 20$, since the energy of the harmonic content over this value can be assumed to be very low. Consequently, the minimum frequency of the overtone grid is the first harmonic of the lowest note, and the maximum frequency is the twentieth harmonic of

the highest note. The frequencies of all the partials of the chord are sorted in a single array in order to define $\widehat{\mathbf{f}^*_{\mathbf{H}\,\text{whole}}}$.

In Fig. 5, this process is illustrated for a C major chord in root inversion ($\widehat{\mathbf{f}^*_{\mathbf{0}}} \rightarrow [C4, E4, G4, C5]$). These notes correspond to the following frequency values in Hz:

$$\widehat{\mathbf{f}^*_{\mathbf{0}}} = [261.63, 329.63, 392.00, 523.25] \text{ Hz}$$

According to the explained procedure, the overtone grid of the whole chord would be:

$$\widehat{\mathbf{f}^*_{\mathbf{H}\,\text{whole}}} = [261.63, 329.62, 392.00, 523.25, 659.24, 784.00, 1046.5, \ldots, 10465] \text{ Hz} \quad (8)$$

Note that all the input notes comprising the chord are supposed to be harmonic, i.e. the overtones are placed in multiples of the fundamental frequency. Typically, major and minor chords contain harmonically related notes, so there is a large number of overlapped overtones. Indeed, this overlapping reduces the perceived dissonance, because beating partials are avoided [21].

The overtone grid generation is robust to certain types of errors in the multiple-f_0 estimation. For instance, an error in the degree of polyphony is not critical if the overlap between harmonics within the chord is relatively high. This is especially noticeable when one note and its octave are present in the chord, since the upper octave does not contribute to a more complete overtone grid. In the same way, octave errors in the multiple-f_0 estimation process (specially when $\widehat{f_{01}} = f_{01}/2$) or fifth errors do not introduce significant changes to the harmonic structure of the whole chord.

C. Beating Reduction of Modulated Partials

In this section, the behavior of merged beating partials (usually found in out-of-tune sounds at low frequencies) is analyzed and a method to avoid them is proposed.

Recall that, as shown in eq. (2), the frequency resolution achieved in the Analysis stage is around 44 Hz. This is less than one semitone for notes below $F\#5$ (due to the logarithmic scaling of the frequency axis). This resolution is definitely too low to resolve beating sinusoids in low frequency, out-of-tune sounds. When two partials are not independently estimated, a single peak with periodic oscillations of amplitude and frequency is detected instead of two stable peaks (i.e. the partials are *merged* during the analysis). These oscillations are usually found in out-of-tune sounds and they increase the perceived dissonance [6].

1) Mathematical Analysis of Beating Sinusoids: We present a mathematical analysis of two common situations in order to justify the proposed beating reduction method.

a) Beating sinusoids with the same amplitude: Let $x(t)$ be the sum of two sinusoids with similar frequencies and equal amplitude. As shown in eq. (9), this is equivalent to a product of a carrier and a modulating sinusoid (amplitude modulated tone):

$$\begin{aligned} x(t) &= A\cos((\omega_0 + \Delta\omega)t) + A\cos((\omega_0 - \Delta\omega)t) \\ &= 2A \cdot \cos(\Delta\omega_0 t) \cdot \cos(\omega_0 t) \end{aligned} \quad (9)$$

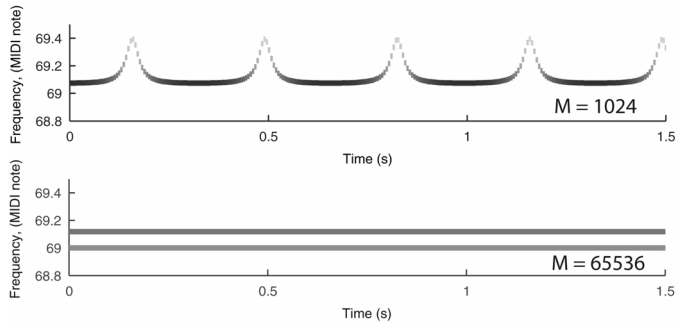

Fig. 6. Peak frequency spectrogram of two beating sinusoids (440 Hz and 443 Hz) with different amplitude (-12 dB and -9 dB respectively) for two different window sizes: $M = 1024$ and $M = 65536$.

If the analysis window used with the STFT is too narrow, the representation of the result becomes close to a amplitude modulated tone. On the other hand, if the window is large enough, two different peaks can be distinguished.

The instantaneous frequency of $x(t)$ in this case is constant and equals ω_0. Due to this, small window sizes provide a constant frequency value but a modulated amplitude.

b) Beating sinusoids with different amplitude: Let $x(t)$ be a sum of two sinusoids with similar frequencies and different amplitudes: $x(t) = A_1\cos((\omega_0+\Delta\omega)t)+A_2\cos((\omega_0-\Delta\omega)t)$. In this case, according to [22], this signal can be expressed as:

$$\begin{aligned} x(t) = &\sqrt{A_1^2 + A_2^2 + 2A_1A_2\cos(2\Delta\omega\, t)} \\ &\cdot \cos\left(\omega_0\, t + \arctan\left(\Delta\omega\, t \cdot \frac{A_1 - A_2}{A_1 + A_2}\right)\right) \end{aligned} \quad (10)$$

The instantaneous frequency, $\hat{\omega}(t)$, can be extracted by taking the derivative of the instantaneous phase in eq. (10):

$$\hat{\omega}(t) = \omega_0 + \frac{(A_1^2 - A_2^2)\Delta\omega}{A_1^2 + A_2^2 + 2A_1A_2\cos(2\Delta\omega t)} \quad (11)$$

The instantaneous frequency is constant when $A_1 = A_2$. However, if $A_1 \neq A_2$, a type of periodic modulation in frequency appears, which explains the presence of strange periodic patterns in the STFT of polyphonic sounds out-of-tune. Fig. 6 shows the peak frequency spectrogram of two beating sinusoids as an example of this situation. The peak frequency spectrogram only shows the local maxima over a given threshold (-80 dB in our case) of the common spectrogram.

2) Proposed Method for Beating Reduction: The proposed beating reduction method removes amplitude and/or frequency modulations. Several assumptions about the input chord are made:

- It is a stable out-of-tune chord whose f_0s do not vary along time. Vibratos or glissandos are not addressed.
- Attack-Decay-Sustain-Release envelope (ADSR) [23] is assumed in the amplitude of the partials. Tremolo or atypical envelope patterns are not addressed.

The contribution of the beating reduction stage to the perceived consonance strongly depends on the type of signal. The improvement attained is especially noticeable when the chord is simple and stable (e.g. perfect major/minor synthetic chords in our evaluation dataset). In contrast, if the input is a significantly time-varying signal in terms of timbre and frequency, the

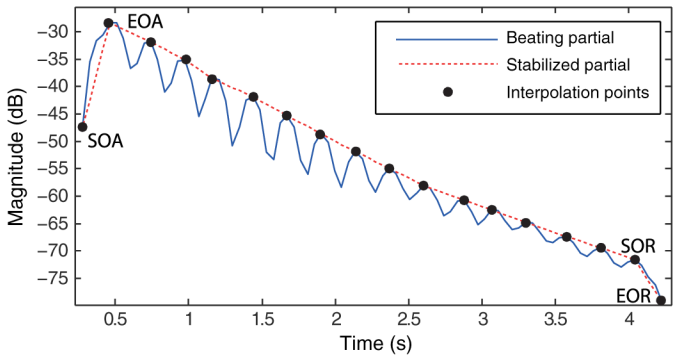

Fig. 7. Magnitude envelope stabilization of a beating partial. The attack and the release of the note are fixed, and only beatings are removed.

beating reduction scheme can produce a lack of naturalness or introduce artefacts, in such case, better results are achieved if it is disabled. Three steps are performed in our beating reduction scheme: Envelope reconstruction, Frequency stabilization and Phase reconstruction.

a) Envelope reconstruction: First, the amplitude envelope is processed to fit the ADSR model by using a variation of the envelope reconstruction procedure presented in [23]. The ADSR model considers several split-points in the envelope of the signal: start of attack (*SOA*), end of attack (*EOA*), start of release (*SOR*), and end of release (*EOR*) (see Fig. 7). Our approach defines each split-point as follows:

- SOA: First amplitude value above -80 dB (noise threshold).
- EOA: First local maximum of the envelope.
- SOR: Last local maximum of the envelope.
- EOR: Last amplitude value above -80 dB.

The envelope reconstruction proposed in [23] only considers these four split-points. In our approach, we also take every local maximum between the EOA and the SOR (interval called sustain or decay) as a set of tracking points in order to faithfully fit the original envelope. The interval between each split-point is modeled by an exponential curve, as proposed in [23]. Note that, if the amplitude is represented in dBs (logarithmic scale), exponential curves become straight lines. Our approach for envelope reconstruction performs linear interpolation of the split and tracking points in a logarithmic scale.

b) Frequency stabilization: Frequency modulations might appear in beating sinusoids, as shown in eq. (11). The proposed technique removes such modulations by time-averaging the measured frequencies of the partial. From an analytical point of view, this can be understood as averaging the instantaneous frequency $\hat{\omega}(t)$ calculated in (11). Note that the average of $\hat{\omega}(t)$ can be bounded in the following interval:

$$\left[\omega_0 + \frac{A_1 - A_2}{A_1 + A_2}\Delta\omega,\ \omega_0 + \frac{A_1 + A_2}{A_1 - A_2}\Delta\omega\right] \tag{12}$$

simply substituting $\cos(2\Delta\omega t)$ in (12) by its maximum and minimum values, 1 and -1, respectively:

This result is useful to know the approximate range of the computed average. Later, this average will be moved to a fixed grid to finally generate the output, as it will be explained in later sections.

c) Phase reconstruction: If the frequency of a partial is changed, the phase evolution has to be adapted to guarantee the continuity of the sinusoid. The phase state for every frame l is estimated as follows:

$$\phi_r^l = \phi_r^{l-1} + \frac{H \cdot 2\pi f_r}{f_s} \tag{13}$$

Where ϕ_r^l is the new phase value for partial r in frame l, H is the hop-size, f_r is the new frequency value and f_s is the sampling rate.

D. Harmonic Reorganization

The final step in the processing is *harmonic reorganization*, which shifts each partial of the input chord in order to fit the overtone grid $\widehat{\mathbf{f}^*_{\mathbf{H}\,\text{whole}}}$ defined in Section III-B, thus, the perceived consonance in sustained chords is improved. We observed that harmonic reorganization is responsible for the highest dissonance reduction in most of the cases evaluated. This process is performed in several steps:

1) **Parametrization of individual partials:** As explained in Section II-A, each partial r is defined by three vectors: $(\hat{A}_r^l, \hat{f}_r^l, \hat{\phi}_r^l)$. We take the average $\overline{\hat{f}_r} = \sum_{l=1}^{l=L} \hat{f}_r^l / L$ as a single representative value of the frequency of the partial. The complete array of average frequencies corresponding to all the partials is called $\widehat{\mathbf{f}_{\mathbf{H}\,\text{whole}}}$, according to the notation used in Section III-B.
2) **Search for the target frequency of each partial:** The vector $\widehat{\mathbf{f}_{\mathbf{H}\,\text{whole}}}$ and the overtone grid $\widehat{\mathbf{f}^*_{\mathbf{H}\,\text{whole}}}$ (defined in Section III-B) are expressed in MIDI numbers, using eq. (3). Then, for each element in $\widehat{\mathbf{f}_{\mathbf{H}\,\text{whole}}}$, the nearest element in $\widehat{\mathbf{f}^*_{\mathbf{H}\,\text{whole}}}$ in semitones is taken as its target frequency.
3) **Shifting the partials:** Let $f_{diff} = \widehat{f^{*r}_{H\,\text{whole}}} - \overline{\hat{f}_r}$ be the distance, in semitones, to the target frequency of the partial r, where $\overline{\hat{f}_r}$ is the average frequency value of the partial r, and $\widehat{f^{*r}_{H\,\text{whole}}}$ is the closest frequency value in the overtone grid (target frequency). Then, the frequency shift applied to each partial is:

$$\Delta f = \left\{ \begin{array}{cc} f_{\text{diff}} & \text{si} \quad f_{\text{diff}} < 3 \text{ semitone} \\ 0 & \text{si} \quad f_{\text{diff}} > 3 \text{ semitone} \end{array} \right\} \tag{14}$$

This condition is applied to avoid shifts of partials to excessively distant positions. In addition, the partials out of the minimum and the maximum frequencies of the overtone grid (respectively, the first harmonic of the lowest note and the 20th harmonic of the highest note of the chord, respectively; see Section III-B) remain unchanged.

Finally, the frequency vector of the corrected partial is defined as $\hat{f}_r^{*l} = \hat{f}_r^l + \Delta f_r$, and the complete set of parameters for each partial r is: $(\hat{A}_r^l, \hat{f}_r^{*l}, \hat{\phi}_r^l)$.

In Fig. 8, the peak frequency spectrogram of an out-of-tune guitar chord at different stages of the system developed is shown. A reference spectrum obtained analyzing the same chord played with an in-tune guitar is also presented (Fig. 8(d)). Frequency and amplitude oscillations along time are noticeable

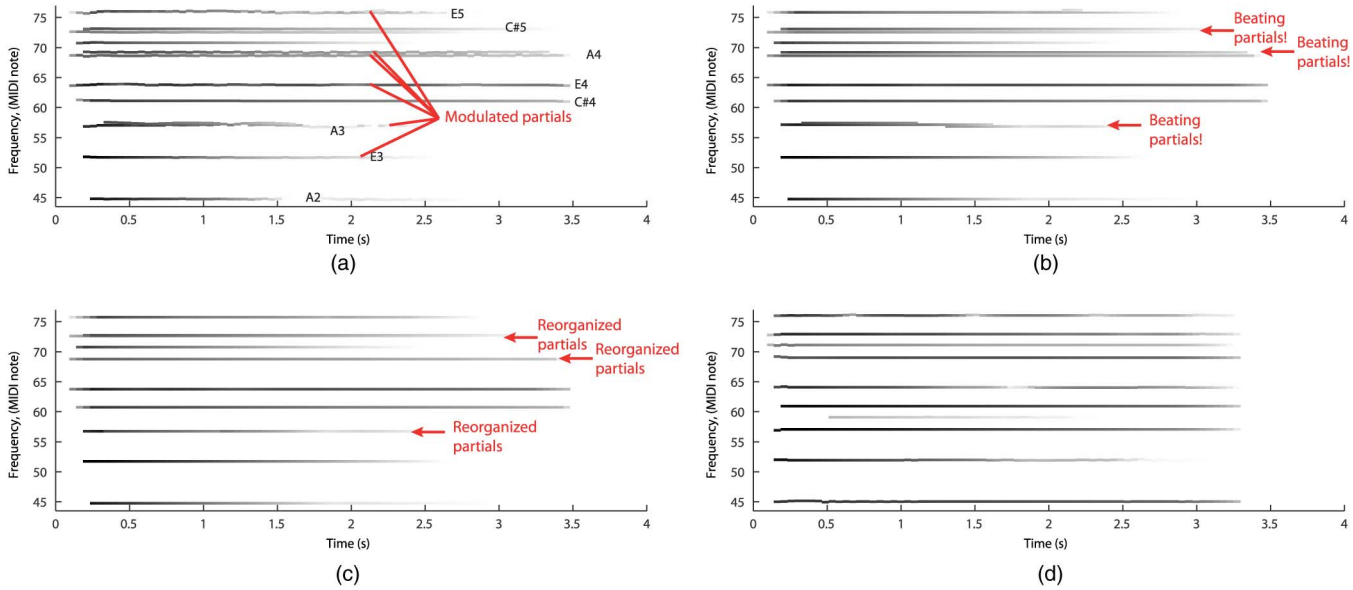

Fig. 8. Detail of the peak frequency spectrograms of several versions of a A major chord played with acoustic guitar. **(a)** Original out-of-tune chord, whose notes are (name and MIDI number): $A2-33$ cents $=44.77$, $E2-33$ cents $=51.77$, $A3+27$ cents $=57.27$, $C\#4+16$ cents $=61.16$, $E4-20$ cents $=63.80$. **(b)** Original chord after the *beating reduction* stage **(c)** Original chord after the *beating reduction* and the *harmonic reorganization* stages. **(d)** A major chord played with a real in-tune guitar.

in the partials of the original chord (the reason for this was discussed in Section III-C). The beating reduction stage removes these oscillations, and then the harmonic reorganization shifts the partials in order to produce a spectrogram more similar to a real in-tune chord. As long as the most important f_0s are correctly estimated, the harmonic reorganization stage is almost free of artifacts for sustained and stable chords.

In the case of time-varying signals, the analysis stage produces less representative parameters, and beating reduction together with harmonic reorganization stages can produce artefacts. In Section V, we provide some comments about specific examples.

IV. Synthesis Stage

With the parameters of the partials modified to be in-tune, a resynthesis stage of the sinusoidal component is performed. This process is carried out frame by frame through an *overlap-add* process. Every frame is synthesized in the frequency domain and converted into a short windowed waveform making use of the inverse FFT. Finally, all the frames are overlapped and added to give rise to the final output waveform. Some previous methods are directly based on the inverse FFT operation, such as [10] or [24].

In the chosen approach [10], the synthesis of one single sinusoid in the frequency domain is straightforward due to the use of Blackman-Harris 92 dB window. In this type of window, secondary lobes are negligible and only the main lobe is needed to accurately represent a sinusoid in the spectrum. For each sinusoid, a lobe containing frequency, magnitude and phase information is placed in the spectrum.

The synthesis of the complete waveform is performed overlapping all the frames. For the case of the Blackman-Harris window, the constant overlap factor is 75%. The spectrum of each frame is used to create a short windowed waveform through the IFFT, and this process is repeated for all the frames. Then all the short waveforms are overlapped and added to create the final sinusoidal output [12]. The last step of the synthesis is adding the residual component (which was extracted during the analysis stage) to give rise to the final output waveform.

V. Evaluation Methodology

The evaluation methodology is based on questionnaires that have been answered by 31 expert musicians. Such musicians have been asked to rate the perceived consonance of a set of chords in a dataset.

A. Dataset

The dataset contains 18 different chords. More specifically, there are 3 versions of 6 different types of out-of-tune chords[3]. The sounds are increasingly complex (from synthetic stable sounds to real chamber ensembles). Most of the chosen sounds are major chords, because they are very common in Western music and the difference between in-tune and out-of-tune chords is quite noticeable:

- Type of out-of-tune chords
  1) C Major played with 6 harmonic complex tones with ADSR envelope. Notes: $C4$, $E4+11$ cents, $G4-21$ cents, $C5+30$ cents. The single notes were artificially synthesized and then combined.
  2) C minor played with 6 harmonic complex tones with ADSR envelope. Notes: $C4$, $E\flat4+13$ cents, $G4+17$ cents, $C5-32$ cents. As the previous case, the notes were artificially synthesized and then combined.
  3) A Major played with a real acoustic guitar. Notes: $A2-33$ cents, $E2-33$ cents, $A3+27$ cents, $C\#4+16$ cents, $E4-20$ cents. The guitar was deliberately left out-of-tune to sound strongly dissonant, and all the strings

[3]These sounds are available at http://www.atic.uma.es/polytuning

were played together. Then, each note was separately analyzed to find out its accurate frequency value.

4) D Major played with a real acoustic guitar. Notes: $D3-30$ cents, $A3+28$ cents, $D4+15$ cents, $F\#4+3$ cents. The recording procedure was the same as in the previous case.
5) $B\flat$ Major played with a real woodwind quartet: $B\flat 2$, $F3-44$ cents, $B\flat 3-50$ cents, $D5+31$ cents. The notes of the chord were extracted from RWC database [25], carefully pitch-shifted and then combined.
6) C Major played with a real string quartet: $C3-6$ cents, $E3-7$ cents, $C4+30$ cents, $G4-73$ cents. This chord was generated in the same way as the previous one.

- Versions
  (A) Unprocessed chord.
  (B) Processed (developed approach).
  (C) Processed (Melodyne Editor).

In version B, we have used the following parameters for all the sounds: sampling rate $= 44100$ Hz, window size $M = 8001$ samples, FFT size $N = 8192$ samples, number of partials per note $R = 30$ and degree of polyphony $n = 5$. In version C, the degree of polyphony and the notes of the chord have been manually adjusted for each case in order to achieve the best results. In next sections, sounds will be identified by combining the number of the chord and the type of version, i.e. *1.A* would be the first chord in the unprocessed version.

B. Evaluation

1) Subjects: For the evaluation, 31 musicians were interviewed. All of them have passed a minimum of 7 years of formal music education. There were 16 male and 15 female individuals, and most of the subjects' age is below 25. The interviewed musicians play very different instruments (woodwind, piano, percussion…), so there is no predominant instrument.

2) Questionnaires: The subjects were asked to rate from 1 to 10 the perceived consonance of 18 sounds. For every group of three versions (A,B and C), they were also asked to choose, globally, the best version if they had to use such chord in a musical context.

3) Statistics: Different measures have been taken from the questionnaires for each sound in the dataset.

- Mean perceived consonance μ_c.
- Standard deviation of the perceived consonance σ_c.
- Percentage of times that each version has been chosen as the best option among the three versions.

VI. Results & Discussion

The results obtained are shown in Table I. In all cases, the multiple-f_0 estimation stage (with a degree of polyphony set to $n = 5$) perfectly identified the most important f_0s of the chord, and so the target overtone grid was correct. In the chords 1.x, 2.x, 3.x, 4.x and 5.x, the chosen f_S is the tempered chromatic scale (no musical assumptions are made about the input). In 6.x, the note $G4-73$ cents could be incorrectly rounded to $F\#4$ because of the large deviation of the partials, so the user must correct f_S with the major scale built upon $\widehat{f}_{01} = C3$, i.e. C major scale.

TABLE I
Questionnaires Results. **x.A:** Unprocessed Sound; **x.B:** Developed Approach; **x.C:** Melodyne Editor

Chord version	*Perceived consonance [1-10]*	*Chosen as best result*
1.A Original	$\mu_c = 3.48$ $\sigma_c = 1.48$	3.2%
1.B Our approach	$\boldsymbol{\mu_c = 6.64}$ $\boldsymbol{\sigma_c = 2.05}$	**77.4%**
1.C Melodyne	$\mu_c = 5.48$ $\sigma_c = 1.80$	19.35%
2.A Original	$\mu_c = 2.67$ $\sigma_c = 1.30$	6.45%
2.B Our approach	$\boldsymbol{\mu_c = 5.35}$ $\boldsymbol{\sigma_c = 2.25}$	**74.2%**
2.C Melodyne	$\mu_c = 3.96$ $\sigma_c = 1.87$	19.3%
3.A Original	$\mu_c = 4.61$ $\sigma_c = 1.89$	3.2%
3.B Our approach	$\boldsymbol{\mu_c = 7.19}$ $\boldsymbol{\sigma_c = 1.86}$	**83.9%**
3.C Melodyne	$\mu_c = 5.83$ $\sigma_c = 2.35$	9.7%
4.A Original	$\mu_c = 4.32$ $\sigma_c = 1.81$	3.2%
4.B Our approach	$\boldsymbol{\mu_c = 7.09}$ $\boldsymbol{\sigma_c = 1.68}$	**71%**
4.C Melodyne	$\mu_c = 6.19$ $\sigma_c = 1.99$	25.8%
5.A Original	$\mu_c = 2.19$ $\sigma_c = 1.27$	0%
5.B Our approach	$\boldsymbol{\mu_c = 4.03}$ $\boldsymbol{\sigma_c = 2.33}$	**32%**
5.C Melodyne	$\mu_c = 4.64$ $\sigma_c = 2.38$	68%
6.A Original	$\mu_c = 1.54$ $\sigma_c = 0.80$	0%
6.B Our approach	$\boldsymbol{\mu_c = 5.54}$ $\boldsymbol{\sigma_c = 2.15}$	**77.4%**
6.C Melodyne	$\mu_c = 4.77$ $\sigma_c = 1.96$	22.6%

In the case of synthetic sounds (chords 1.x and 2.x) the results show a clear improvement in the consonance of the processed sounds. Unprocessed sounds were strongly perceived as dissonant, whereas the processed ones improved the consonance rating around 3 points. Moreover, the developed approach provides better results than Melodyne Editor for the case of synthetic sounds, since in this case noticeable beating partials are still present in the processed chords. In all comparisons, a *t-Student* test (with $p < 5\%$) revealed statistical validity [26].

The case of the acoustic guitar (chords 3.x and 4.x) is especially interesting, since it is a very common instrument and the results are quite satisfactory. More than 70% of the subjects considered the selected approach to be better than Melodyne Editor. We conclude that plucked string instruments are very appropriate to be processed with the selected approach, since the assumed partial stability holds true for most of the cases.

In the case of a woodwind quartet (5.x), Melodyne performs better than our approach, with a perceived consonance of 4.63 and 4.02 respectively. If both versions are carefully compared, it can be noticed that the difference between them in terms of dissonance is mild, but Melodyne produces a more natural result.

In the case of the strings quartet (6.x) Melodyne does not properly separate the various notes of the chord, so 6.C is still dissonant and unnatural compared to 6.B.

VII. Conclusions

In this paper, a novel method for automatic reduction of dissonance in recorded out-of-tune chords has been proposed. The method shown manipulates the harmonic structure of the input chord as a whole in order to make it fit onto a previously estimated overtone grid. This scheme has applications for professional post-processing tasks of recorded audio. The most prominent commercial tool for polyphonic sound processing is *Melo-*

dyne Editor, which is based on source separation and note-level processing. However, Melodyne is not really suitable for out-of-tune sounds, since it is not able to effectively separate two notes very close in frequency [5].

Additionally, the method proposed reduces amplitude and/or frequency modulations due to unresolved close partials by means of a novel beating reduction algorithm. This algorithm produces a noticeable performance improvement for simple and sustained chords.

The selected approach is based on an *analysis-resynthesis* schema with a *sinusoidal plus residual* model. The system is composed of three stages: *Analysis*, *Harmonic reorganization* and *Synthesis*. In the *Analysis stage*, the sinusoidal and residual components are separated, the partials of the signal are tracked, and the various f_0 comprising the input chord are estimated. In the *Harmonic reorganization stage*, the partials are stabilized and shifted to generate the parameters of the in-tune output chord. Finally, in the *Synthesis stage*, the final waveform is generated using an overlap-add process.

Our approach assumes that the f_0s of the input do not change along time (i.e. it is an isolated chord), the envelope of the signal corresponds to the ADSR model [27] and the notes of the chord are relatively harmonic (i.e. the overtones are placed in multiples of the f_0). Therefore, input chords with vibrato and/or tremolo are not addressed. However, plucked strings instruments, such as the guitar, are very appropriated to be processed with our approach, as demonstrated in Section VI. The achieved results with other type of instruments are varied, but we observed they are quite acceptable as long as the signal is stable in terms of timbre and frequency.

The performance of the system has been evaluated by 31 expert musicians and it has been compared against the performance of the professional reference tools for this task (Melodyne Editor). In the results, the proposed approach shows an important reduction of the inner dissonance of the chords. For most of the cases evaluated, our method provides better results than Melodyne Editor. The most interesting results are found with acoustic guitar recordings, which are almost free of artefacts after processing.

In our future work, we intend to overcome the current limitations of our approach. For instance, the developed system can benefit from one of the existing chord-segmentation methods [28] to deal with sequences of chords. Additionally, the analysis of time-varying sounds (e.g. vibrato or tremolo) can be addressed with predictive time-tracking algorithms (e.g. HMM-based approaches [29]). Moreover, further research is needed to really address the importance of beating reduction in time-varying chords. Furthermore, the system can be easily adapted to process inharmonic sounds if the overtone grid is adapted to the specific inharmonicity of the input notes [30], [31]. Finally, our system can be also adapted to other temperaments, such as Pythagorean or Zarlino [2], if the scale vector $\mathbf{f_s}$ is redefined.

REFERENCES

[1] O. Gurney, "An old Babylonian treatise on the tuning of the harp," *Iraq*, vol. 30, no. 2, pp. 229–233, 1968.

[2] J. Barbour, *Tuning and temperament: a historical survey*. New York, NY, USa: Dover, 2004 [Online]. Available: http://books.google.com/books?id=G-pG77pmlp4C

[3] R. Clark, "Mixing, recording and producing techniques of the pros," Thomson Course Technology PTR, 2006 [Online]. Available: http://bks9.books.google.co.ke/books?id=l4oJAQAAMAAJ

[4] P. Neubäcker, "Sound-object oriented analysis and note-object oriented processing of polyphonic sound recordings," U.S. patent 8,022,286, Sep. 20, 2011.

[5] J. Akin, "Celemony melodyne editor review," *Mix Mag. Profess. Audio and Music Product.*, Feb. 2010 [Online]. Available: http://mixonline.com/gear/reviews/celemony-melodyne-editor-0210

[6] R. Plomp and W. Levelt, "Tonal consonance and critical bandwidth," *J. Acoust. Soc. Amer.*, vol. 38, no. 4, pp. 518–560, 1965.

[7] W. Piston and M. DeVoto, *Harmony*. New York, NY, USA: Norton, 1987 [Online]. Available: http://books.google.com/books?id=MozDQgAACAAJ

[8] N. Cazden, "Sensory theories of musical consonance," *J. Aesthetics Art Criticism*, vol. 20, no. 3, p. 301, 1962.

[9] J. A. Swets, D. M. Green, and W. P. Tanner, "On the width of critical bands," *J. Acoust. Soc. Amer.*, vol. 34, no. 1, p. 108, 1962 [Online]. Available: http://link.aip.org/link/JASMAN/v34/i1/p108/s1&Agg=doi

[10] X. Serra, "A system for sound analysis/transformation/synthesis based on a deterministic plus stochastic decomposition," Ph.D. dissertation, Univ. of Stanford, Stanford, CA, USA, 1989.

[11] A. Oppenheim, R. Schafer, and J. Buck, *Discrete-Time Signal Processing*, ser. ser. signal processing series, A. V. Oppenheim, Ed. Englewood Cliffs, NJ, USA: Prentice-Hall, 1999, vol. 23 [Online]. Available: http://link.aip.org/link/ELPWAQ/v23/i2/p157/s1&Agg=doi, no. 2

[12] J. Smith, *Spectral Audio Signal Processing, October 2008 Draft*. Stanford, CA, USA: Stanford Univ., Oct., 2013 [Online]. Available: http://ccrma.stanford.edu/~jos/sasp/

[13] M. Abe and J. O. Smith, "Design criteria for the quadratically interpolated FFT method (i): Bias due to interpolation," no. STAN-M-114, 2004 [Online]. Available: https://ccrma.stanford.edu/files/papers/stanm114.pdf

[14] B. Moore, "Frequency difference limens for short-duration tones," *J. Acoust. Soc. Amer.*, vol. 54, no. 3, pp. 610–619, 1973 [Online]. Available: http://www.ncbi.nlm.nih.gov/pubmed/4754385

[15] C. Yeh, A. Roebel, and X. Rodet, "Multiple fundamental frequency estimation and polyphony inference of polyphonic music signals," *IEEE Trans. Audio, Speech, Lang. Process.*, vol. 18, no. 6, pp. 1116–1126, Aug. 2010.

[16] A. P. Klapuri, "A perceptually motivated multiple-f0 estimation method," in *Proc. IEEE Workshop Applicat. Signal Process. Audio Acoust.*, 2005, pp. 291–294.

[17] A. P. Klapuri, "Multiple fundamental frequency estimation based on harmonicity and spectral smoothness," *IEEE Trans. Speech Audio Process.*, vol. 11, no. 6, pp. 804–816, Nov. 2003.

[18] T. Tolonen and M. Karjalainen, "A computationally efficient multipitch analysis model," *IEEE Trans. Speech Audio Process.*, vol. 8, no. 6, pp. 708–716, Nov. 2000.

[19] M. Slaney, "An efficient implementation of the Patterson-Holdsworth auditory filter bank," Apple Computer, Perception Group, Tech. Rep, 1993.

[20] "Complete MIDI 1.0 Detailed Specification," M. M. Association, 1999/2008 [Online]. Available: http://www.midi.org/techspecs/gm.php

[21] H. Helmholtz, *Die Lehre den Tonempfindungen als physiologische Grundlage fur die Theorie der Musik / von H. Helmholtz*. Braunschweig, Germany: F. Vieweg und Sohn, 1877.

[22] R. Maher, "An approach for the separation of voices in composite musical signals," Ph.D. dissertation, Univ. of Illinois, Urbana, IL, USA, 1989.

[23] K. Jensen, "Envelope model for isolated musical sounds," in *Proc. 2nd COST-G6 Workshop Digital Audio Effects (DAFx99)*, Trondheim, Norway, Dec. 1999, pp. 35–39.

[24] P. Rodet and X. Depalle, "Spectral envelopes and inverse FFT synthesis," *Audio Eng. Soc. Conv. 93*, vol. 10, 1992 [Online]. Available: http://www.aes.org/e-lib/browse.cfm?elib=6740

[25] M. Goto, H. Hashiguchi, T. Nishimura, and R. Oka, "RWC music database: Music genre database and musical instrument sound database," in *Proc. ISMIR*, 2003, vol. 3, pp. 229–230.

[26] S. L. Zabell, "On student's 1908 article the probable error of a mean," *J. Amer. Statist. Assoc.*, vol. 103, no. 481, pp. 1–7, 2008 [Online]. Available: http://amstat.tandfonline.com/doi/abs/10.1198/016214508000000030

[27] G. Torelli and G. Caironi, "New polyphonic sound generator chip with integrated microprocessor-programmable ADSR envelope shaper," *IEEE Trans. Consumer Electron.*, vol. CE-29, no. 3, pp. 203–212, Aug. 1983.
[28] W. De Haas, "Music information retrieval based on tonal harmony," Ph.D. dissertation, Utrecht Univ., Utrecht, The Netherlands, 2012.
[29] P. Depalle, G. Garcia, and X. Rodet, "Tracking of partials for additive sound synthesis using hidden Markov models," in *Proc. ICASSP*, 1993, vol. 1, pp. 225–228 [Online]. Available: http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=319096
[30] I. Barbancho, L. Tardon, S. Sammartino, and A. Barbancho, "Inharmonicity-based method for the automatic generation of guitar tablature," *IEEE Trans. Audio, Speech, Lang. Process.*, vol. 20, no. 6, pp. 1857–1868, Aug. 2012.
[31] S. Dixon, M. Mauch, and D. Tidhar, "Estimation of harpsichord inharmonicity and temperament from musical recordings," *J. Acoust. Soc. Amer.*, vol. 131, pp. 878–887, 2012.

Emilio Molina received his degrees in Technical Telecommunications Engineering and Telecommunications Engineering from University of Málaga, Spain in 2007 and 2011, respectively. In 2013, he received his MSc in 'Sound and Music Computing' from Universitat Pompeu Fabra, Barcelona, Spain. He obtained the Professional Degree of Classic Piano at Conservatori del Liceu, Barcelona, Spain, in 2012. During his studies, he obtained the Best Final Year Project award from University of Málaga in 2007. He was nominated as finalist for the Best Final Year Project by the Official National Telecommunications Engineering Board in 2013. Currently, he is a PhD candidate at the Application of Information and Communication Technologies Research Group (ATIC) under the supervision of Lorenzo J. Tardón. His main research topic is the automatic analysis and processing of audio signals with special focus on singing voice and its applications.

Ana M. Barbancho received her degree in telecommunications engineering and her Ph.D. degree from University of Málaga, Málaga, Spain, in 2000 and 2006, respectively. In 2001, she also received her degree in solfeo teaching from the Málaga Conservatoire of Music. Since 2000, she has been with the Department of Communications Engineering, University of Málaga, as an Assistant and then Associate Professor. Her research interests include musical acoustics, digital signal processing, new educational methods, and mobile communications. Dr. Barbancho was awarded with the Second National University Prize to the Best Scholar 1999/2000 by the Spanish Ministry of Education in 2000 and with the 'Extraordinary Ph.D. Thesis Prize' by ETSI Telecomunicación of University of Málaga in 2007.

Lorenzo J. Tardón received his degree in Telecommunications Engineering from University of Valladolid, Valladolid, Spain, in 1995 and his Ph.D. degree from Polytechnic University of Madrid, Madrid, Spain, in 1999. In 1999 he worked for ISDEFE on air traffic control systems at Madrid-Barajas Airport and for Lucent Microelectronics on systems management. Since November 1999, he has been with the Department of Communications Engineering, University of Málaga, Málaga, Spain. Lorenzo J. Tardón is currently the head of the Application of Information and Communications Technologies (ATIC) Research Group. He has worked as main researcher of different projects on audio and music analysis. He is a member of several international journal committees on communications and signal processing. In 2011, he has been awarded the 'Premio Málaga de Investigación' by the Academies 'Bellas Artes de San Telmo' and 'Malagueña de Ciencias'. His research interests include serious games, audio signal processing, digital image processing and pattern analysis and recognition.

Isabel Barbancho (SM'10]) received her degree in telecommunications engineering and her Ph.D. degree from the University of Málaga (UMA), Málaga, Spain, in 1993 and 1998, respectively, and her degree in piano teaching from the Málaga Conservatoire of Music in 1994. Since 1994, she has been with the Department of Communications Engineering, UMA, as an Assistant and then Associate Professor. During 2013, she has been a Visiting Scholar at University of Victoria, Victoria, BC, Canada. She has been the main researcher in several research projects on polyphonic transcription, optical music recognition, music information retrieval, and intelligent content management. Her research interests include musical acoustics, signal processing, multimedia applications, audio content analysis, and serious games. Dr. Barbancho received the Severo Ochoa Award in Science and Technology, Ateneo de Málaga-UMA in 2009 and the 'Premio Málaga de Investigación 2011' Award from the Academies 'Bellas Artes de San Telmo' and 'Malagueña de Ciencias.'