MULTIRESOLUTION STFT PHASE ESTIMATION WITH FRAME-WISE POSTERIOR WINDOW-LENGTH DECISION

Volker Gnann and Martin Spiertz

Institut für Nachrichtentechnik, RWTH Aachen University
D-52056 Aachen, Germany
{gnann, spiertz}@ient.rwth-aachen.de

ABSTRACT

This paper presents an extension to the dual-window-length Real-Time Iterative Spectrogram Inversion phase estimation algorithm (RTISI). Instead of a transient detection in advance, the phase estimator itself determines the correct window length when the phase information for all window lengths have already been estimated. This way, we get significant improvements compared with the previous method. Additionally, we extend this estimator to configurations with three or more window lengths.

1. INTRODUCTION

The reconstruction of missing phase information is an important step to get an audio signal from a magnitude short-time Fourier transform (STFT) spectrogram and enables audio effects to work just on the spectrogram magnitude. However, the quality of this approach suffers from the time/frequency resolution tradeoff, similarly to other spectrogram-based audio manipulation methods. In audio coding, window switching is a well-known method to improve this tradeoff. In a previous paper, [1], we already presented an implementation of a phase estimator using window switching between two analysis windows.

Lukin and Todd [2] proposed a different method to perform arbitrary spectrogram-based audio effects with different spectrogram lengths: They process audio frame-wise in parallel with different window length and decide afterwards, frame by frame, which process has performed best. However, this approach needs the knowledge of the phase information. This paper proposes a method to estimate it.

Dual-resolution phase estimation with window switching has some problems which do not occur with the Lukin/ Todd approach:

- Errors in transient detection lead to a sub-optimal time/ frequency resolution for the recent audio frame. The decision which resolution is correct is not trivial.
- Algorithms which modify the resulting spectrum do not get the whole spectrogram for processing. Instead, they get only the frames the transient detector has allocated to them. This can be a disadvantage, e. g. when they need complete statistics for correct processing.
- It is not guaranteed that an optimal decision before the spectrum modification remains optimal during the modification.

This motivates us to create a multi-window-length STFT phase estimator not based on window switching, but on parallel processing, similar to the Lukin/Todd processing scheme. One important difference: In the original paper [2], a coefficient mixing is proposed to determine the final signal. Since such a mixing can lead to phase cancellations, we perform a strong decision for every frame which time-frequency resolution is most appropriate.

As phase estimation algorithm, we use the Real-Time Iterative Spectrogram Inversion (RTISI) [3], with the improvements from [4]. To our knowledge, there are no other phase estimators available that could handle multi-window-length spectrograms. RTISI itself is a localized variation of the classic Griffin/Lim algorithm [5]. An alternative would be the phase estimator of Le Roux et al [6]. Unfortunately, unlike RTISI, this phase estimator operates only in frequency domain, while we need the time-domain representation of intermediate signals to synchronize the buffers (see below for details). Unlike in [1], this synchronization must happen after every committed frame to avoid drifting phase estimations.

This paper is organized as follows. Section 2 gives an overview over the whole processing scheme. Section 3 introduces the phase estimation and synchronization scheme. Section 4 shows how the best estimation is selected. Section 5 generalizes the scheme to more window lengths than two. Section 6 contains the experiments and results. Here, the new method is also compared with [1]. The paper finishes with the conclusions.

2. PROCESSING OVERVIEW

Figure 1 shows how the overall processing is performed. From the original waveform $s[n]$, we generate two magnitude spectrograms with different window lengths L_1 and L_2 and the same overlap (typically 75%) using the STFT. We call the resulting hop lengths S_1 and S_2. The ratio L_1/L_2 must be a power of two. As window function, we use the scaled Hamming window from [5].

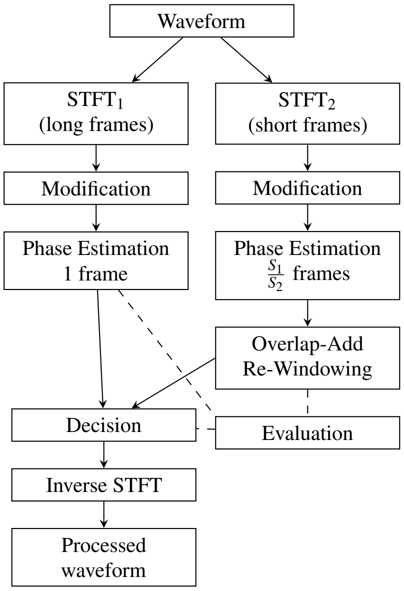

Figure 1: General overview. The evaluation step compares all phase estimations frame-wise with all spectrograms and decides with a minimax principle.

Those magnitude spectrograms can be modified in an arbitrary way. After that, two RTISI algorithms [3] re-estimate the phase spectra. As explained in Section 3, one RTISI buffer is used for each window length, respectively. For each frame of length L_1 except the first one, $\frac{S_1}{S_2}$ frames of length L_2 are estimated, so that the number of estimated samples is the same.

The result of the short-window-length phase estimation is re-windowed such that it is implicitly windowed with the long window. When the estimated samples are available for both window lengths, we rate them using the procedure explained in Section 4 to get the best estimation. The best estimated frame is committed, so that a final overlap-add procedure can collect the committed frames to construct the final modified audio signal.

To ensure that the initialization data of the RTISI buffers are the same, the data of the chosen RTISI buffer must be copied to the other one every time when a frame is committed. Finally, the buffers are set to the next frame. Thus, the next long-window frame, and the next $\frac{S_1}{S_2}$ short-window-frames are processed, and so on.

3. PHASE ESTIMATION

Central data structure for our phase estimator is a combination of two two-dimensional buffers ${}^{\text{long}}\mathbf{B}$ and ${}^{\text{short}}\mathbf{B}$, one for each window length. The buffers are basically illustrated in Figure 3 with two modifications: The number of rows in the long-window-length RTISI buffer is actually 7, with the commit frame in the center. Additionally, each buffer row stores the target magnitude spectrum.

Mathematically, we can interpret these buffers as two-dimensional arrays (not matrices) $(\mathbf{B}_{MN})$, which are illustrated in Figure 2. On each buffer $\mathbf{B}$, we define the following operations. To help understanding, operations returning a complete row are overlined, whereas operations returning only a vector of the window size are marked with a degree symbol. The symbol i denotes a row index, a denotes an external vector.

- Addition, Subtraction, Multiplication, Division are defined element-wise.
- $\mathbf{B}.\text{CR}\overset{\circ}{\text{O}}\text{P}(i, a)$: Shortens the row vector $(a_1, \cdots, a_N)$ to $(a_l, \cdots, a_r)$, with $l := iS$ and $r := l + L$.
- $\mathbf{B}.\overline{\text{GET}}(i)$: Returns the row vector $(B_{i1}, \cdots, B_{iN})$.
- $\mathbf{B}.\text{G}\overset{\circ}{\text{E}}\text{T}(i) := \mathbf{B}.\text{CR}\overset{\circ}{\text{O}}\text{P}(i, \mathbf{B}.\overline{\text{GET}}(i))$
- $\mathbf{B}.\text{S}\overset{\circ}{\text{E}}\text{T}(i, a)$: Sets the row vector $(B_{il}, \cdots, B_{ir})$ to $\mathbf{B}.\text{CR}\overset{\circ}{\text{O}}\text{P}(i,a)$, with l and r defined as for the $\text{CR}\overset{\circ}{\text{O}}\text{P}$ operation.
- $\mathbf{B}.\overline{\text{SET}}(i, a)$: Sets the row vector $(B_{i1}, \cdots, B_{iN})$ to a.
- $\mathbf{B}.\overline{\text{SUM}}$: Calculates the sum of the matrix rows as a row vector $\left(\sum_{i=1}^{M} B_{i1}, \cdots, \sum_{i=1}^{M} B_{iN}\right)$.
- $\mathbf{B}.\text{S}\overset{\circ}{\text{U}}\text{M}(i) := \mathbf{B}.\text{CR}\overset{\circ}{\text{O}}\text{P}(i, \mathbf{B}.\overline{\text{SUM}}(i))$

For convenience, we also associate following functions with the buffer which return additional data:

- $\mathbf{B}$.MAG(i): Returns the target magnitude spectrum of row i.
- $\mathbf{B}$.W: Returns the discrete window function the magnitude spectra are inherently calculated with.

RTISI, introduced in [3], is an online-capable algorithm, which works on a frame-after-frame basis. The phase estimation consists of two steps: an initialization and several iterations of an (improving) update rule.

The initialization depends on the window length: The short-window buffer is always initialized with zeros. The long-window buffer is initialized with a propagated phase of the previous frames, as proposed in [4], to exploit the phase continuity of steady-state signals.

The update rule is the same on both buffers. It is processed on a buffer row and works as follows:

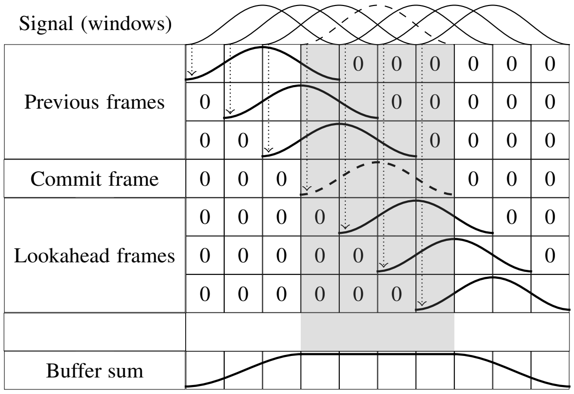

Figure 2: Single phase estimation buffer. Every sketched cell contains S elements, whereas S denotes the hop length between adjacent frames.

- Calculate the sum frequency spectrum for the current row *i* by taking the according part of the buffer sum, re-windowing this part and applying an Discrete Fourier Transform.
- Project all coefficients of this spectrum onto the unit circle. The result is equivalent to the phase. Multiply this result with the target magnitude (M-Constrained transform).
- Transform the result of this combination into the time domain, window it, and store it back into the current row.

In short:

$$\mathbf{B}.\text{S}\mathring{\text{E}}\text{T}(i, \mathbf{B}.W \cdot \left(\text{IDFT}\left(\mathbf{B}.\text{MAG}(i) \cdot \frac{\text{DFT}(\mathbf{B}.\text{S}\mathring{\text{U}}\text{M}(i))}{|\text{DFT}(\mathbf{B}.\text{S}\mathring{\text{U}}\text{M}(i))|}\right)\right)), \tag{1}$$

where DFT denotes the Discrete Fourier Transform, and IDFT its inverse.

The order of the buffer row updates is determined by the energy of the rows — first the loudest row is updated, then the second loudest, and so on, until all rows between the last and the commit row have been processed [4]. After a certain number of iterations, the frame is committed, and the buffer is synchronized to the next audio frame.

After each commit of the long-window buffer and $\frac{S_1}{S_2}$ commits of the short-window buffer, we must decide which configuration is better (see Section 4). After that, the result is copied from the "winning" buffer ${}^{\text{src}}\mathbf{B}$ to the "losing" buffer ${}^{\text{tgt}}\mathbf{B}$ as follows:

- Calculate the buffer sum.
- Divide the buffer sum by the sum of the squared window functions for all rows so that the buffer sum is implicitly windowed with a rectangle window.

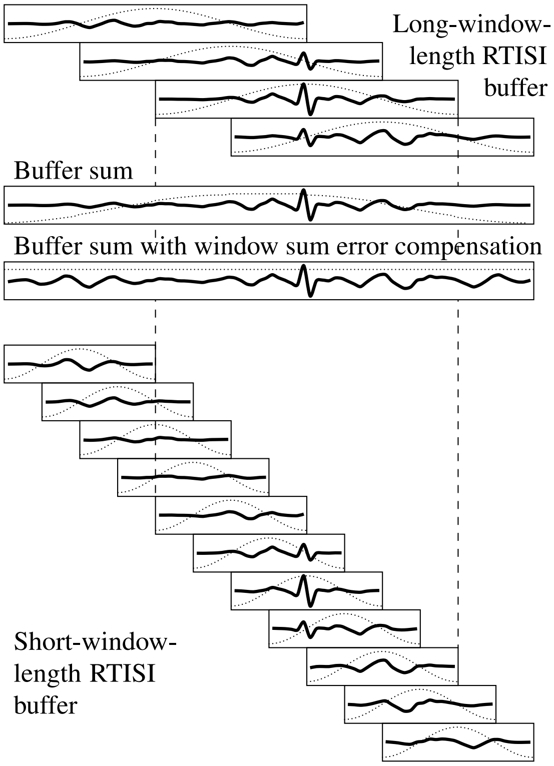
Long-window-length RTISI buffer
Buffer sum
Buffer sum with window sum error compensation
Short-window-length RTISI buffer

Figure 3: Two RTISI buffers with different window lengths without target magnitude spectra and without look-ahead frames. To transport audio data between the buffers, the algorithm calculates the buffer sum, compensates the window sum error, and windows the result for each target buffer row. The dotted lines denote the window function the audio data in the buffer are implicitly multiplied with.

- For each target buffer row, window the according part of the buffer sum. Copy the windowed buffer sum part into the target row.

To express this mathematically, we consider the RTISI buffers ${}^{\text{long}}\mathbf{W}$ and ${}^{\text{short}}\mathbf{W}$ which are filled with the squared windows such that ${}^{\text{long}}\mathbf{W}.\text{G}\mathring{\text{E}}\text{T(i)} = {}^{\text{long}}\mathbf{B}.\text{W}^2$ for all i, and ${}^{\text{short}}\mathbf{W}.\text{G}\mathring{\text{E}}\text{T(i)} = {}^{\text{short}}\mathbf{B}.\text{W}^2$ for all i. We denote the "window buffers" with the source and target window length as ${}^{\text{src}}\mathbf{W}$ and ${}^{\text{tgt}}\mathbf{W}$, respectively. Then the synchronization becomes

$${}^{\text{tgt}}\mathbf{B}.\text{S}\mathring{\text{E}}\text{T}(i, \left({}^{\text{tgt}}\mathbf{W}.\text{G}\mathring{\text{E}}\text{T}(i) \cdot {}^{\text{tgt}}\mathbf{B}.\text{CR}\mathring{\text{O}}\text{P}(i, \frac{{}^{\text{src}}\mathbf{B}.\overline{\text{SUM}}}{{}^{\text{src}}\mathbf{W}.\overline{\text{SUM}}})\right)) \tag{2}$$

for all i.

4. DETERMINING THE OPTIMAL WINDOW SIZE

To find the optimal window size, we have to compare the phase estimation for each window length with the magnitude spectrograms, also for each window length. Let $\tilde{M}$ be the position index of the commit frame. As comparison criterion, we use the signal-to-error ratio (in dB) in the magnitude spectrogram domain:

$$\mathrm{SER} = 10\log_{10} \frac{\sum\limits_{m=\tilde{M}-\xi}^{\tilde{M}+\xi} \left(\sum\limits_{k=0}^{L-1} |X[mS,k]|^2 \right)}{\sum\limits_{m=\tilde{M}-\xi}^{\tilde{M}+\xi} \left(\sum\limits_{k=0}^{L-1} (|X[mS,k]| - |X'[mS,k]|)^2 \right)} \tag{3}$$

From the RTISI buffer with the window length L_u to evaluate, we calculate the actual magnitude spectrogram X' from the buffer sum on the commit frame $\pm\xi$ additional rows; see Section 6.2 why a choice $n = 2$ can be considered optimal. For this spectrogram calculation, we use the window length L_v of the target spectrogram to compare. We denote this SER as $\mathrm{SER}_{u,v}$ which means: "Calculate the signal-to-error ratio of the spectrogram of the phase estimation with the length L_u, but use length L_v for the spectrogram calculation and thus set $S = L_v/4$ for 75% overlap. Compare the resulting spectrogram with the target magnitude stored in the RTISI buffer of the length L_v."

The four signal-to-error ratios $\mathrm{SER}_{u,v}$ for one long and one short window can be summarized in a matrix:

$$\begin{pmatrix} \mathrm{SER}_{1,1} & \mathrm{SER}_{1,2} \\ \mathrm{SER}_{2,1} & \mathrm{SER}_{2,2} \end{pmatrix} \tag{4}$$

To get a final decision from these four SER values, we derive a minimax approach comparable to the Haussdorff distance [7] as follows: Low SER values for a *short* reference window length L_v correspond to artifacts manifesting in the time domain, usually time-domain smearing. These artifacts are audible especially in transient regions. On the other hand, low SER values for a *long* reference window length correspond to errors in the frequency domain, e. g. modulation effects. They are usually audible at steady-state signals, especially at low frequencies. Since our goal is the reduction of audible artifacts, we dump the estimation with the worst SER and favor the time-domain signal estimation which does *not* produce this worst SER.

As the SER is also the optimization criterion for RTISI, the SERs on the main diagonal of the matrix are optimized, so they are always better than other SER values. For that reason, the worst SER is either $\mathrm{SER}_{1,2}$ or $\mathrm{SER}_{2,\,1}$. Thus, we can simply decide: If $\mathrm{SER}_{1,2} > \mathrm{SER}_{2,1}$, we choose L_1, otherwise L_2.

5. GENERALIZATION TO MORE WINDOW LENGTHS

The extension of this approach to multiple window lengths is now straightforward. For every used window length except the longest one, we employ one separate RTISI buffer and treat it as a short-window case. The longest-window-length buffer is treated as the long-window-length buffer in the previous sections.

To determine the correct window length after estimation, we also generalize the avoid-the-worst-SER approach. For every window length, we compare the phase estimation result with each reference spectrogram and find the worst SER. We choose the window length where the worst SER is best:

$$U = \operatorname*{argmax}_u \left(\min_v (\mathrm{SER}_{L_u,L_v}) \right) \tag{5}$$

6. EXPERIMENTS AND RESULTS

In order to evaluate these improvements, we use a test set based on the Sound Quality Assessment Material (SQAM) from the EBU [8]. Our test set consists of 70 files containing speech, singing vocals, and instruments. The sampling frequency is 48 kHz. For spectrogram generation and phase estimation, we use the scaled Hamming window from [5] with $L = 4S$, yielding an overlap of 75%. For phase estimation, we use RTISI with three look-ahead frames (see also the remarks to Fig. 3). As evaluation measure, we use the mean signal-to-error ratio (in dB) of the magnitude spectrograms of the phase-reestimated signal versus the original, respectively, as defined in Equation (3).

Like in Section 4, this SER measure operates on STFT magnitudes and thus depends on its own window length. Also, this SER window length determines the operating point of the *measured* time-frequency resolution tradeoff. To analyze the phase estimation performance over the whole range of time/frequency resolutions, the SER values are plotted against this window length. A high-quality phase reestimation should achieve high SER values for all window lengths to show that it avoids artifacts in both time and frequency domain.

6.1. General Results

Figure 4 shows the SER results of a single-resolution RTISI phase reestimation with the window lengths 512, 1024, and 2048 samples, respectively. Note the peaks at the window lengths; here the evaluation criterion is identical to the optimization criterion. Additionally, this figure shows the results of dual-length phase estimation with transient detection and with post-estimation decision. As a transient detection measure, we use the maximal energy compaction principle described in [2]. We see that our post-estimation decision

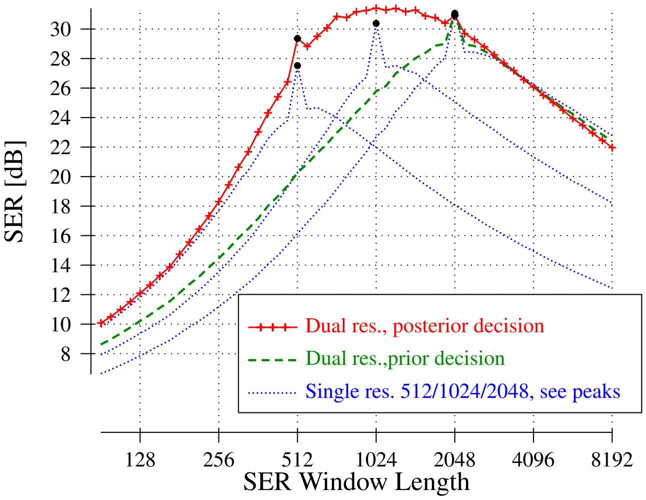

Figure 4: Average SER *for different phase estimation methods on the standard EBU set against the measure window length. The crosses on the solid (red) line denote the measure points. The peaks marked with black-filled balls result from the fact that, on this window length, the evaluation criterion matches the optimization criterion.*

outperforms the transient detection approach except on very large windows. It delivers also better results than each of the single-resolution estimations, even when evaluation and estimation window length match. The outlier peaks here are smaller than in the single-resolution case; an obvious reason is that the optimization result is a compromise between two window lengths.

6.2. Number of frames to analyze in a buffer

Figure 5 shows which number of frames ξ to analyze in the buffer should be chosen. As a general rule we can follow, the higher the number of frames, the better the estimation. An interesting exception arises when we choose the complete buffer (seven frames, e. g. commit frame ±3 frames). Here the results are *worse* compared with the five-frames evaluation.

To explain this decline, we recall that all magnitude spectra are implicitly windowed with $w[n]$. In the state of convergence, the M-constrained transform does not change anything, so due to the following windowing step the data in the buffer are implicitly windowed with $w^2[n]$. This window is compensated in the rectification step.

If the state of convergence is not achieved, the M-constrained transform *does* perform changes and tends to neutralize the previous windowing step. As a result, after the following windowing step, the data in the buffer are implicitly windowed with something between $w[n]$ and $w^2[n]$. The rectification over-compensates this windowing step, thus distorting the buffer sum especially at the edges. When this edge is also considered for determining the correct window size, this error leads in some cases to wrong decisions and finally to lower SER values than possible.

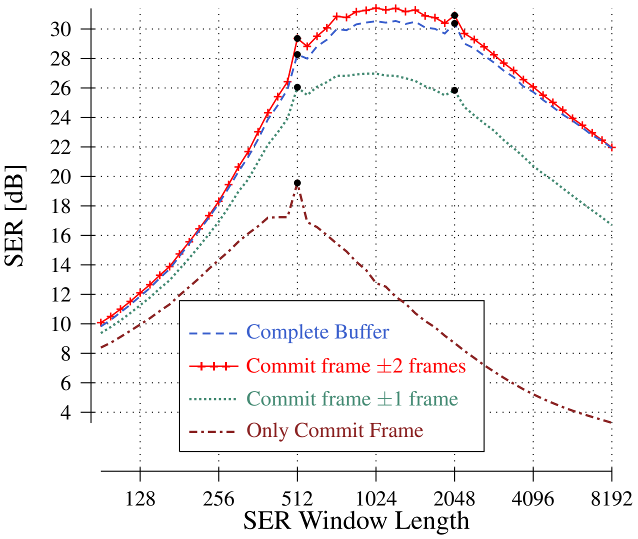

Figure 5: Average SER *with respect to the number of long-size buffer frames to take into account for minimax evaluation.*

6.3. Distance between Window Lengths

Figure 6 shows the influence of the difference between the window lengths. One window length – 2048 samples – is fixed, the other one varies between 256 and 2048 samples. As we can expect, the temporal resolution becomes better with increasing difference. On the other hand, in this case the SER values above a window length of 2048 become worse. We can conclude that the time/frequency resolution tradeoff in a certain way also holds for multi-resolution phase estimation.

6.4. Extension to multiple window lengths

The results of using more than two window lengths are presented in Figure 7 and 8. Generally, it makes sense to compare the 512/2048 dual-resolution curve with the 512/1024/2048 triple-resolution curve, and the 256/4096 dual-resolution curve with the 5-resolution curve. In both cases, the SER for the reference window lenghts between L_1 and L_2 are better when more than two resolutions are considered. On the other hand, the SER at very long windows becomes smaller. A comparison between the 5-resolution and dual-resolution 256/2048 samples (green, solid curve) shows that the SER gain for the 5-resolution for long window lengths is also limited; for short window lengths the dual-resolution is even better.

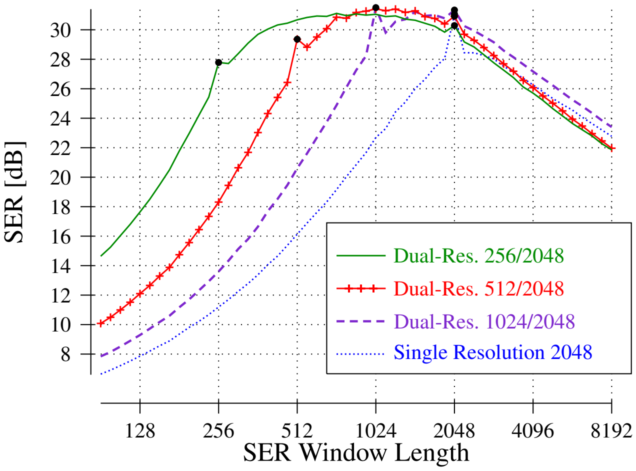

Figure 6: SER *for dual window length with one window length (2048 samples) kept fixed.*

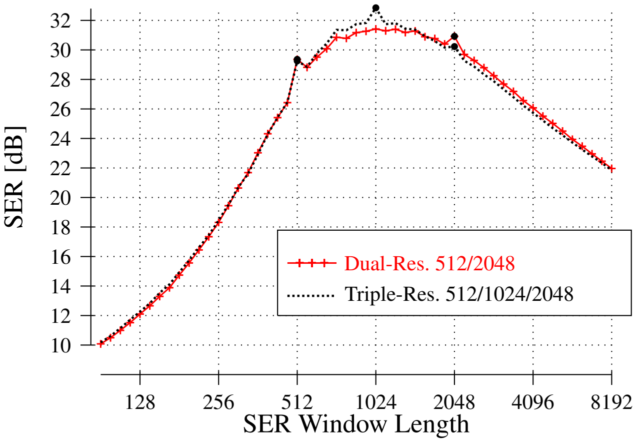

Figure 7: Average SER *for triple resolution.*

7. CONCLUSIONS

We have generalized the method of [1] to a signal processing scheme that allows switching between different window lengths without relying on a (better or worse) transient detector. Additionally this scheme allows the usage of more than two window lengths. Applications for this technique are all algorithms which operate on magnitude-spectrums only, like time/pitch modification or source separation. While the new approach performs significantly better than the previous method [1], the results for three or more window lengths indicate that in this case the window length decision method gives room for additional research.

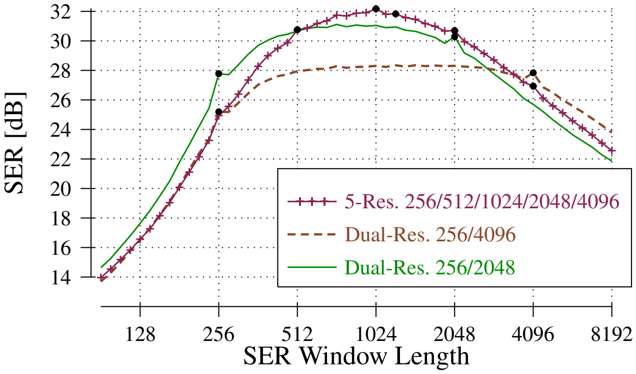

Figure 8: Average SER *for 5 different window lengths compared with 256/2048 and 256/4096 dual-window length.*

8. REFERENCES

[1] V. Gnann and M. Spiertz, “Inversion of Magnitude Spectrograms with Adaptive Window Lengths,” in *Proc. IEEE Int. Conference on Acoustic Speech and Signal Processing ICASSP ’09*, 2009, pp. 325–328.

[2] A. Lukin and J. Todd, “Adaptive time-frequency resolution for analysis and processing of audio,” in *AES 120th Convention Paper*, 2006.

[3] X. Zhu, G. Beauregard, and L. Wyse, “Real-Time Signal Estimation From Modified Short-Time Fourier Transform Magnitude Spectra,” *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 15, no. 5, pp. 1645–1653, 2007.

[4] V. Gnann and M. Spiertz, “Improving RTISI Phase Estimation With Energy Order and Phase Unwrapping,” in *Proc. Int. Conference of Digital Audio Effects DAFx 10*, 2010, pp. 367–371.

[5] D. Griffin and J. Lim, “Signal Estimation From Modified Short-Time Fourier Transform,” *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol. 32, no. 2, pp. 236–243, 1984.

[6] J. Le Roux, H. Kameoka, N. Ono, and S. Sagayama, “Fast signal reconstruction from magnitude STFT spectrogram based on spectrogram consistency,” in *Proc. Int. Conference of Digital Audio Effects DAFx 10*, 2010, pp. 397–403.

[7] J.-R. Ohm, *Multimedia Communication Technology: Representation, Transmission and Identification of Multimedia Signals*, Springer, 2004.

[8] European Broadcasting Union, “Sound Quality Assessment Material,” Tech 3253, 1988.