Research Letter

An Adaptive Resolution Computationally Efficient Short-Time Fourier Transform

Saeed Mian Qaisar, Laurent Fesquet, and Marc Renaudin

TIMA, CNRS UMR 5159, 46 avenue Felix-Viallet, 38031 Grenoble Cedex, France

Correspondence should be addressed to Saeed Mian Qaisar, saeed.mian-qaisar@imag.fr

Received 31 January 2008; Accepted 22 April 2008

Recommended by Sos Agaian

The short-time Fourier transform (STFT) is a classical tool, used for characterizing the time varying signals. The limitation of the STFT is its fixed time-frequency resolution. Thus, an enhanced version of the STFT, which is based on the cross-level sampling, is devised. It can adapt the sampling frequency and the window function length by following the input signal local characteristics. Therefore, it provides an adaptive resolution time-frequency representation of the input signal. The computational complexity of the proposed STFT is deduced and compared to the classical one. The results show a significant gain of the computational efficiency and hence of the processing power.

Copyright © 2008 Saeed Mian Qaisar et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

1. INTRODUCTION

Most of the real-life signals like speech, Doppler, seismic, and biomedical signals are time varying in nature. The spectral contents of these signals vary with time, which is a direct consequence of the signal generation process [1]. The STFT is a classical tool for characterizing such signals [2]. The limitation with the STFT is that it provides a fixed resolution time-frequency representation of the input signal. This fixed resolution is the reason for the creation of the multiresolution analysis (MRA) techniques [3–5], which provide a good frequency but a poor time resolution for the low-frequency events and a good time but a poor frequency resolution for the high-frequency events. This type of analysis is well suited for most of the real-life signals [3].

In this article, the fixed resolution dilemma is resolved to a certain extent by revising the STFT. The motivation behind the proposed STFT is to achieve a smart time-frequency representation of the time varying signals. The idea is to adapt the time-frequency resolution along with the computational load by following the input signal local characteristics. An efficient solution is proposed by smartly combining the features of both uniform and nonuniform signal processing tools.

2. PROPOSED ADAPTIVE RESOLUTION STFT

The block diagram of the proposed STFT is shown in Figure 1. The description of different blocks is given below.

2.1. Asynchronous analog to digital converter (AADC)

According to [6], the sampling instants of a nonuniformly sampled signal obtained with the level crossing sampling scheme (LCSS) are defined by (1). Where \( t_n \) is the current sampling instant, \( t_{n-1} \) is the previous one, and \( dt_n \) is the time delay between the current and the previous sampling instants (cf. (2)).

The LCSS drastically reduces the activity of the post processing chain, because it only captures the relevant information [7–9]. In this context, analog to digital converters based on the LCSS have been developed [10–12]. The AADC [10] is employed for digitizing \( x(t) \). An \( M \)-bit resolution AADC has \( 2^M - 1 \) quantization levels which are uniformly disposed according to \( x(t) \) amplitude dynamics. The AADC has a finite bandwidth. Thus, to assure a proper signal capturing a band pass filter with pass band \( [f_{min}; f_{max}] \) is employed at the AADC input. Let \( \Delta V_{in} \) and \( \Delta x(t) \) be the AADC and \( x(t) \) amplitude dynamics, respectively. In order to avail the complete AADC resolution in the studied case,
2.2. Enhanced activity selection algorithm (EASA) and window selector

The relevant parts of the nonuniformly sampled signal obtained with the AADC are selected—corresponds to the variable length rectangular window—by the EASA. The EASA is defined as shown in Algorithm 1. \( T_0 = 1/f_{\text{min}} \) is the fundamental period of \( x(t) \). \( T_0 \) and \( dt_n \) detect parts of the nonuniformly sampled signal with activity. The condition on \( dt_n \) is chosen in order to satisfy the Nyquist criterion for \( f_{\text{min}} \), when sampling \( x(t) \) nonuniformly with the AADC [13]. \( N^i \) represents the number of nonuniform samples lie in the \( i \)th selected window, which lie on the \( j \)th active part of the nonuniformly sampled signal. Where, \( i \) and \( j \) both belong to the set of natural numbers \( \mathbb{N}^+ \). \( N_{\text{ref}} \) represents the upper bound on \( N^i \). The choice of \( N_{\text{ref}} \) depends on \( x(t) \) characteristics and on system parameters. The above described loop repeats for each selected window, which occurs during the observation length of \( x(t) \). Every time before repeating the loop, \( i \) is incremented and \( N^i \) is initialized to zero.

The EASA displays interesting features with the LCSS, which are not available in the classical case. It selects only the active parts of the nonuniformly sampled signal, obtained with the AADC. Moreover, it correlates the length of the selected window with the signal local characteristics.

The window selector implements the condition given by expression (5). Jointly, the EASA and the window selector provide an efficient spectral leakage reduction in the case of transient signals [13]. Indeed, spectral leakage occurs due to the signal truncation problem. Usually an appropriate smoothing (cosine) window function is employed to reduce the signal truncation. For the proposed case, as long as the condition 5 is true, the leakage problem is resolved by avoiding the signal truncation. As no signal truncation occurs so no cosine window is required. In this case the window decision \( D^i = 1 \), which makes the switch state 1 (cf. Figure 1)). Otherwise, an appropriate cosine window is employed to reduce the signal truncation problem. In this case \( D^i = 0 \), which makes the switch state 0. In expression 5, \( t_1^i \) represents the 1st sampling instant of the \( i \)th selected window and \( t_{\text{end}}^{i-1} \) represents the last sampling instant of the \( (i-1) \)th selected window.

For proper spectral representation, the condition given by expression (6) should be satisfied [13]. Where, \( L^i \) is the length in seconds of the \( i \)th selected window. In order to satisfy this condition for the worst case, which occurs for \( F_{\text{max}}, N_{\text{ref}} \) is calculated for an appropriate reference window length \( L_{\text{ref}} \). Where, \( L_{\text{ref}} \) satisfies the condition \( L_{\text{ref}} \geq T_0 \). The process is given by (7) as follows:

$$
\text{if } \left( N^i \leq N_{\text{ref}} \text{ and } (T_d^i = t_1^i - t_{\text{end}}^{i-1}) > \frac{T_0}{2} \right),
$$

$$
L^i \geq T_0,
$$

$$
N_{\text{ref}} = L_{\text{ref}} \cdot F_{\text{max}}.
$$

The lower and the upper bounds on \( L_{\text{ref}} \) are posed, respectively, by \( T_0 \) and the system resources (the maximum sample frame which the system can process at once). For \( N_{\text{ref}} \) (cf. (7)), the condition 6 holds for all selected windows except for the case when the actual length of the \( j \)th activity is less than \( T_0 \).

2.3. Adaptive sampling rate

The AADC sampling frequency is correlated to \( x(t) \) local variations [7, 13]. Let \( F^i \) represent the AADC sampling frequency for the \( i \)th selected window. \( F^i \) can be calculated
by using (8). Where, \( t_{\max}^{i} \) and \( t_{\min}^{i} \) are the final and the initial times of the \( i \)th selected window. The upper and the lower bounds on \( F_s^{i} \) are posed by \( F_{s_{\max}} \) and \( F_{s_{\min}} \), respectively:

\[
L_i = t_{\max}^{i} - t_{\min}^{i}, \\
F_s^{i} = \frac{N_{i}}{L_i}.
\]

The selected data obtained with the EASA can be used directly for further nonuniform digital processing [8, 14]. However in the studied case, the selected data is resampled uniformly. It enables to take advantage of both nonuniform and uniform signal processing tools [7, 13]. Due to this resampling there will be an additional error. Nevertheless, prior to this transformation, one can take advantage of the inherent oversampling of the relevant signal parts in the system [7]. Hence, it adds to the accuracy of the post resampling process [11]. The nearest neighbour resampling interpolation (NNRI) is employed for data resampling. The reasons of inclination towards NNRI are discussed in [13, 15].

A reference sampling frequency \( F_{ref} \) is chosen, such as itremains greater than and closest to the \( F_{S_{Nyq}} = 2 \cdot f_{max} \). Depending upon the values of \( F_{ref} \) and \( F_{s}^{i} \), the resampling frequency \( Frs^{i} \) (cf. (Figure 1)) can be adapted for the \( i \)th selected window. For the case, \( F_s^{i} > F_{ref} \), \( Frs^{i} \) is chosen as: \( Frs^{i} = F_{ref} \). It is done in order to resample the selected data, lie in the \( i \)th selected window closer to the Nyquist frequency. It avoids the unnecessary interpolations during the data resampling process and so reduces the computational load of the proposed technique.

For the case, \( F_s^{i} \leq F_{ref} \), \( Frs^{i} \) is chosen as: \( Frs^{i} = F_s^{i} \). In this case, it appears that the data lie in the \( i \)th selected window may be resampled at a frequency which is less than \( F_{S_{Nyq}} \) and so it can cause aliasing. Since, the sampling rate of the AADC varies according to the slope of \( x(t) \) [10]. A high-frequency signal part has a high slope and the AADC samples it at a higher rate and vice versa. Hence, a signal part with only low-frequency components can be sampled by the AADC at a subNyquist frequency of \( x(t) \). But still this signal part is locally sampled in time with respect to its local bandwidth [7]. Hence, there is no danger of aliasing. This statement is further illustrated with the results summarized in Table 1.

### 2.4. Adaptive resolution analysis

The STFT of a sampled signal \( x_n \) is determined by computing the discrete Fourier transform (DFT) of an \( N \) samples segment centred on \( \tau \), which describes the spectral contents of \( x_n \) around the instant \( \tau \). Where \( N \) is defined as: \( N = L \cdot F_s \). Here, \( L \) is the effective length in seconds of the window function \( w_n \) and \( F_s \) is the sampling frequency. The STFT can be expressed mathematically by (9). In Equation (9), \( f \) is the frequency index, which is normalized with respect to \( F_s \).

\[
X[\tau, f] = \sum_{n=\tau-L/2}^{\tau+L/2} [x_n \cdot w_{n-\tau}] \cdot e^{-j \cdot 2\pi \cdot f \cdot n}.
\]

The proposed STFT is a smart alternative of the MRA techniques. It performs adaptive time-frequency resolution analysis, which is not attainable with the classical STFT. It is achieved by adapting the \( Frs^{i}, L_{i}, \) and \( Nr^{i} \) according to the local variations of \( x(t) \). \( Nr^{i} \) is the number of resampled data points that lie in the \( i \)th selected window. Thus, the time resolution \( \Delta t^{i} \) and the frequency resolution \( \Delta f^{i} \) of the proposed STFT can be specific for the \( i \)th selected window and are defined by (12) and (13), respectively. Because of this adaptive resolution, the proposed STFT will be named as the adaptive resolution STFT, (ARSTFT) throughout the following parts of this article. The adaptation of \( Frs^{i}, L_{i}, \) and \( Nr^{i} \) also adds to the computational gain of the ARSTFT, compared to the classical one. It is achieved firstly by avoiding the unnecessary samples to process and secondly by avoiding the use of the cosine window function as far as the condition 5 is true. The ARSTFT is defined by (14).

\[
\begin{align*}
\Delta t^{i} & = L_{i}, \\
\Delta f^{i} & = \frac{F_{s_{\max}}}{Nr^{i}}.
\end{align*}
\]

\[
X[\tau^{i}, f^{i}] = \sum_{n=\tau^{i}-Nr^{i}/2}^{\tau^{i}+Nr^{i}/2} [\text{Re sample}(x_n, t_n)] \cdot w_{n-\tau^{i}}^{t^{i}} \cdot e^{-j \cdot 2\pi \cdot f^{i} \cdot n}.
\]

### 3. Illustrative Example

In order to illustrate the ARSTFT an input signal \( x(t) \), shown on the left part of Figure 2 is employed. Its total duration is 30 seconds and it consists of three active parts. Each activity is a sinusoid of 0.9 v amplitude and of 50, 200, and 500 Hz frequency, respectively. The time length of each activity is 5, 0.5, and 1.6 seconds, respectively. \( x(t) \) is band limited between 50 to 500 Hz and it is sampled by employing a 3-bit resolution AADC. Thus, \( F_{s_{\max}} \) and \( F_{s_{\min}} \) become 7 kHz and 0.7 kHz, respectively (3), (4). \( F_{ref} = 1.25 \text{kHz} \) and \( \Delta V_{in} = 1.8 \text{v} \) are chosen. The selected data obtained with the EASA is shown on the right part of Figure 2. By following the criteria
The ARSTFT also adapts the window shape (rectangle or cosine) for the \( i \)th selected window. The condition 5 remains true for the first two selected windows, which sets \( D_i = 1 \). As no signal truncation occurs so no cosine window is required in this case. On the other hand, the number of samples for the fourth activity is 11200. Therefore, \( N_{\text{ref}} = 4096 \) leads to the three selected windows for the time span of the fourth activity. The condition 5 becomes false in this case, which sets \( D_i = 0 \). As signal truncation occurs so suitable length cosine (Hanning) windows are employed to reduce this effect.

In the classical case, if \( F_s = F_{\text{ref}} \) is chosen, in order to satisfy the Nyquist sampling criterion for \( x(t) \). Then the whole signal will be sampled at 1.25 kHz, regardless of its local variations. It will produce unnecessary samples than required. Moreover, the windowing process is not able to select only the active parts of the sampled signal. In addition, \( L \) remains static and is not able to adapt with the signal local variations. Thus, it causes the system to process needless samples and so causes an increased computational activity than the proposed case. For classical case, fixed \( N = 4096 \) will produce nine fixed \( L = 3.3 \) second windows, for the total \( x(t) \) time span of 30 seconds. It will lead to fix \( \Delta t = 3.3 \) seconds and \( \Delta f = 0.31 \text{Hz} \) for all nine windows (cf. (10) and (11)).

### 4. COMPUTATIONAL COMPLEXITY

This section compares the computational complexity of the ARSTFT with the classical STFT. The complexity evaluation is made by considering the number of operations executed to perform the algorithm.

In the classical case, \( F_s \) is fixed. In this case, a time invariant, fixed \( L \), cosine window function is employed to window the sampled data. If \( N \) is the number of samples lie in the window then the windowing operation will perform \( N \) multiplications between \( w_n \) and \( x_n \) (cf. (9)). The spectrum of the windowed data is obtained by computing its DFT. A complex term is involved in the DFT computation. The DFT complexity is calculated by taking the real and the imaginary parts separately. The DFT performs \( 2 \cdot (N)^2 \) additions and \( 2 \cdot (N)^2 \) multiplications, thus operations count becomes \( 4 \cdot (N)^2 \) for \( N \) output frequencies. The combined computational complexity \( C_i \) of the STFT is given by (15). Where, \( A \) is the total number of windows occurs for the observation length of \( x(t) \).
For the proposed ARSTFT, $F_{si}$, $F_{rsi}$, and $w_{ni}$ are not fixed and are adapted according to the local variations of $x(t)$. The EASA performs $2 \cdot N_{i}$ comparisons and $N_{i}$ increments for the $i$th selected window (cf. (Section 2)). The choice of $F_{rsi}$ and window shape requires three comparisons. The selected signal is resampled before computing its DFT. The NNRI is employed for the resampling purpose. The NNRI only requires a comparison operation for each resampled observation. Therefore, the resampler performs $N_{r} \cdot i$ comparisons. If $D = 0$, then a cosine window function is applied on the resampled data, which performs $N_{r} \cdot i$ multiplications (cf. (Figure 1)). The DFT performs $4 \cdot (N_{r} \cdot i)^{2}$ operations for the $i$th selected window. The combine computational complexity $C_{2}$ of the ARSTFT is given by (16). Where $j = 1, 2, \ldots, K$ represents the index of the selected window. $a$ is a multiplying factor, its value is 1 for $D = 0$ and 0 for $D = 1$. The computational gain of the ARSTFT over the classical one is calculated by employing the results of the illustrative example. The results are summarized in Table 3.

$$C_{1} = A \cdot [N + 4 \cdot (N)^{2}].$$

$$C_{2} = K \cdot 3 + \sum_{i=1}^{K} 3 \cdot N_{i} + a \cdot N_{r} \cdot i + N_{r} \cdot i + 4 \cdot (N_{r} \cdot i)^{2}. $$

Table 3 shows the computational gain of the ARSTFT over the STFT for each $x(t)$ activity. It shows that the ARSTFT leads to a significant reduction of the total number of operations as compared to the classical one. This reduction in operations is achieved by adapting $F_{si}$, $F_{rsi}$, and $w_{ni}$ according to the local variations of $x(t)$.

### 5. CONCLUSIONS

A new tool for the adaptive resolution time-frequency analysis is proposed. The ARSTFT is especially well suited for the low activity sporadic signals like electrocardiogram, phonocardiogram, seismic signals, and so forth. It is shown that $F_{si}$ and $L_{j}$ adapt by following the $x(t)$ local variations. Criteria to choose the appropriate $F_{ref}$ and $N_{ref}$ are developed. A complete methodology of adapting $F_{rsi}$ and $w_{ni}$ for the $i$th selected window has been demonstrated.

The ARSTFT outperforms the STFT. The advantages of the ARSTFT over the STFT are the adaptive time-frequency resolution and the computational gain. These smart features of the ARSTFT are achieved due to the joint benefits of the AADC, the EASA, and the resampling as they enable to adapt $F_{si}$, $F_{rsi}$, $N_{i}$, $N_{r} \cdot i$, and $w_{ni}$ by exploiting the local variations of $x(t)$. The employment of fast algorithms in place of the DFT for the spectrum computation is in progress, it will further add up to the computational efficiency of the ARSTFT. Moreover, the performance comparison of the ARSTFT with other MRA techniques, in terms of computational complexity and quality, opens the way to new research prospective.

### REFERENCES

1. S. C. Sekhar and T. V. Sreenivas, “Adaptive window zero-crossing-based instantaneous frequency estimation,” EURASIP Journal on Applied Signal Processing, vol. 2004, no. 12, pp. 1791–1806, 2004.
2. D. Gabor, “Theory of communication,” Journal of the IEE, vol. 93, no. 3, pp. 429–457, 1946.
3. R. Polikar, “The engineer’s ultimate guide to wavelet analysis,” Rowan University, College of Engineering, retrieved in 2006.
4. M. Vetterli and C. Herley, “Wavelets and filter banks: theory and design,” IEEE Transactions on Signal Processing, vol. 40, no. 9, pp. 2207–2232, 1992.
5. H. K. Kwock and D. L. Jones, “Improved instantaneous frequency estimation using an adaptive short-time Fourier,” IEEE Transactions on Signal Processing, vol. 48, no. 10, pp. 2964–2972, 2000.
6. J. Mark and T. Todd, “A nonuniform sampling approach to data compression,” IEEE Transactions on Communications, vol. 29, no. 1, pp. 24–32, 1981.
7. S. M. Qaisar, L. Fesquet, and M. Renaudin, “Computationally efficient adaptive rate sampling and filtering,” in Proceedings of the 15th European Signal Processing Conference (EUSIPCO ’07), pp. 2139–2143, Poznan, Poland, September 2007.
8. M. Gretains, “Time-frequency representation based chirp like signal analysis using multiple level crossings,” in Proceedings of the 15th European Signal Processing Conference (EUSIPCO ’07), pp. 2154–2158, Poznan, Poland, September 2007.
9. K. M. Guan and A. C. Singer, “Opportunistic sampling by level-crossing,” in Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP ’07), vol. 3, pp. 1513–1516, Honolulu, Hawaii, USA, April 2007.
10. E. Allier, G. Sicard, L. Fesquet, and M. Renaudin, “A new class of asynchronous A/D converters based on time quantization,” in Proceedings of the 9th IEEE International Symposium on Asynchronous Circuits and Systems (ASYNC ’03), pp. 196–205, Vancouver, BC, Canada, May 2003.
11. N. Sayiner, H. V. Sorensen, and T. R. Viswanathan, “A level-crossing sampling scheme for A/D conversion,” IEEE Transactions on Circuits and Systems II, vol. 43, no. 4, pp. 335–339, 1996.
12. F. Akopyan, R. Manohar, and A. B. Apsel, “A level-crossing flash asynchronous analog-to-digital converter,” in Proceedings of the 12th IEEE International Symposium on Asynchronous Circuits and Systems (ASYNC ’06), pp. 12–22, Grenoble, France, March 2006.
13. S. M. Qaisar, L. Fesquet, and M. Renaudin, “Spectral analysis of a signal driven sampling scheme,” in Proceedings of the 14th European Signal Processing Conference (EUSIPCO ’06), Florence, Italy, September 2006.
14. F. Aeschlimann, E. Allier, L. Fesquet, and M. Renaudin, “Asynchronous FIR filters: towards a new digital processing chain,” in Proceedings of the 10th IEEE International Symposium on Asynchronous Circuits and Systems (ASYNC ’04), pp. 198–206, Crete, Greece, April 2004.
15. S. de Waele and P. M. T. Broersen, “A time domain error measure for resampled irregular data,” in Proceedings of the 16th IEEE Instrumentation and Measurement Technology Conference (IMTC ’99), vol. 2, pp. 1172–1177, Venice, Italy, May 1999.
Submit your manuscripts at http://www.hindawi.com