Supplement

S1 Impulse response and gain of commonly-used smoothing filters

Here we briefly review, only for reference, a few commonly-used smoothing filters. Providing recommendations for the use of specific filters is beyond the scope of this article. However, inspection of the many transfer functions shown in this supplement can help the reader in the design of a filter optimized for his application.

S1.1 Modified Least squares fitting

Starting from example 1 in **section 2.3** of the main manuscript, we modify the shape of the transfer function by changing the two terms at the end of the summation (**Eq. (24)** of the main manuscript), more specifically, taking half of the value of the end coefficients instead of their full value. We also need to re-normalize the sum of the coefficients to $2N$ instead of $2N$+1, and we obtain the transfer function for the so-called *modified least-squares*:

$$\lambda(\omega) = \frac{1}{2N}\left[\frac{1}{2}e^{-Ni\omega} + e^{-(N-1)i\omega} + ... + e^{-i\omega} + 1 + e^{i\omega} + ... + e^{(N-1)i\omega} + \frac{1}{2}e^{Ni\omega}\right] \tag{S1}$$

Leading to:

$$G(f) = H(f) = \frac{1}{2N}\left[\frac{\sin\left((2N+1)\pi f\right)}{\sin(\pi f)} - \cos(2\pi N f)\right] = \frac{1}{2N}\left[\frac{\sin(2\pi N f)\cos(\pi f)}{\sin(\pi f)}\right] \tag{S2}$$

With the presently modified coefficients, the frequency f_0 of the position of the first zero-gain node is now:

$$f_0 = \frac{1}{2N} \tag{S3}$$

Figure S1 shows the impulse response (left) and the gain (right) for the modified least-squares filters with a full-width comprised between 3 and 25 points. As can be seen on the right-hand side plot, changing the two end coefficients has the effect of producing a slightly less efficient low-pass filter (f_0 is increased) but a more efficient high-stop filter (i.e., smaller amplitude of the Gibbs ripples).

S1.2 Lanczos window

The windowing procedure consists of applying well-chosen additional weights to the original filter coefficients in order to change the shape of the transfer function (in our case, to reduce the amplitude of the Gibbs ripples). For example it can be shown (Hamming, 1989) that a discrete Fourier series truncated at its M^{th} term could be efficiently smoothed (and therefore Gibbs ripples attenuated) if its coefficients were multiplied by the so-called *sigma factors*. For a smoothing filter with even symmetry, an unsmoothed, M-terms truncated discrete Fourier series can be written:

$$F(f) = c_0 + 2\sum_{n=1}^{M} c_n \cos(2\pi n f) \tag{S4}$$

The sigma factors can be written:

$$\sigma(M,n)=\frac{\sin(\pi n/M)}{(\pi n/M)} \qquad 1\le n\le M \tag{S5}$$

The smoothed Fourier series can then be written:

$$F_S(f)=c_0+2\sum_{n=1}^{M}\sigma(M,n)c_n\cos(2\pi nf) \tag{S6}$$

Considering the low-pass filter case with 2N+1 coefficients (example 2 of **section 2.4**), the sigma factors are then:

$$w_n=\frac{\sin(\pi n/N)}{(\pi n/N)} \qquad -N\le n\le N \tag{S7}$$

Note that the sigma factor at the central location (n=0) is $\sigma(N,0)=1$. The new filter coefficients and transfer function can now be written:

$$c_n=2f_C\frac{\sin(2\pi nf_C)}{2\pi nf_C}w_n=2f_C\frac{\sin(2\pi nf_C)}{2\pi nf_C}\frac{\sin(\pi n/N)}{\pi n/N} \qquad -N\le n\le N \tag{S8}$$

$$G(f)=H(f)=2f_C+2\sum_{n=1}^{N}\frac{\sin(2\pi nf_C)}{\pi n}\frac{\sin(\pi n/N)}{(\pi n/N)}\cos(2\pi nf) \tag{S9}$$

Figure S2 shows the impulse response (left) and gain (right) of the low-pass filter introduced in **section 2.4** of the main manuscript, this time with its coefficients weighted by the Lanczos window (full-width comprised between 3 and 25 points). The convolution of the low-pass filter coefficients by the Lanczos window reduces greatly the Gibbs ripples (compare with **Fig. 3** of the main manuscript). Note that the 3-point Lanczos window consists of two null coefficients and one unity coefficient, which is equivalent to no filtering and results into a gain equal to 1 at all frequencies. We kept it on the figure only for the sake of completeness.

S1.3 Von Hann window (or Hanning, or raised cosine window)

Another window commonly used is the von Hann window (also called Hanning window or the raised cosine window). For a window of 2N+1 terms, the window weights in this case are defined by:

$$w_n=\frac{1+\cos(\pi n/N)}{2} \qquad -N\le n\le N \tag{S10}$$

Figure S3 shows the impulse response (left) and gain (right) of the box car average filter after it is convoluted by the von Hann window. Using this window causes the transition region to be much wider, but the Gibbs ripples to have a much smaller amplitude. The frequency f_0 of the first node (zero-gain) is now:

$$f_0=\frac{1}{N} \tag{S11}$$

S1.4 Hamming window

The sign of the lobes of the Von Hann window transfer function is opposite to those of the least-square transfer function (not shown). The Hamming window consists of finding the optimized linear combination of these two transfer functions that will minimize the magnitude of those lobes. The result is a slightly modified version of the von Hann window:

$$w_n = \alpha + \beta \cos(\pi n / N) \qquad \text{with } \alpha + 2\beta = 1 \qquad -N \le n \le N \tag{S12}$$

Contrary to common belief, α and β are not constants. They represent only approximations of the best solution for the minimization of the lobes amplitude, and their value depends on N. For large values of N, we find $\alpha = 0.54$ and $\beta = 0.23$, but for small values ($N < 6$) we find $\alpha > 0.55$ and $\beta < 0.225$ (Hamming, 1989).

S1.5 Blackman window

We can continue to follow the same approach to minimize further the amplitude of the Gibbs ripples by taking optimized linear combinations of the rectangular and cosine window functions, this time using higher harmonics. One common window obtained this way is the Blackman window, defined by its weights:

$$w_n = \alpha + \beta \cos(\pi n / N) + \chi \cos(2\pi n / N) \qquad -N \le n \le N \tag{S13}$$

This time, we have α=0.42, β=0.42, and χ=0.08.

S1.6 Kaiser window and NER filter

An alternate set of window weights was suggested by Kaiser and Reed (1977). These weights have the main function of spreading the large amplitude of the first Gibbs ripples (those near the transition region) into all the ripples between the transition region and the two ends of the frequency range (f=0 and f=0.5). The weights are based on the Bessel function I_0, and can be written:

$$w_n = \frac{I_0\left(\alpha\sqrt{1-(n/N)^2}\right)}{I_0(\alpha)} \qquad -N \le n \le N \tag{S14}$$

with the Bessel function:

$$I_0(\alpha) = 1 + \sum_{m=1}^{\infty} \left(\frac{(\alpha/2)^m}{m!} \right)^2 \tag{S15}$$

The convolution of the Kaiser window weights with the boxcar filter coefficients results in the so-called Near-Equal-Ripple (NER) filter:

$$c_n = 2 f_C \frac{\sin(2\pi n f_C)}{2\pi n f_C} w_n = 2 f_C \frac{\sin(2\pi n f_C)}{2\pi n f_C} \frac{I_0\left(\alpha\sqrt{1-(n/N)^2}\right)}{I_0(\alpha)} \tag{S16}$$

The advantage of this filter is the ability to fine-tune the cut-off frequency, the bandwidth and the amplitude of the Gibbs ripples, all at the same time. Obviously the method does not produce a “perfect” filter, but it allows the optimization of at least two filter parameters at the expense of the third one. For example, we can prescribe the bandwidth of the transition region Δf_C (full-width) with $\Delta f_C < 2f_C$ and $\Delta f_C < 1\text{-}2f_C$, and the amplitude of the Gibbs ripples δ (half-width), and deduce the number of filter coefficients needed. Following the formulation of Kaiser and Reed (1977), the amplitude of the Gibbs ripples can be expressed in terms of attenuation A (in decibel):

$$A = -20\log_{10}(\delta) \qquad \text{(S17)}$$

After we fix the attenuation A and bandwidth Δf_C, an optimal Kaiser filter will be designed by calculating the required number of points N (half-width) using:

$$N = \text{int}\left(\frac{0.13927(A-7.95)}{4\Delta f_C} + 0.75\right) \quad \text{for } A > 21$$

$$N = \text{int}\left(\frac{1.8445}{4\Delta f_C} + 0.75\right) \quad \text{for } A < 21 \qquad \text{(S18)}$$

The α parameter used in argument of the Bessel function is then computed using:

$$\alpha = 0.1102(A-8.7) \quad \text{for } A > 21$$

$$\alpha = 0.5842(A-21)^{0.4} + 0.07886(A-21) \quad \text{for } 21 < A < 50$$

$$\alpha = 0 \quad \text{for } A < 21 \qquad \text{(S19)}$$

An example of optimized low-pass filter using a Kaiser window with 50-dB attenuation is provided in **Fig. S4.** Once again the impulse response is on the left, and the gain is on the right. The right-hand plot shows that the total number of coefficients $2N+1$ must be 7 or larger to produce an optimized filter for this particular value of attenuation.

S1.7 Noise reduction and number of filter coefficients

Figure S5 shows, for four of the filters introduced in this section, the amount of noise reduction as a function of the number of filter coefficients used. The noise reduction values are computed using **Eq. (2)** in the main manuscript. The black dotted curve on each plot shows the noise reduction expected from an arithmetic average of multiple samples containing Poisson-distributed noise (i.e., square root of the number of samples used for the average). Not surprisingly, it is identical to the red symbols on the top-left figures (boxcar average). The bottom-left and top-right plots show that higher orders polynomials, or filters convolved with windows, yield a noisier signal (less noise reduction) than in the case of the simple boxcar average. The bottom-right plot shows that noise reduction for low-pass filters designed with a prescribed cut-off frequency does not increase with the number of coefficients used.

S2 Impulse response and gain of commonly-used derivative filters

All filters considered here have the double function of smoothing and differentiating.

S2.1 Least squares fitting derivative filters (or Savitsky-Golay derivative filters)

In **section 2.3** of the main manuscript, we derived the coefficients of a 5-point boxcar function which was equivalent to fitting the signal using the least-squares technique with a polynomial of degree 1 (straight line). We can indeed use the second normal equation (second equation of the system of **Eqs. (21)** in the main manuscript**)**) to compute c_1, which is the value of the slope of the fitting function. Applying Faulhaber's summing formula to a polynomial of degree 1 (Knuth, 1993), we find the values of the filter coefficients as a function of the total number of terms N_T to be:

$$c_n = \frac{12}{N_T\left(N_T^2-1\right)}n = \frac{3}{N\left(N+1\right)\left(2N+1\right)}n \qquad -N \le n \le N \text{ and } N_T = 2N+1 > 2 \tag{S20}$$

For $N_T = 2N+1 = 3$ points, that corresponds to the central difference:

$$c_n = \frac{n}{2} \qquad -1 \le n \le 1 \tag{S21}$$

For $N_T = 2N+1 = 5$ points, that corresponds to:

$$c_n = \frac{n}{10} \qquad -2 \le n \le 2 \tag{S22}$$

For $N_T = 2N+1 = 7$ points, that corresponds to:

$$c_n = \frac{n}{28} \qquad -3 \le n \le 3 \tag{S23}$$

Using a similar mathematical development, the filter coefficients corresponding to the least-squares fitting technique by higher order polynomials can also be obtained. For polynomials of degrees 3 and 4 (cubic and quartic) we have:

$$c_n = 225\frac{(3N_T^4-18N_T^2+31)n-28(3N_T^2-7)n^3}{N_T(N_T^2-1)(3N_T^4-39N_T^2+108)} \qquad -N \le n \le N \text{ and } N_T = 2N+1 > 3 \tag{S24}$$

For $N_T=2N+1=5$ points, that corresponds to:

$$c_n = \frac{455-119n^2}{504}n \qquad -2 \le n \le 2 \tag{S25}$$

For $N_T=2N+1=7$ points, that corresponds to:

$$c_n = \frac{397-49n^2}{1512}n \qquad -3 \le n \le 3 \tag{S26}$$

The coefficients of the smoothing and derivative filters associated with the least squares fitting by polynomials of degrees 1 through 6 are provided by Savitsky and Golay (1964) with corrected values in Steinier et al. (1972). The impulse response and gain of these filters are plotted in **Fig. S6** for polynomials of degree 1 and 2 and **Fig. S7** for polynomials of degree 3 and 4, and for full widths ranging between 3 and 25 points.

S2.2 Low-pass derivative filters

Just as we did for the low-pass smoothing filters (**section S1**), we want to design a derivative low-pass filter with a prescribed cut-off frequency f_C. We therefore start with the initial conditions defining an ideal derivative low-pass filter:

$$H(f) = 2i\pi f \qquad \text{for } 0 < |f| < f_C$$

$$H(f) = 0 \qquad \text{for } f_C < |f| < 0.5 \tag{S27}$$

We find that these conditions are always true for a family of un-truncated Fourier series with the following transfer function:

$$H(f) = 2\sum_{n=1}^{\infty}\left(\frac{i}{n}\left(\frac{\sin(2\pi n f_C)}{\pi n} - 2 f_C \cos(2\pi n f_C)\right)\right)\sin(2\pi n f) \tag{S28}$$

Again, we truncate the series to a finite number of terms at the expense of producing Gibbs ripples. The actual low-pass filter thus created has the following $2N$+1 coefficients and transfer function:

$$c_n = \frac{2 f_C}{n}\left(\frac{\sin(2\pi n f_C)}{2\pi n f_C} - \cos(2\pi n f_C)\right) \qquad -N \le n \le N$$

$$H(f) = 2i\sum_{n=1}^{N}\left(\frac{f_C}{n}\left(\frac{\sin(2\pi n f_C)}{2\pi n f_C} - \cos(2\pi n f_C)\right)\right)\sin(2\pi n f) \tag{S29}$$

S2.3 Lanczos low-pass derivative filters

The low-pass filter coefficients will simply be multiplied by the sigma factors, as defined in **section S1.2**, to obtain the smooth derivative filter coefficients:

$$c_n = \frac{2 f_C}{n}\left(\frac{\sin(2\pi n f_C)}{2\pi n f_C} - \cos(2\pi n f_C)\right)\frac{\sin(\pi n / N)}{\pi n / N} \qquad -N \le n \le N \tag{S30}$$

S2.4 Kaiser window and NERD filter

The low-pass filter coefficients are multiplied by the Kaiser window weights to obtain the coefficients of the Near-Equal-Ripple Derivative (NERD) filter:

$$c_n = \frac{2 f_C}{n}\left(\frac{\sin(2\pi n f_C)}{2\pi n f_C} - \cos(2\pi n f_C)\right)\frac{I_0\left(\alpha\sqrt{1-(n/N)^2}\right)}{I_0(\alpha)} \qquad -N \le n \le N \tag{S31}$$

Figure S8 shows the impulse response (left) and gain (right) of a low-pass derivative filter with f_C=0.2 before any convolution. **Fig. S9** is similar to **Fig. S8** but after convolution by a Lanczos window. **Figure S10** is similar to **Fig.**

S8, but after convolution by a Kaiser window (50-dB attenuation). These figures show that the filters and the windows used are not optimized for all values of *N*. Therefore, the choice of filter must be carefully made together with the number of filter coefficients used

Table S1: Noise reduction factor (normalized to sqrt(2N+1)) for the least-squares fitting smoothing filters and windows introduced in this appendix

Noise reduction/sqrt(2N+1)	LS and MLS deg. 0-1	LS deg. 2-3
No window	1.00	0.66
w/ Lanczos window	0.84*	0.74*
w/ von Hann window	0.78*	0.71*
w/ Blackman window	0.73*	0.67*
w/ Kaiser 50-dB window	0.84*	0.72*

* Valid for N>3 only. For N<3, values depend on N and are 10-40% lower

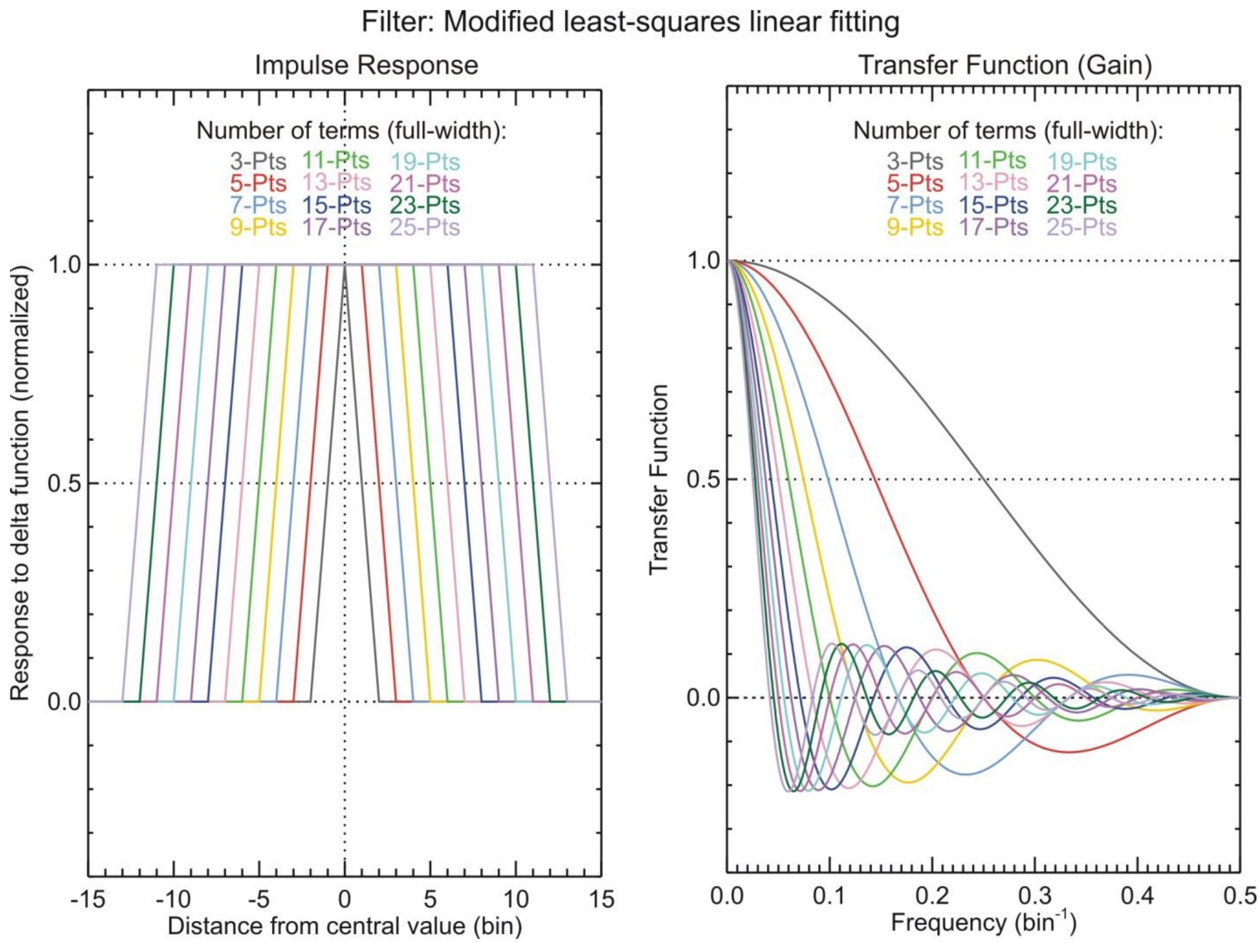

Figure S1: Impulse response (left) and gain (right) of a digital filter equivalent to the modified least-squares linear fitting method over an interval comprising 2*N*+1 points (full width). Full widths represented in this figure range from 3 to 25 points.

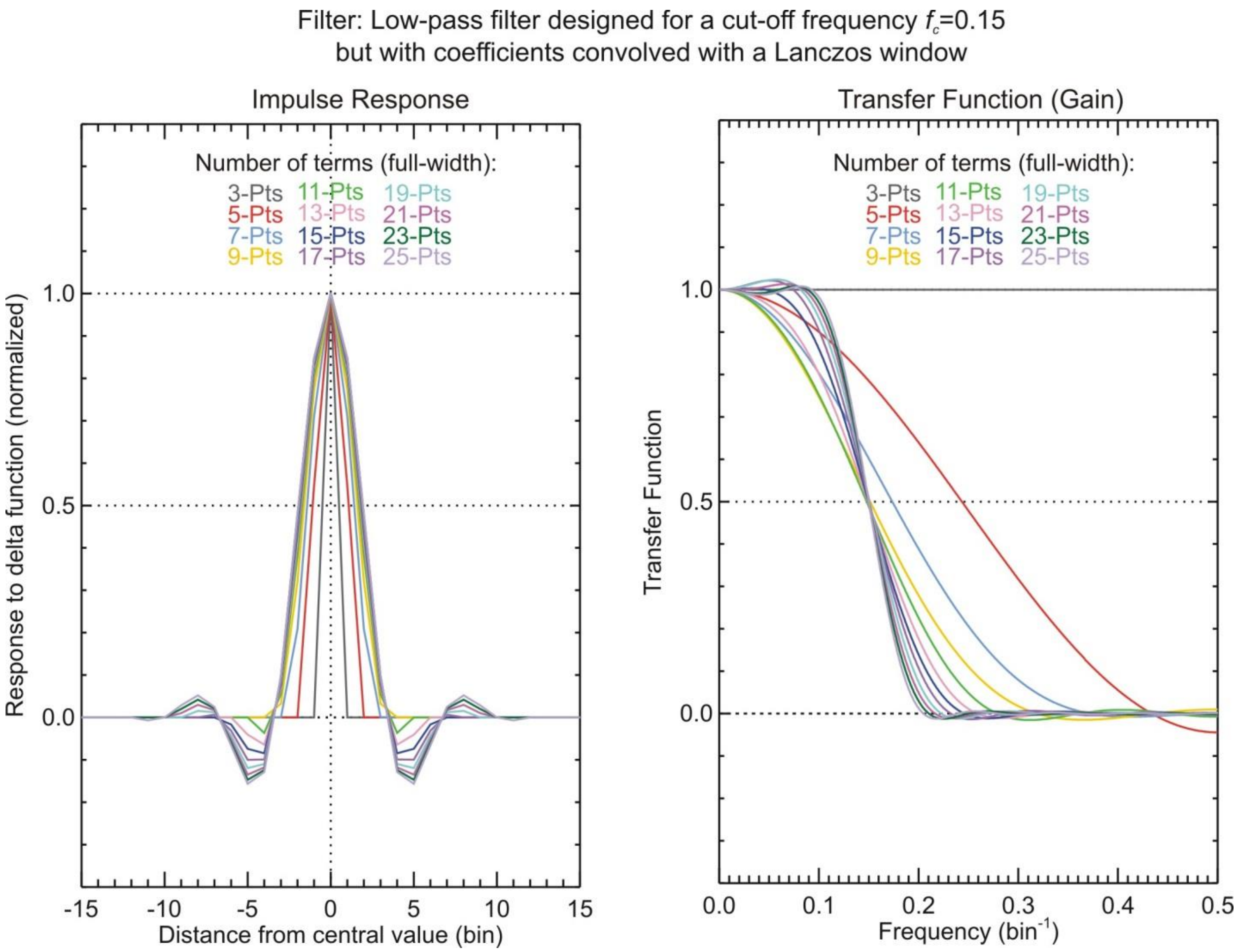

Figure S2: Impulse response and gain of a low-pass filters using 2*N*+1 coefficients (full width) convolved with a Lanczos window, and designed to have a cut-off frequency f_c=0.15. Full widths range from 3 to 25 points

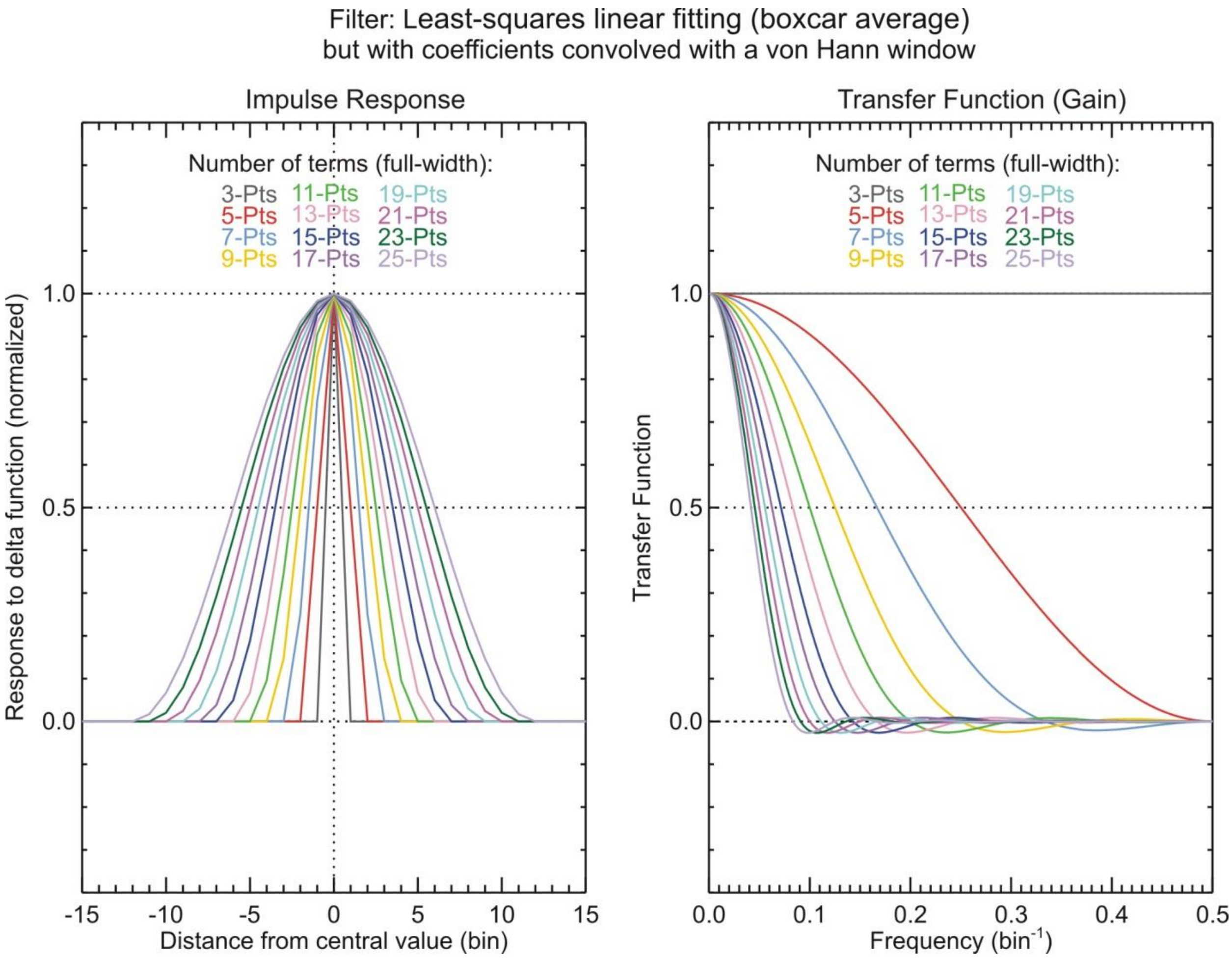

Figure S3: Impulse response and gain of least-squares linear fitting filters using 2*N*+1 coefficients (full width) and convolved with a von Hann window. Full widths range from 3 to 25 points

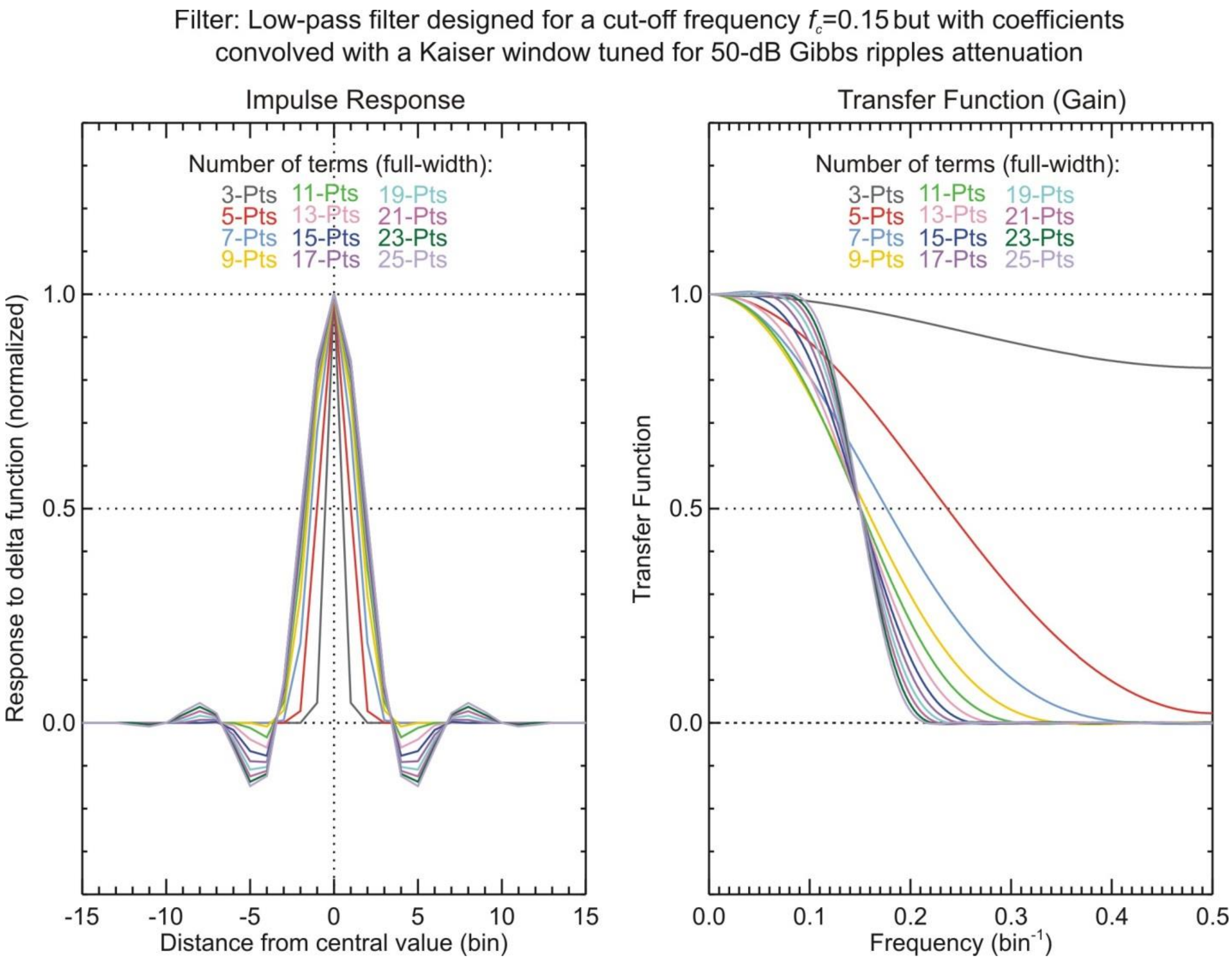

Figure S4: Impulse response and gain of a low-pass filters using 2*N*+1 coefficients (full width) convolved with a Kaiser window, and designed to have a cut-off frequency f_c=0.15 with a 50-dB attenuation. Full widths range from 3 to 25 points

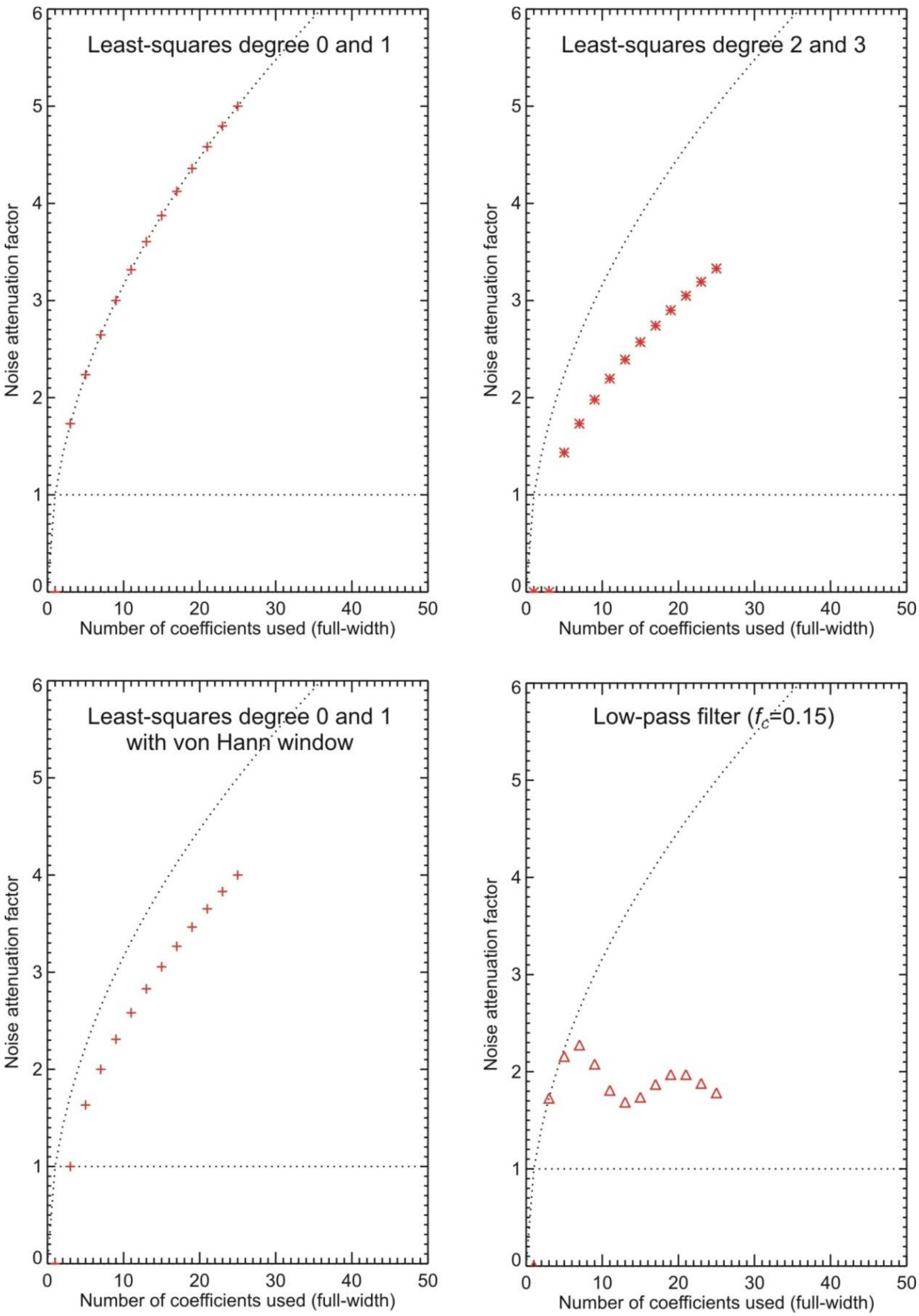

Figure S5: Noise reduction factor as a function of the number of coefficients, for a selected number of filters introduced in the previous section (see text for details)

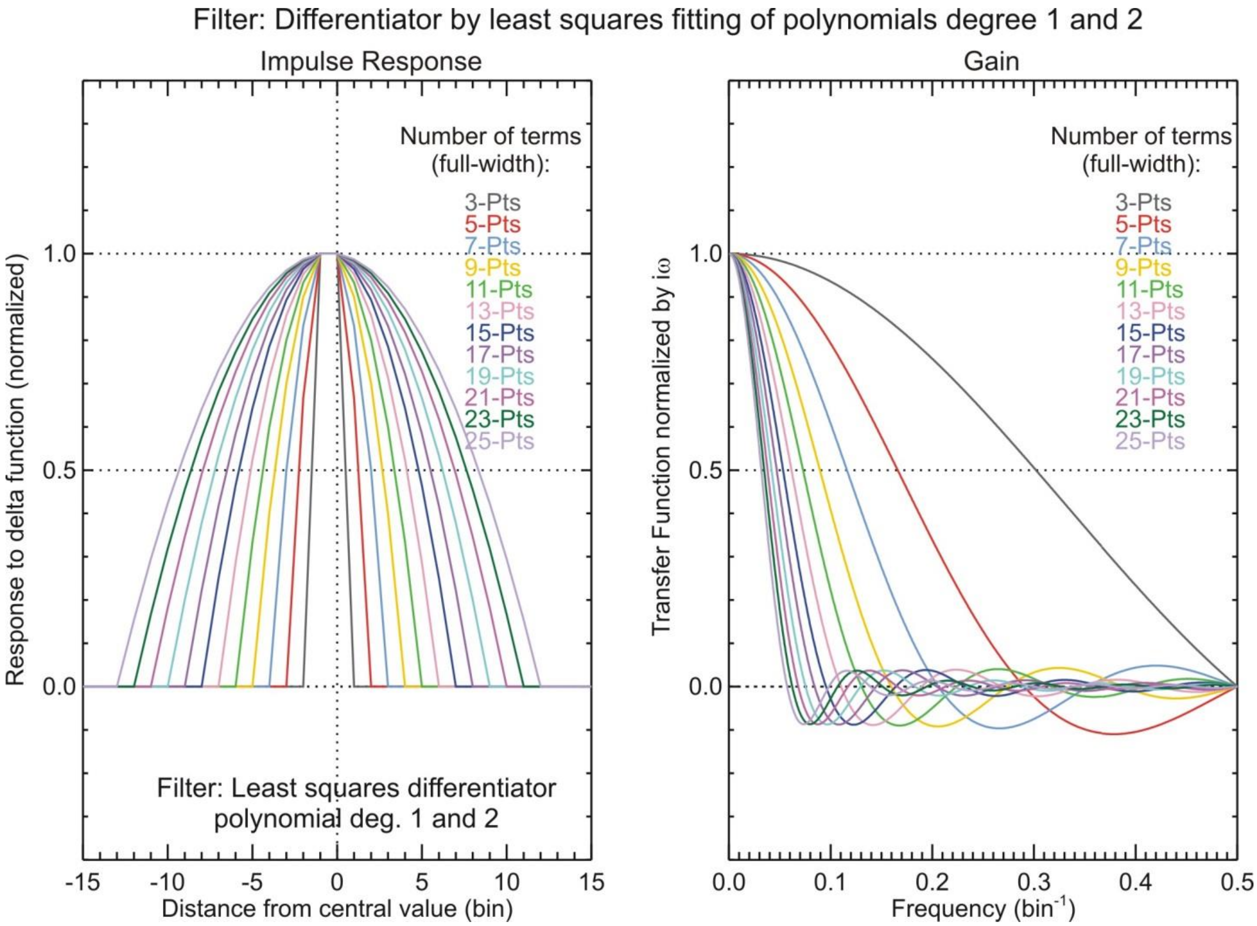

Figure S6: Impulse response (left) and gain (right) of derivative filters obtained from the calculated slope of a polynomial of degrees 1 and 2 using the least-squares fitting method over an interval comprising 2*N*+1 points (full width). The gain is the transfer function normalized by 2π*f*. Full widths range from 3 to 25 points

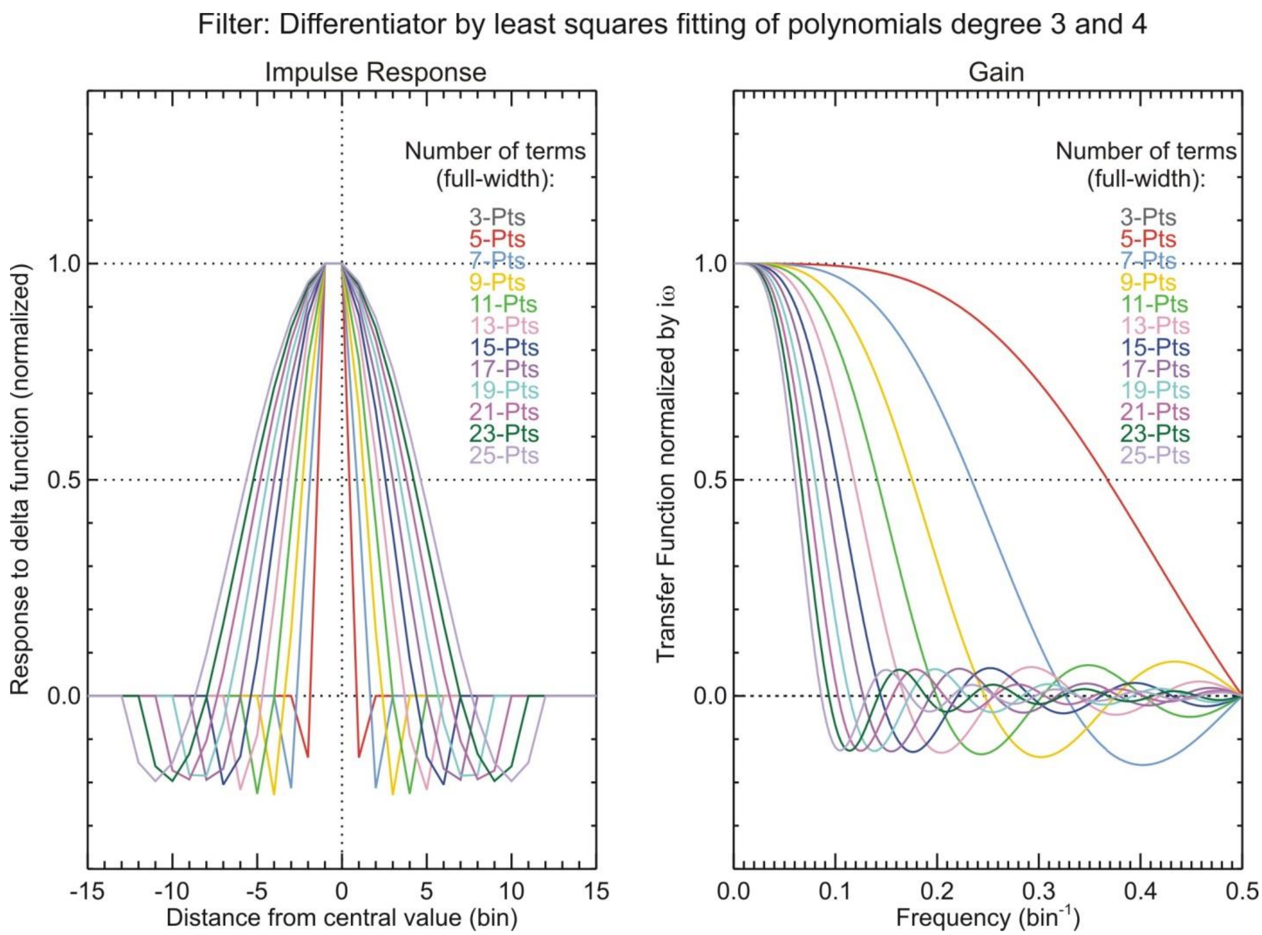

Figure S7: Same as Fig. S6, but fitting with polynomials of degree 3 and 4 instead of 1 and 2

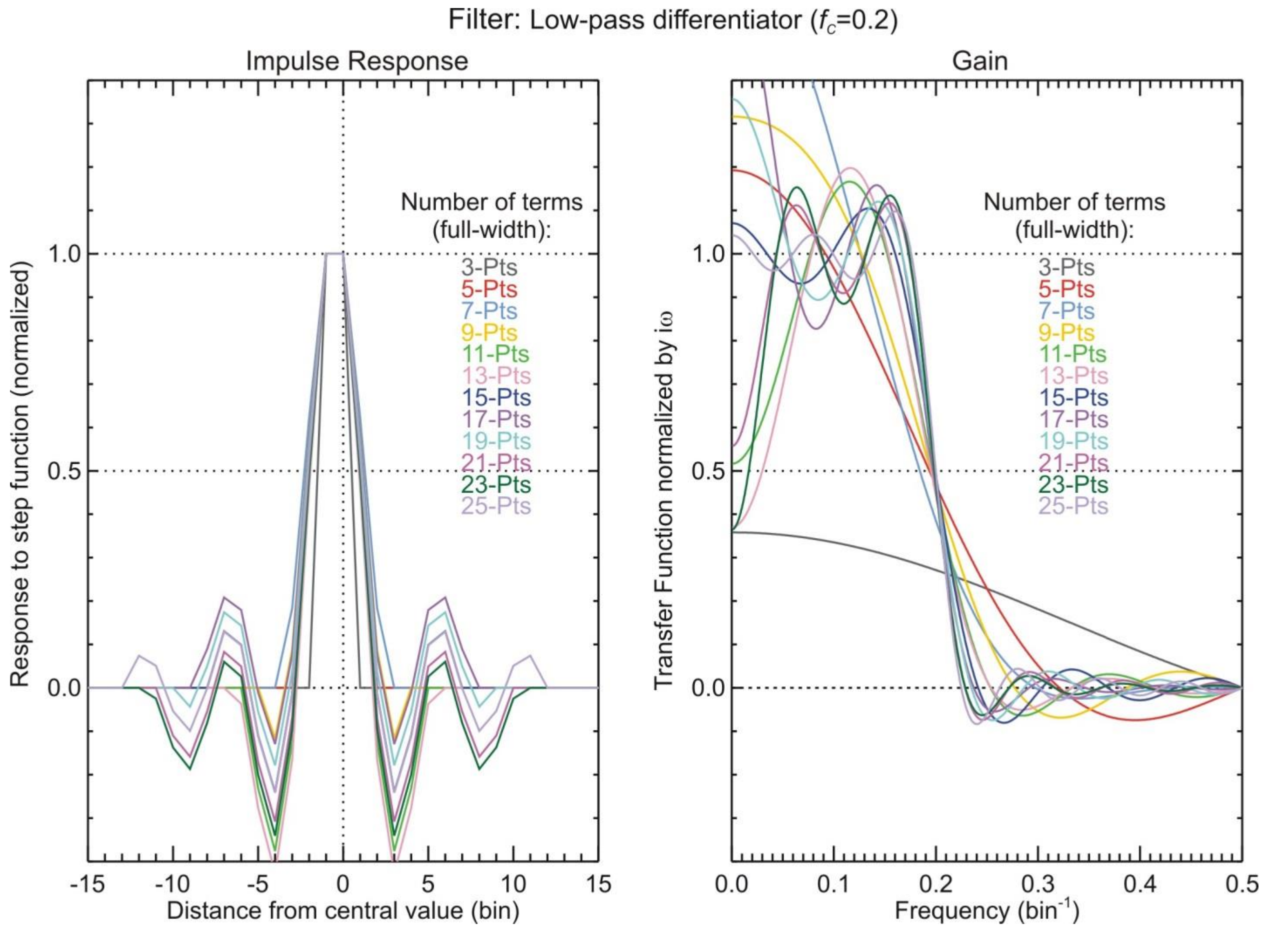

Figure S8: Impulse response (left) and gain (right) of a low-pass derivative filter (f_C=0.2). The gain is the transfer functions normalized by 2π*f*. Full widths range from 3 to 25 points

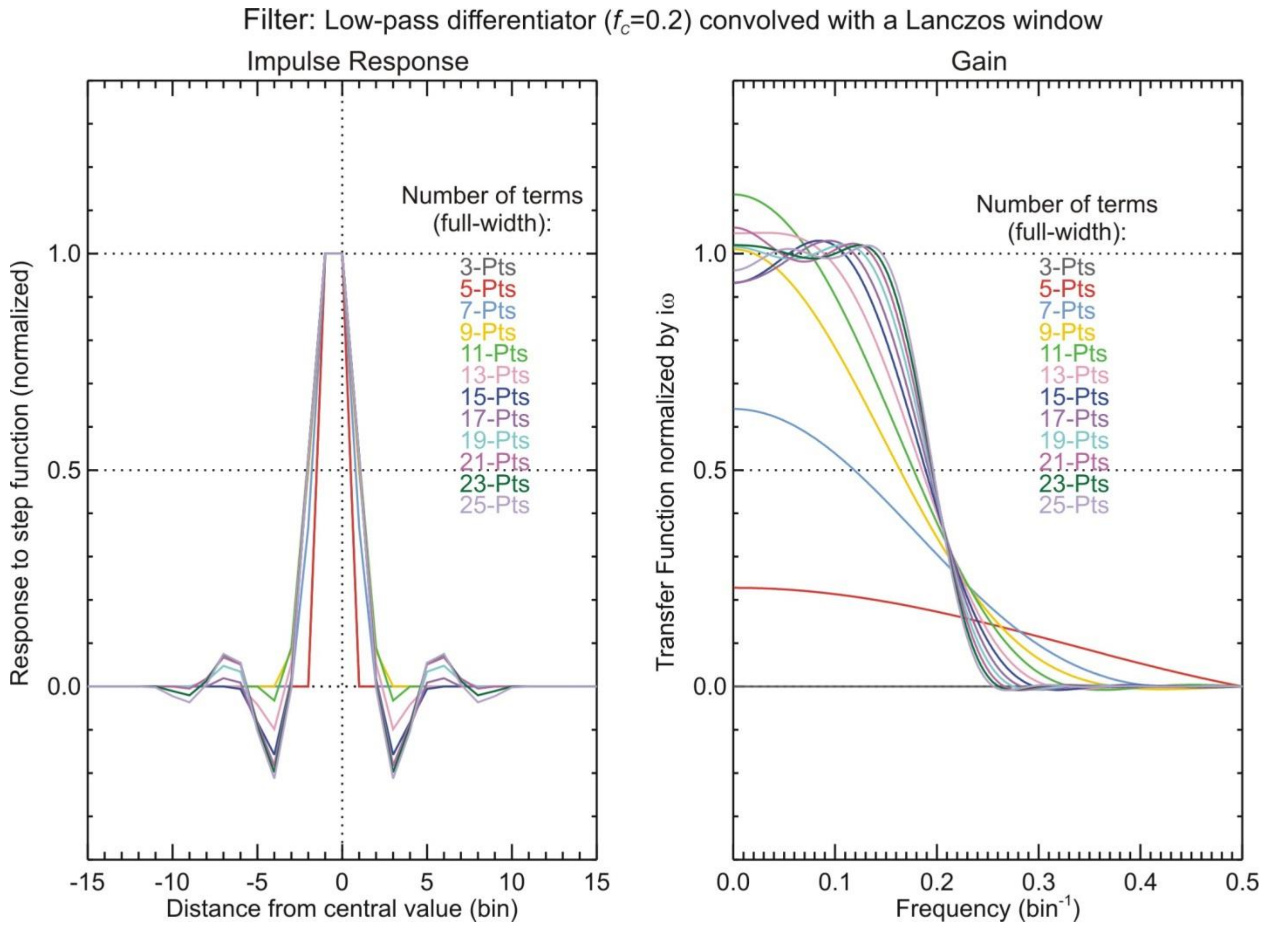

Figure S9: Same as Fig. S8 but after convolution by a Lanczos window

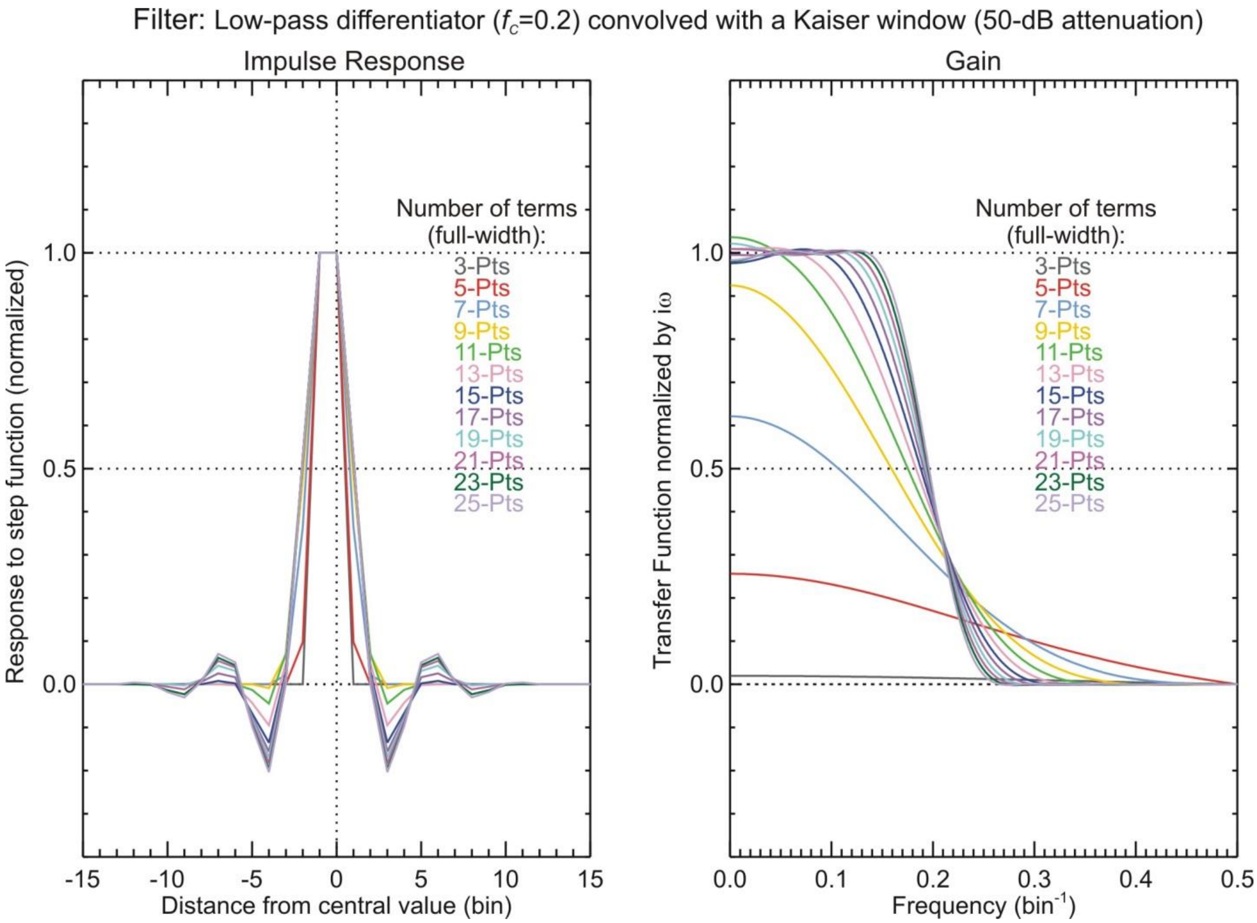

Figure S10: Same as Fig. S8 but after convolution by a Kaiser window tuned for a 50-dB attenuation