On the Frequency-Domain Properties of Savitzky-Golay Filters

Ronald W. Schafer

HP Laboratories
HPL-2010-109

Keyword(s):

Savitzky-Golay filter, least-squares polynomial approximation, smoothing

Abstract:

This paper is concerned with the frequency-domain properties of the so called Savitzky-Golay lowpass filters, which are based on the principle of local least-squares fitting of a polynomial. A summary of the important frequency-domain properties is given along with an empirically-derived formula for 3 dB cutoff frequency as a function of polynomial order N and impulse response half-length M.

External Posting Date: September 6, 2010 [Fulltext] Approved for External Publication
Internal Posting Date: September 6, 2010 [Fulltext]

ON THE FREQUENCY-DOMAIN PROPERTIES OF SAVITZKY-GOLAY FILTERS

Ronald W. Schafer

Hewlett-Packard Laboratories
1501 Page Mill Road
Palo Alto, CA 94304

ABSTRACT

This paper is concerned with the frequency-domain properties of the so called Savitzky-Golay lowpass filters, which are based on the principle of local least-squares fitting of a polynomial. A summary of the important frequency-domain properties is given along with an empirically-derived formula for 3 dB cutoff frequency as a function of polynomial order N and impulse response half-length M.

***Index Terms*—** Savitzky-Golay filter, least-squares polynomial approximation, smoothing

1. INTRODUCTION

Savitzky and Golay [1] proposed a method of data smoothing based on local least-squares polynomial approximation. They showed that fitting a polynomial to a set of input samples and then evaluating the resulting polynomial at a single point within the approximation interval is equivalent to discrete convolution with a fixed impulse response. The lowpass filters obtained by this method are widely known as Savitzky-Golay (S-G) filters. Savitzky and Golay were interested in smoothing of noisy data obtained from chemical spectrum analyzers, and they demonstrated that least-squares smoothing reduces noise while maintaining the shape and height of waveform (in their case, spectral) peaks. Subsequently, this property of the S-G filters has been found to be attractive in other applications such as ECG processing [2], and the basic concept has been generalized to two-dimensions [3] and applied in processing images such as ultra sound and SAR.

Most discussions of S-G filters emphasize their time-domain properties without reference to such frequency-domain features as passband width, stopband attenuation, etc. The purpose of this paper is to examine the S-G filters from the frequency-domain viewpoint and to quantify some of the frequency-domain properties of the S-G filters.

2. LEAST-SQUARES SMOOTHING OF SIGNALS

The basic idea behind least-squares smoothing is depicted in Figure 1, which shows a sequence of samples $x[n]$ of a discrete signal as solid dots. Considering for the moment the

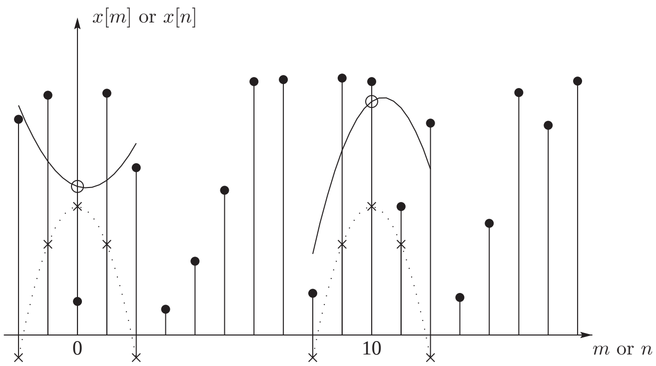

Fig. 1. Illustration of least-squares smoothing by locally fitting a second-degree polynomial (solid line) to five input samples: • denotes the input samples, ○ denotes the least-squares output sample, and × denotes the effective impulse response samples (weighting constants). (Dotted line denotes polynomial approximation to centered unit impulse.)

.

group of $2M+1$ samples centered at $n=0$, we obtain (by a process to be described) the coefficients of a polynomial

$$p(n) = \sum_{k=0}^{N} a_k n^k \qquad (1)$$

that minimize the mean-squared approximation error,

$$\mathcal{E}_N = \sum_{n=-M}^{M} \left(\sum_{k=0}^{N} a_k n^k - x[n] \right)^2 . \qquad (2)$$

In Figure 1, where $N = 2$ and $M = 2$, the solid curve on the left in Figure 1 is the polynomial $p(n)$ evaluated on a fine grid between -2 and $+2$, and the smoothed output value is obtained by evaluating $p(n)$ at the central point $n = 0$.[1] That is, $y[0]$, the output at $n = 0$, is $y[0] = p(0) = a_0$. The value of the output at the next sample is obtained by shifting the analysis interval to the right by one sample and repeating the polynomial fitting and evaluation at the central location.

[1] In general, the approximation interval need not be symmetric about the evaluation point. This leads to nonlinear phase filters, that may be useful for smoothing at the ends of finite-length input sequences.

This is repeated at each sample of the input, each time producing a value of the output sequence $y[n]$. Another example is shown on the right where the center of the interval is shifted to sample $n = 10$ and the new polynomial fit to the samples $8 \leq n \leq 12$ is shown again by the solid curve and the output at $n = 10$ is the value of the new polynomial evaluated at the center location.

The original paper by Savitzky and Golay [1] observed that at each position, the smoothed output value obtained by sampling the fitted polynomial is identical to a fixed linear combination of the the local set of input samples; i.e., the set of $2M+1$ input samples within the approximation interval are effectively combined by a fixed set of weighting coefficients that can be computed once for a given polynomial order N and approximation interval of length $2M + 1$. That is, the output samples can be computed by a discrete convolution of the form

$$y[n] = \sum_{m=-M}^{M} h[m]x[n-m] = \sum_{m=n-M}^{n+M} h[n-m]x[m] \quad (3)$$

The values marked with $\times$ in Figure 1 are the shifted impulse responses $h[0 - m]$ and $h[10 - m]$ that could be used to compute the output samples labeled with $\circ$, thus replacing the polynomial fitting process at each sample with a single evaluation of (3).

To show that we can find a single FIR impulse response that is equivalent to least-squares polynomial smoothing, we must first determine the optimal coefficients of the polynomial in (1) by differentiating $\mathcal{E}_N$ in (2) with respect to each of the unknown coefficients and setting the corresponding derivative equal to zero. This yields, for $i = 0, 1, \ldots, N$,

$$\frac{\partial \mathcal{E}_N}{\partial a_i} = \sum_{n=-M}^{M} 2n^i \left(\sum_{k=0}^{N} a_k n^k - x[n] \right) = 0, \qquad (4)$$

which, by interchanging the order of the summations, becomes the set of $N + 1$ equations in $N + 1$ unknowns

$$\sum_{k=0}^{N} \left(\sum_{n=-M}^{M} n^{i+k} \right) a_k = \sum_{n=-M}^{M} n^i x[n] \quad i = 0, 1, \ldots, N. \qquad (5)$$

The equations in (5) are known as the *normal equations* for the least-squares approximation problem. It is important to note before we proceed further that a unique solution requires that we have at least as many data samples as we have coefficients in the polynomial approximation. That is, we require $2M \geq N$. In fact, the equations in (5) become ill-conditioned if M and N are large and $2M$ is close to N.

Additional insight can be obtained by expressing the equations in (5) in matrix form. To do this it is helpful to define a $(2M + 1)$ by $(N + 1)$ matrix $\mathbf{A} = \{\alpha_{n,i}\}$ as the matrix with elements

$$\alpha_{n,i} = n^i, \quad -M \leq n \leq M, \;\; i = 0, 1, \ldots, N.$$

This matrix is called the *design matrix* for the polynomial approximation problem [5]. The transpose of $\mathbf{A}$ is $\mathbf{A}^T = \{\alpha_{i,n}\}$ and the product matrix $\mathbf{B} = \mathbf{A}^T\mathbf{A}$ is an $(N + 1) \times (N + 1)$ symmetric matrix with elements

$$\beta_{i,k} = \sum_{n=-M}^{M} \alpha_{in}\alpha_{nk} = \sum_{n=-M}^{M} n^{i+k} = \beta_{k,i},$$

for $i = 0, 1, \ldots, N$ and $k = 0, 1, \ldots, N$, which we see are the coefficients for the set of equations in (5). If we further define the vector of input samples as

$$\mathbf{x} = [\, x[-M], \ldots, x[-1], x[0], x[1], \ldots, x[M] \,]^T,$$

and define $\mathbf{a} = [a_0, a_1, \ldots, a_N]^T$ as the vector of polynomial coefficients, then it follows that the equations in (5) can be represented in matrix form as

$$\mathbf{B}\mathbf{a} = \mathbf{A}^T\mathbf{A}\mathbf{a} = \mathbf{A}^T\mathbf{x}.$$

Therefore, the solution for the polynomial coefficients can be written as

$$\mathbf{a} = (\mathbf{A}^T\mathbf{A})^{-1}\mathbf{A}^T\mathbf{x} = \mathbf{H}\mathbf{x}.$$

Now recall that the output for the group of samples centered on $n = 0$ is $y[0] = a_0$; i.e., we only need to obtain the coefficient a_0. Furthermore, we see that we only need the 0^{th} row of the $(N+1) \times (2M+1)$ matrix $\mathbf{H} = (\mathbf{A}^T\mathbf{A})^{-1}\mathbf{A}^T$, which by the definition of matrix multiplication gives a_0 as a linear combination of the $(2M + 1)$ elements of the $(2M + 1) \times 1$ column vector $\mathbf{x}$. The important observation is that the matrix $\mathbf{H}$ depends only on N and M and is independent of the input samples. Thus, the same weighting coefficients will be obtained at each group of $2M + 1$ samples, and so we can think of least-squares smoothing as a shift-invariant discrete convolution process.

One approach to finding the impulse response of the equivalent LTI system is to compute the matrix $\mathbf{H}$. Then, by the definition of matrix multiplication, the output will be

$$y[0] = a_0 = \sum_{m=-M}^{M} h_{0,m} x[m]$$

where $h_{i,n}$ denotes the elements of the $(N + 1) \times (2M + 1)$ matrix $\mathbf{H}$ and $h_{0,n}$ is an element of the 0^{th} row. Therefore, comparing this equation to the second term of (3) with $n = 0$, we observe that

$$h[-m] = h_{0,m} \quad -M \leq m \leq M.$$

Note that this equation gives $h[-m]$ since, as shown in (3), the impulse response is flipped with respect to the input in evaluating discrete convolution. Efficient matrix inversion techniques can be employed [5] to compute only this first row rather than the entire matrix H.

Another approach is to note that since the same weighting coefficients are obtained irrespective of the signal vector, we can set $\mathbf{x}$ equal to a unit impulse centered in the interval $-M \leq n \leq M$, and solve for all the coefficients of the corresponding polynomial approximation.[2] Then, the impulse response can be obtained by evaluating the corresponding polynomial at locations $-M \leq n \leq M$.

To show that this statement is true, we denote the coefficient vector for approximation of the impulse as $\tilde{\mathbf{a}}$, which is given by

$$\tilde{\mathbf{a}} = (\mathbf{A}^T\mathbf{A})^{-1}\mathbf{A}^T\mathbf{d}.$$

where $\mathbf{d} = [0, 0, \ldots, 0, 1, 0, \ldots, 0, 0]^T$ is a $(2M+1) \times 1$ column vector. Then for the impulse input $\mathbf{d}$, it follows that $\mathbf{A}^T\mathbf{d}$ is the $(N+1) \times 1$ column vector

$$\mathbf{A}^T\mathbf{d} = [1, 0, \ldots, 0]^T.$$

This means that $(\mathbf{A}^T\mathbf{A})^{-1}$ must have the form

$$(\mathbf{A}^T\mathbf{A})^{-1} = \begin{bmatrix} \tilde{a}_0 & \tilde{a}_1 & \cdots & \tilde{a}_N \\ \tilde{a}_1 & \bullet & \cdots & \bullet \\ \vdots & \vdots & \vdots & \vdots \\ \tilde{a}_N & \bullet & \cdots & \bullet \end{bmatrix},$$

where the matrix entries denoted $\bullet$ do not enter into the computation of $\tilde{\mathbf{a}}$. Now, since $\mathbf{A}^T$ is

$$\mathbf{A}^T = \begin{bmatrix} (-M)^0 & \cdots & 1 & \cdots & M^0 \\ (-M)^1 & \cdots & 0 & \cdots & M^1 \\ (-M)^2 & \cdots & 0 & \cdots & M^2 \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ (-M)^N & \cdots & 0 & \cdots & M^N \end{bmatrix},$$

it follows from the definition of matrix multiplication that the 0^{th} row of the matrix $\mathbf{H} = (\mathbf{A}^T\mathbf{A})^{-1}\mathbf{A}^T$ is

$$\begin{aligned} &\left[\begin{array}{ccccc} h_{0,-M} & h_{0,-M+1} & \cdots & h_{0,0} & \cdots & h_{0,M} \end{array}\right] \\ &= \left[\begin{array}{ccccc} \tilde{p}(-M) & \tilde{p}(-M+1) & \cdots & \tilde{p}(0) & \cdots & \tilde{p}(M) \end{array}\right] \end{aligned}$$

where $\tilde{p}(n)$ is the polynomial fit to the unit impulse,

$$\tilde{p}(n) = \sum_{k=0}^{N} \tilde{a}_k n^k \quad -M \leq n \leq M. \tag{6}$$

Therefore, the impulse response of the S-G filter is

$$h[-n] = h_{0,n} = \tilde{p}(n).$$

As before, this equation gives $h[-n]$ since the impulse response is flipped around $n = 0$ in evaluating discrete convolution. Henceforth, we shall refer to $\tilde{p}(n)$ as the *impulse response design polynomial*. As we will discuss in Section 5, (6) is the basis for a simple method for design of S-G filters using the polynomial fitting functions in MATLAB.

[2]Note that these polynomial coefficients, denoted $\tilde{\mathbf{a}}$, will not in general be equal to those of any of the local approximations that are implicitly generated for each group of $2M+1$ input samples.

3. MOVING AVERAGE FILTERING AS LEAST-SQUARES FILTERING

An often used expedient for data smoothing is symmetrical moving average (MA) filtering defined as the convolution

$$y[n] = \frac{1}{2M+1}\sum_{m=-M}^{M} x[n-m] = \frac{1}{2M+1}\sum_{m=n-M}^{n+M} x[m], \tag{7}$$

from which we see that the impulse response for the symmetrical MA, filter is

$$h[n] = \begin{cases} \dfrac{1}{2M+1} & -M \leq n \leq M \\ 0 & \text{otherwise.} \end{cases} \tag{8}$$

Symmetrical MA filtering is identical to polynomial smoothing with a polynomial of degree $N = 0$ (i.e., a constant). To see this, we simply assume $N = 0$ in (5) to obtain the single equation

$$\left(\sum_{n=-M}^{M}\right) a_0 = \sum_{n=-M}^{M} x[n]$$

from which we obtain for the output value at the center of the interval

$$y[0] = a_0 = \frac{\displaystyle\sum_{n=-M}^{M} x[n]}{\displaystyle\sum_{n=-M}^{M}} = \frac{1}{2M+1}\sum_{n=-M}^{M} x[n],$$

which is the average over the $2M+1$ samples. Thus, we see that MA filtering is entirely equivalent to least-squares polynomial smoothing with a polynomial of degree $N = 0$.

It is also interesting to consider least-squares smoothing with a polynomial of degree $N = 1$. Again from (5), the coefficients of the fitted polynomial must satisfy the equations[3]

$$\begin{bmatrix} \displaystyle\sum_{n=-M}^{M} & 0 \\ 0 & \displaystyle\sum_{n=-M}^{M} n^2 \end{bmatrix} \begin{bmatrix} a_0 \\ a_1 \end{bmatrix} = \begin{bmatrix} \displaystyle\sum_{n=-M}^{M} x[n] \\ \displaystyle\sum_{n=-M}^{M} n x[n] \end{bmatrix}$$

from which it follows easily that

$$a_0 = \frac{\displaystyle\sum_{n=-M}^{M} x[n]}{\displaystyle\sum_{n=-M}^{M}} = \frac{1}{2M+1}\sum_{n=-M}^{M} x[n],$$

[3]We have used the fact that $\displaystyle\sum_{n=-M}^{M} n = 0$.

and

$$a_1 = \frac{\sum_{n=-M}^{M} n x[n]}{\sum_{n=-M}^{M} n^2}.$$

Since the output of the smoothing process is simply $y[0] = a_0$, it follows that least-squares polynomial smoothing with a first-degree polynomial (straight line) is identical to smoothing by least-squares fitting a constant, which we have shown above to be equivalent to MA filtering. Clearly, the reason for this is that the sum of the integers $-M, \ldots, -1, 0, 1, \ldots, M$ is zero, making the output independent of a_1.

4. A SUMMARY OF PROPERTIES OF S-G FILTERS

Figure 2 shows the impulse response of a S-G filter with $N = 6$ and $M = 16$. Although this is a specific example, its properties are representative of the entire class of symmetric S-G filters. Figure 3 shows the frequency response of several S-G filters designed by MATLAB statements given in Section 5. The impulse response lengths are all $(2M+1) = 2 \cdot 16 + 1 = 33$ with implicit polynomial orders of $N = 0, 2, 4, 6, 12$. Figures 2 and 3 illustrate properties that are shared by all S-G filters. These properties, which can be shown to be due to the structures of the matrices $\mathbf{B}$ and $\mathbf{H}$ are summarized below:

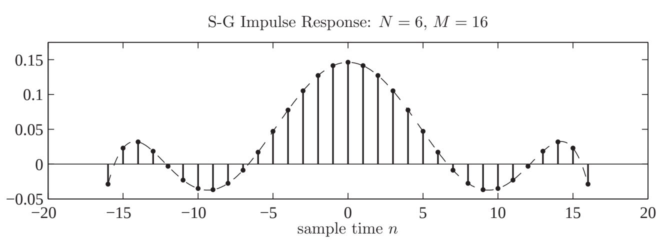

Fig. 2. Impulse response of a S-G filter with $M = 16$ and $N = 6$. Dashed curve is the polynomial $\tilde{p}(n)$ evaluated on a dense grid.

.

- The odd-indexed coefficients of the impulse response design polynomial all zero so that we can express $\tilde{p}(n)$ as

$$\tilde{p}(n) = \sum_{k=0}^{\lfloor N/2 \rfloor} \tilde{a}_{2k} n^{2k} \qquad (9)$$

where $\lfloor \; \rfloor$ means rounding down.

- The impulse response is symmetric since $h[-n] = \tilde{p}(n) = \tilde{p}(-n) = h[n]$. Therefore, the frequency response is purely real. (The shifted impulse response $h[n-M]$ is causal and the corresponding frequency response has linear phase corresponding to the time delay of M samples.) S-G filters are type I FIR lowpass filters [6] with nominal passband gain of unity.

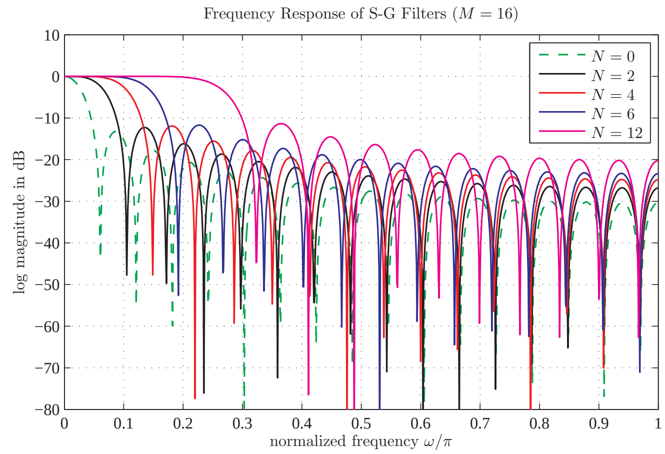

Fig. 3. Frequency response of S-G filters with $M = 16$ and various polynomial orders.

.

- The zeros of the system function $H(z)$ of a S-G filter are either on the unit circle of the z-plane or they occur in complex conjugate reciprocal groups [6]. The unit circle zeros are, of course, responsible for the sharp dips (high attenuation) in the stopband of the frequency response.

- The nominal normalized cutoff (3 dB) frequency, $f_c = \omega_c/\pi$, depends on both the implicit polynomial order N and the length of the impulse response, $(2M+1)$. If M is fixed as in Figure 3, the passband of the filter gets wider approximately in proportion to N. Although not illustrated in Figure 3, the cutoff frequency depends inversely on M. Section 5 gives an approximate empirical relation for f_c as a function of N and M.

- S-G filters have very flat frequency response in their passbands since it can be shown that $H(e^{j\omega})\big|_{\omega=0} = 0$ and

$$\left.\frac{d^r H(e^{j\omega})}{d\omega^r}\right|_{\omega=0} = (-j)^r \sum_{n=-M}^{M} n^r h[n] = 0, \qquad (10)$$

for $r = 1, 2, \ldots, N$.

- The S-G filters have mediocre attenuation characteristics in their stopband regions (except at the frequencies corresponding to zeros on the unit circle). Defining the stopband as the frequency range from the first zero up to π radians, we see from Figure 3 that for the MA filter ($N = 0$), the minimum attenuation in the stopband (amplitude of first peak after the first zero) is approximately 13 dB. For $N \geq 2$, the minimum attenuation in the stopband is approximately 11 dB. Also illustrated in Figure 3 is the fact that the peak stopband gain tends to increase with increasing N for fixed M. Figure 3 also shows that the frequency response decreases in gain as frequency increases above the nominal cutoff frequency.

- A final property not explicitly illustrated by Figures 2 and 3 is that for the symmetric case that we have emphasized, the S-G filters for polynomial orders N and $N+1$ (N even) are identical. This was illustrated in Section 3 where it was shown that the cases $N = 0$ and $N = 1$ give the same impulse response. It can be shown this is true in general due to the symmetry of the approximation region.

5. DESIGN OF S-G FILTERS

Recall from (6) that the impulse response of a S-G filter can be computed as samples of the N^{th} degree polynomial fit to the unit impulse. This method of computing the S-G filters is easily implemented using MATLAB's polynomial functions as in the following MATLAB statements:

```
a=polyfit( -ML:MR,...
    [zeros(1,ML),1,zeros(1,MR)],N );
h=fliplr( polyval(a,-ML:MR) )
```

The MATLAB function `polyfit( )` computes the coefficients of the impulse response design polynomial and `polyval( )` evaluates the polynomial at a discrete set of points. Note that these statements can be use to compute non-symmetric S-G filter by setting ML≠MR.

There are some important constraints in the use of polynomial fitting in general. Specifically, the number of data points (in this case $2M+1$) must be at least as large as the number of undetermined coefficients $N+1$. Furthermore, if the order of the polynomial, N, is too large, the approximation problem is badly conditioned and the solution will be of no value. (The function `polyfit( )` issues an alert when the approximation problem is ill-conditioned.) Although these factors are significant limitations, a wide range of frequency-domain characteristics can be achieved by choosing M and N appropriately.

In order to quantify the frequency-domain behavior of S-G filters, impulse responses were computed for various values of M and N within the constraints mentioned above, and the corresponding frequency responses were computed for $0 \leq \omega \leq \pi$. The passband of the filter was defined by the frequency where $20\log_{10}|H(e^{j\omega})|$ is "3 dB down" from the value of 0 dB, the gain of the filter at $\omega = 0$. The results for measurements on filters with $M = 25, 50, 100, 200$ and even orders $N = 2, 4, \ldots, 32$ are displayed in Figure 4. The points marked with * and connected by a blue line are the

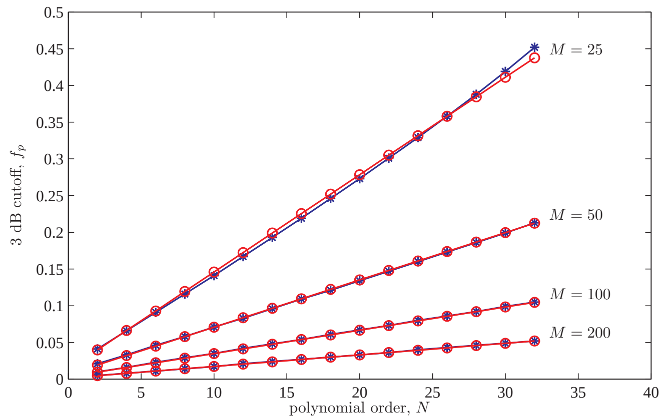

Fig. 4. Relationship between 3 dB cutoff frequency f_c, polynomial length N, and impulse response half-length M.)

.

measured cutoff frequencies. It is seen that f_c varies almost linearly with N with the slope being dependent inversely on M. A reasonably accurate approximation to this behavior for the indicated range of parameters is the equation

$$f_c = \frac{N+1}{3.2M - 4.6}. \qquad (11)$$

The values of $f_c = \omega_c/\pi$ predicted by this equation are marked with $\circ$ and connected by a red line. Figure 4 shows that this simple formula fits the measurements quite well except for the case $M = 25$ where the measurements deviate somewhat from linear over the entire range of N. The relative error in predicting the measured cutoff frequency is less than 4% over the range $M = 25, 50, 100, 200$ and $N = 4, \ldots, 32$, and the relative error is within 8% for the cases $M = 25, 50, 100, 200$ and $N = 2$. As can be seen, large values of M and $N = 2$ lead to extremely narrow passbands, which would be of limited usefulness except when the signal components are greatly oversampled. It was found that even though the function `polyfit( )` gave an ill-conditioned warning for the larger values of N, the resulting filters remained acceptable for values of N up to 40. The formula in (11) becomes increasingly accurate for larger values of M and N. The formula does not fit as well for values of M less than 25. However, the dependence of f_c on N is still linear. For $M < 25$ and N suitably restricted, a formula similar to (11) with 4.6 replaced by 2 gives more accurate predictions. While a more complicated functional form based on more measurements could provide more accurate predictions over a wider range of M and N, (11) should be adequate for most applications of S-G filters.

6. DISCUSSION

Savitzky-Golay filters are based on the principle of fitting of an N^{th} degree polynomial to a set of input samples in a finite-length interval around the output sample time. There is value in knowing that a single impulse response implicitly achieves this local polynomial fitting for every output sample. However, in many applications, signals are not characterized in terms of their ability to be modeled by polynomials but rather in terms of their frequency spectra. Thus, we have focused in this paper on the frequency-domain properties of the S-G filters.

Savitzky-Golay filters are often preferred because, when they are appropriately designed to match the waveform of an oversampled signal corrupted by noise, they tend to preserve the width and height of peaks in the signal waveform. While such performance features are often explained in terms of the implicit polynomial fitting process (where it is assumed that the fitted polynomial slopes are matched to those of the signal) the reason for this behavior is also obvious from the frequency domain properties of the filters. Specifically, they have extremely flat passbands with modest attenuation in their stopbands. Furthermore, the symmetric S-G filters have zero phase so that features of the signal are not shifted. Thus, if the signal has most of its energy in the passband of the filter (implying significant over-sampling), the signal components are undistorted while some high-frequency noise is reduced but not completely eliminated. Of course, assuming that the signal is lowpass is equivalent to assuming that the signal is smooth enough to be represented by a polynomial of high enough degree. However, S-G filters are often used in situations where a direct frequency-domain specification is more precise or more easily related to models for signal production. Toward the end of quantifying the design of S-G filters, we have given an empirical relationship in (11) between 3 dB frequency and the parameters M and N.

If one adopts the frequency-domain point of view, the question naturally arises as to whether the main desirable property of the S-G filters (very flat passband) could be achieved with another design method, and perhaps with greater attenuation in the stopband region. Figure 5 shows the frequency response of an S-G filter with $M = 16$ (impulse response length $L = 2M + 1 = 33$) and $N = 6$. Also shown is the frequency response of a length $L = 33$ filter designed by the Parks-McClellan (P-M) algorithm. In this example, the passband and stopband cutoff frequencies were adjusted by trial and error so that the transition region and the location of the first zero of the frequency response were approximately in the same location as those of the corresponding S-G filter. The measured 3 dB cutoff frequency of the S-G filter was $f_c = 0.143$ (the formula of (11) predicts $f_c = 0.15$). A very flat passband is achieved with the P-M design algorithm by imposing a 10:1 ratio between the passband equiripple approximation error and the stopband approximation error.[4] In the case of the S-G filter, the gain at the first local maximum beyond the first zero of the frequency response is -11.73 dB, while the equiripple maxima of the P-M filter have gains of -19.9 dB. The lower part of the plot shows that the passband gain of the P-M filter has small ripple about 0 dB, and the flat region is in fact wider than that of the S-G filter. It should be noted that due to the tendency of S-G frequency responses to fall off at high frequencies, the S-G filter has lower peak stopband gain than the P-M filter after about $\omega/\pi = 0.5$.

Given the close similarity of the two frequency responses in Figure 5, it is clear that for the case of a signal confined to the band $|\omega| < 0.143\pi$ with additive white noise, the performance of the two systems should be nearly identical.

[4]Larger ratios will make the passband even flatter.

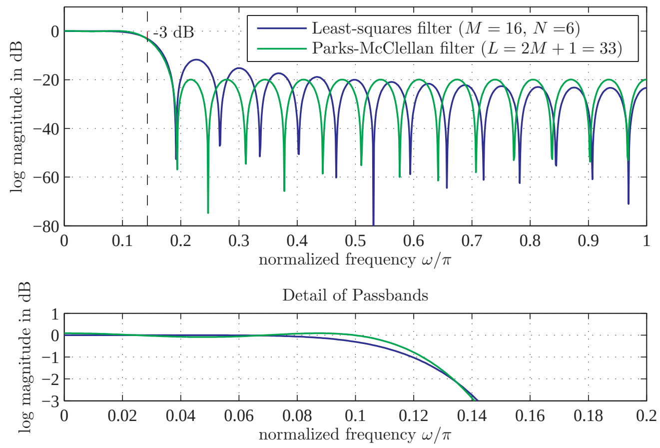

Fig. 5. Comparison of S-G filter ($N = 6$ and $M = 16$) with an equal-length equiripple filter designed with the Parks-McClellan algorithm: upper plot is entire frequency response and lower plot is only the passband region.)

.

7. REFERENCES

[1] A. Savitzky and M. J. E. Golay, "Soothing and Differentiation of Data by Simplified Least Squares Procedures," *Analytical Chemistry*, vol. 36, pp. 1627-1639, 1964.

[2] K. Pandia, S. Revindran, R. Cole, G. Kovacs, and L. Giaovangrandi, "Motion Artifact Cancellation to Obtain Heart Sounds from a Single Chest-wWorn Accelerometer," *Proc. ICASSP-2010*, pp. 590-593, 2010.

[3] M. Sühling, M. Arigovindan, P. Hunziker, and M. Unser, "Multiresolution Moment Filters: Theory and Applications," *IEEE Trans. on Image Processing*, vol. 13, No. 4, pp. 484-495, April, 2004.

[4] R. W. Hamming, *Digital Filters*, 3E, Prentice-Hall, Inc., Englewood Cliffs, NJ, 1989.

[5] W. H. Press, S. A. Teukolsky, W. T. Vertterling, and B. P. Flannery, *Numerical Recipes*, 3e, Cambridge University Press, 2007.

[6] A. V. Oppenheim and R. W. Schafer, Discrete-Time Signal Processing, 3e, Pearson, Upper Saddle River, NJ, 2010.