

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.

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

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).

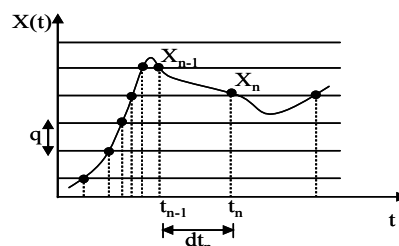


Fig. 1 Level-crossing sampling scheme

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

Where, AF is the adjustment factor, which is chosen by V_{in} and F_{ref} , the process of keeping them coherent leads towards the following two filtering cases.

* Frs^i should maintain the symmetry in the interpolation process.

* Frs^i should be greater than and closest to F_{Nyq}^i .

Here, $F_{Nyq}^i = 2 \cdot f_{max}^i$ and f_{max}^i represents the local bandwidth of the i^{th} selected window [11, 13]. The 1^{st} criterion adds to the system accuracy by reducing the interpolation error. The 2^{nd} 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.

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

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.

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

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 $Frs^i = 4 \cdot f^i$.

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: $F_{ref} = 4 \cdot f_{max}$. 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

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.

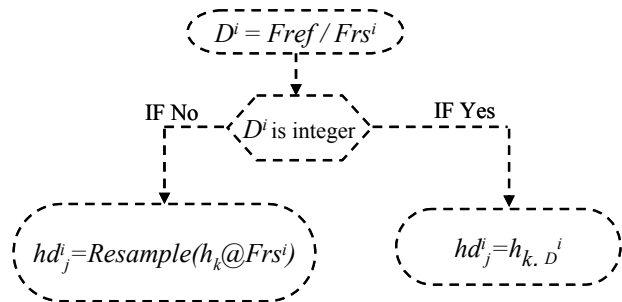


Fig. 3 Flowchart of calculating D^i and decimating h_k

In Fig. 3 hd_j^i represents the decimated filter for the i^{th} selected window. Here j represents the index of the decimated filter coefficients. For integral D^i , hd_j^i 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_j^i 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_j^i with D^i . The proc-

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

$$hw_j^i = hd_j^i \times D^i. \quad (13)$$

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 $F_{S_{max}}=15.4$ kHz and $F_{S_{min}}=2.8$ kHz respectively (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.

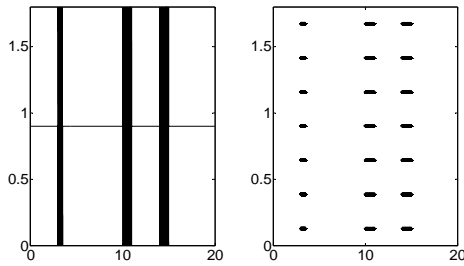


Fig. 4 Input signal (left) and the selected signal (right)

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 $T_{ref} = 1$ second 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	T_s^i (Sec)	F_s^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	F_{rs}^i (kHz)	N_{r^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

Tables 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. F_s^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]. F_{rs}^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. N_{r^i} shows that how the adjustment of F_{rs}^i avoids the processing of unnecessary samples during the filtering process. T_s^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 $F_s = 3$ 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 $F_s = 3$ 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, F_s 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.

$$C_1 = \underbrace{H \cdot N}_{\text{Additions}} + \underbrace{H \cdot N}_{\text{Multiplications}} \quad (14)$$

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 F_{rs}^i requires a division between F_s^i and AF (cf. Equation 12). The selected data lie in the i^{th} selected window is resampled uniformly at F_{rs}^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 N_{r^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.

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

In Equation 18, $i = 1, 2, \dots, L$ 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 the choice of $Frs^i \geq 2$. f is enough, but in order to achieve symmetric interpolation $Frs^i = 4$. f is chosen, in the studied case. Although this choice of Frs^i affects the proposed technique computational efficiency,

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.

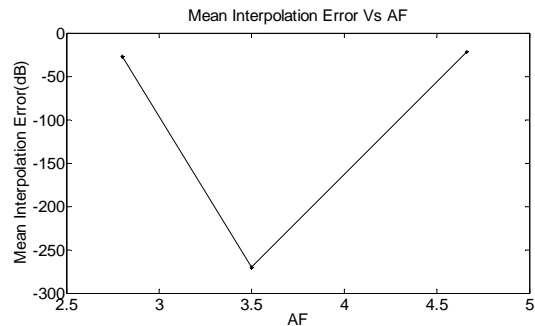


Fig. 5 Mean interpolation error (dB) versus AF

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_j^i , 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.

