International Journal of Electrical, Electronic and Communication Sciences
ISSN: 2517-9438
Vol:2, No:5, 2008

An Improved Quality Adaptive Rate Filtering Technique Based on the Level Crossing Sampling

Saeed Mian Qaisar, Laurent Fesquet, and Marc Renaudin

***Abstract*—**Mostly the systems are dealing with time varying signals. The Power efficiency can be achieved by adapting the system activity according to the input signal variations. In this context an adaptive rate filtering technique, based on the level crossing sampling is devised. It adapts the sampling frequency and the filter order by following the input signal local variations. Thus, it correlates the processing activity with the signal variations. Interpolation is required in the proposed technique. A drastic reduction in the interpolation error is achieved by employing the symmetry during the interpolation process. Processing error of the proposed technique is calculated. The computational complexity of the proposed filtering technique is deduced and compared to the classical one. Results promise a significant gain of the computational efficiency and hence of the power consumption.

***Keywords*—**Level Crossing Sampling, Activity Selection, Rate Filtering, Computational Complexity, Interpolation Error.

I. Context of the Study

THIS work is a contribution in the development of smart mobile systems. The goal is to reduce their size, cost, processing noise, electromagnetic emission and especially power consumption as they are remotely powered by batteries. This can be done by smartly reorganizing their associated signal processing theory and architecture. The idea is to combine the signal event driven processing with the asynchronous circuit design in order to reduce the system processing activity.

Most of the real life signals like speech, seismic, Doppler and biological 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 [5].

Classical systems are based on the Nyquist signal processing architectures. They do not take advantage of the input signal local variations. These systems are highly constrained due to the Shannon theory especially in the case of low activity sporadic signals like electrocardiogram, phonocardiogram, seismic signals etc. It causes a large number of samples without any relevant information, a useless increase of the system activity and so a useless increase of the power consumption.

Saeed Mian Qaisar is a PhD candidate in Laboratory TIMA, CNRS UMR 5159, 46 Avenue Felix-Viallet, 38031 Grenoble Cedex, France (phone: +33-476574646; fax: +33-476574981; e-mail: saeed.mian-qaisar@imag.fr).

Laurent Fesquet is an associate professor at INPG and is working with Laboratory TIMA, CNRS UMR 5159, 46 Avenue Felix-Viallet, 38031 Grenoble Cedex, France (e-mail: laurent.fesquet@imag.fr).

Marc Renaudin is a professor at INPG and is working with Laboratory TIMA, CNRS UMR 5159, 46 Avenue Felix-Viallet, 38031 Grenoble Cedex, France (e-mail: marc.renaudin@imag.fr).

This problem is resolved by employing a signal driven sampling scheme, which is sensitive to the input signal local variations [12, 17]. It is based on the principle of "level-crossing" that provides a non-uniform time repartition of the samples [1], consequently it is named as the LCSS (level crossing sampling scheme). This sampling scheme drastically reduces the activity of the post processing chain because it only captures the relevant information [11, 13]. In this context, analog to digital converters based on the LCSS have been developed [2, 4, 18]. Algorithms for processing [3, 11, 13] and analysis [8, 12, 19] of the non-uniformly spaced out in time sampled data obtained with the LCSS have also been developed.

The focus of this work is to develop an efficient online FIR (Finite Impulse Response) filtering technique. The idea is to extract the input signal local features and then use them to improve the quality and to reduce the computational load of the post processing chain. An efficient solution is proposed by combining the features of both non-uniform and uniform signal processing tools.

II. LCSS (Level Crossing Sampling Scheme)

In the case of LCSS, a sample is captured only when the input analog signal *x(t)* crosses one of the predefined threshold levels [1]. The samples are not uniformly spaced in time because they depend on *x(t)* variations as it is clear from Fig. 1. Thus, the non-uniformity in the sampling process reflects the local characteristics of *x(t)* [12].

According to [1], the sampling instants of a non-uniformly sampled signal obtained with the LCSS are defined by Equation 1.

$$t_n = t_{n-1} + dt_n \,. \quad (1) \qquad dt_n = t_n - t_{n-1} \,. \quad (2)$$

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. Equation 2).

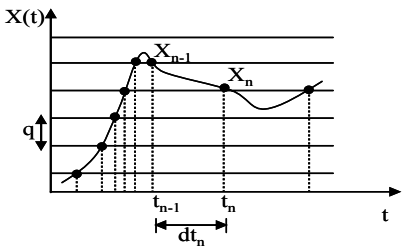

Fig. 1 Level-crossing sampling scheme

III. Proposed Adaptive Rate Filtering Technique

The activity selection and local features extraction [8] is the base of the proposed technique. It makes to achieve the

adaptive rate sampling (only relevant number of samples to process) along with adaptive rate filtering (only relevant number of operations to deliver per filtered sample). These achievements assure a drastic computational gain of the proposed filtering technique compared to the classical one. The approaches to realize it are detailed in the following subsections.

The block diagram of the proposed filtering technique is shown in Fig. 2. The description of different blocks is given in the following subsections. This technique is splitted into two filtering cases, which are explained in Section III-D.

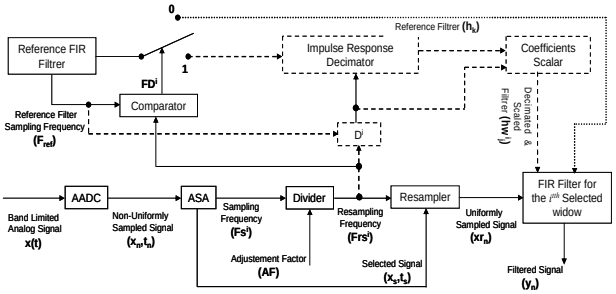

Fig. 2 Block diagram of the Proposed Filtering Technique. '—' represents the common blocks and signal flow used in both filtering cases,'·····' represents the signal flow used only in case 1 and '---' represents blocks and the signal flow used only in case 2

A. AADC (Asynchronous Analog to Digital Converter) and ASA (Activity Selection Algorithm)

The AADC [2], 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.

Reconstruction issue of the non-uniformly sampled signal has been discussed in [7, 14]. In [14], author showed that a bandlimited signal can be ideally reconstructed from its non-uniformly spaced samples, provided that the average number of samples satisfies the Nyquist criterion. In the case of AADC, the number of samples is directly influenced by *M* and the signal characteristics [2, 4, 18]. Thus, for a given application an appropriate *M* should be chosen in order to respect the reconstruction criterion [14].

Let ΔV_{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, $\Delta x(t)$ is always adapted to match ΔV_{in}. For a given *M*, The AADC maximum sampling frequency Fs_{max} and minimum sampling frequency Fs_{min} [11] are defined by Equations 3 and 4 respectively. Where, f_{max} and f_{min} are the bandwidth and the fundamental (lowest) frequencies of *x(t)* respectively.

$$Fs_{max} = 2.f_{max}.(2^M\text{-}1). \quad (3) \qquad Fs_{min} = 2.f_{min}.(2^M\text{-}1). \quad (4)$$

The relevant (active) parts of the non-uniformly sampled signal are selected and windowed by the ASA. The complete procedure of activity selection has been explained in [8]. The ASA displays interesting features with the LCSS, which are not available in the classical case. It selects only the interesting parts of the non-uniformly sampled signal obtained with the AADC. Moreover, it correlates the length of the selected window with the signal activity, lies in it. In addition, it also efficient reduction of the phenomenon of spectral leakage in the case of transient signals. This is done by avoiding the signal truncation with a simple and efficient algorithm instead of a smoothening window function, which is used in the classical scheme [8].

B. Adaptive Rate Sampling

For a fixed *M*, the temporal density of the AADC sampling operation is a function of the input signal variations [11, 13]. Let Fs^i represents the AADC sampling frequency for the i^{th} selected window. Fs^i can be specific for each selected window, depending upon the window length Ts^i in seconds and the slope of *x(t)* part lying within this window [8]. Fs^i can be calculated by using the following equations.

$$Ts^i = tmax^i - tmin^i \ . \ (5) \qquad Fs^i = N^i / Ts^i \ . \qquad (6)$$

In Equation 5, $tmax^i$ and $tmin^i$ are the final and the initial times of the i^{th} selected window. The upper and the lower bounds on Fs^i are posed by Fs_{max} and Fs_{min} respectively.

C. Adaptive Rate Resampling

The non-uniformly sampled signal obtained with the AADC can be used directly for further non-uniform digital processing [3, 12]. However in the studied case, the AADC output is first selected with the ASA and then resampled uniformly before proceeding towards further processing. It enables to take advantage of both non-uniform and uniform signal processing tools [8, 11, 13]. An additional error occurs due to this resampling. Nevertheless, prior to this transformation, one can take advantage of the inherent over-sampling of the relevant signal parts in the system [11, 13]. Hence it improves the accuracy of the post resampling process [4].

The resampling process requires interpolation, which changes the properties of the resampled signal compared to the original one. The properties of the resampled signal depend upon the interpolation technique used to resample it [9, 10]. The NNRI (nearest neighbour resampling interpolation) is employed for data resampling. The reasons of inclination towards NNRI are discussed in [8, 9, 10].

A study on the interpolation error is made by taking into account an academic signal, known formally. If (tr_n, xr_n) represent the time-amplitude pairs of the n^{th} interpolated sample. Then the original sample amplitude xo_n can be mathematically calculated for tr_n, as the input signal is analytically known. Now the interpolation error per interpolated point Ie_n is given by the absolute difference between xo_n and xr_n. The process is given by Equation 7. The mean interpolation error for the i^{th} selected window MIe^i can be calculated by using Equation 8. In Equation 8, Nr^i is the number of resampled data points lie in the i^{th} selected window.

$$Ie_n = \left| xo_n - xr_n \right| \ . \ (7) \qquad MIe^i = \frac{1}{Nr^i} . \sum_{n=1}^{Nr^i} Ie_n \ . \qquad (8)$$

The interpolation error can be reduced by making the interpolation interval as symmetrical as possible [6], by having equal number of samples on either side of the interpolation instant. This type of interpolation is also known as the central point interpolation. In order to achieve the symmetry in the interpolation interval, a resampling frequency for the i^{th} selected window Frs^i, is calculated by employing Equation 9.

Where, *AF* is the adjustment factor, which is chosen by respecting the following criteria.

** Frs^i should maintain the symmetry in the interpolation process.*
** Frs^i should be greater than and closest to $F_{Nyq}{}^i$.*

Here, $F_{Nyq}{}^i=2.f_{max}{}^i$ and $f_{max}{}^i$ represents the local bandwidth of the i^{th} selected window [11, 13]. The *1st* criterion adds to the system accuracy by reducing the interpolation error. The *2nd* criterion keeps the system computationally efficient, firstly by avoiding the unnecessary interpolations during the resampling process and secondly by avoiding the processing of unnecessary samples during the filtering operation.

In this article the performance of the proposed filtering technique in terms of computation and quality is studied by employing a modelled input, consists of different frequency sinusoids. For a monotone sinusoid $f_{max}{}^i=f^i$, here f^i is the frequency of the sinusoid lie in the i^{th} selected window. In this case, Fs^i can be calculated by employing Equation 10. Where, *NRC* represents the relevant number of level-crossing thresholds crossed by the sinusoid.

$$Frs^i = Fs^i/AF. \quad (9) \qquad Fs^i = 2.f^i.(NRC)\,. \quad (10)$$

In the studied case $\Delta x(t)= \Delta V_{in}$, so *NRC* becomes equal to 2^M-1. Thus, Fs^i can be expressed mathematically by Equation 11. The over sampling ratio for the i^{th} selected window OSR^i is given by Equation 12.

$$Fs^i = 2.f^i.(2^M-1) \quad (11) \qquad OSR^i = \frac{Fs^i}{F_{Nyq}{}^i = 2.f^i} = 2^M - 1\,. \quad (12)$$

As $\Delta x(t)=\Delta V_{in}$ so *AF* is a function of *M*, in the studied case. By following the above discussed criteria, for $M \geq 2$ the *AF* is chosen equal to $(2^M-1)/2$. The replacement of *AF* and Fs^i in Equation 9 gives $Frs^i=4.f^i$.

D. Adaptive Rate Filtering

It is known that for fixed design parameters (cut-off frequency, transition-band width, pass-band and stop-band ripples) the FIR filter order varies as a function of the operational sampling frequency. For high sampling frequency, the order is high and vice versa. In the classical case, the sampling frequency and filter order both remains unique regardless of the input signal variations, so they have to be chosen for the worst case. This time invariant nature of the classical filtering causes a useless increase of the processing load. This drawback has been resolved up to a certain extent by employing the multirate filtering techniques [15, 16]. They achieve computational efficiency which is not attainable with the classical approach.

The proposed filtering technique is a smart alternative of the multirate filtering techniques. It adapts the sampling frequency and the filter order by following the input signal local variations, which leads to a drastic computational gain of the proposed technique over the classical one.

The idea is to offline design a reference filter for a reference sampling frequency F_{ref}, which satisfies the *AF* criteria, discussed in Section III-C. In the studied case F_{ref} is chosen as: $F_{ref} = 4.f_{max}$. The reference filter impulse response h_k is sampled at F_{ref} during offline processing. Here *k* is the index of the reference filter coefficients.

Frs^i can be equal to or less than F_{ref}, depending upon Fs^i. In order to perform a proper filtering operation Frs^i should match to F_{ref}, the process of keeping them coherent leads towards the following two filtering cases.

Filtering Case 1

This case is true if $Frs^i = F_{ref}$. For this case, the filtering decision for the i^{th} selected window FD^i is set to *0* and it drives the switch to state *0,* in Fig. 2. The reference filter impulse response h_k remains unaltered for this case.

Filtering Case 2

This case is true if $Frs^i < F_{ref}$. For this case, FD^i is set to *1* and it drives the switch to state *1,* in Fig. 2.

In this case, it appears that the data lies in the i^{th} selected window can be resampled at a frequency which is less than the Nyquist frequency of *x(t)* and so it can cause aliasing. Since the AADC sampling frequency varies according to the slope of *x(t)* [2]. 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 sub-Nyquist frequency of *x(t)*. But still this signal part is locally over sampled in time with respect to its local bandwidth [11, 13]. It is valid as far as $\Delta x(t) = \Delta V_{in}$, because it makes the relevant signal part to cross all thresholds of the AADC, therefore it is locally oversampled in time. This statement is further illustrated with the results summarized in Table II. Hence, there is no danger of aliasing, when the low frequency relevant signal parts are locally over-sampled in time at overall sub-Nyquist frequencies.

As in order to perform a proper filtering operation Frs^i should match to F_{ref}, so in filtering case 2, h_k is online decimated as a function of Frs^i for the i^{th} selected window. The decimation factor for the i^{th} selected window D^i can be specific depending upon Frs^i. The process of calculating D^i and decimating h_k for the integral and the fractional D^i is shown in Fig. 3.

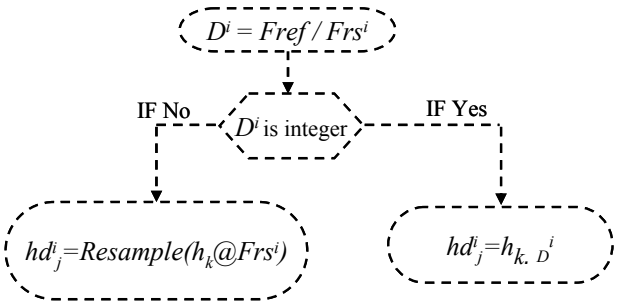

Fig. 3 Flowchart of calculating D^i and decimating h_k

In Fig. 3 hd^i_j represents the decimated filter for the i^{th} selected window. Here *j* represents the index of the decimated filter coeficients. For integral D^i, hd^i_j is obtained by picking every $(D^i)^{th}$ coefficient from h_k. For fractional D^i, the fractional decimation is achieved by resampling h_k at Frs^i. The NNRI is employed to resample h_k. If the order of h_k is *H* then the order of hd^i_j is $P^i=H/D^i$.

A simple decimation causes a reduction of the decimated filter energy compared to the reference one. It will lead to an attenuated version of the filtered signal. D^i is a good approximate of the ratio between the energy of the reference filter and that of the decimated filter. Thus, this effect of decimation is compensated by scaling hd^i_j with D^i. The proc-

ess of scaling hd^i_j in order to obtain the decimated and the scaled impulse response hw^i_j for the i^{th} selected window is given by Equation 13.

$$hw^i_j = hd^i_j \times D^i . \quad (13)$$

IV. Illustrative Example

In order to illustrate the proposed filtering technique a modelled signal $x(t)$, shown on the left part of Fig. 4 is employed. Its total duration is 20 seconds and it consists of three active parts. Each activity is a sinusoid of amplitude 0.9v and of frequency 1100, 200 and 350 Hz respectively. The time span of each activity is 0.5, 1.0 and 1.0 second respectively.

As $x(t)$ is band limited up to 1.1 kHz so the value of F_{ref} becomes 4.4 kHz in this case. $x(t)$ is sampled by employing a 3-bit resolution AADC, it results into Fs_{max}=15.4 kHz and Fs_{min}= 2.8 kHz respecttively (c.f. Equations 3 and 4). AF becomes 3.5 in this case. ΔV_{in} is set to 1.8v, thus AADC quantum $q=\Delta V_{in}/(2^M-1)$ becomes 0.2571v.

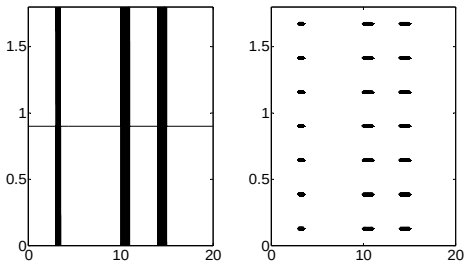

Fig. 4 Input signal (left) and the selected signal (right)

The 1^{st} and the 3^{rd} active parts of $x(t)$ are of higher frequencies compared to the 2^{nd} one. In order to filter out the high frequency components from $x(t)$ a low pass reference filter is designed by using standard Parks-McClellan algorithm. The filter parameters are summarized in Table I. The purpose of this filtering process is to demonstrate that how the proposed filtering technique adapts its processing load by following the local variations of $x(t)$.

TABLE I
Summary of Reference Filter Parameters

Cut-off Freq (Hz)	Transition Band (Hz)	Pass Band Ripples	Stop Band Ripples	F_{ref} (kHz)	R
250	250~310	-25 (dB)	-80 (dB)	4.4	186

In order to apply the ASA, the reference window length T_{ref} = 1second is chosen for this studied example. The selected signal obtained with the ASA is shown on the right part of Fig. 4. The parameters of each selected window are summarized in Tables II and III.

TABLE II
Summary of Parameters of the Selected Windows

Selected Window	Ts^i (Sec)	Fs^i (kHz)	N^i (smp)	F_{ref} (kHz)
1^{st}	0.4995	15.4	7693	4.4
2^{nd}	0.9994	2.8	2799	4.4
3^{rd}	0.9995	4.9	4898	4.4

TABLE III
Summary of Parameters of the Selected Windows

Selected Window	Frs^i (kHz)	Nr^i (smp)	OSR^i	D^i	P^i
1^{st}	4.4	2198	7	1	186
2^{nd}	0.8	800	7	5.5	33
3^{rd}	1.4	1400	7	3.1	60

Table II and III exhibit the interesting features of the proposed filtering technique, which are achieved due to the smart combination of the non-uniform and the uniform signal processing tools. Fs^i represents the sampling frequency adaptation by following the local variations of $x(t)$. N^i and OSR^i show that the relevant signal parts are locally over-sampled in time with respect to their local bandwidths [11, 13]. Frs^i shows the adaptation of the resampling frequency for each selected window. It further adds to the computational gain of the proposed technique by avoiding the unnecessary interpolations during the resampling process. Nr^i shows that how the adjustment of Frs^i avoids the processing of unnecessary samples during the filtering process. Ts^i exhibits the dynamic feature of ASA, which is to correlate the reference window length [8] with the signal activity lying in it. On the contrary, in the classical case, the reference window length remains static and is not able to adapt according to the signal activity lying in it. Moreover, the windowing process does not select only the active parts of the sampled signal. For this studied example T_{ref} = *1* Sec. would lead to twenty 1-second windows for the whole signal span (20 seconds), in the classical case. It follows that the system has to process more than the relevant information part in $x(t)$.

From Table III, P^i represents the adaptation of h_k for the i^{th} selected window. It is another advantage of the proposed technique over the classical one. In the classical case, the filter remains time invariant so it has to be designed for the worst case. In this example the input signal is band limited to 1.1 kHz. Therefore, if the Fs = 3 kHz is chosen in order to respect the Shannon sampling theorem. Then for the same filter parameters, summarized in Table I, Parks-McClellan design algorithm provides a 127^{th} order filter. As in the classical case, signal regardless of its activity is sampled at a fixed sampling frequency Fs = 3 kHz, so a fixed order filter H = *127* has to be employed for the whole signal length, leads towards the extra system activity compared to the proposed case.

V. Computational Complexity

This section compares the computational complexity of the proposed filtering technique with the classical one. The complexity evaluation is made by considering the number of online operations executed to perform the algorithm.

In the classical case, Fs and H remains constant, regardless of the input signal local variations. It is known that for an R order FIR filter, H multiplications and H additions are computed for each output sample. The total computational complexity C_1 for N samples can be calculated by employing Equation 14.

$$C_1 = \underbrace{H.N}_{Additions} + \underbrace{H.N}_{Multiplica\ tions} . \quad (14)$$

In the proposed filtering technique, the sampling frequency and the filter order both are not fixed and are adapted for each selected window according to the local variations of $x(t)$. In comparison to the classical case this approach locally requires some extra operations for each selected window.

Calculation of Frs^i requires a division between Fs^i and AF (cf. Equation 12). The selected data lie in the i^{th} selected window is resampled uniformly at Frs^i before filtering. The NNRI is employed for the resampling purpose. The NNRI requires only a comparison operation for each resampled observation. Therefore the interpolator performs Nr^i comparisons. Moreover, the filtering case selection requires a com-

parison between F_{ref} and Frs^i. If $FD^i=0$ then the reference filter impulse response h_k remains the same. Otherwise, the decimation of h_k is required. D^i is calculated for this purpose. The calculation of D^i requires one division (cf. Fig. 3). For an integral D^i the decimation of h_k is simple and has a negligible complexity as compare to operations like addition or multiplication. This is the reason that for this case the complexity of the decimator is not taken into consideration, during the complexity evaluation. For a fractional D^i, it is required to resample h_k at Frs^i, which which perform P^i comparisons. Here, P^i is the order of decimated filter for the i^{th} selected window. The filter coefficients scalar performs P^i multiplications (cf. Equation 16). Finally a P^i order filter, performs $P^i.Nr^i$ multiplications and $P^i.Nr^i$ additions for the i^{th} selected window. The processes of designing the reference filter is performed offline, so is not included in the online algorithm complexity calculation. The combined computational complexity C_2 of the proposed filtering technique is given by Equation 15.

$$C_2 = \underbrace{L(1+\alpha)}_{Division} + \sum_{i=1}^{L} \underbrace{\left(Nr^i + \alpha\beta P^i + 1\right)}_{Comparisons} + \underbrace{P^i Nr^i}_{Additions} + \underbrace{\left(P^i Nr^i + \alpha P^i\right)}_{Multiplications} . \quad (15)$$

In Equation 18, $i = 1, 2, ..., L$ represents the index of selected window. α is a multiplying factor and its value is 0 for $Frs^i = F_{ref}$ and 1 otherwise. β is also a multiplying factor, it is 0 for an integral D^i and 1 for a fractional D^i.

From C_1 and C_2 it is clear that there are uncommon operations between both filtering techniques. In order to make them approximately comparable the following assumptions are made.

**A comparison has same processing cost as that of an addition.*
**A division has same processing cost as that of a multiplication.*

By following these assumptions, comparisons are merged into additions count and divisions are merged into multiplications count, during the complexity evaluation process. The computational gain of the proposed filtering technique over the classical one is calculated for results of the illustrative example, for different time spans of *x(t)*. The results are summarized in Table IV.

TABLE IV
SUMMARY OF THE COMPUTATIONAL GAIN

Time Span (Sec)	Gain in Additions	Gain in Multiplications
Ts^1	0.93	0.93
Ts^2	13.99	13.41
Ts^3	4.45	4.53
Total signal span (20)	14.55	14.61

Table IV shows 14.55 and 14.61 times gain in additions and multiplications respectively for the total *x(t)* span of 20 seconds. It shows that the proposed filtering technique leads towards a drastic reduction in the number of operations compared to the classical one. It is achieved by adapting the sampling frequency and the filter order by following the input signal local variations.

It is clear that by knowing f_i the choice of $Frs^i \geq 2.f_i$ is enough, but in order to achieve symmetric interpolation $Frs^i = 4.f_i$ is chosen, in the studied case. Although this choice of Frs^i affects the proposed technique computational efficiency, but it can be accepted to achieve a significant processing accuracy (c.f. Section VI).

VI. PROCESSING ERROR

A. Interpolation Error

The interpolated samples are calculated by employing the level-crossing samples. For the practical AADC there exist uncertainties in time-amplitude pairs of the level-crossing samples [2]. These uncertainties accumulate in the interpolation process and cause to increase the error [4]. Therefore in order to have a fair idea of the interpolation error an ideal AADC is employed, which provides the exact time-amplitude pairs of the level-crossing samples.

For the studied example, $AF=3.5$ leads to the interpolation symmetry (cf. Section III-C). In order to observe the effect of symmetry on the interpolation error, values of MIe^2 are calculated by varying AF around 3.5. The process of calculating MIe^2 is clear from Equation 8. The values of AF are chosen in such a way that Frs^2 remains greater than $F_{Nyq}{}^2$ for all AFs. The results are plotted in Fig. 5.

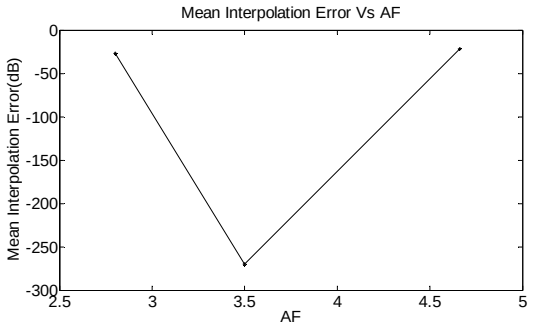

Fig. 5 Mean interpolation error (dB) versus AF

Fig. 5 shows that a drastic reduction in the interpolation error can be achieved by availing the symmetry during the interpolation process. Interpolation symmetry may leads to some latencies in terms of the real time implementation and the computational efficiency. However depending on the application requirements, these latencies can be tolerated for the benefit of maintaining a small error.

B. Filtering Error

In the proposed filtering technique a reference filter h_k is employed and then it is adapted for the i^{th} selected window, depending upon the chosen Frs^i. It makes the sampling and filtering of the i^{th} activity at Frs^i. In order to calculate the relative filtering error, of the proposed technique with the classical one, the following procedure is adopted.

In the classical case, instead of adapting h_k to obtain h^i_j, a specific filter is designed for the i^{th} selected window, by employing the Parks-McClellan algorithm. It is designed for Frs^i by employing the same design parameters, summarized in Table I. The i^{th} activity is sampled at Frs^i and is filtered by employing the designed filter. The filtered signal obtained in this case is used as a reference and is compared with the one, obtained with the proposed technique.

Let y_n be the n^{th} reference filtered sample and $\hat{y_n}$ the n^{th} filtered sample obtained with the proposed filtering technique. Then the relative filtering error for the n^{th} filtered point can be calculated by using Equation 16. The mean filtering error for the i^{th} activity MFe^i can be calculated by employing Equation 17.

$$Fe_n = \left| y_n - \tilde{y}_n \right| . \quad (16) \qquad MFe^i = \frac{1}{Nr^i} \cdot \sum_{n=1}^{Nr^i} Fe_n \quad . \quad (17)$$

By following the above method MFe^2 is calculated and it is 5.6 %. It shows that the output obtained with the proposed filtering technique is of comparable quality to the classical one. It is obvious that the online decimation of h_k in the proposed filtering technique, leads to a loss of the filtering quality. The measure of this quality loss can be used to decide the upper bound on D^i. By performing offline calculations the maximum value of D^i can be decided for which the decimated and scaled filter provides results with an acceptable level of accuracy. The level of accuracy is application dependent.

VII. Conclusion

An adaptive rate filtering technique is devised. This technique is well suited for the low activity sporadic signals like electro-cardiogram, phonocardiogram, seismic signals, etc. A reference filter is offline designed by taking into account the *AF* criteria. A complete methodology to obtain Frs^i and decimating h_k for the i^{th} selected window has been demonstrated. It is shown that how the data resampling rate and the reference filter order are smartly adapted for the i^{th} selected window, by following the input signal local variations. The computational gain of the proposed adaptive rate filtering technique is shown over the classical one. It is achieved due to the joint benefits of the AADC, the ASA and the resampling as they enable to adapt the Fs^i, Frs^i, N^i, Nr^i, D^i and P^i by exploiting the input signal local variations.

The interpolation error is calculated. It is shown that a drastic reduction in the interpolation error can be achieved by employing the symmetry in the interpolation process. A method to calculate the relative filtering error of the proposed technique with respect to the classical one is also proposed. It is shown that the result obtained with the proposed filtering technique is of comparable quality to the classical one. The reference filter decimation is required in the proposed technique. The complete procedure of decimating and scaling the pre-calculated reference filter during online computation is demonstrated. This online decimation reduces the quality of the decimated filter as compare to the reference one. The upper bound on the decimation factor can be determined by offline calculations, for which the decimated and scaled filter provides a response with an acceptable level of accuracy. Moreover, for high precision applications, an appropriate filter can be directly calculated online for each selected window at the cost of an increased computational load.

A detailed study of the computational complexity of the proposed filtering technique by taking into account the real processing cost at circuit level is in progress. Further research focuses on the optimization of the proposed techniques and on its performance study in the case of real life applications.

References

[1] J.W. Mark and T.D. Todd, "A nonuniform sampling ap-proach to data compression", IEEE Transactions on Communica-tions, vol. COM-29, pp. 24-32, January 1981.

[2] E. Allier, G. Sicard, L. Fesquet and M. Renaudin, "A new class of asyn-chronous A/D converters based on time quantization", ASYNC'03, pp.197-205, May 2003.

[3] F. Aeschlimann, E. Allier, L. Fesquet and M. Renaudin, "Asynchronus FIR filters, towards a new digital processing chain", ASYNC'04, pp. 198-206, April 2004.

[4] 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, pp. 335-339, April 1996.

[5] S.C. Sekhar and T.V. Sreenivas, "Adaptive window zero-crossing based instantaneous frequency estimation", EURASIP Journal on Applied Signal Processing, pp.1791-1806, Volume 2004, Issue 1, January 2004.

[6] D.M. Klamer and E.Marsy, "Polynomial interpolation of randomly sam-pled band-limited functions and processes", SIAM Review, vol.42, No.5, pp. 1004-1019, October 1982.

[7] F.J. Beutler, "Error free recovery from irregularly spaced samples", SIAM Review, vol. 8, pp. 328-335, 1996.

[8] S.M. Qaisar, L. Fesquet and M. Renaudin "Spectral Analysis of a signal Driven Sampling Scheme", EUSIPCO'06, September 2006.

[9] S. de Waele and P.M.T.Broersen, "Time domain error measures for resampled irregular data", IEEE Transactions on Instrumentation and Meas-urements, pp.751-756, May 1999.

[10] S. de Waele and P.M.T. Broersen, "Error measures for resampled irregu-lar data", IEEE Transactions on Instrumentation and Measurements, pp.216-222, April 2000.

[11] S.M. Qaisar, L. Fesquet and M. Renaudin, "Computationally efficient adaptive rate sampling and filtering", EUSIPCO'07, pp.2139-2143, Septem-ber 2007.

[12] M. Gretains, "Time-frequency representation based chirp like signal analysis using multiple level crossings", EUSIPCO'07, pp.2154-2158, Sep-tember 2007.

[13] S.M. Qaisar, L. Fesquet and M. Renaudin, "Adaptive rate filtering for a signal driven sampling scheme", ICASSP'07, pp.1465-1468, April 2007.

[14] F. Marvasti, "Nonuniform sampling theory and practice", Kluwer academic/Plenum Publisher, New York, 2001.

[15] Martin Vetterli, "A theory of multirate filter banks", IEEE transaction on Acoustic, Speeh and Signal Processing, vol. 35, pp.356-372, March 1987.

[16] S. Chu and C.S. Burrus, "Multirate filter designs using comb filters", IEEE transaction on Circuits and Systems, vol. 31, PP. 913-924, November 1984.

[17] K. M. Guan et al. "Oppertunistic Sampling by Level-Crossing", ICASSP'07, pp.1513-1516, April 2007.

[18] F. Akopyan et al. "A level-crossing flash analog-to-digital converter", ASYNC'06, pp.12-22, March 2006.

[19] S.M. Qaisar, L. Fesquet and M. Renaudin, "An Adaptive resolution computationally efficient short time Fourier transform", EURASIP, RLSP, April 2008.