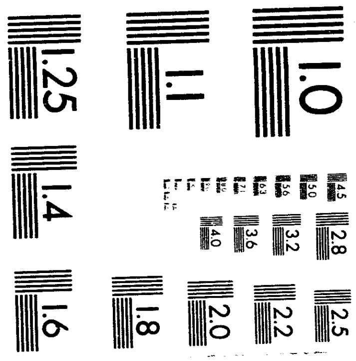


1.0
1.1
1.25
1.4
1.6
1.8
2.0
2.2
2.5
2.8
3.2
3.6
4.0
4.5
5.0
5.6

1 of 1

SAND93-3979C
RS, A Comment Copy
Conf-940581--1

ARITHMETIC AVERAGING – A VERSATILE TECHNIQUE FOR SMOOTHING AND TREND REMOVAL*

Edward L. Clark
Sandia National Laboratories
Albuquerque, NM 87185-0825

KEYWORDS

Data Smoothing, Digital Filters, Time Series Analysis

ABSTRACT

Arithmetic averaging is simple, stable, and can be very effective in attenuating the undesirable components in a complex signal, thereby providing smoothing or trend removal. An arithmetic average is easy to calculate. However, the resulting modifications to the data, in both the time and frequency domains, are not well understood by many experimentalists. This paper discusses the following aspects of averaging: (1) types of averages–simple, cumulative, and moving; and (2) time and frequency domain effects of the averaging process.

INTRODUCTION

By far the most common measure of central tendency for a set of data representing a finite sample of size n is the sample mean, $\bar{x}$. It can be shown[1] that the expected value of $\bar{x}$ is the population mean, μ. Thus, we can expect $\bar{x}$ to provide a good approximation to the population mean.

This statistic is so common–it is usually the first to be encountered in a college statistics course and, indeed, we were aware of it long before then–that we may have forgotten what a powerful tool it can be for experimentalists. In this paper, it will be assumed that the data sets of interest are time series consisting of two components, denoted as signal and noise. The signal portion is deterministic, that is, it is repeatable and can be defined by a mathematical function. The signal may be constant, periodic, or transient. Usually, it is the signal that we desire to enhance. The noise portion may be periodic or random. Although we are usually trying to eliminate the noise, in some cases, such as determination of the frequency content of periodic noise, it is the noise component that we want to recover.

Arithmetic averaging techniques will be defined which allow the user to: (1) define the signal mean within a specified accuracy; (2) smooth the data, that is, attenuate undesired noise to emphasize the trend;

*This work was supported by the United States Department of Energy under Contract DE-AC04-94AL85000.

and (3) eliminate the trend for enhanced spectral analysis of the noise. The first application will employ operations called simple and cumulative averaging. The second and third applications will use moving averages. In most cases, it will be assumed that the time series samples are uniformly spaced with a sampling interval, T. However, for many of the results, uniform sampling is not required, and of course, the simple average is not restricted to time series. Of particular interest will be the effects of the averaging processes in both the time and frequency domains. The principal effects in the time domain are variance reduction and sample correlation. In the frequency domain, attenuation of selected frequency components is the main effect.

Simple and cumulative averages are unique and do not require justification for their use. However, the moving average is one of two types of digital filters and it is necessary to explain why the moving average is the best choice for the present applications. Digital filters are computational algorithms which accomplish desired modifications to selected frequency components of a time series. They may be divided into two categories–nonrecursive and recursive. A nonrecursive filter uses only the input data set in computing the output set. It is also called a finite impulse response (FIR) filter because the effect of an impulse function (also call a delta function) in the input data has only a limited influence on the output data. A weighted moving average is a nonrecursive filter. In filter notation, the weights are called coefficients. A recursive filter uses both the input data set and previous values of the output set in computing the current output, therefore it has feedback. It is also called an infinite impulse response (IIR) filter because an impulse function in the input data set will influence the entire output data set for times greater than that at which the impulse occurred. Both filter types can be equally effective and both have advantages and disadvantages. There are three primary advantages of nonrecursive filters. First, they are simple to implement and the filter coefficients can be easily determined without use of a design code. Recursive filters require fairly complex design codes. Second, they are always stable because there is no feedback from the output series. Therefore, the output series cannot diverge from the input series as may happen with recursive filters. Third, nonrecursive filters can be easily designed to provide zero phase (no time delay) or linear phase (constant time delay). A zero phase filter allows direct comparison of the input and output data to insure that the desired smoothing occurs. Zero or linear phase is also desirable to avoid distortion of the signal in the passband, i.e., for the frequencies between zero and the cutoff frequency (defined as the frequency at which some specified attenuation, usually 30% or -3 db, of the signal occurs). Most recursive filters either distort the signal (nonlinear phase) or tend to be unstable if designed for linear phase. Although there are techniques to achieve zero phase with any recursive filter by direct and then reverse filtering of the data[2], this approach changes the frequency response function (to be defined later) of the filter (it is squared) and can significantly distort the signal at each end of the series. It has been the author's experience that, although recursive filters such as a Butterworth design, are very useful with periodic signals which have a relatively large signal-to-noise ratio (SNR) and many cycles of data, a nonrecursive filter is far superior with very noisy (small SNR) transient data. Additional advantages of nonrecursive filters include that the time and frequency domain characteristics (effects) of the filters can be easily determined and the filters are less sensitive to computational inaccuracy. The disadvantages of nonrecursive filters include the need for higher order filters, i.e., more filter coefficients, to accomplish the same rolloff characteristics in the transition region between the passband and stopband. This results in increased computation time and larger memory requirements. If there is not a requirement to filter the data in real time, for example, if it can be filtered post-test, the increased time is not usually a significant problem. Also, because the nonrecursive filter equation is a discrete convolution of the coefficients and data, fast Fourier transforms (FFT) can be used to greatly reduce computation time although at the cost of added coding complexity. Probably the major disadvantage of the zero phase nonrecursive filter is that the first and last output samples are offset from the ends of the input series by T(n-1)/2 seconds, where n is the number of coefficients in the filter. In other words, (n-1)/2 samples at each end of the time series cannot be filtered. Finally, although recursive digital filters can be matched to analog filters for system simulation, this is not true for nonrecursive filters. To summarize the advantages of arithmetic averaging (nonrecursive filtering) for data smoothing and trend removal, it is a simple, low-risk procedure. With the few easy rules to be defined in the following sections, the experimentalist can perform the smoothing operation, based on desired variance reduction and/or frequency component attenuation, without recourse to handbooks and complicated design and implementation codes.

Before the various averaging operations and their effects are described, a warning is necessary. Averaging is a very powerful technique for reducing the precision error present in a set of data. However, averaging of poor data should never be substituted for an improved process which will provide better data. "It always pays more to make a better measurement than to repeat an old one[3]". Aside from the obvious ethical requirement for an experimentalist to obtain the best measurements possible, many time series measurements of physical processes are correlated, that is, the samples are not independent. This correlation may occur from the physical process (turbulence, periodic phenomena, etc.) or the data acquisition (analog filters, pneumatic lag in pressure systems, etc.). It cannot be emphasized enough that the reduction in precision error that occurs when measurements are averaged, will be far less for correlated data than for independent data.

SIMPLE AND CUMULATIVE AVERAGES

Simple and cumulative averages are used to determine a single value which defines the zero frequency (dc) level of the signal. The simple average is used when the mean of the data set is not varying significantly with time. Equations are developed for this process which allow the user to estimate the number of samples required to define the mean within a specified uncertainty. This feature is valuable when multiple sets of data, having similar mean and variance, are to be acquired, since it will allow the user to avoid the expense of acquiring unneeded samples. The cumulative average is used when the short-term mean of the data is varying. It provides a graphical indication of the time at which a satisfactory mean value can be obtained.

SIMPLE AVERAGE

The simple average is used to reduce the noise so that the signal level is defined within specified error limits. A typical application would be to estimate the mean pressure or velocity from measurements at a fixed point in a turbulent flow. As will be shown later, a simple average can also be used effectively to eliminate a single frequency and its harmonics, for example, to remove 60 Hz noise from a constant signal. The basic equation for the simple average is,

$$\bar{x} = \frac{1}{n}\sum_{i=1}^{n} x_i \qquad (1)$$

where $\bar{x}$ is the average value, x_i is the set of data samples, and n is the number of samples. For large n and/or large values of x_i, Eq. (1) may present numerical problems (lack of precision) unless the summation is performed with double precision arithmetic. An algorithm which is widely recommended is,

$$\bar{x} = \bar{x}_o + \frac{1}{n} \sum_{i=1}^{n} (x_i - \bar{x}_o) \tag{2}$$

where $\bar{x}_o$ is given by Eq. (1). This is a "two-pass" algorithm requiring two passes through the data to get $\bar{x}$. Thus, the increased accuracy is achieved with increased code complexity and computation time. For large values of n, computation time for Eq. (2) can be reduced with little sacrifice in accuracy by using a subset of x_i, say 100 samples, to calculate $\bar{x}_o$.

If the variance of the data set is constant as well as the mean, it can be shown[4] that the variance of $\bar{x}$ is given by

$$\mathrm{Var}[\bar{x}] = \frac{\mathrm{Var}[x]}{n} \tag{3}$$

where the samples are assumed to be independent, that is, not correlated. Since the standard deviation, σ, is equal to the square root of the variance, Eq. (3) can be written as,

$$\sigma_{\bar{x}} = \frac{\sigma_x}{\sqrt{n}} \tag{4}$$

The goal in performing the simple average is to obtain a good estimate of the population mean, μ. The standard deviation, σ_x, is not known, but we can estimate it with the sample standard deviation, s_x, defined by

$$s_x^2 = \frac{1}{n-1} \sum_{i=1}^{n} (x_i - \bar{x})^2 \tag{5}$$

where $\bar{x}$ can be the value obtained with Eq. (1), requiring two passes through the data to calculate s_x, or with Eq. (2), requiring three passes. Some texts recommend a single-pass version of Eq. (5), frequently called the "computational form," which is,

$$s_x^2 = \frac{1}{n-1} \left(\sum_{i=1}^{n} x_i^2 - \frac{\left(\sum_{i=1}^{n} x_i \right)^2}{n} \right) \tag{6}$$

This equation is generally considered to be a very poor numerical algorithm and should be avoided.

When only the sample standard deviation is known, the t distribution is used to define the uncertainty in the estimate of μ. Thus,

$$t = \frac{\bar{x} - \mu}{s_{\bar{x}}} = \frac{\bar{x} - \mu}{s / \sqrt{n}} \tag{7}$$

For the large number of samples usually averaged in a time series application, the t distribution approaches the normal distribution. If the sample set has a normal distribution, the means of subsets from this set will also have a normal distribution. Even if the sample set does not have a normal distribution, the central limit theorem[5] states: "Regardless of the distribution of the parent population (as long as it has a finite mean, μ, and variance, σ^2), the distribution of the means of random samples will approach a normal distribution (with mean μ and variance σ^2/n) as the sample size n goes to infinity." In fact, for most sample distributions encountered in experimental measurements, the distribution of $\bar{x}$ closely approximates a normal distribution for values of $n \geq 5$. From Eq.(7), for $100(1-\alpha)\%$ confidence in μ, the number of samples to average is given by,

$$n = \left(\frac{t_{(\alpha/2,n-1)}\, s_x}{\bar{x} - \mu} \right)^2 \tag{8}$$

Although s_x and t are not known until n is defined, s_x can be adequately estimated from the current or previous data sets and for large n, t is insensitive to n (within 1% of the normal distribution value for $n > 120$). For example, for a 95% confidence level that the relative error in μ, defined as $(\bar{x} - \mu)/\bar{x}$ does not exceed some level, ε_μ,

$$n \geq \left(\frac{1.96\, s_x}{\varepsilon_\mu \bar{x}} \right)^2 \tag{9}$$

It must be emphasized that Eq. (9) is valid *only* when the data samples are independent, i.e., uncorrelated.

As an example, consider the data set shown in Figure 1 which was generated with the function

$$x_i = 10.0 + R\,(0.0,\ 1.0)$$

where R is a set of normally distributed pseudo-random numbers with a mean of 0.0 and a standard deviation of 1.0. It is desired to estimate μ with no more than a 1% error, i.e., $\varepsilon_\mu = 0.01$. Estimates from the first 100 samples give $\bar{x} = 10.122$ and $s_x = 1.0733$. Assuming large n, and a 95% confidence level, Eq. (9) gives

$$n \geq \left[\frac{(1.96)\ (1.0733)}{(0.01)\ (10.122)} \right]^2 = 432$$

The average of the first 432 samples is 9.989 which is well within 1% of the input mean value.

CUMULATIVE AVERAGE

The cumulative, or running, average is less often used than the simple or moving average. It is most valuable when the data mean is needed, but the variance cannot be defined. This occurs when there is an initial transient in the data, or the data includes a periodic component. In these cases, the sample standard deviation calculated with Eq. (5) does not accurately represent the standard deviation of the noise (random) component and Eq. (9) cannot be used to estimate the number of samples to average because the data are not independent. The cumulative average provides a graphical method of estimating the mean of the data set. Unlike the simple average which produces a single output value, $\bar{x}$, the cumulative average produces an output time series, y, which has the same time interval, T, as the input series, x. The average is given by,

$$y_n = \frac{1}{n}\sum_{i=1}^{n} x_i \qquad ; n = 1, N \tag{10}$$

To avoid calculating the sum at each time, a recursion equation can be used,

$$\begin{aligned} y_n &= \frac{1}{n} S_n \\ S_n &= S_{n-1} + x_n \ ; \ S_o = 0.0 \end{aligned} \tag{11}$$

As an example of this type of average, data were generated with the model,

$$x_i = 10.0 + \sin(2\pi t) + R(0.0, 1.0)$$

where $t = (i-1)T$ and $T = 0.010$ sec. The data are shown in Figure 2 with the cumulative average. It is seen that the cumulative average predicts the mean with reasonable accuracy (1%) for $t > 2.5$ sec.

MOVING AVERAGE

Although the simple and cumulative averages smooth the data in the sense that they reduce the variance in a noisy dc signal, they cannot be used to reduce the noise which is compromising a time-varying signal. To achieve this, a moving average is used. The moving average is accomplished by averaging a subset of the series, n samples in length, advancing j samples and repeating the operation until the series is exhausted. Although n and j can be any integers, for experimental data n is usually taken to be odd with $j = 1$. To avoid a phase shift (time delay) in the averaging operation, the average value is assigned to the time corresponding to the middle of the subset. Finally, for purposes of the frequency response analysis in the next section, it is assumed that the time interval, T, between consecutive samples is a constant. It should be noted that if $j > 1$, the smoothed time series may be aliased. An example calculation for n=5 and j=1 is given below, where x_i is the input series and y_i is the output (filtered) series.

$$\begin{aligned} y_1 &= y_2 = \text{undefined} \\ y_3 &= \frac{1}{5}(x_1 + x_2 + x_3 + x_4 + x_5) \\ y_4 &= \frac{1}{5}(x_2 + x_3 + x_4 + x_5 + x_6) \\ y_5 &= \frac{1}{5}(x_3 + x_4 + x_5 + x_6 + x_7) \\ y_6 &= \ldots \end{aligned} \tag{12}$$

The algorithm for a general moving average is,

$$y_i = \sum_{k=-m}^{m} w_k x_{i-k} \quad ; i = 1+m, N-m \tag{13}$$

where

$$m = \frac{n-1}{2} \tag{14}$$

Equation (13) is the basic equation for a nonrecursive digital filter[6]. The weights, w_k, are usually called coefficients in filtering applications. For the present applications, the weighting function will be subject to a number of constraints. First, to insure that a time series consisting of constant values is not modified by the smoothing operation, it is necessary that,

$$\sum_{k=-m}^{m} w_k = 1.0 \tag{15}$$

Second, we will consider only moving averages with symmetrical weights, that is,

$$w_{-k} = w_k \tag{16}$$

Finally, we will always use an odd number of weights, as indicated in Eq. (14), so that the output value, y_i, can assume the same time value as the corresponding input value, x_i. This gives a zero-phase filter.

Weighted moving averages will be discussed in a later section. For a simple moving average, the weights are constant and equal to $1/n$ as shown in the $n=5$ example above. These weights satisfy the constraints given above. From Eq. (13) and the example, it is seen that the first and last m values from the series are lost in the averaging process. Since moving averages are most often used with large data sets, this is not usually a problem. If it is, the data set can be extended with m points at each end. Schemes for extending the data at the start of the series include: (1) odd symmetry $[x_{-i} - x_1 = -(x_i - x_1)]$; (2) constant values $(x_{-i} = x_1)$; and (3) even symmetry $(x_{-i} = x_i)$. A similar extension is performed at the end of the series. All are artificial and should be avoided if possible.

The moving average is excellent for removing a specific frequency component and its harmonics from a complex signal. Thus, it behaves as a notch filter. It can also be used to smooth data, that is, as a lowpass filter with noise variance reduction given by Eq. (3). Finally, it can be used as a highpass filter to remove the low frequency trend in a signal. Examples of the second and third applications are shown in Figure 3. The input signal was a smooth trend line with superimposed random noise ($\sigma = 0.2$) and a sampling interval of 0.001 sec. Figure 3a shows a composite of the original signal (x) and the smoothed output (y) of a simple moving average with $n = 51$. Although the original trend can never be perfectly recovered after random noise is added, the moving average did greatly reduce the noise with only minor distortion of the input signal. The input series was generated for $t = 0$ to 1.0, note that 25 samples are lost at each end of the output series.

Low frequency trends such as the one in this example cause problems with low-frequency resolution in power spectral density (PSD) calculations. The moving average can be used as a highpass filter to remove the trend. The highpass filtering operation is accomplished simply by subtracting the average from the original data, that is,

$$y_i' = x_i - y_i \tag{17}$$

where y_i' is the highpass filter output. The results are shown in Figure 3b and would permit a much better PSD calculation of the frequency content of the noise than the original series shown in Figure 3a.

TIME DOMAIN EFFECTS

Averaging has a number of effects on the data set in the time domain. First, the data are smoothed, that is, the variance is reduced. Assuming that the noise component is random, the reduction in variance of the noise is given by [7]

$$V[y_R] = V[x_R] \sum_{k=-m}^{m} w_k^2 \tag{18}$$

where x_R and y_R are the noise components in the input and output series, respectively. That is, the variance is reduced by the sum of the weights squared. For a simple moving average, $w_k = 1/n$ and Eq. (18) reduces to Eq. (3).

Second, averaging can cause a time delay in the output. For the simple and cumulative averages, it is assumed that the desired output of the averaging process is a constant and time delay is not a concern. For the moving average, a time delay makes it difficult to compare the input and output signals, but the delay is eliminated by placing the average value at the time corresponding to the midpoint of the averaged subset as in Eq. (13).

Finally, when a moving average is applied to a time series, the output series is correlated by the averaging process. Examination of Eq. (12) shows that y_3 and y_4 have four common input samples in the two subsets which are averaged. Outputs y_3 and y_5 have three common input samples, and so on. Thus, for a simple moving average, there is a linear correlation in the output series which extends for n samples. Correlation of the output series has three consequences. First, the appearance of the output series is modified. The random component, when smoothed, may appear as an oscillatory component. This effect is known as the Slutsky-Yule effect[8,9] and produces what appears to be a periodic signal. However, the oscillations have random period and amplitude. The experimentalist should be aware of this phenomenon and not attribute significance to the oscillations. Second, the autocorrelation function of the output series will be modified. Third, the correlation reduces the effect of any subsequent smoothing operations. Equation (18) applies only for independent data, and if the output series is smoothed a second time, the reduction in variance is much less than that given by the equation.

FREQUENCY DOMAIN EFFECTS

Frequency domain effects are of most interest for moving averages. However, an understanding of the filter characteristics is valuable even for the simple average since they define how frequency components in the input series are altered by the smoothing operation. It is beyond the scope of the present work to develop the theory necessary to understand filtering theory. However, the definition of a frequency response function (also called a transfer function) is essential to understanding the effects of averaging on a time series. The equation for a moving average, Eq. (13), is also the definition of a discrete convolution of the two time functions, w(t) and x(t), and is written symbolically as,

$$y(t) = w(t) * x(t) \tag{19}$$

By the convolution theorem,

$$Y(f) = W(f) \cdot X(f) \tag{20}$$

where Y(f), W(f), and X(f) are the Fourier transforms of y(t), w(t), and x(t), respectively. The transform W(f) is called the frequency response function and defines how the averaging operation effects the signal component at a specific frequency. That is,

$$W(f) = \frac{Y(f)}{X(f)} \tag{21}$$

In general, W(f) is a complex function. However, for real, symmetrical weights, W(f) is also real. The filtering operation described by Eq. (13) is linear and does not change the frequency of an input sinusoidal wave. However, the amplitude and phase are changed. For example, if the input signal is defined by,

$$x = A\sin(2\pi ft + \varphi) \tag{22}$$

the output signal is,

$$y = A'\sin(2\pi ft + \varphi') \tag{23}$$

The magnitude, |W(f)|, of the frequency response function is called the amplitude response function or, gain factor, and defines the output-to-input amplitude ratio, A'/A,

$$\frac{A'}{A} = |W(f)| \tag{24}$$

For simple or weighted moving averages, the phase difference, $\varphi' - \varphi$, is equal to 0 or 180 deg, depending upon frequency.The frequency range over which these relations apply is defined by sampling theory and is $f = 0$ to f_N, the Nyquist frequency, which is defined by,

$$f_N = \frac{1}{2T} \quad (25)$$

where T is the sampling interval of the time series. For arithmetic averaging with real, symmetrical weights, w_k, the frequency response function is given by[10],

$$W(f) = w_o + 2\sum_{k=1}^{m} w_k \cos(2\pi kfT) \quad (26)$$

For the simple moving average, $w_k = 1/n$, and Eq. (26) can be evaluated[11] to give,

$$|W(f)| = \left|\frac{\sin(n\pi fT)}{n\sin(\pi fT)}\right| \quad ; \quad f = 0 \text{ to } f_N \quad (27)$$

where $|W(0)| = 1.0$. A plot of W(f) for a simple moving average with selected values of n is shown in Figure 4. Several features of moving averages are apparent in the figure. First, the average acts as a lowpass filter and passes dc ($f = 0$) signals unchanged. Also, the gain is zero at frequency f_o so averaging provides a notch filter. This frequency, f_o, is also called the mainlobe width, and subsequent regions between zeros are called sidelobes. The mainlobe and even-numbered sidelobes have positive gain and zero phase. The odd-numbered sidelobes have negative gain or 180 deg phase shift. Obviously, the phase shift will distort the output waveform relative to the input waveform. However, the series at frequencies greater than f_o is considered to be noise and the phase shift is not significant. As is the case with all digital filters, the frequency response function is periodic and $W(f) = 1$ at $f = 2f_N = 1/T$ and multiples of this frequency.Therefore, any signal component which has a frequency of 1/T will not be altered by the averaging process. This problem is avoided by using an analog anti-aliasing filter during data acquisition to attenuate frequency components at $f \geq f_N$.

Examination of Eq. (27) shows that for $|W(f)|$ to equal zero at the notch frequency, f_o, then,

$$f_o nT = 1 \quad (28)$$

Equation (28) is fundamental to moving average smoothing and provides a means of collapsing the gain curves for arbitrary n into a single curve by using the frequency parameter fnT as the independent variable instead of f. This is shown in Figure 5. Note that for a given value of n, this curve is valid only to $f = f_N$ or for $fnT \leq n/2$. For $n > 5$, the curves are nearly coincident for $f = 0$ to f_o ($fnT = 0$ to 1). Rearranging Eq. (28) gives

$$n = \frac{1}{f_o T} \quad (29)$$

This equation permits easy selection of the number of samples to be used in a simple moving average to eliminate a particular frequency and its harmonics. It should be noted that the amplitude response function given by Eq. (27) also applies to the simple average. For the example of Figure 3, with $n = 432$, the notch frequency is $f_o T = 0.0023$ which is very low ($f_N T = 0.5$). This is typical for simple averages. The author has found that a simple average with $n = 10$ and $T = 0.00167$ sec ($f_o \approx 60$ Hz) is very effective at removing residual ac noise which enters the data acquisition system after the analog filter section.

Although the simple moving average is a very effective notch filter and does attenuate high frequency (noise) components in a signal, it is not a "great" lowpass filter for two reasons. First, it attenuates all signal components at frequencies above zero. This means that any nonconstant signal will be modified by the moving average. Typical values of the amplitude ratio for a simple moving average are given in the table below.

$\lvert W(f) \rvert$	fnT
0.99	0.078
0.95	0.176
0.90	0.250
0.71 (-3db)	0.442

Second, the sidelobes have a higher gain than is desired and, therefore, the moving average does not reduce high frequency components as much as the user may desire. The first problem is best addressed by using a weighted moving average which has been designed using digital filter techniques. This is beyond the scope of this paper, but can be found in many digital filter texts, including Ref. 12. The problem of reducing sidelobes can be handled by two approaches. First, the output series can be smoothed a second time using the same or a different value of n. For the same value of n, this operation gives an effective frequency response function which is equal to the square of the basic function. Thus, the first sidelobe amplitude, which is equal to 0.217 for large n, will be reduced to 0.047 after the second average. Unfortunately, frequencies in the passband will also be attenuated. A better approach is to use a weighted average as will be discussed in the next section.

WEIGHTED MOVING AVERAGES

A general weighted moving average is defined by Eq. (13). The constant weights of the simple moving average give a weight function which is discontinuous at each end, that is, the function is a rectangle. To improve the frequency response function, it is necessary to replace this weight function with one which decreases smoothly from $k = 0$ to $k = \pm m$. These weight functions are called windows in filtering and spectral analysis applications and are widely discussed in the literature[13,14]. The relative merits of various windows is beyond the scope of this paper. A suggested window, which combines simplicity with effectiveness, is the Hann or cosine-squared window. The weight function for this window is,

$$w_k' = \frac{1}{2}\left[1 + \cos\left(\frac{\pi k}{m}\right)\right] = \cos^2\left(\frac{\pi k}{2m}\right) \qquad (30)$$

Equation (30) is the form of the window function as it is usually given and the sum of the weights is not equal to unity. The proper form to use with a moving average is,

$$w_k = \frac{1}{m} w_k' \qquad (31)$$

The Hann weights for n = 51 are plotted in Figure 6 with the rectangular weights used for a simple moving average. When implementing a weighted moving average in a data analysis code, the symmetry of the weights, that is, $w_{-k} = w_k$, should be used to reduce storage and computation time.

The amplitude response function for this weighted moving average is compared in Figure 7 to that for a simple moving average with n = 51 in both cases. The sidelobes are greatly diminished for the weighted average, but the notch frequency has doubled. Equation (29) provided a means for estimating n for a simple moving average when f_0 and T are given. An equivalent equation for the Hann weighting function is

$$n = 1 + \frac{2}{f_o T} \tag{32}$$

Therefore, for the same f_0, smoothing with the Hann window will require approximately twice the number of weights that the simple moving average requires. By increasing n to 101 for the Hann window, the amplitude response functions shown in Figure 8 are obtained. This figure shows the greatly reduced sidelobe amplitudes for the Hann window and emphasizes that the improved filter is obtained at the expense of increased computation time since n is doubled. Also, twice as many samples are lost at each end of the series. The reduction in variance achieved by each of these moving averages is defined by Eq. (18) as the sum of the weights squared. For these three averages, the reduction was: simple average (n = 51), $V[y]/V[x] = 0.0196$; Hann-weighted average (n = 51), $V[y]/V[x] = 0.0300$; and Hann-weighted average (n = 101), $V[y]/V[x] = 0.0150$. For n = 51, even thought the Hann average reduced the sidelobe amplitudes, it greatly widened the mainlobe width resulting in less reduction in variance. As would be expected, for a given f_0 the Hann-weighted average resulted in a greater variance reduction than the simple average. It was mentioned earlier that the simple moving average eliminates data components at the notch frequency and its harmonics. The Hann average also has this characteristic, although many other window functions do not. Returning to the example of Figure 3, the input series is filtered with a Hann weighted moving average (n = 101) and the results are shown in Figure 9. Note that 50 samples are lost at each end of the output series in this case. The improvement in the smoothed trend is obvious.

CONCLUSIONS

The simple arithmetic average can be used to define a signal level, in the presence of uncorrelated noise, within specified error limits if the signal is not varying with time. An equation was provided which allows the user to easily estimate the number of samples to be averaged for a given error level and probability. For a varying signal, the cumulative average provides a graphical solution to estimating the mean. Moving averages, both simple and weighted, provide a versatile technique for smoothing and trend removal. These averages, which are nonrecursive digital filters, are simple to use and do not require the elaborate design and implementation software associated with recursive filters. Furthermore, they can be more stable than recursive filters, since they do not require feedback. A basic equation was developed that allows the user to accurately define the number of samples to be averaged for a specified filter passband. Finally, a weighting function was suggested which greatly improves frequency rejection in the filter stopband without complicating the averaging operation.

REFERENCES

1. Harnett, D. L., *Statistical Methods*, 3rd Ed., Section 6.4, Addison-Wesley, Reading, MD, 1982.
2. Stearns, S. D. and Hush, D. R., *Digital Signal Analysis*, 2nd Ed., Section 8.8, Prentice Hall, Englewood Cliffs, NJ, 1990
3. Meyer, S. L., *Data Analysis for Scientists and Engineers*, p. 23, John Wiley & Sons, New York, 1975.
4. Harnett, *loc. cit.*
5. Harnett, *op. cit.*, Section 6.6
6. Stearns and Hush, *op. cit.*, Section 8.2.
7. Hamming, R. W., *Digital Filters*, Section 1.7, Prentice Hall, Englewood Cliffs, NJ, 1977.
8. Kendall, M., Stuart, A., and Ord, J. K., *The Advanced Theory of Statistics, Vol. 3, Design and Analysis, and Time-Series*, 4th Ed., Section 46.18, Oxford University Press, New York, 1983.
9. Kendall, M. and Ord, J. K., *Time Series*, 3rd Ed., Example 5.5, Oxford University Press, New York, 1990.
10. Stearns and Hush, *op. cit.*, Section 8.3
11. Koopmans, L. H., *The Spectral Analysis of Time Series*, Example 6.2, Academic Press, New York, 1974.
12. Stearns and Hush, *op. cit.*, Section 8.4.
13. Stearns and Hush, *op. cit.*, Section 8.5.
14. Harris, F. J., "On the Use of Windows for Harmonic Analysis with the Discrete Fourier Transform," *Proceedings of the IEEE*, Vol. 66, No. 1, pp. 51-83, January 1978.

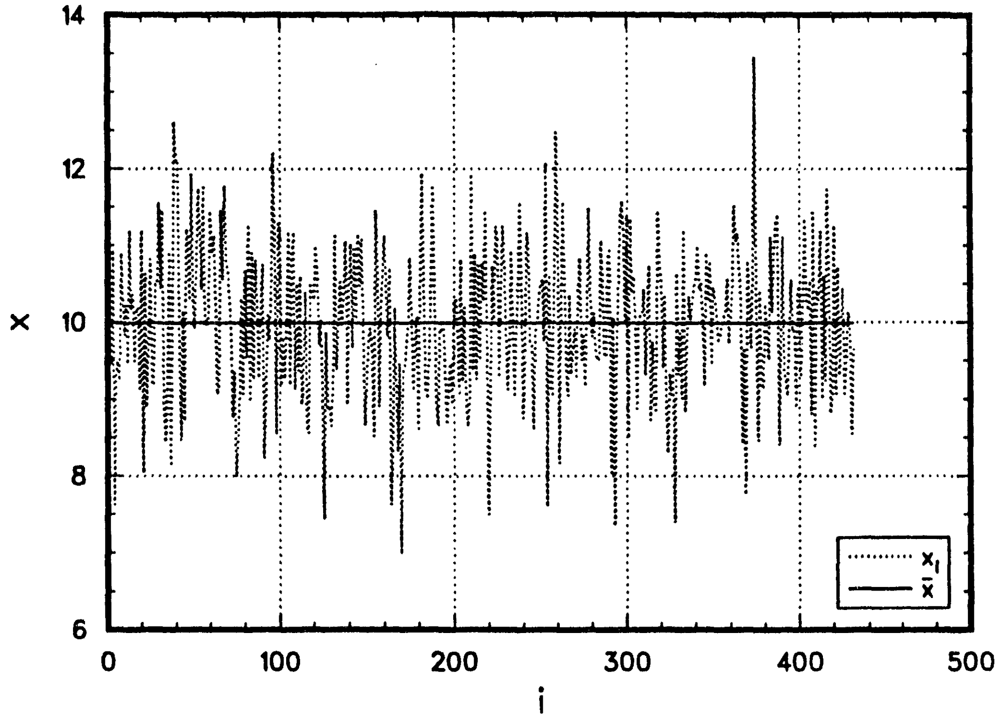


Figure 1. Example of simple average

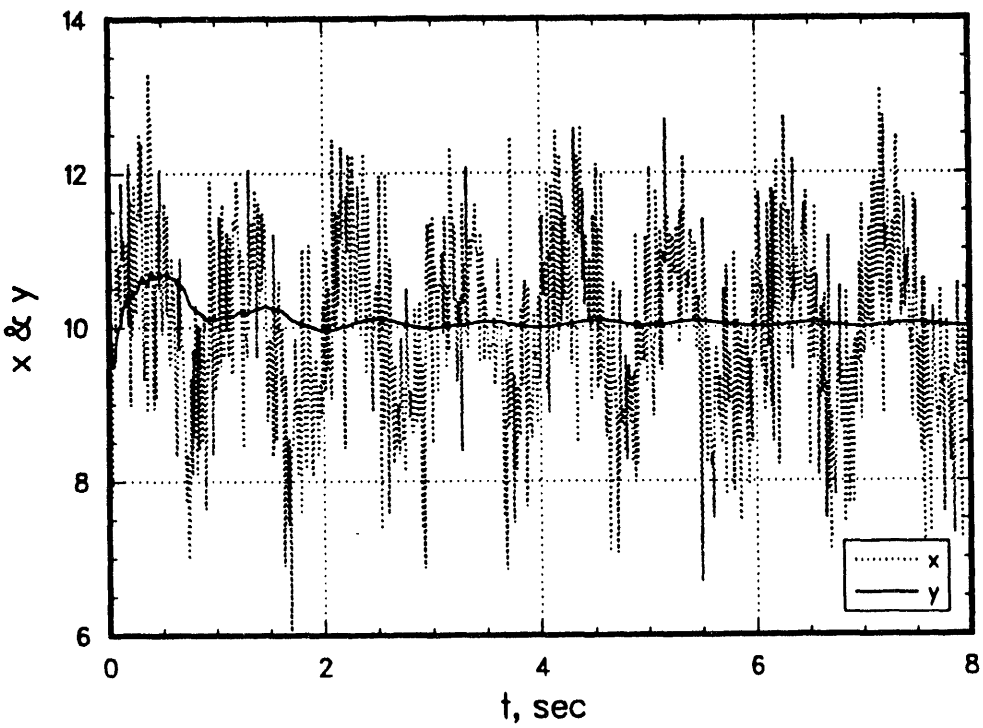
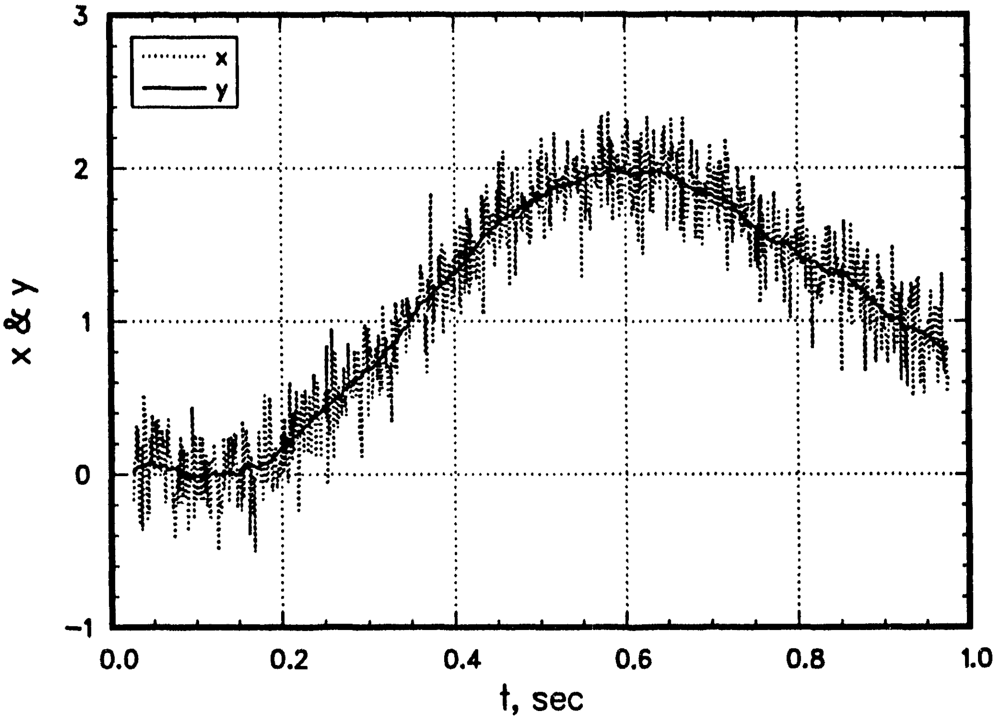
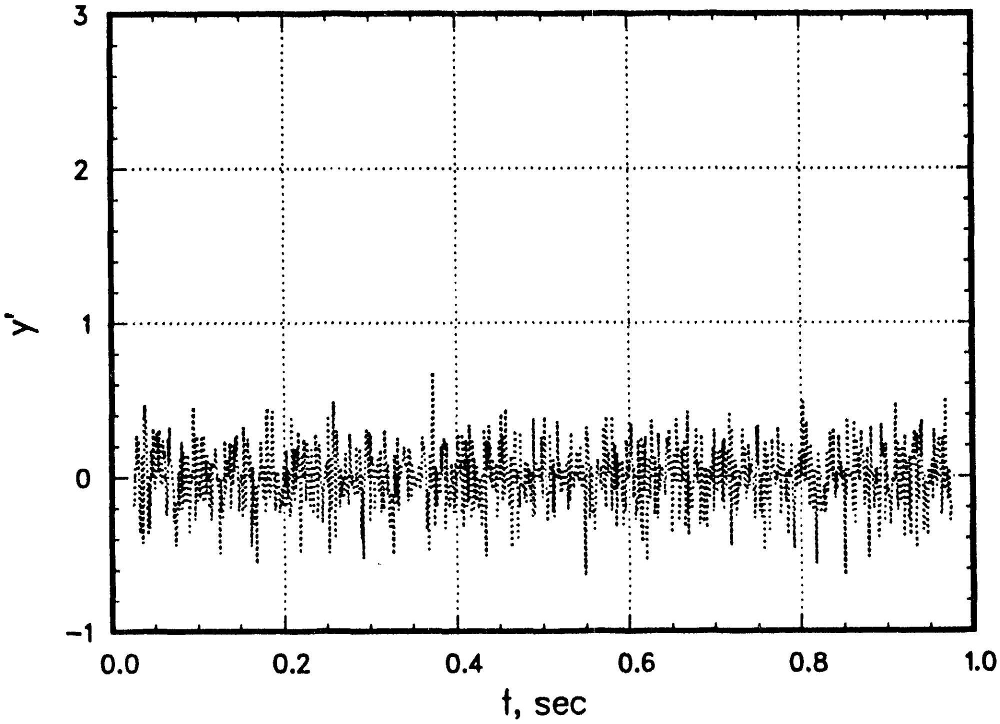


Figure 2. Example of cumulative average



a. Smoothing (lowpass filter)



b. Trend removal (highpass filter)

Figure 3. Smoothing and trend removal with a simple moving average, n = 51

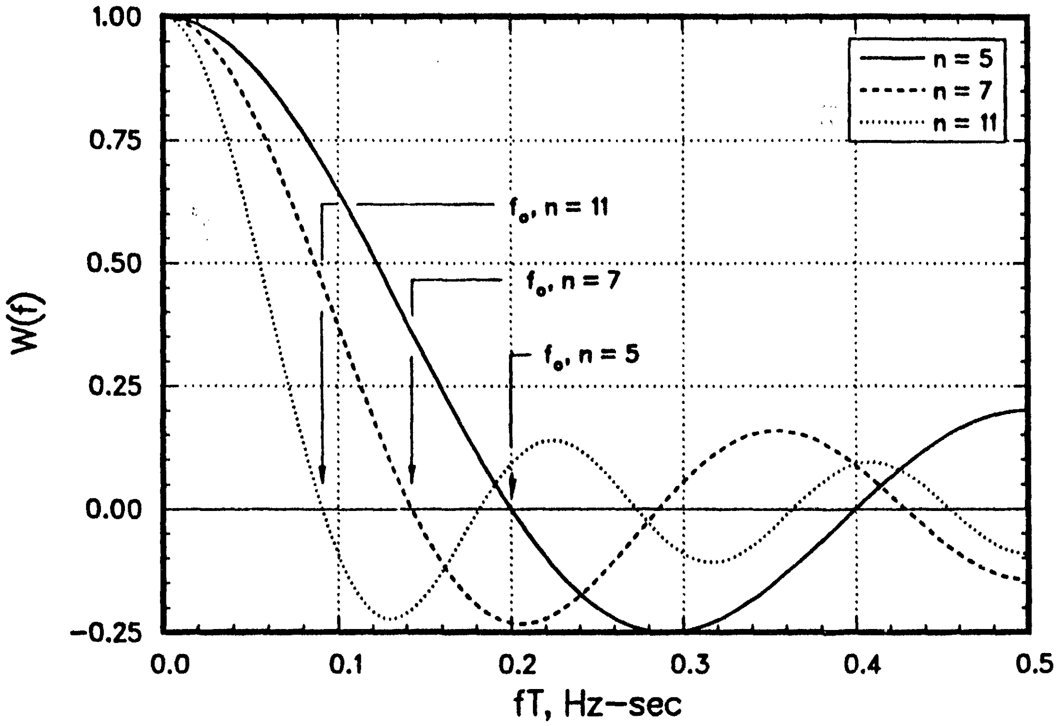


Figure 4. Frequency response functions for simple moving averages

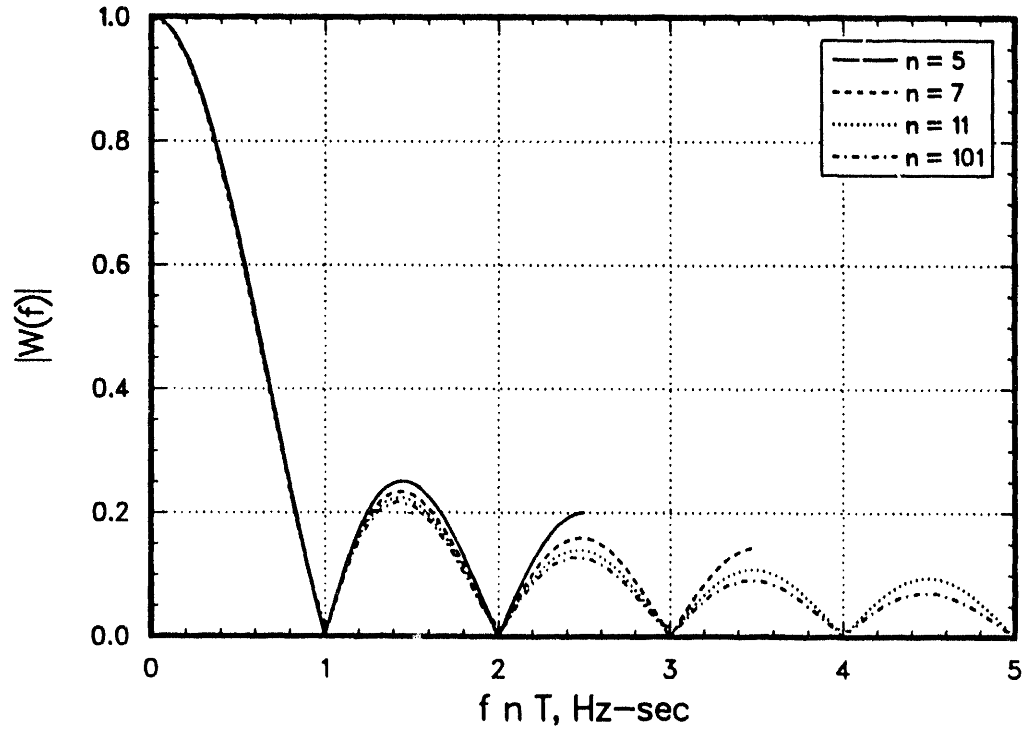


Figure 5. Amplitude response function for simple moving averages with frequency parameter, fnT

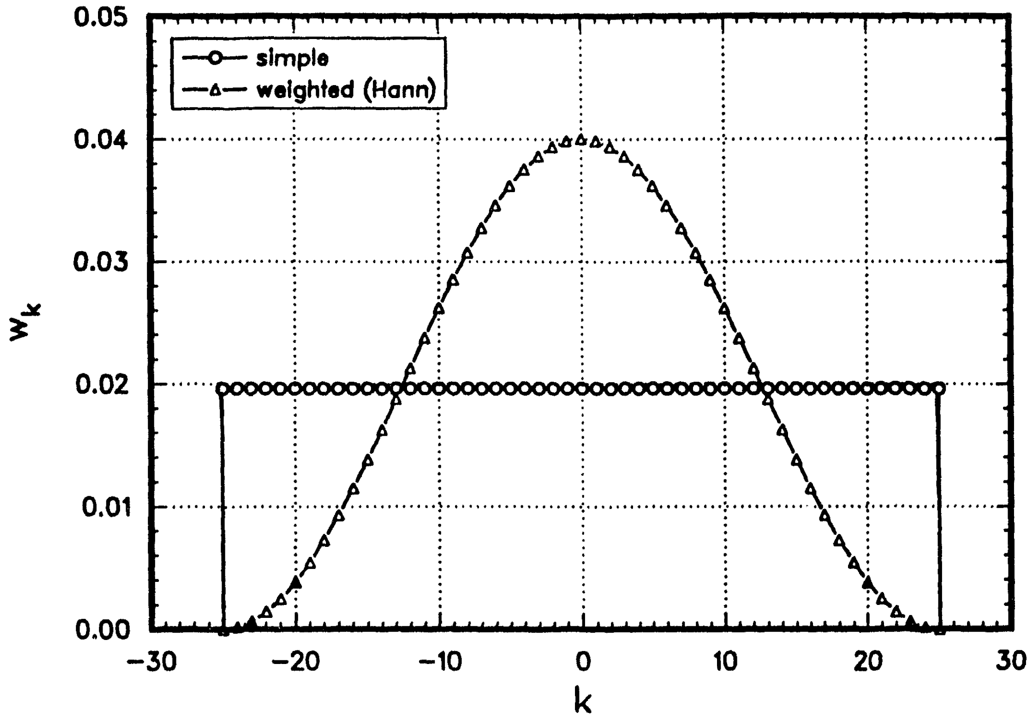


Figure 6. Weight functions for simple and weighted moving averages, n = 51

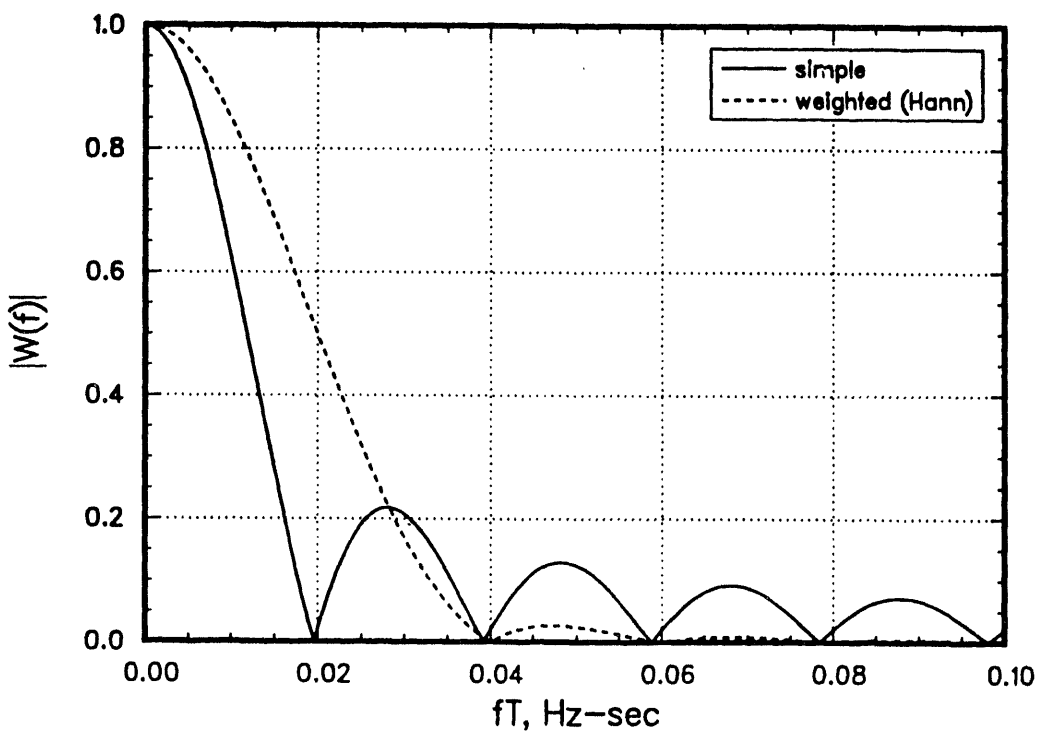


Figure 7. Amplitude response functions for simple and weighted averages, n = 51

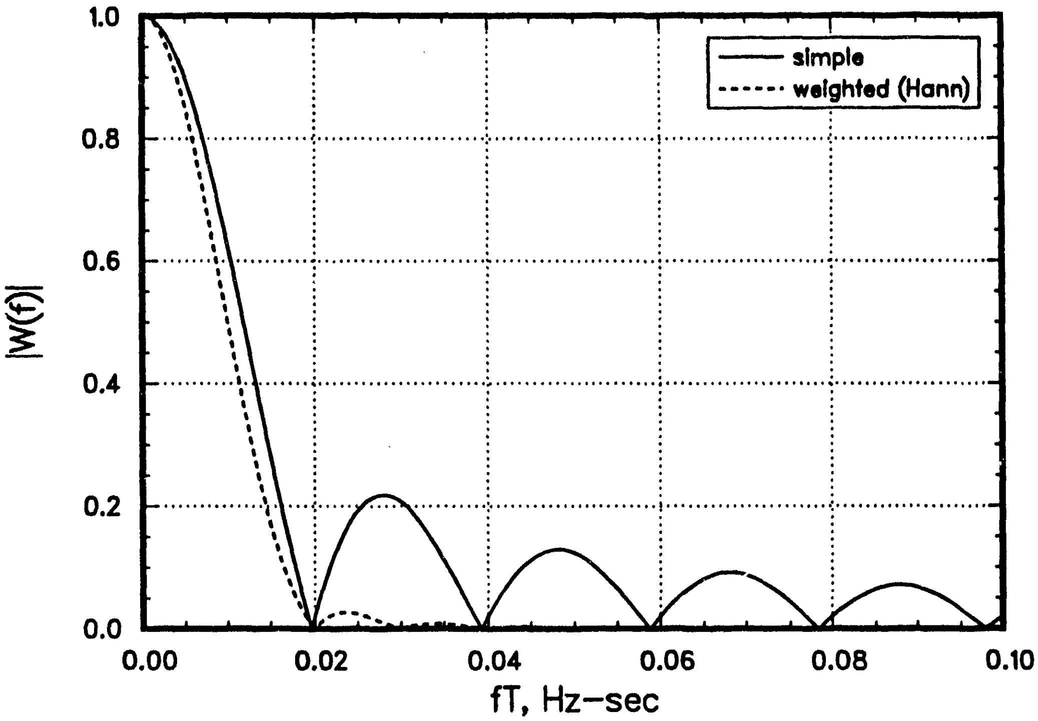


Figure 8. Amplitude response functions for simple (n = 51) and weighted averages (n = 101)

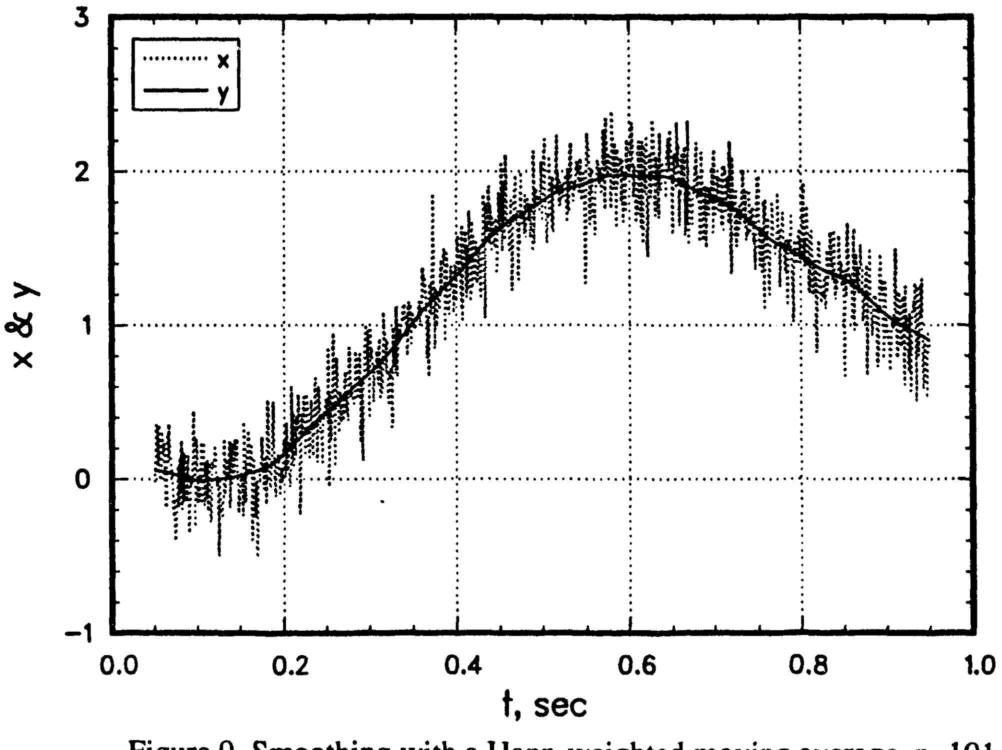


Figure 9. Smoothing with a Hann-weighted moving average, n=101

DATE FILMED

4/7/94

END