DETECTION OF CLICKS USING SINUSOIDAL MODELING FOR THE CONFIRMATION OF THE CLICKS

Esteban Alvarez, Rafael Mendez and Guillermo Langwagen

Faculty of Engineering,
University ORT Uruguay

ABSTRACT

This article presents methods for clicks detection in degraded audio recordings. It begins with a brief description of the method implemented in first instance for the detection of clicks in audio sources based on linear prediction. Looking for an improvement of the results obtained with this method, we propose a method based on sinusoidal modeling for the confirmation of the clicks. This method discards clicks that were wrongly detected. This allows the detection of clicks of small amplitude avoiding wrong detections. The results obtained by this method are shown, confirming the good operation. Finally, the method implemented for detection of clicks in naturally degraded audio sources is presented.

1. INTRODUCTION

There are several distinct types of degradation common in audio sources. These can be broadly classified into two groups: localized degradations and global degradations. Localized degradations are discontinuities in the waveform, which affect only certain samples. The most common of this type of degradation are the clicks. Global degradations affect all samples of the waveform being hiss the most common.

The first step in the restoration of an audio source is the detection of the clicks. The algorithm to be implemented must fulfill the following requirements: detect most true clicks, detect properly the width of the clicks and avoid false detections that would degrade the signal even more. The detection methods are based on linear predictors.

2. DETECTION METHOD

In the following Sections we will explain some of the methods implemented to detect the clicks. These methods will be used before and in conjunction to sinusoidal modeling.

2.1. Linear Prediction

The audio signal will be modeled with a linear predictor that minimizes the mean squared error (see [1], [2] and [3]). With this method, the value of the predicted sample is based on a linear combination of the preceding samples:

$$\tilde{s}_n = -\sum_{k=1}^{p} a_k s_{n-k} \quad , \tag{1}$$

where

$\tilde{s}_n =$ predicted sample

$s_n =$ signal sample

$a_k =$ coefficient of the AR model

The prediction error (the difference between the value of the predicted sample and the original sample) is given by:

$$e_n = s_n - \tilde{s}_n = s_n + \sum_{k=1}^{p} a_k s_{n-k} \tag{2}$$

Considering σ_e as the standard deviation of the error signal, a sample belongs to a click if

$$|e_n| > K \cdot \sigma_e \tag{3}$$

The product $K \cdot \sigma_e$ is called the *detection threshold*. The value of K (real number) must be sufficiently great to avoid false detections and small enough so that clicks are detected. The value that multiplies K may be other that the standard deviation of the error, since the module of the average or the module of the median can be used [1].

2.2. Determination of the width of the click

Because of the linear prediction, there is a problem with the detection inside the click. The detector works quite well in the borders of the click, however in the middle of it, the detector consider that some samples are not affected by the click. Figure 1a shows in thin line a click and in thick line the detection vector (clicks vector). This vector takes the value 1 when a sample is consider like click, otherwise takes the value 0.

2.2.1. Zeros consecutive parameter

In order to solve the problem previously mentioned, the parameter consecutive zeros was defined. This parameter counts the number of samples not being considered as clicks that are between two samples detected like clicks. If the value of the parameter consecutive zeros is less than 20, then the samples between clicks will be considered also as clicks. In fact, all these consecutive samples will be considered as pertaining to the same click (Figure 1b).

2.2.2. Detection in inverse sense of the music

The detection of the end of clicks is generally not correct: clicks finish before the detector indicates their end (Figure 1b). In order to solve this problem a detection of clicks is made in inverse sense of the music. This detection is made in the same way as previously explained, with the difference that the inverted musical signal vector is processed (Figure 1c).

2.2.3. Double-threshold method

From ideas obtained from [1] we implemented a system of double threshold. The first threshold (called detection threshold) will be used for the determination of the location of clicks, and a second threshold (called small threshold) lower than the first one will be used to determine the width of each click. The variant which we developed respect to [1] was to perform an additional detection in inverse sense of music with the small threshold to determine the correct width of the click.

In summary, to detect clicks the following steps are followed: first, detection of clicks with the detection threshold without taking in consideration the width of the click. After that, detection with the small threshold is made and next detection with this same threshold but in the inverse sense of music is made. The detections with the small threshold do not add new clicks, only widen or shorten them.

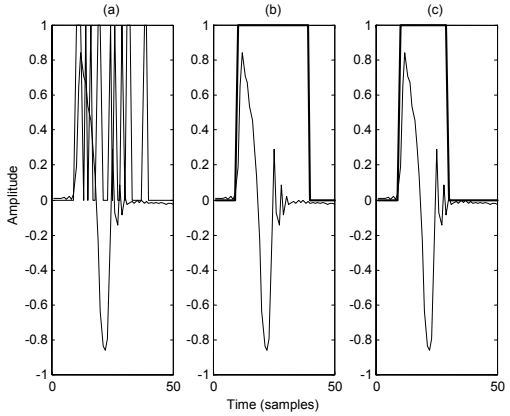

Figure 1: *Detected click: (a) without zeros consecutive parameter; (b) with zeros consecutives parameter; (c) with zeros consecutive parameter and detection in inverse sense of the music.*

2.3. Iteration

In signals affected by clicks, there are clicks of high amplitude and others with low amplitude. The detector sometimes does not detect the clicks of low amplitude because the error is smaller than the threshold. The implemented procedure to overcome this disadvantage is explained next. After the first detection, the sections affected by clicks are recovered, using the interpolator algorithm LSAR (Least Squares Auto Regressive [1], [2], [3]). The new signal will have a smaller standard deviation error vector. Therefore the detection threshold will be also smaller. In addition clicks that before were considered small now will be more representative. Then, the detection of clicks is again applied to the signal recovered in first instance, improving the detection of small clicks.

2.4. Use of windows to calculate the threshold

Because of the variable behavior of the musical signals, it becomes necessary to use a dynamic system of thresholds that changes its value throughout the recording. In order to implement the variation of the thresholds, the standard deviation of the error is calculated for each window (440 samples). A vector of the thresholds whose value changes from window to window is obtained. So the threshold adapts to the variations of the signal.

2.5. Detection of clicks in artificially degraded recordings

Having implemented a generic method for the detection of clicks, we used it to detect clicks of a recording and to interpolate these zones (for the interpolation LSAR was used).

The qualitative results of the recovered audio signals are in last instance a subjective question related to the perceptual results produced by the processing. In this part of the work we used non degraded signals to which artificial clicks were added to allow the use of quantitative measures that could validate the obtained results. For this purpose, the rate of wrongly detected clicks (clicks detected in samples not affected by click) and the rate of non detected clicks (clicks where the detector determined erroneously that the samples did not belong to click) were measured. In order to make these calculations, recordings with clicks artificially inserted, obtained from [4], were used, so that the location of clicks on the recordings were known.

The equations for obtaining the mentioned parameters are the following ones:

$$\%\ wrongly\ detected = \left(1 - \frac{real\_clicks - wrongly\_detected\_clicks}{real\_clicks}\right) \cdot 100 \quad (4)$$

$$\%\ non\ detected = \left(1 - \frac{real\_clicks - non\_detected\_clicks}{real\_clicks}\right) \cdot 100 \quad (5)$$

These rates were used to determine the values of the parameter used for the detection of clicks in artificially degraded recordings.

2.6. Detection of clicks in naturally degraded recordings

Once all the parameters for the detection of clicks were determined, we made tests in naturally degraded recordings. The results in first instance were not satisfactory since the detector fails in samples clearly affected by clicks. A possible cause of this behavior is that the waveform of clicks inserted artificially differs from clicks that appear in real recordings. Another cause may be that as the signal also is affected by hiss, the model of linear prediction is not as effective as in the signals that only have clicks. In order to solve this problem some parameters of the detector were modified and a method of confirmation of clicks was introduced. This will be explained next.

3. METHOD FOR CLICK CONFIRMATION USING SINUSOIDAL MODELING

The methods implemented for the detection of clicks are based on the comparison of the real signal with an estimated one. Thus we are looking for a threshold that let us detect all the clicks of the signal and avoid false clicks detection. Looking for a method to discard false click detection, we propose the method of click confirmation. We can use a smaller detection threshold than previously in order to detect all the clicks of the signal. Since the

threshold is small, there will be a lot of false detection clicks, but these will be discarded with the method presented in the following Sections.

3.1. Introduction

The objective of sinusoidal modeling (see [5], [6] and [7]) is to represent a signal based on variable sinusoids of frequency and amplitude in time. In order to do that, we perform the spectral analysis in windows of time. So it is possible to calculate the functions that describe the variations of frequency, amplitude and phase of each component of the signal.

These methods enable us to model a signal $x[n]$ as a sum of evolutional sinusoids [5]:

$$x[n] \approx \hat{x}[n] = \sum_{q=1}^{Q[n]} A_q[n] \cdot \cos \theta_q[n] \quad , \tag{6}$$

where $Q[n]$ is the component number in time *n*. The components will have variable amplitude $A_q[n]$ and phase $\theta_q[n]$ in time.

Sinusoidal modeling can be considered as a generalization of the Fourier series that allows us to describe the signal as the sum of sinusoids that changes in dependence with its behavior.

The sum of sinusoids that have slow variation in time is not effective for the modeling of impulsive events or noise. A term grouping these processes must be added to the model. That term is called *residual*, $r[n]$. So

$$x[n] = \hat{x}[n] + r[n] \tag{7}$$

Sinusoidal modeling can be interpreted as the evolution of the STFT (Shot Time Fourier Transform). In the following Sections we will present a brief description of the STFT and its relationship with sinusoidal modeling.

3.2. Short Time Fourier Transform

The goal of the STFT is to derive a time-localized representation of the frequency-domain behavior of a signal. The STFT is carried out by applying a sliding time window to the signal; this process isolates time-localized regions of the signal. Each of them is analyzed using a discrete Fourier transform (DFT).
Typically, the STFT is given by

$$\tilde{X}[k,n] = \sum_{m=n}^{n+N-1} w[n-m] \cdot x[m] \cdot e^{-j\omega_k m} \quad , \tag{8}$$

The analysis presented in this article will be based in the following definition of the STFT. The reference of time is changed to facilitate its interpretation as a filter bank and its relationship with sinusoidal modeling:

$$X[k,n] = \sum_{m=0}^{N-1} w[m] \cdot x[n+m] \cdot e^{-j\omega_k m} \, , \tag{9}$$

where $\omega_k = 2\pi k / K$ and $w[m]$ is a window in the time domain with zero value outside of the interval $[0,\ N-1]$.
Equation (9) can be expressed as:

$$X[k,i] = \sum_{m=0}^{N-1} w[m] \cdot x[m+iL] \cdot e^{-j\omega_k m} \quad , \tag{10}$$

where *L* is the time distance between successive applications of the window to the data.

In the first case, the reconstruction of the signal is obtained by calculating the inverse DFT for each window of the spectrum.
From equation (8), we get:

$$\tilde{X}[k,n] = \sum \tilde{w}[n-m] \cdot x_k[m] \tag{11}$$

where

$$x_k[m] = x[m] \cdot e^{-j\omega_k m} \, .$$

$\tilde{X}[k,n]$ can be interpreted as the output of a low-pass filter. In time domain it can be interpreted as the envelope of a sinusoid of frequency ω_k. The idea is straightforward: the signal can be reconstructed by modulating each of these envelopes to the appropriate frequency and summing the resulting signals. This construction is given by

$$\hat{x}[n] = \sum_k \tilde{X}[k,n] \cdot e^{j\omega_k n} \tag{12}$$

$$\hat{x}[n] = \sum_k X[k,n] \quad . \tag{13}$$

Equation (12) allows us to perform the STFT as a heterodyne filter bank: each envelope is modulated at the corresponding frequency.

Equation (13) corresponds to a modulated filter bank, where each sub-band of the signal is the complete component of the signal in that band.

The most immediate limitation of the short-time Fourier transform results from its fixed structure. A sinusoid with time-varying frequency will move across bands; this evolution leads to delocalization of the representation and a no-compact model, as can be seen in Figure 2 [5].

Sinusoidal modeling provides a more compact representation of the signal. The parameters of this model ($Q[n]$, $A_q[n]$, $\theta_q[n]$) can be estimated using the STFT and any procedure of search of the spectrum peaks.

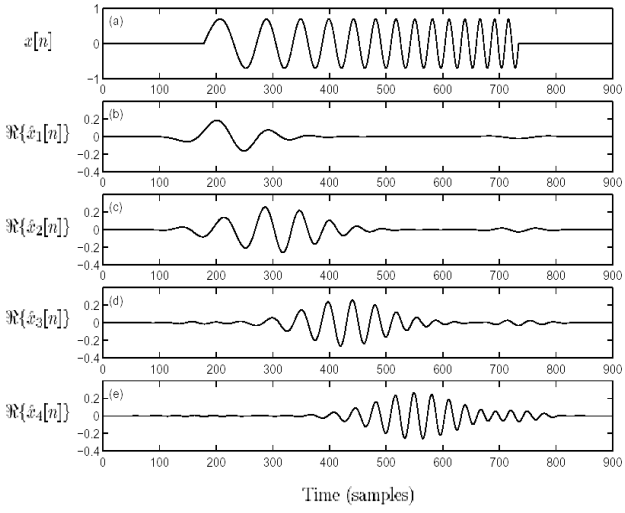

Figure 2: *Reconstructed sub-band signals in a non-subsampled STFT filter bank model of a chirp signal.*

3.3. Implemented method for click confirmation

Figure 3 shows samples of a song that includes a click. Also there is the reconstruction of the signal using five components with sinusoidal modeling.

It can be seen that the modeled signal does not follow the click closely because the reconstruction uses only few components. We propose to use this property to discard clicks that were detected incorrectly.

For each detected click we will reconstruct the signal using 50 samples previous to the start of the click and until 50 samples after its end. That signal will be compared with the real signal. If there is no sample that is bigger than the threshold (the definition will be explained later) we will consider that the samples involved are not a real click.

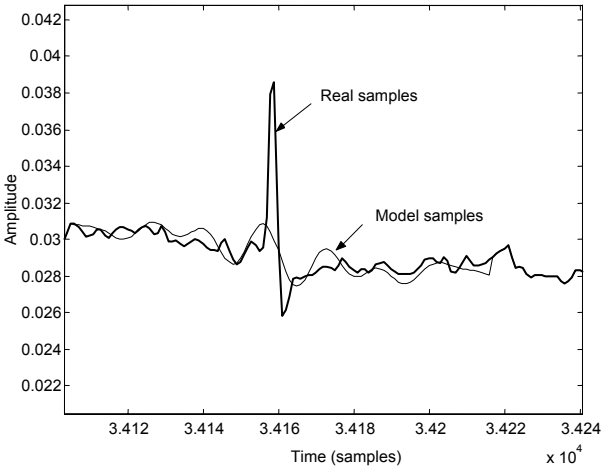

Figure 3: *Signal modeled in the presence of a click.*

3.3.1. Algorithms

Figure 4 shows the flow chart of the algorithms that was implemented.

The program works as follows: first, it takes the vector that shows the samples that are affected with clicks. This is done using the detection algorithms previously explained with a small threshold. Because of that condition, we expect that some of the detected clicks are not real clicks.

Secondly, we define the threshold and model the signal. The idea is to include close to 100 samples without clicks so the modeled signal become similar to the music but not to the click (if it exists). For that purpose we model using 50 samples previous to the start of the click until 50 samples after the end of the click and the samples affected by clicks are substituted by a straight line. Then, the signal is reconstructed using five components with sinusoidal modeling.

To confirm the click we calculate the difference between the reconstructed signal and the original one in the samples that are not affected by clicks, that is to say in the 50 samples before the click and the 50 samples after the click. Then we calculate the comparison threshold using the following difference:

$$Threshold = k \cdot \sqrt{(Var(difference))} \qquad (14)$$

where k is 3 (determined experimentally) and *Var(difference)* is the variance of the difference of the mentioned signals.

For each click we calculate the threshold and then we compare the original signal with it. If any of the samples is bigger than the threshold, then we consider that the detected click is a real click. Otherwise the click is not confirmed and the vector that indicates the affected samples is modified *(Clicks Vector)*.

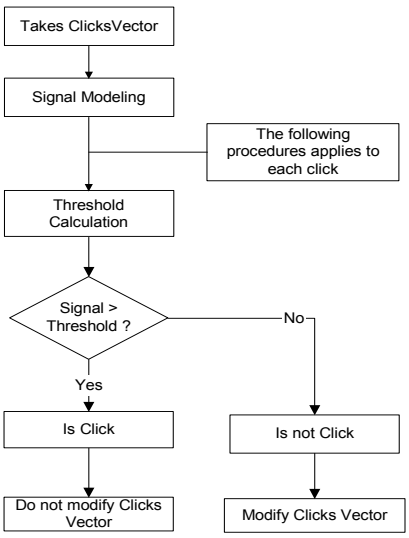

Figure 4: *Flow chart of the confirmation click algorithm.*

3.3.2. Results

To evaluate the method we use the parameters previously explained in this article. We call "*wrongly detected clicks,*" to the detected clicks in samples that are not affected. With the term "*non detected clicks*" we mean the real clicks that are not detected.

The Table 1 shows the results of the click detection without using sinusoidal modeling (just the detection explained in the previous Sections) and using sinusoidal modeling.

The parameters used for the comparison are:

- Number of iterations: 6, the first 3 with a threshold of 8 and the other 3 with a threshold of 7.
- The value of k is 3.

	WITHOUT Sinusoidal Modeling		**WITH Sinusoidal Modeling**	
	% non detected	**% wrongly detected**	**% non detected**	**% wrongly detected**
Signal1	0	42,98	0	23,14
Signal2	1,65	9,92	1,65	2,48
Signal3	0	9,92	0	2,48
Signal4	1,65	8,26	2,48	2,48
Signal5	0,44	2,31	1,87	0,99
Signal6	0,88	4,5	0,88	2,31

Table 1

Given that the algorithm of confirmation sometimes discard clicks that really are, it can be seen that there is a small increase in the value of "*non detected clicks.*" However, the percentage of *"wrongly detected clicks"* has been substantially decreased. The algorithm results efficient in the "no-confirmation" of detected clicks in samples that are not affected by clicks, so it achieves its primary objective.

4. ERROR VECTOR

To improve the detection in naturally degraded songs, we introduced an important modification: in the analysis shown in [1], there is a variant for the calculation of the threshold error. The objective is to improve the threshold calculation so that it becomes independent of the power of the signal and immune to the degradations of the song. In that way, we follow these steps to determine the prediction threshold error:

- Obtain the vector error of each window
- Square the elements of the vector
- Delete a determined percentage of the highest values of the vector
- Calculate the root of the elements of the vector
- The threshold error is obtained calculating the mean of the absolute value of the vector

The number of degraded samples can vary from a window to other. This can not be known beforehand, so we had to determine an arbitrary number of samples to be eliminated. In [1] the author proposes a value close to 10% of the highest samples of the error vector. In this article we propose the elimination of 5% of such samples. The statistic that we will use is the standard deviation because we got better results than with the mean of the error vector.

5. IMPLEMENTED CONFIGURATION

In songs with real clicks the behavior of the detection can only be determined by qualitative results. By qualitative results we mean comparing the signal with the detected click vector or listening to the restored song.

After several tests of the different configurations, we determined the following parameters for the detection of clicks:

- Number of coefficients of AR model: 20
- Length of the window: 440 samples
- Detection threshold: 7 · Standard deviation of the error vector without 5% of the highest samples.
- Small threshold: 5.5 · Standard deviation of the error vector without 5% of the highest samples.
- Number of iterations: 6

The detection threshold was not easy to determine. If the threshold is high (7), the clicks of high amplitude are detected but there are still many clicks not detected. If the threshold is small, there are a lot of wrong detections. Because of that we use sinusoidal modeling to discard clicks. As we explained previously, the algorithm that implements sinusoidal modeling takes the vector click with the detected clicks and discards the ones that are not confirmed.

The detection threshold will have two possible values, in accordance with the number of iterations that the process has. In the first four iterations the threshold will be 7, there the clicks of highest amplitude will be detected. In the following two iterations the threshold will be 6 and we use sinusoidal modeling to discard those clicks wrongly detected. The small threshold has a value of 5.5 for the first four iterations and 5 for the last two. In the following flow chart, the process of detection and interpolation of the clicks it can be seen.

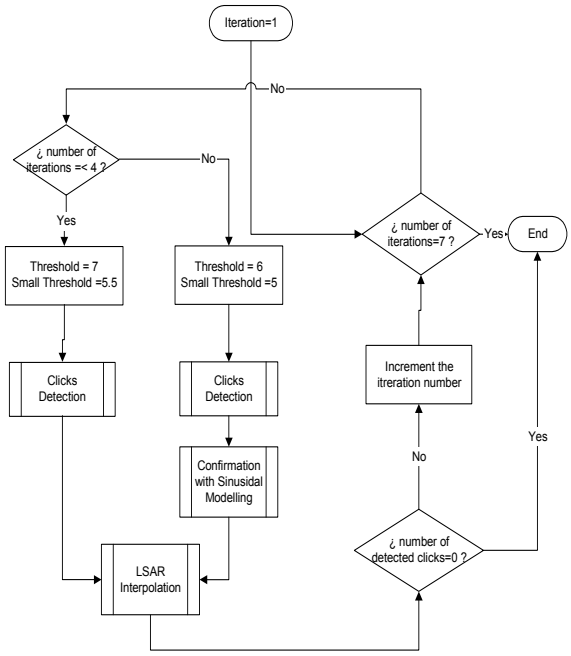

Figure 5: *Flow chart of the detector's process.*

Next, we can see a click detected with the mentioned method.

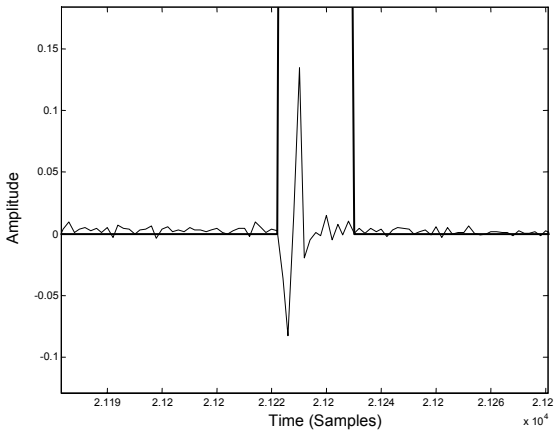

Figure 6: *Real click detected.*

To ensure the detection of the whole click, we decided to increase in two samples the start and the end of the click. Other option is to decrease the small threshold, but in this case there are risks to overestimate the width of the click. So with this option we get

better results in the detection in the edge of the clicks and do not add many samples to each click.

Using this configuration there is an average of 25.1 clicks detected per second in songs that do not have degradations. In spite of the numerical value, the important point is to determine if a song without clicks is degraded when it is submitted to the procedure of detection-interpolation. Thus, we applied the mentioned method to many songs without degradations. It can be seen that the difference between the original song and the restored song is practically null, so we confirm that using these parameters no degradations are introduced to the songs.

6. CONCLUSIONS

Respect to the determination of the width of the clicks, the implementation of the detection in inverse sense of the music allows us to adjust properly the end of the click. The double-threshold method allows us to reduce the number of samples to increase at the start and the end of the click. This improves noticeably the quality of the restored song.

The use of windows enables us to adapt the value of the detection threshold to the variations of the signal, following improvements in click detection. This is because the threshold adjusts to the statistic of the signal.

The click confirmation method based on sinusoidal modeling introduced in this article allows us to reduce the detection thresholds to detect the small amplitude clicks, while avoiding an increase in the wrongly detected click value. This method improves substantially the restoration of the songs that have clicks with small amplitude.

Finally, the complete methods implemented for the detection of clicks achieve excellent results in songs both natural and artificial degraded.

7. REFERENCES

[1] P. A. A. Esquef, *Restauração de Sinais de Áudio Degradados por Ruído Impulsivo,* Master's Thesis, Federal University of Rio de Janeiro, 1999. http://www.acoustics.hut.fi/~esquef/mypapers/esquef_tese.pdf

[2] S. J. Godsill, M. J. Rayner, *Digital Audio Restoration*. Great Britain: Springer, 1988.

[3] R. Gadea, R. Laureiro, F. Vilaró, *Recuperación de grabaciones analógicas antiguas*. Facultad de ingeniería Universidad de la República. Montevideo, 1996.

[4] P. A. A. Esquef, L. W. P. Vizcaino, P. S. R. Diniz, P. F. Freeland, "A Double-Threshold-Based Approach to Impulsive Noise Detection in Audio Signals." http://www.lps.ufrj.br/audio/

[5] M. M. Goodwin, *Adaptive Signal Models: Theory, Algorithms, and Audio Applications*, PhD Thesis, University of California, Berkeley, 1997. http://ptolemy.eecs.berkeley.edu/papers/97/mgoodwinThesis/mgoodwinThesis.pdf

[6] T. Tolonen, *Methods for Separation of Harmonic Sound Sources using Sinusoidal Modeling,* Helsinky University of Technology, Laboratory of Acoustics and Audio Signal Processing. http://lib.hut.fi/Diss/2000/isbn9512251965/article7.pdf

[7] T. Virtanen, A. Klapuri, *Separation of Harmonic Sound Sources using Sinusoidal Modeling.* Tampere University of Technology, Signal Processing Laboratory. http://www.cs.tut.fi/sgn/arg/music/tuomasv/sssep.pdf