REAL-TIME ITERATIVE SPECTRUM INVERSION WITH LOOK-AHEAD

Xinglei Zhu
Institute for Infocomm Research, Singapore
xzhu@i2r.a-star.edu.sg

Gerald T. Beauregard
muvee Technologies
g.beauregard@ieee.org

Lonce Wyse
Institute for Infocomm Research, Singapore
lonce@i2r.a-star.edu.sg

ABSTRACT

In this paper, we present an algorithm for Real-time Iterative Spectrogram Inversion (RTISI) with Look-Ahead (RTISI-LA). RTISI-LA reconstructs a time-domain signal from a given sequence of short-time Fourier transform magnitude (STFTM) spectra without phase information. Whereas RTISI [1] reconstructs the current frame using only magnitude spectra information for previous frames and the current frame, RTISI-LA also uses magnitude spectra for a small number future frames. This allows RTISI-LA to achieve substantially higher signal-to-noise (SNR) performance than either RTISI or the Griffin & Lim method [2][3] with an equivalent computational load, while retaining the real-time properties of RTISI.

1. INTRODUCTION

Magnitude and power spectra, and their time sequences in the form of spectrograms, are widely used to represent the time-frequency structure of audio signals such as speech and music. They are also used where the frequency domain representation of a signal is modified before being transformed back into a time-domain signal, in a number of applications such as noise reduction, signal enhancement and signal source separation.

In general however, even when a given magnitude spectrum is calculated from a real signal, there is no way to exactly convert the magnitude spectrum back into the original time-domain real signal. Furthermore, in some applications we have only a modified (or arbitrary) magnitude spectrum, which may be not a valid representation of an audio signal in the sense that there may be only a complex, but no real signal whose STFTM exactly matches the modified one. In such cases, we would like to find a real-valued signal with an STFTM as close as possible to the modified or target STFTM. Griffin and Lim [2][3] developed an iterative least-squares error method for estimating a real audio signal from a modified STFTM (that we abbreviate 'G&L'). G&L generally reaches high quality construction after a large number of iterations. Slaney [4] developed techniques to reconstruct time-domain audio signals from cochleagrams and correlograms exploiting the G&L technique.

Based on G&L, the RTISI algorithm [1] was developed to invert spectrograms in real-time. RTISI generates the initial phase estimation for a new frame from the partially-constructed frame. With a better initial phase estimation, RTISI needs far fewer iterations than G&L to achieve acceptable quality.

RTISI also considers only the previous frames when estimating the phases for the current frame and immediately "commits" to those phases before considering the next frame. Since it does not rely on information from future frames, it has the advantage of being suitable for real-time applications, unlike G&L. However, because it does not consider any future information, the quality of reconstruction quickly plateaus as the number of iterations is increased. Depending on the signal, generally after 20 to 50 iterations, the SNR of G&L exceeds that of RTISI.

We propose a revised RTISI algorithm that relies on information in a small number of future frames. We call this method Real-Time Iterative Spectrogram Inversion with Look-Ahead (RTISI-LA). It leaves the current frame "uncommitted" until a number of future frames are considered. By choosing the look-ahead length and the number of iterations appropriately, RTISI-LA can provide excellent SNR performance while still providing low-latency.

The rest of this paper is organized as follows. In section 2 we review related work including the G&L and the RTISI algorithms. In Section 3 we present the details of the RTISI-LA algorithm. In Section 4 we evaluate RTISI-LA and compare the results with RTISI and G&L. In Section 5 we draw conclusions.

2. BACKGROUND

A discrete signal $x(n)$ can be represented as a sequence of STFT's as follows:

$$X(mS, \varpi) = \sum_{n=-\infty}^{\infty} x(n) w(n - mS) e^{-j\varpi n} \quad (1)$$

where w is the analysis window, S is the analysis step size and m is the index of the frames of STFT's. The STFT can be considered to be generated by sliding a window w across

the time domain signal with step size S. From $X(mS,\varpi)$ we can exactly reconstruct the time-domain signal $x(n)$. However in many applications we need to recover the time-domain from the magnitude spectrum $|X(mS,\varpi)|$, or a modified version $|X'(mS,\varpi)|$.

2.1. G&L algorithm

The structure of the G&L algorithm is illustrated in Figure 1. Starting with an initial estimate $x^0(n)$ of the original time-domain signal $x(n)$, the G&L algorithm iteratively renews the estimate $x^i(n)$ at the ith iteration so that the STFTM of the new estimate is monotonically closer to the STFTM of the original signal $x(n)$ in terms of the distance measure function $D_M[x(n),x^i(n)]$. The distance measure is defined as

$$D_M[x(n),x^i(n)] = \sum_{m=-\infty}^{\infty} \frac{1}{2\pi} \int_{\varpi=-\pi}^{\pi} [|X(mS,\varpi)| - |X^i(mS,\varpi)|]^2 d\varpi \quad (2)$$

where $|X(mS,\varpi)|$ is the STFTM of original signal $x(n)$ and $|X^i(mS,\varpi)|$ is the STFTM of the i^{th} estimate $x^i(n)$.

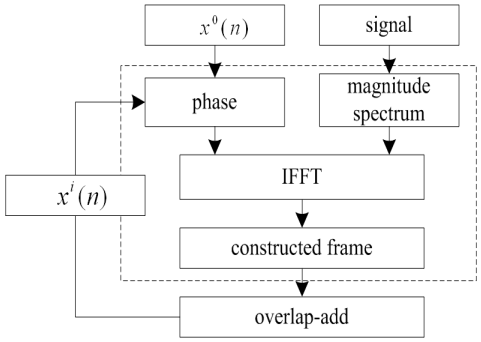

Figure 1. *The Basic iterative process in G&L. The dashed square is the magnitude spectrum constrained transform ("M-constrained transform")*

G&L uses the following function to update the estimate in each iteration,

$$x^{i+1}(n) = \frac{\sum_{m=-\infty}^{\infty} w(n-mS) \frac{1}{2\pi} \int_{\varpi=-\pi}^{\pi} \hat{X}^i(mS,\varpi) e^{-j\varpi n} d\varpi}{\sum_{m=-\infty}^{\infty} w^2(n-mS)} \quad (3)$$

where $\hat{X}^i(mS, \varpi)$ is the STFT of $x^i(n)$ with the magnitude constraint:

$$\hat{X}^i(mS,\varpi) = X^i(mS,\varpi) \frac{|X(mS,\varpi)|}{|X^i(mS,\varpi)|} \quad (4)$$

By the magnitude constraint, $\hat{X}^i(mS, \varpi)$ has the same phase as $|X^i(mS,\varpi)|$ and the same magnitude as $|X(mS,\varpi)|$. A scaled Hamming window w is used, with 75% overlap and scaling such that the sum of the squares of the overlapping windows is always 1. This simplifies the update function in Equation (3), as the denominator will be 1 for all n. In this paper we simply use w to refer to this modified Hamming window.

We will refer to the main part of Figure 1, shown in the dashed box, as the magnitude spectrum constrained transform (M-constrained transform). In each iteration the G&L algorithm concurrently applies the M-constrained transform to all the frames and overlap-adds them to get a new estimate of $x(n)$.

2.2. RTISI algorithm

In G&L, the phase estimate for a given frame is dependent on all future and all past frames in the original signal, so the algorithm is inherently non real-time. RTISI is a variant of the G&L algorithm in which a given frame's phase estimate is dependent only upon the current and previous frames of the spectrogram so that it can be used for real-time applications (where by definition the magnitude spectra of the future frames are unavailable).

RTISI is also considerably faster than G&L, thanks to better initial phase estimates. Instead of applying the M-constrained transform on all the frames concurrently, RTISI performs the M-constrained transform on a frame-by-frame basis. Before constructing frame m, RTISI uses the phase of the partial frame shown in Figure 2 as the initial phase and applies the M-constrained transform on the frame m alone. After each iteration, the constructed frame is overlap-added with the partial frame and the phase of the summation is used as the input phase of the next M-constrained transforms iteration. When frame m is generated, it is overlap-added with the partial frame and we get a new partial frame for frame m+1 and the process moves on.

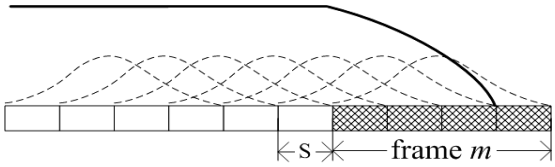

Figure 2. *An illustration of the partially reconstructed frames of signal y(n). Before frame m is estimated, there exists an overlap-added result of the frames m-1, m-2, m-3 in the range of the frame m window. The solid line shows the time domain contour of the previously constructed signal and the dotted lines show the overlap-added windows. S is the synthesis stepsize between two adjacent frames.*

3. THE RTISI-LA ALGORITHM

In RTISI only the information from previous frames is used when constructing frame m. Although RTISI generates high quality results with a small number of iterations, the lack of information from future frames prevents RTISI from further

performance improvement, even as the iteration number is increased. In this section we present RTISI with Look-Ahead (RTISI-LA), a versatile algorithm which uses information from a small number of future frames before committing the current frame.

Before the presentation of the structure, we first explain the Look-*K*-Ahead concept. In RTISI frame *m* is committed except for future frame overlap immediately after it is generated by the iterative process. By contrast, in RTISI with Look-*K*-Ahead, after we generate the frame *m,* it is kept uncommitted until the frame *m*+*K* is generated.

In Figure 3 the commitment of frame *m* is shown with *K*=3. The position of frame *m* is shown in Figure 3(a) in shade. We use a frame buffer (shown in Figure 3(b)) to hold the committed frames overlapped with frame *m,* frame *m* itself, and *K* future frames. We use a fixed 75% overlap rate between the adjacent frames in our system so the number of committed frames overlapped with frame *m* is 3. In Figure 3 the number of look ahead frames *K* is 3 but it can be any non-negative integer (Note that if *K*=0, RTISI-LA is identical to RTISI).

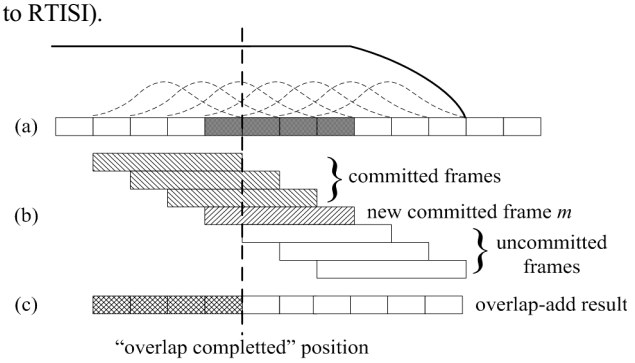

Figure 3. *RTISI with Look-Ahead after committing frame m. (a) Constructed signal with the indication of the contour (solid line) and overlapped windows (dashed line). (b) The frames being processed in the frame buffer. There are three kinds of frames in the buffer: frames committed in the previous process; the newly- committed frame m and the uncommitted frames. (c)The overlap-add result of the frames in the buffer. The shaded part is the "overlap completed" signal.*

When frame *m* is initially generated, we leave it uncommitted in the frame buffer and move forward until we reach the frame *m*+*K*. Then we use the partial frame to estimate the initial phase for frame *m*+*K* and concurrently apply the M-constrained transform to all the uncommitted frames in the frame buffer (frames m to *m*+*K*) using the corresponding magnitude spectra. We overlap-add all the frames in the frame buffer and obtain an overlap-add result, as shown in Figure 3(c). Note at that time frame *m* is still noted as uncommitted. We read in all the uncommitted frames (frame *m* to *m*+*K*) using the scaled Hamming window from the overlap-add result. Then we concurrently apply the M-constrained transform again. This process is repeated for a given number of iterations and frame m is noted to be committed. The committed position is moved one step forward and set at the position of the first quarter of frame *m*.

In the above process, a committed frame in the frame buffer will not be changed. However it still plays its role in the frames with which it overlaps. As shown in Figure 3(c), only the "overlap completed" frame (meaning the segment where all overlapping fames have been committed) is output. Then we remove frame *m*-3 from the frame buffer and read in the partial frame *m*+*K*+1 and repeat the above iterative process for frame *m* to *m*+*K*+1.

The basic framework of the RTISI-LA is shown in Figure 4. In each step we concurrently apply the M-constrained transform to all the uncommitted frames in the frame buffer and update the uncommitted frames by overlap-adding and window. This process is repeated until the maximum iteration number is reached. Then the "overlap completed" signal is output and the commitment information in the frame buffer is revised. Then we move one step forward and pursue the same process and go on until we reach the end of the given magnitude spectra.

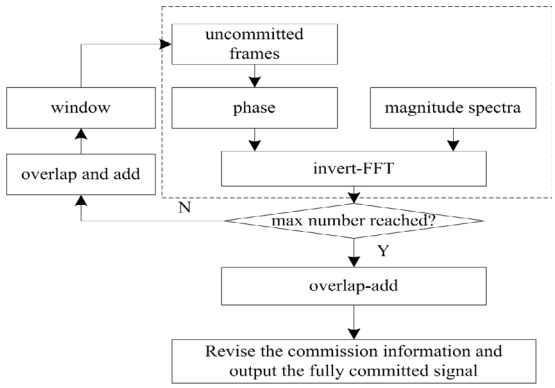

Figure 4. *Framework of RTISI-LA algorithm. The dashed square is the M-constrained transform.*

For the initial step there are no frames in the frame buffer so we start with zero phase for the first frame and then pursue the same process as above and leave all the frames in the frame buffer uncommitted until we finish frame *K*, then we commit the first frame and output the first *L*/4 samples of the overlap-add result.

The computational load is determined mainly by the total number of M-constrained transforms required per frame. For a given frame, the number of M-constrained transforms is the number of iterations times *K*+1 (i.e. the current frame plus the number of look-ahead frames).

4. EVALUATION

We evaluate the phase reconstruction result using a SNR function similar to the one in [5], comparing the spectrogram of the reconstructed signal to that of the target:

$$SNR = 10\log\frac{\frac{1}{E}\sum_w\sum_f |s_w(f)|^2}{\sum_w\sum_f (\frac{1}{\sqrt{\hat{E}}}|\hat{s}_w(f)| - \frac{1}{\sqrt{E}}|s_w(f)|)^2} \quad (5)$$

where $|S_w(f)|$ is the STFTM of the original signal, $|\hat{S}_w(f)|$ is the STFTM of the reconstructed signal, E is the total energy in the original signal and $\hat{E}$ is the total energy in the reconstructed signal, and summations over w and f are over all windows and frequencies respectively.

In RTISI-LA, the total iteration number for a frame is determined by the number of look-ahead frames and the given number of iterations in each step. With 75% overlap between adjacent frames, frame m+4 has no overlap with frame m so that the magnitude spectrum of frame m+4 has no direct relationship with the magnitude spectrum of frame m. In our experiments, looking more than 3 frames ahead gives limited performance improvement, generally less than 1dB.

Figure 5 shows the SNR of the construction result of a male speech signal using G&L, RTISI and RTISI-LA with look-ahead number K=3. We can see that RTISI achieves better SNR than G&L in only a very few iterations but the SNR remains almost stable after that. RTISI-LA achieves almost the same performance level as RTISI in 4 iterations and is substantially improved when the iteration number is 8 or 12. Like G&L, the SNR continuous to increase with the iteration number. (Note that the total iteration number of RTISI-LA can only be a multiple of K+1.)

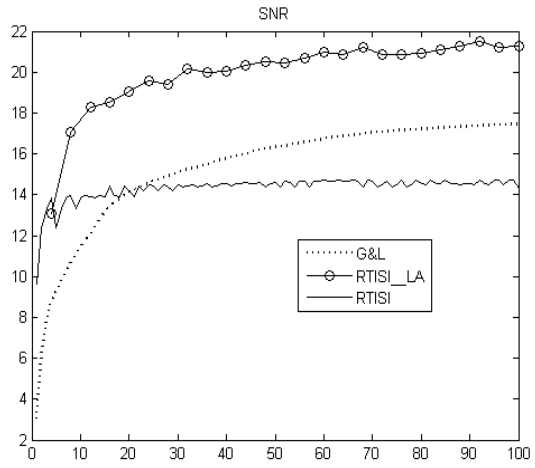

Figure 5. *SNR of the construction result of a male speech signal using G&L, RTISI and RTISI-LA (with look-ahead number of 3) respectively.*

We evaluate the RTISI-LA with look-ahead number K=3 and G&L and RTISI algorithm on a test set containing 15 musical and speech signals. The average SNR is shown in Table 1. From the result we can see that under few iterations, RTISI generates the best quality construction signal but stays stable with more iterations; under a greater number of iterations (>4), RTISI-LA generates better result than both of the other algorithms and the SNR keeps growing with the increase of iteration numbers. The performance of RTISI-LA at 12 iterations is at the same level or even better than the result from G&L at 100 iterations.

Table 1. average SNR (in dB) of construction result

M-constrained Transforms per Frame	G&L	RTISI	RTISI-LA
4	9.42	15.31	13.58
12	12.78	15.48	19.17
100	19.06	15.81	23.12

.

5. CONCLUSIONS

A versatile real-time magnitude spectrum inversion algorithm, RTISI-LA, is presented. Based on RTISI, the new algorithm utilizes the information of a number of future frames to achieve a better match between adjacent frames. With zero look-ahead frames, RTISI-LA is identical to the normal RTISI algorithm. The performance of RTISI-LA with a few iterations is comparable with RTISI. Increasing iteration improves performance even beyond that of G&L. RTISI-LA inherits advantages from both the G&L algorithm and the RTISI algorithm, although the error does not decrease strictly monotonically as it dose for G&L.

Future work includes the quantification of the frame transition artifacts that are not reflected in spectral SNR measures for this family of signal reconstruction techniques. Further work is also necessary to improve the ability to capture and incorporate transient behavior from source signals in applications where such information is available.

6. REFERENCES

[1] G.T. Beauregard, X. Zhu, L. Wyse, " An Efficient Algorithm for Real-Time Spectrogram Inversion", *Proc. of the 8th Int. Conference on Digital Audio Effects (DAFX-05)*, Madrid, Spain, Sep 2005.

[2] D.W. Griffin, J.S. Lim, " Signal Estimation from Modified Short-Time Fourier Transform", *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol.32, no. 2, Apr, 1984.

[3] S.H. Nawab, T.F. Quatieri, J.S. Lim, " Signal Reconstruction from Short-Time Fourier Transform Magnitude", *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol.31, no. 4, Aug 1983.

[4] M. Slaney, D.Naar, R.F. Lyon, " Auditory Model Inversion for Sound Separation" , *IEEE International Conference on Acoustics, Speech, and Signal Processing*, vol. 2, pp. 77-80, Apr 1994.

[5] K. Achan, S.T. Roweis, B.J. Frey, "Probabilistic Inference of Speech Signals from Phaseless Sepctrograms" , *Neural Information Processing Systems*, pp. 1393-1400, 2003.