FILTER DESIGN FOR A RATE-CONVERSION SYSTEM

Pablo Martínez Nuevo

Massachusetts Institute of Technology
Department of Electrical Engineering and Computer Science

ABSTRACT

We present different IIR and FIR filter designs for a rate-conversion system detailing their various characteristics. We carry out a comparison of their efficiency in terms of multiplies per input and output sample considering cascade and folded direct form structures as well as polyphase implementations.

***Index Terms*—** Multirate signal procesing, Filter design, IIR, FIR.

1. INTRODUCTION

Sampling rate conversion is at the heart of discrete-time signal processing: changing the sampling rate of a discrete-time signal to obtain a new discrete-time representation of the underlying continuous-time signal (see Fig. 1). A/D and D/A systems converters use it to exploit oversampling and noise shaping. It is also necessary when the sampling rates of a interconnection of systems are different, or when we want to reduce the amount of computation in an intermediate filtering stage.

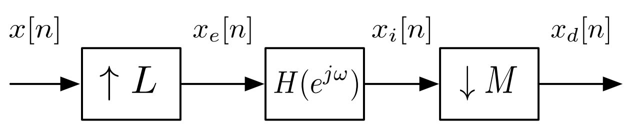

Fig. 1. Block diagram of a sample rate conversion system.

The goal of the project is to change the sampling rate of signal from 20 kHz to 28 kHz by using different filter approximation designs for $H(e^{j\omega})$. In order to do so, we use the sampling rate conversion system shown in Fig. 1. The filter specifications are summarized in Table 1.

2. SAMPLE RATE CONVERSION SPECIFICATIONS

The correct choice for the expander and compressor is $L = 7$ and $M = 5$. In particular, for $f_s = 20$ kHz and $f_s' = 28$ kHz, we have that $f_s' = (L/M)f_s$. The cutoff frequency of the ideal filter in the discrete domain is the minimum of π/L and π/M, which is $\omega_c = \pi/7$ [1]. In this part of the processing, the underlying continuous-time sampling frequency is $f_{e,s} = Lf_s = 140$ kHz; therefore, the equivalent continuous-time cutoff frequency is $f_c = (\pi/7)(f_{e,s}/2\pi) = 10$ kHz.

Table 1. Filter Specifications.

Ideal filter cutoff frequency	Average of band edges
Width of transition band	5 kHz
Maximum passband gain	0 dB
Minimum passand gain	-1 dB
Maximum stopband gain	-50 dB

Now, we can complete the specifications by setting the passband edge to $f_p = 7.5$ kHz and the stopband edge to $f_s = 12.5$ kHz. In the discrete domain after the expander, these frequencies correspond to $\omega_p \approx 0.1071\pi$ and $\omega_s \approx 0.1786\pi$. We use Matlab for filter design and plots generation [2]. In doing so, we decomposed the filter in second order sections to avoid potential numerical problems caused by coefficient quantization. For each design, a table summarizes the filter characteristics and a plot displays its pole-zero locations, group delay, and frequency magnitude response—the band edges as well as the required attenuations in each band are marked by dashed lines according to Table 1.

3. IIR FILTER DESIGN

For IIR filter design, Matlab uses the bilinear transform and considers the specifications of the passband in linear scale from $1 - \delta_{p,\mathrm{IIR}} = 10^{-1/20}$ to 1, and the maximum deviation in the stopband as $\delta_{s,\mathrm{IIR}} = 10^{-50/20}$. Thus, we can directly use these parameters in decibels with its built-in functions for the design procedure.

3.1. Butterworth

Butterworth filters posses two degrees of freedom: the order and the cutoff frequency. Its frequency magnitude response is maximally flat at zero and decreases monotonically with frequency; thus, it does not present any ripples. As it has all the zeros concentrated at $\omega = \pi$, the attenuation approaches infinity as we get closer to π.

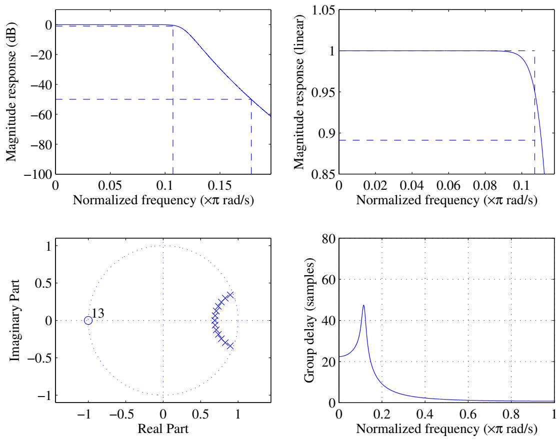

Fig. 2. Butterworth magnitude frequency response, pole-zero plot, and group delay.

Table 2. Butterworth characteristics.

Order	Passband Gain (dB)		Stopband Gain (dB)
	max	min	max
13	0	-0.45	-50

3.2. Chebyshev type I

It allows three degrees of freedom: order, cutoff frequency, and passband ripple. It has equiripple in the passband and decreases monotonically in the stopband. Similar to the Butterworth, it has all its zeros located at $z = -1$.

Table 3. Chebyshev Type I characteristics.

Order	Passband Gain (dB)		Stopband Gain (dB)
	max	min	max
7	0	-1	-56.23

3.3. Chebyshev type II

The number of degrees of freedom is the same as in Chebyshev I. However, it has a ripple in the stopband and is monotonic in the passband. Note that, in general, the order is the same as the type I for the same set of specifications.

3.4. Elliptic

The monotonic behavior of Chebyshev design suggests that ripple in both bands might lower the order of the filter. Elliptic filters have equiripple in both bands and usually present the rational function that requires the lowest order for the same

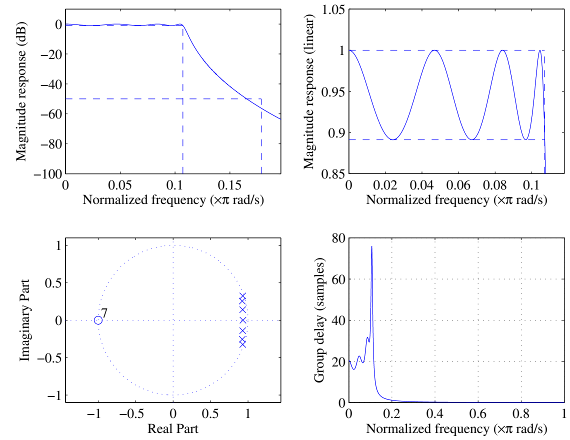

Fig. 3. Chebyshev Type I magnitude frequency response, pole-zero plot, and group delay.

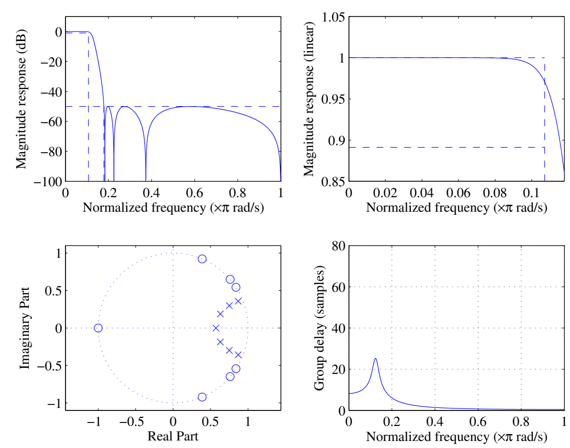

Fig. 4. Chebyshev Type II magnitude frequency response, pole-zero plot, and group delay.

Table 4. Chebyshev Type II characteristics.

Order	Passband Gain (dB)		Stopband Gain (dB)
	max	min	max
7	0	-0.2611	-50

set of specifications. They have four degrees of freedom: order, passband edge, and passband and stopband ripple. They are optimal in terms of minimizing the transition bandwidth which is controlled by the order.

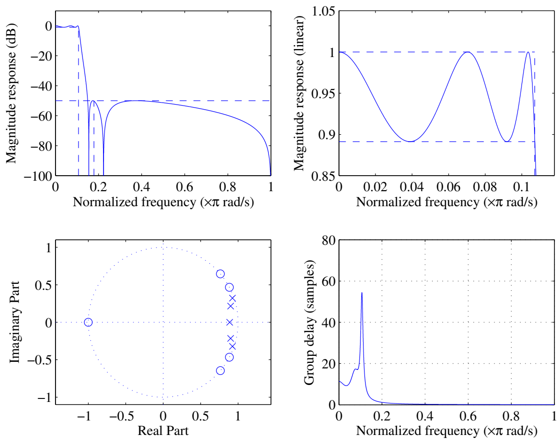

Fig. 5. Ellip characteristics

Table 5. Elliptic specifications.

Order	Passband Gain (dB)		Stopband Gain (dB)
	max	min	max
5	0	-1	-50

4. FIR FILTER DESIGN

As opposed to the IIR case, the passband deviations for the FIR in Matlab are defined from $1 - \delta_{p,\text{FIR}}$ to $1 + \delta_{p,\text{FIR}}$, and $\delta_{s,\text{FIR}}$ for the stopband. We know that $\delta_{p,\text{IIR}} = 1 - 10^{-1/20}$; thus, by a simple proportional rule $\delta_{p,\text{FIR}} = \delta_{p,\text{IIR}}/(2 - \delta_{p,\text{IIR}}) \approx 0.0575$, and $\delta_{s,\text{FIR}} = \delta_{s,\text{IIR}}(1 + \delta_{p,\text{FIR}}) \approx 0.003344$. After designing the filter, we only have to divide the coefficients by $(1 + \delta_{p,\text{FIR}})$ to get the correct specifications as shown in Table 1. Both designs produced type I generalized linear phase systems for the problem at hand.

4.1. Kaiser Window

The windowing technique does not permit us to have separate control over the deviations in the passband and stopband , in fact, they are approximately equal; thus, the order of the filter will be determined by the most restrictive specification, in this case, the stopband. The main lobe of the window frequency response controls the transition bandwidth. Conversely, sidelobes control passband and stopband ripples: the larger the area under the sidelobes, the larger the ripples. We can see in Fig. 6 that the specifications are greatly exceeded in the passband and tightly met in the stopband. The Kaiser family allows controlled tradeoffs between the sidelobe amplitudes and mainlobe widths, and it can easily replicate the rest of the windows by varying one of its parameters. Notice the location of the zeros in conjugate reciprocal locations that ensures constant group delay.

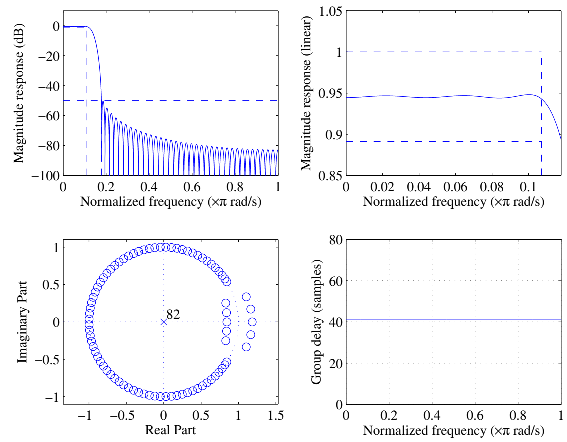

Fig. 6. Kaiser characteristics

Table 6. Kaiser specifications.

Order	Passband Gain (dB)		Stopband Gain (dB)
	max	min	max
82	-0.4623	-0.5110	-50.2016

4.2. Parks-McClellan

These filter approximations have equiripple in the passband as well as in the stopband. They minimize the maximum error in both bands for a given weighting function, i.e. we have separate control of the deviations in both bands. In this case, the Matlab routine underestimated the order of the filter and output 47. We had to increase the order, and the minimum one that met the specifications was 54. As in the previous case, note the particular location of the zeros suffices to produce constant group delay.

Table 7. Parks-McClellan specifications.

Order	Passband Gain (dB)		Stopband Gain (dB)
	max	min	max
54	-0.0252	-0.9717	-50.4403

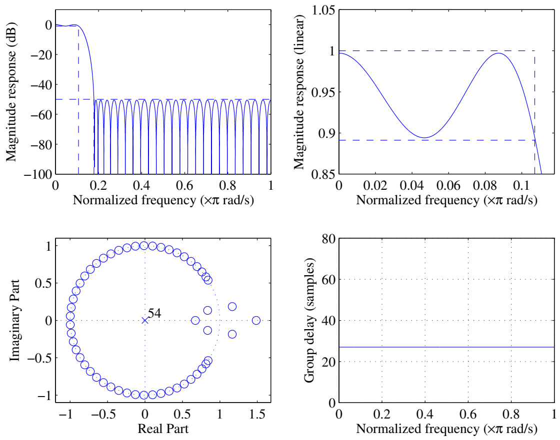

Fig. 7. Parks-McClellan characteristics

5. DISCUSSION

It is clear to observe that the phase in the IIR filters presented is difficult to control, in principle, it is not possible to achieve linear phase for causal IIR filters. Additionally, they can be unstable due to coefficient quantization. We observe that the greatest deviation from constant group delay occurs in all IIR cases at either the edge of the passband, or within the transition band. Chebyshev type II has the widest region of the passband over which the delay is approximately constant and possesses the smallest delay in the passband. In the FIR case, we are able to get constant group delay when we use generalized linear phase filters as in the cases presented. As FIR filters do not have poles, we avoid the risk of instability due to quantization.

5.1. Computational Cost

We measure the computational cost by looking at the number of multiplications per input and output sample. It is important to note that a lower order does not guarantee less multiplies, it strongly depends on the kind of implementation we are using. We have used a cascade of first and second order sections in the IIR case and folded direct form structures in the FIR case. We do not count as multiplies the factors ± 1 or 0. For Butterworth and Chebyshev I, the zeros are located at $\omega = \pi$; therefore, we can implement a cascade of systems of the form $(1 + z^{-1})$ for the zeros—which will not count as multiplies—followed by the poles. In the FIR case, the folded structure allows us to save half of the multiplies that we had used in a regular direct form.

Over the IIR filters, the elliptic approximation gives the lowest order rational system function and Buttwerworth the highest. Note that, in general, both types of Chebyshev filters yield the same order for the same set of specifications. Chebyshev type II zeros are not all located at $z = -1$; then, more multiplications are required in comparison to the Chebyshev type I design considering the cascade form structure. This phenomenon allows Chebyshev I match the number of multiplies of the Elliptic case despite having a higher order. Clearly, Butterworth has the worst performance within the IIR scenario. If we consider a polyphase implementation of the nonrecursive part of these filters, we no longer are able to benefit from the convenient cascade structure of the Butterworth and Chebyshev I designs. This causes that the number of multiplies increases in both cases. However, in the Elliptic and Chebyshev II desing we observe the improvement due to this implementation.

The separate control of the passband and stopband ripples in the Parks-McClellan case produces a lower order filter than the windowing technique for FIR. As a result, Parks-McClellan is more efficient in terms of multiplies than the design using Kaiser windowing considering folded direct form structures. In the FIR case, we cannot take advantage of saving up multiplies when the coefficients are unity due to the gain factor. Clearly, the polyphase implementations of both filters significantly reduces the number of multiplies.

Table 8. Computational cost based on number of multiplies per input and output sample. We considered a folded direct form structure for FIR filters and a cascade structure of first and second-order sections for IIR filters.

Filter Type	Order	Cascade/Folded		Polyphase	
		in	out	in	out
Butterworth	13	98	70	105	75
Chebyshev I	7	56	40	57	40.71
Chebyshev II	7	77	55	57	40.71
Elliptic	5	56	40	41	29.28
Kaiser	82	287	205	83	59.28
Parks-McClellan	54	189	135	55	39.28

If we do not used polyphase, the IIR designs perform better in terms of number of multiplies. However, when polyphase is considered, the number of multiplies is reduced more proportionally in the FIR case. The Elliptic approximation requires the least number of multiplies followed by the Parks-McClellan design with a polyphase implementation—notice that both allow separate ripple control in both bands. Even the Kaiser approximation requires less multiplies than the most efficient implementation of the Butterworth filter.

6. CONCLUSION

From the analysis, we can conclude that the performance in terms of multiplies per input or output sample strongly depends on the implementation. Polyphase greatly impacts the FIR performance in this sense.

7. REFERENCES

[1] Alan V. Oppenheim and Ronald W. Schafer, *Discrete-time signal processing*, Prentice Hall signal processing series. Upper Saddle River, N.J., 2010.

[2] The Mathworks Co., Natick, MA, *Matlab and Simulink*.