A fully non-uniform approach to FIR filtering

Brigitte Bidégaray-Fesquet [(1)] and Laurent Fesquet [(2)]

(1) LJK, CNRS / Grenoble University, B.P. 53, 38042 Grenoble Cedex 9, France.
(2) TIMA, 46 avenue Félix Viallet, 38031 Grenoble Cedex, France.
Brigitte.Bidegaray@imag.fr, Laurent.Fesquet@imag.fr

Abstract:

We propose a FIR filtering technique which takes advantage of the possibility of using a very low number of samples for both the signal and the filter transfer function thanks to non-uniform sampling. This approach leads to a summation formula which plays the role of the discrete convolution for usual FIR filters. Here the formula is much more complicated but it can be implemented and the evaluation of more elaborate expressions is compensated by the very low number of samples to process.

1. Introduction

Reducing the power consumption of mobile systems – such as cell phones, sensor networks and many others electronic devices – by one to two orders of magnitude is extremely challenging but will be very useful to increase the system autonomy and reduce the equipment size and weight. In order to reach such a goal, this paper proposes a solution applicable to FIR filtering which completely rethinks the signal processing theory and the associated system architectures.

Today the signal processing systems uniformly sample analog signals (at Nyquist rate) without taking advantage of their intrinsic properties. For instance, temperature, pressure, electro-cardiograms, speech signals significantly vary only during short moments. Thus the digitizing system part is highly constrained due to the Shannon theory, which fixes the sampling frequency at least twice the input signal frequency bandwidth. It has been proved in [4] and [6] that Analog-to-digital Converters (ADCs) using a non equi-repartition in time of samples leads to interesting power savings compared to Nyquist ADCs. A new class of ADCs called A-ADCs (for Asynchronous ADCs) based on level-crossing sampling (which produces non-uniform samples in time) [2, 3] and related signal processing techniques [1, 5] have been developed.

This work suggests an important change in the FIR filter design. As sampling analog signals is usually performed uniformly in time, sampling the filter transfer function is also done in a regular way with a constant frequency step. Non-uniform sampling leads to an important reduction of the weight-function coefficients. Combined with a non-uniform level-crossing sampling technique performed by an A-ADC, this approach drastically reduces the computation load by minimizing the number of samples and operations, even if they are more complex.

2. Principle and notations

For a large class of signal, non-uniform sampling leads to a reduced number of samples, compared to a Nyquist sampling. This feature has already been used in [1] to design non-uniform filtering techniques based on interpolation. In this work the authors however used a classical (uniform) filter, that is a usual discretization in time of the impulse response.

Here we want to go further and take advantage of the fact that the filter transfer function (the Fourier transform of the impulse response) is a very smooth function with respect to frequency. It can therefore be well approximated by the linear interpolation of quite few samples.

2.1 Level crossing sampling

The initial signals are supposed to be analog ones. The signal which we want to filter is given in the time domain and is denoted by $s(t)$. The filter transfer function is given in the frequency domain and is denoted by $H(\omega)$. The result of the filtering process $x(t)$ is then theoretically the convolution of $s(t)$ with the impulse response $h(t)$ which is the inverse Fourier transform of $H(\omega)$:

$$\begin{aligned} x(t) &= \int_{-\infty}^{+\infty} h(t-\tau)s(\tau)d\tau, \\ h(t) &= \frac{1}{2\pi}\int_{-\infty}^{+\infty} H(\omega)e^{-i\omega t}d\omega. \end{aligned}$$

These signal are sampled in their initial domain using a level crossing scheme. This technique has to be adapted for the filter transfer function. Indeed level crossing has a sense if an order can be defined, for example for a real valued function. The filter transfer function is complex valued, therefore we can choose to sample either when the amplitude crosses some predefined values, or the phase, or both. The samples read $(s_n, \delta t_n)$ for the signal and $(H_k, \delta\omega_k)$ for the filter transfer function. These samples are formed of a value and the (time or frequency) interval length "elapsed" since the last sample. To give results or describe algorithms we will use the sample times or frequencies defined as $t_n = t_0 + \sum_1^n \delta t_{n'}$ and $\omega_k = \omega_0 + \sum_1^k \delta\omega_{k'}$ but computations will be performed using only the time and frequency intervals δt_n and $\delta\omega_k$. We will also denote by $I_n = [t_{n-1}, t_n]$ and $J_k = [\omega_{k-1}, \omega_k]$ the time and frequency intervals.

2.2 Linear interpolation

To derive the FIR algorithm and approximate the theoretical integral formula, we form new analog functions from the previously described samples. To this aim we choose linear interpolation and we have

$$\begin{aligned}\bar{s}(t) &= \sum_n [a_n + b_n t]\chi_{I_n}, \\ \bar{H}(\omega) &= \sum_k (\alpha_k + \beta_k\omega)e^{i(\gamma_k+\delta_k\omega)}\chi_{J_k},\end{aligned}$$

where χ denotes the indicator function of the set given in index. The coefficients a_n and b_n can be expressed in terms of s_n, s_{n-1}, t_n and δt_n. The coefficients α_k, β_k, γ_k and δ_k can be expressed in terms of H_k, H_{k-1}, ω_k and $\delta\omega_k$.

In fact these formulae cover the piecewise constant case (only take $b_n = \beta_k = \delta_k = 0$) in three possible forms: constant on intervals I_n or nearest neighbor interpolation, with a possible need to modify the definition of t_n and δt_n in the algorithms. They also cover two ways to linearly interpolate the complex valued filter transfer function: either interpolate separately the amplitude and the phase (α_k and β_k are real) or interpolate in the complex plane (α_k and β_k are complex, γ_k and δ_k are zero).

The digital filter then consists in computing (possibly) for all time

$$\begin{aligned}\bar{x}(t) &= \int_{-\infty}^{+\infty} \bar{h}(t-\tau)\bar{s}(\tau)d\tau, \\ \bar{h}(t) &= \frac{1}{2\pi}\int_{-\infty}^{+\infty} \bar{H}(\omega)e^{-i\omega t}d\omega.\end{aligned}$$

3. Deriving a filtering formula in the general context

3.1 A summation formula

The impulse response $\bar{h}(t)$ can be split in contributions for each frequency sample $\bar{h}(t) = \sum_k h_k(t)$ with

$$h_k(t) = \frac{1}{2\pi}\int_{\omega_{k-1}}^{\omega_k} (\alpha_k + \beta_k\omega)e^{i(\gamma_k+\delta_k\omega)}e^{-i\omega t}d\omega$$

for which we will give an explicit expression in Section 3.2. Although the piecewise linear function $\bar{H}(\omega)$ has a compact support (we only have a finite number of samples), the functions $h_k(t)$ have an infinite support. This is not a problem since the convolution will involve $\bar{s}(t)$ which has a compact support. The convolution reads

$$\begin{aligned}\bar{x}(t) &= \int_{-\infty}^{+\infty} \bar{h}(t-\tau)\bar{s}(\tau)d\tau \\ &= \sum_n \int_{t_{n-1}}^{t_n} h(t-\tau)s_n(\tau)d\tau \\ &= \sum_n\sum_k \int_{t_{n-1}}^{t_n} h_k(t-\tau)(a_n + b_n\tau)d\tau \\ &= \sum_n \left(a_n \sum_k h^0_{nk}(t) + b_n \sum_k h^1_{nk}(t)\right)\end{aligned}$$

where

$$h^0_{nk}(t) = \int_{t_{n-1}}^{t_n} h_k(t-\tau)d\tau,$$

$$h^1_{nk}(t) = \int_{t_{n-1}}^{t_n} h_k(t-\tau)\tau d\tau.$$

We obtain a summation formula as in the classical FIR filtering case where it takes the form of a discrete convolution. To be closer to this classical case, we should write this as

$$\bar{x}(t) = \sum_n s_n \sum_k h_{nk}(t),$$

which is possible but the effective expression depends on the type of interpolation used (piecewise constant or linear).

There remains to make explicit these two types of elementary contributions.

3.2 Elementary impulse responses

A straightforward computation of the integral formulation for $h_k(t)$ yields

$$\begin{aligned}h_k(t) &= \frac{\alpha_k e^{i\gamma_k}}{2\pi}\int_{\omega_{k-1}}^{\omega_k} e^{i(\delta_k - t)\omega}d\omega \\ &+ \frac{\beta_k e^{i\gamma_k}}{2\pi}\int_{\omega_{k-1}}^{\omega_k} e^{i(\delta_k - t)\omega}\omega d\omega \\ &= \frac{\alpha_k e^{i\gamma_k}\left(e^{i(\delta_k-t)\omega_k} - e^{i(\delta_k-t)\omega_{k-1}}\right)}{2\pi i(\delta_k - t)} \\ &+ \frac{\beta_k e^{i\gamma_k}\left(\omega_k e^{i(\delta_k-t)\omega_k} - \omega_{k-1}e^{i(\delta_k-t)\omega_{k-1}}\right)}{2\pi i(\delta_k - t)} \\ &+ \frac{\beta_k e^{i\gamma_k}\left(e^{i(\delta_k-t)\omega_k} - e^{i(\delta_k-t)\omega_{k-1}}\right)}{2\pi(\delta_k - t)^2}.\end{aligned}$$

These formulae seem singular when $t = \delta_k$. This is not the case and has no reason to be since the function we integrate is smooth with respect to all parameters and variables. The limiting value for $t = \delta_k$ is clearly

$$\begin{aligned}h_k(\delta_k) &= \frac{\alpha_k e^{i\gamma_k}}{2\pi}\int_{\omega_{k-1}}^{\omega_k} d\omega + \frac{\beta_k e^{i\gamma_k}}{2\pi}\int_{\omega_{k-1}}^{\omega_k}\omega d\omega \\ &= \frac{e^{i\gamma_k}}{2\pi}\delta\omega_k(\alpha_k + \beta_k\frac{1}{2}(\omega_{k-1} + \omega_k)).\end{aligned}$$

3.3 Elementary summation coefficients

A quick glance at the explicit expression of $h_k(t)$ clearly provides the impression that the explicit formulae for $h^0_{nk}(t)$ and $h^0_{nk}(t)$ will not fit in the columns here. We will give only their flavor. Indeed we want to compute the time integrals of of $h_k(t-\tau)$ and $h_k(t-\tau)\tau$ for $\tau \in I_n$. This leads to integrate the product of a rational function with a complex exponential function. The results cannot be given in terms of simple functions but only in terms of the exponential integral function

$$\mathrm{Ei}(ix) = -\int_x^\infty e^{iy}\frac{dy}{y} + i\frac{\pi}{2}.$$

We give in the next section a simple example of elementary summation coefficient calculation in the piecewise linear context.

4. A simple and ideal example

4.1 Computation of the coefficients

Our sampling for the filter transfer function yields a particularly simple formulation for the ideal low-pass filter which is 1 on the frequency interval $[-\omega_c, \omega_c]$ and zero elsewhere. This yields a single sample $(1, 2\omega_c)$ and linearly interpolated coefficients $\alpha_1 = 1$, $\beta_1 = 0$, $\gamma_1 = 0$ and $\delta_1 = 0$. The expression for the elementary impulse response is

$$h_1(t) = \frac{\left(e^{-i\omega_c t} - e^{i\omega_c t}\right)}{-2\pi i t} = \frac{\omega_c}{\pi}\,\mathrm{sinc}(\omega_c t).$$

Then we have to compute

$$\begin{aligned} h^0_{n1}(t) &= \int_{t_{n-1}}^{t_n} h_1(t-\tau)d\tau = -\int_{t-t_{n-1}}^{t-t_n} h_1(\tau)d\tau \\ &= -\frac{1}{\pi}(\mathrm{Si}(\omega_c(t-t_n)) - \mathrm{Si}(\omega_c(t-t_{n-1})), \end{aligned}$$

where Si is the special function known as sine integral and defined by

$$\mathrm{Si}(x) = \int_0^x \sin(y)\frac{dy}{y} = \frac{1}{2i}(\mathrm{Ei}(ix) - \mathrm{Ei}(-ix)) + \frac{\pi}{2},$$

and

$$\begin{aligned} h^1_{n1}(t) &= \int_{t_{n-1}}^{t_n} h_1(t-\tau)\tau d\tau \\ &= -\int_{t-t_{n-1}}^{t-t_n} h_1(\tau)(t-\tau)d\tau \\ &= t\,h^0_{n1}(t) + \frac{1}{\pi}\int_{t-t_n}^{t-t_{n-1}} \sin(\omega_c\tau)d\tau \\ &= t\,h^0_{n1}(t) - \frac{1}{\pi\omega_c}(\cos(\omega_c(t-t_n)) \\ &\qquad - \cos(\omega_c(t-t_{n-1}))). \end{aligned}$$

This case is simple due to its minimal number of samples in the frequency domain, but it displays all the difficulties of the general case, i.e. the need to evaluate special functions. These functions are built in many libraries in view of a numerical implementation of these algorithms. Moreover these functions are however very smooth: the Si function for example is almost linear in the neighborhood of 0 and tends to $\pm\pi/2$ at $\pm\infty$ with very gentle oscillations. This feature makes possible the construction of efficient lookup tables in view of a hardware implementation.

4.2 Numerical results

To illustrate this simple example we filter the signal

$$s(t) = 0.45\sin(2\pi t) + 0.45\sin(10\pi t) + 0.9$$

with the ideal low pass filter with the cutoff frequency $\omega_c = 4\pi$. The theoretical result is therefore supposed to be

$$x(t) = 0.45\sin(2\pi t) + 0.9.$$

This is not the typical sort of signal which is supposed to be addressed by our technique since it is not a sporadic one and a relatively large number of samples are taken. We perform the computations within the MATLAB SPASS (Signal Processing for ASynchronous Systems) framework (`http://ljk.imag.fr/membres/Brigitte.Bidegaray/SPASS/`).

This signal is sampled with a M-bit Asynchronous A/D Converter (AADC) which leads to a level crossing sampling over the amplitude range $[0, 1.8]$.

We can choose as we want the times at which the filtered signal is computed. To display the results we choose the sequence of times $t_m = .17m$ (m integer) to have sampling points dispatched irregularly over the obtained solution.

On Figure 1, you can see the result for a linear interpolation of the signal non-uniform samples and a 3-bit AADC. We plot continuous functions with lines: the initial signal $s(t)$ (dashed line) and the theoretical filtered signal $x(t)$ (solid line). We plot the sampled results with markers: the non-uniformly sampled initial signal s_n (asterisk markers) and the computed filtered samples x_m (circle markers) at times t_m.

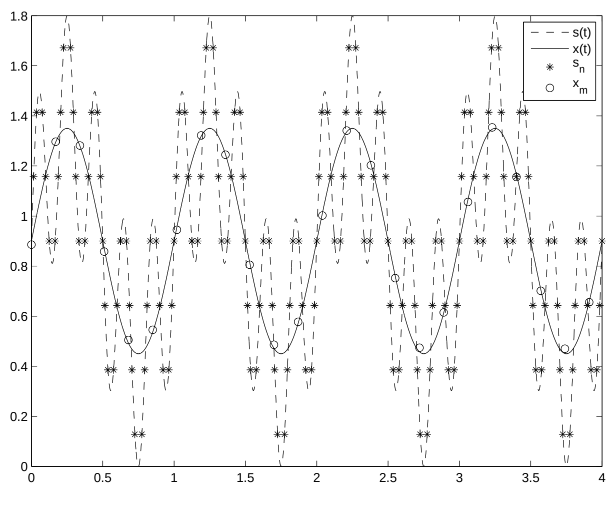

Figure 1: Filtering result. Initial signal (dashed line), theoretical filtered signal (solid line), non-uniformly sampled initial signal (asterisk markers) and computed filtered samples (circle markers).

This very simple test case has quite a low number of parameters compared to the full problem for which we can finely tune the filter transfer function sampling for example. We compare here the results obtained for a zeroth and a first order interpolation of the signal and for different values (2, 3, 4 and 5) of the AADC resolution. On Table 1 we give the relative l^1 error between computed filtered samples x_m at times $t_m = .01m$ (m integer) and the theoretical values $x(t_m)$.

	0th order	1st order
$M = 2$	0.0608	0.0584
$M = 3$	0.0076	0.0046
$M = 4$	0.0052	0.0045
$M = 5$	0.0046	0.0045

Table 1: l^1 error of the filtering method for 0th and first order interpolation of the signal and and M bit resolution of the AADC ($M = 2,\ 3,\ 4,\ 5$).

In the case of the 2-bit AADC, there are 2.8 points per wavelength for the highest frequency part of the signal. This is a very low rate, and we are however able to have only 6% error on the filtered result which is quite sufficient for a large range of applications. The other results all show less than 1% error. The values displayed on Table 1 are very dependent on the choice of the function to filter. Finer results (allowing less than .45% error) should certainly be obtained by using a higher order interpolation for the signal.

5. Conclusions

We have presented a novel approach to FIR filtering based on the non-uniform sampling of the signal but also the non-uniform sampling in frequency of the filter transfer function. The final result is complex but is nonetheless possible to implement in hardware devices and of course in numerical codes. This complexity is balanced by the very low number of samples and the relatively low number of operations needed for each evaluation. This approach is very promising to achieve a lower power consumption in mobile systems.

6. Acknowledgments

This work has been supported by a funding from the Joseph Fourier-Grenoble 1 University: MSTIC project TATIE.

References:

[1] Fabien Aeschlimann, Emmanuel Allier, Laurent Fesquet, and Marc Renaudin. Asynchronus fir filters, towards a new digital processing chain. In *10th IEEE International Symposium on Asynchronous Circuits and Systems (ASYNC'04)*, pages 198–206, Crete, Greece, April 2004.

[2] Filipp Akopyan, Rajit Manohar, and Alyssa B. Apsel. A level-crossing flash asynchronous analog-to-digital converter. In *12th IEEE International Symposium on Asynchronous Circuits and Systems (ASYNC'06)*, pages 11–22, Grenoble, France, March 2006.

[3] Emmanuel Allier, Gilles Sicard, Laurent Fesquet, and Marc Renaudin. A new class of asynchronous A/D converters based on time quantization. In *9th IEEE International Symposium on Asynchronous Circuits and Systems (ASYNC'03)*, pages 197–205, Vancouver, Canada, May 2003.

[4] Jon W. Mark and Terence D. Todd. A nonuniform sampling approach to data compression. *IEEE Trans. on Communications*, COM-29(1):24–32, January 1981.

[5] Saeed Mian Qaisar, Laurent Fesquet, and Marc Renaudin. Adaptive rate filtering for a signal driven sampling scheme. In *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP07)*, pages III–1465–III–1468, Honolulu, Hawaii, USA, April 2007.

[6] N. Sayiner, H.V. Sorensen, and T.R. Viswanathan. A level-crossing sampling scheme for A/D conversion. *IEEE Trans. on Circuits and Systems II*, 43(4):335–339, April 1996.