Signal Processing in Analytical Chemistry

Peter D. Wentzell and Christopher D. Brown

in
Encyclopedia of Analytical Chemistry
R.A. Meyers (Ed.)
pp. 9764–9800

Signal Processing in Analytical Chemistry

Peter D. Wentzell and Christopher D. Brown
Dalhousie University, Halifax, Canada

Signal processing refers to a variety of operations that can be carried out on a continuous (analog) or discrete (digital) sequence of measurements in order to enhance the quality of information it is intended to convey. In the analog domain, electronic signal processing can encompass such operations as amplification, filtering, integration, differentiation, modulation/demodulation, peak detection, and analog-to-digital (A/D) conversion. Digital signal processing can include a variety of filtering methods (e.g. polynomial least-squares smoothing, differentiation, median smoothing, matched filtering, boxcar averaging, interpolation, decimation, and Kalman filtering) and domain transformations (e.g. Fourier transform (FT), Hadamard transform (HT), and wavelet transform (WT)). Generally the objective is to separate the useful part of the signal from the part that contains no useful information (the noise) using either explicit or implicit models that distinguish these two components. Signal processing at various stages has become an integral part of most modern analytical measurement systems and plays a critical role in ensuring the quality of those measurements.

1 INTRODUCTION

The reliability of analytical results is vitally dependent on the quality of the measurements leading to their determination. Signal processing refers to a variety of operations that can be carried out on a continuous or discrete sequence of measurements in order to enhance the quality of information they are intended to convey. The term 'signal' is ordinarily applied to a sequence of measurements that are related by some ordinal variable, such as time or wavelength, and usually obtained via electrical transduction. Operations can be carried out on a continuous electrical signal (analog signal processing) or on discretely sampled numerical values (digital signal processing). Generally, the objective is to separate the desired part of the signal (the pure signal, which is correlated to some physical or chemical property of interest) from the unwanted part of the signal, or noise. Often, the term 'signal processing' implies that operations are carried out in real time, or as the data are acquired, but this is not a requirement. The distinction between signal processing and other forms of data analysis is often open to interpretation, but usually signal processing emphasizes alternative representations of the sequence of measurements as opposed to the direct extraction of secondary information such as analyte concentrations or chemical structures.

The goal of this article is to provide an overview of signal processing methods used in analytical chemistry with an emphasis on their capabilities, weaknesses, and practical implementation. Although both analog and digital signal processing are discussed, a much greater emphasis is placed on the latter because of its greater relevance to the practicing analytical chemist. Owing to the scope of the subject area, some topics will no doubt be neglected or underemphasized, but an attempt has been made to balance coverage with the importance to the field.

Encyclopedia of Analytical Chemistry
R.A. Meyers (Ed.)

2 OVERVIEW

2.1 History

Historically, signal processing in analytical chemistry can be regarded as originating with the first quantitative analytical measurements, such as the end-point of a titration, the potential of an electrochemical cell, or the absorbance of a solution at a single wavelength. Such scalar quantities are now sometimes referred to as zero-order measurements, with reference to the fact that a scalar is a zero-order tensor. Rudimentary signal processing consisted of averaging replicate measurements or damping the response of electrical signals, but these were rather simplistic approaches when compared to today's more sophisticated methods.

The evolution of modern signal processing in analytical chemistry can be traced to three parallel developments. The first was the emergence of analytical instruments capable of producing first-order data, or a vector (first-order tensor) of measurements. The development of instruments such as chromatographs and scanning spectrometers meant that the signal from an instrument could no longer be considered static, but rather changed in a regular fashion with some variable such as time or wavelength, thereby requiring more flexible signal processing methods that could remove the noise without distorting the pure signal. A second important influence was the appearance of the analog-to-digital converter (ADC) and the computers that drove them. This opened the door to more versatile digital signal processing methods. Finally, there was the parallel development of efficient digital signal processing algorithms, such as the Savitzky–Golay (SG) implementation of polynomial least-squares filters,[1] and the Cooley–Tukey algorithm for the fast Fourier transform (FFT),[2] which are still among the most highly cited papers in the literature. Such algorithms made the practical implementation of signal processing methods a reality.

Although first-order instruments are still the mainstay of analytical chemistry, second-order instruments which provide a matrix of data, such as chromatographs with multichannel detection and tandem mass spectrometers, are now routinely employed, and higher-order instruments are commonplace. These, combined with ever more powerful computational platforms, have advanced modern signal processing to yet another level.

2.2 Definitions and Notation

Throughout this article, the term 'signal' will be used to refer to either a continuous or discrete measurement sequence which consists of a pure or undistorted signal corrupted by noise. The signal is implicitly measured as a function of some other variable which will be referred to as the ordinal variable because it correlates directly with the sequence order. Traditionally, this variable is time, but other variables such as wavelength or applied voltage (which may or may not be correlated with time), can be employed without loss of generality. References to representation of the signal in the 'time domain' will therefore refer to this the original measurement sequence even if the ordinal variable is not time. Likewise, references to 'frequency domain' representations (section 3) will refer to a FT (section 6) into the inverse domain of the ordinal variable.

The term 'noise' is used to refer to unwanted fluctuations from the pure signal that obscure its measurement. This definition is quite general and means that what is considered noise can vary with the situation. If a signal consists of contributions from two sources, A and B, then B is considered noise if one is looking for A, and vice versa. Noise, especially random noise, is often characterized by its root-mean-square (rms) amplitude (continuous signals) or by its standard deviation (discrete signals).

Most of this article concerns digital signal processing methods where the signal is a vector of discrete measurements, usually made on equal intervals of the ordinal variable referred to as the sampling interval, t_s. The reciprocal of this interval will be referred to as the sampling frequency, f_s. (Again, analogous definitions hold when the ordinal variable is not time.) In vector notation, the signal $\mathbf{x}$ is the result of combining the pure signal vector $\mathbf{x^0}$ with the noise vector $\mathbf{e}$ (Equation 1):

$$\mathbf{x} = \mathbf{x^0} + \mathbf{e} \tag{1}$$

Throughout this article, boldface lower case letters are used to represent vectors (assumed to be column vectors unless otherwise indicated). Scalar quantities will be represented by lower case italic letters and matrices by boldface upper case letters. Transposes of vectors or matrices will be indicated with a superscript 'T' and inverses of matrices with a superscript '-1'. The identity matrix will be designated '$\mathbf{I}$'.

3 SIGNALS AND NOISE

3.1 Signals and Signal Domains

The features that distinguish a signal from a random series of measurements are: the measurements have a definite order, and the measurements are generally correlated in the time domain. For a discrete signal undistorted by noise, $\mathbf{x^0} = [x_1^0\, x_2^0\, x_3^0 \ldots]$, the correlation of two measurements is described by their covariance (Equation 2),

$$\mathrm{cov}(x_1^0, x_2^0) = \mathrm{E}(x_1^0 x_2^0) \tag{2}$$

where 'E' denotes the expectation value. To say that two measurements are correlated means that knowing one allows us to say something about the other. Although the covariance among measurements in a sequence is not generally known, some knowledge of its characteristics may be known, and it is these characteristics that are often used to distinguish the pure signal from the noise. For example, if the pure signal changes relatively slowly, it exhibits long-range correlations which may not be present in the noise.

Although signals are usually presented in the time domain, the same information can be conveyed by transforming them in alternate domains. The most useful of these is the frequency domain. For a continuous signal, this transformation can be accomplished by using a spectrum analyzer, a device that consists of a continuous series of electronic bandpass filters. The signal, which must be repetitively applied to the input of the device, is filtered by the spectrum analyzer and the rms signal at each frequency is determined. In essence, the power of the signal is plotted as a function of frequency to give a power spectrum. In the age of discrete signals, this approach is rarely used anymore, instead being replaced by the discrete FT, which is the digital counterpart of the analog spectrum analyzer. FTs are discussed in more detail in section 6 but, because of their importance to signal processing, they are introduced here. All of the information conveyed in the original signal is carried in the FT, but it is represented in the frequency domain rather than the time domain. This allows the composition of the signal in terms of sinusoidal frequencies to be analyzed, providing more direct information about the correlations in the time domain. Slowly varying signals in the time domain will have significant low-frequency components, and the shape of the power spectrum conveys important information about optimal signal processing methods. Figure 1 shows some simple signals and their FTs. The first example shows that a pure sinusoid gives rise to a single peak in the frequency domain. In contrast, the sharp edges of the square wave in the second example lead to high-frequency components in the FT. The last two examples demonstrate that more slowly varying signals have fewer high frequency components.

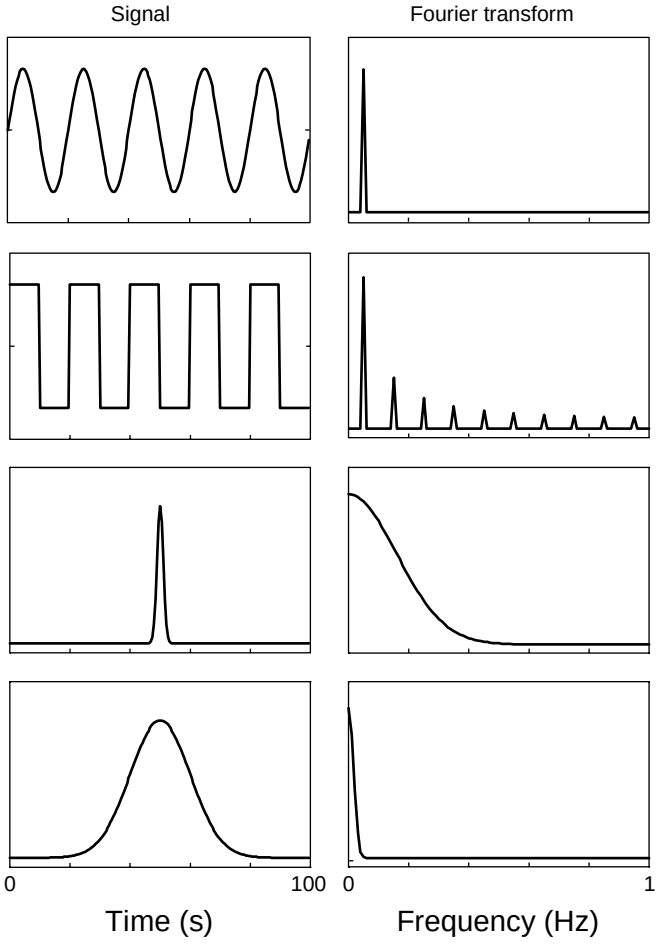

Figure 1 FTs of some simple signals.

3.2 Noise

The noise in an analytical signal can be classified in a number of ways, including (a) its distribution, (b) its source, (c) its characteristics in the time domain, and (d) its characteristics in the frequency domain. Because different classifications are used in different situations and they are not all mutually exclusive, it is necessary to understand the different cases and how they relate to signal processing. In doing so, it is helpful to imagine a signal from which we could subtract the pure signal component leaving only the noise or, alternatively, a situation in which the pure signal is zero so that we are measuring only noise, as shown in Figure 2(a).

If we were to plot a histogram of the magnitude of the noise for a large number of measurements, we might obtain a distribution such as that shown in Figure 2(b). By far the most common noise distribution (assumed or measured) for analytical measurements is the normal, or Gaussian, distribution, shown by the solid line in the figure. The reason for this is the central limit theorem which, simply put, states that if a measurement is the sum of a series of values drawn from arbitrary distributions, the distribution of the measurement will approach a normal distribution as the length of the series approaches infinity. As, in an analytical instrument, the observed noise is a consequence of many smaller random events, the central limit theorem can be rationalized to hold. Other noise distributions (e.g. uniform, log-normal) are also observed but are much less common. One other type which is common, however, is the Poisson distribution, which is observed in cases where the signal arises from

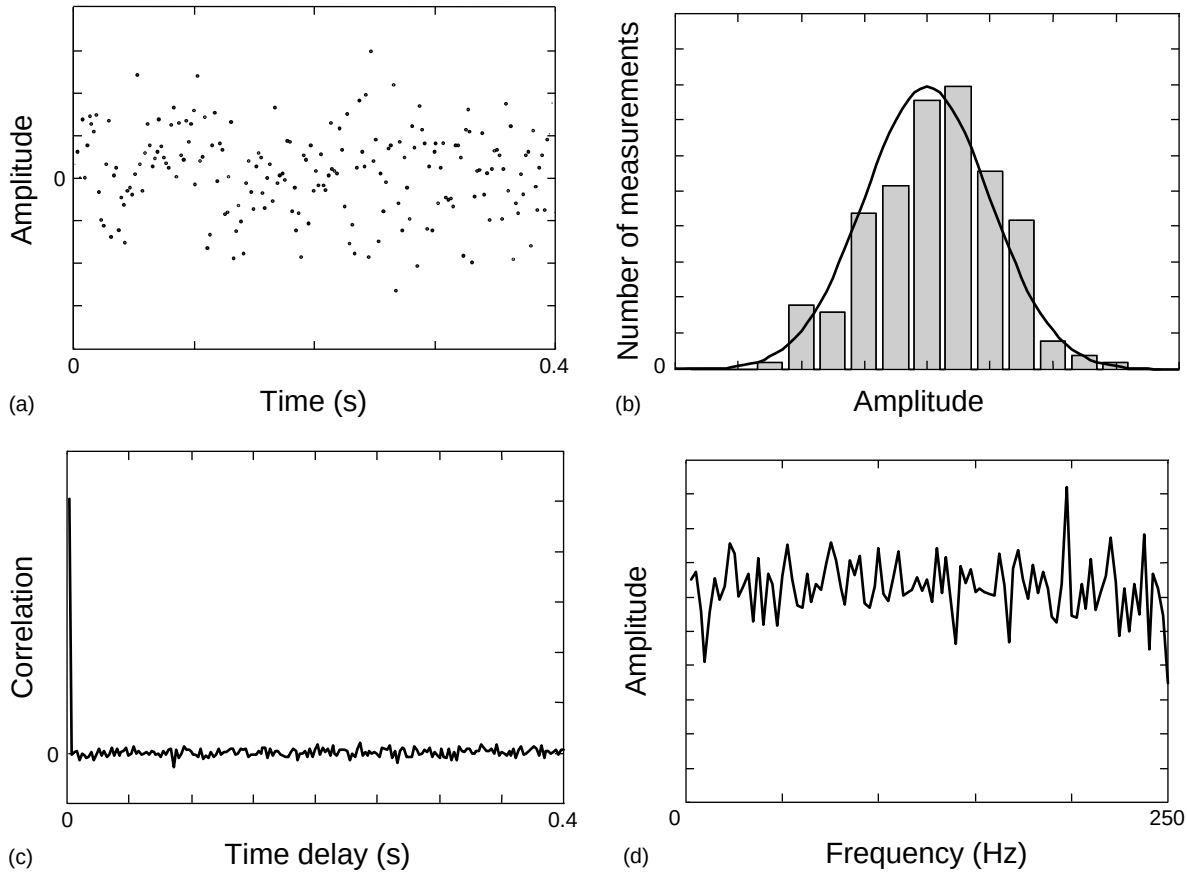

Figure 2 Representations of white noise: (a) noise sequence in the time domain, (b) distribution, (c) autocorrelation function, (d) FT.

a collection of discrete events, such as photons striking a photomultiplier tube. However, in reality a histogram constructed from Poisson noise would look essentially the same as the normal distribution in most cases. The distinction is in how the magnitude of the noise (i.e. its standard deviation) changes with signal intensity. The standard deviation of Poisson noise will increase with the square root of the signal intensity. To say that noise has a normal distribution, however, does not imply anything about how its magnitude changes with signal intensity. In this sense, the Poisson distribution can be regarded as a special case of the normal distribution.

In some cases noise is classified according to its dominant source. Such classifications often imply information about the distribution or temporal characteristics of the noise. For example shot noise, also called Schottky noise or quantum noise, arises in detectors based on discrete events, such as photons striking a photomultiplier, and exhibits a Poisson distribution. Shot noise is a type of fundamental noise because it originates from the random statistical nature of the events themselves and not from any deficiencies of the instrument. This type of noise dominates in cases where the number of events is relatively small, such as in fluorescence measurements. Johnson noise is another type of fundamental noise that arises from the random thermal motion of electrons in resistors. Flicker noise is considered to be a type of non-fundamental or excess noise in which the magnitude of the noise is directly proportional to the signal amplitude, hence it is often referred to as proportional or multiplicative noise. Flicker noise is often associated with variations in source intensity in absorption spectroscopy and can have distinctive frequency characteristics (see below). Other types of noise include interference noise (electrical, optical or other interferences that arise at specific frequencies, such as 60 Hz line interference), detector noise (a general term referring to instruments in which the limiting noise, such as shot noise or thermal noise, occurs at the detector), amplifier-readout noise (the noise observed for the readout circuitry when the input signal is zero) and quantization noise (observed when measurement precision is limited by the A/D conversion step).

In the time domain, noise can be classified in two ways: correlated and uncorrelated. Uncorrelated, or independent, noise implies that the noise observed at one point in the series of measurements is not related

in any way to that at other points. If we consider the sequence of noise to be represented by the vector [e_1 e_2 e_3 ...], then the covariance between the first two noise elements can be represented by Equation (3),

$$\sigma_{12} = \mathrm{E}(e_1 e_2) \tag{3}$$

and will be zero for uncorrelated noise. In many applications, independent and identically distributed noise with a normal distribution, or *iid normal* noise, is assumed, implying uncorrelated noise with a Gaussian distribution and equal variances at all channels. However, in reality correlated noise is commonplace, arising from varied sources such as temporal variations in spectroscopic source intensities, spatial correlations (cross-talk) in array detectors, thermal variations and electronic filtering. Signal processing itself can turn uncorrelated noise into correlated noise, a fact that can be important for subsequent data analysis methods. The correlation of a noise sequence can be examined through its autocorrelation function. This is obtained through an element-by-element multiplication of the noise sequence by a time shifted version of itself and averaging the results of the n multiplications. This is repeated for each time shift. Uncorrelated noise should give a single spike at zero time delay, with the products averaging to zero everywhere else. This is demonstrated in Figure 2(c).

The complete characterization of the correlation among elements of a noise vector of length n is given by the error covariance matrix Σ (Equation 4):

$$\Sigma = \begin{bmatrix} \sigma_1^2 & \sigma_{12} & \cdots & \sigma_{1n} \\ \sigma_{12} & \sigma_2^2 & \cdots & \sigma_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ \sigma_{1n} & \sigma_{2n} & \cdots & \sigma_n^2 \end{bmatrix} \tag{4}$$

where the diagonal elements represent the noise variances and the off-diagonal elements (zero for uncorrelated noise) represent the covariance. If a noise sequence is a stationary process, then its statistical properties remain constant throughout the sequence, which means $\sigma_{12} = \sigma_{23} = \sigma_{34}$ and so on. The terms 'homoscedastic' and 'heteroscedastic' are also used to indicate whether the variance of the noise remains constant or changes with the position in the series, respectively.

As with the signal, the characteristics of the noise can be examined in the frequency domain by using a spectrum analyzer or the FT. The result is called the noise power spectrum (NPS) and it conveys important information about the time domain correlations of the noise. Uncorrelated noise is referred to as white noise and, analogous to white light, it contains equal contributions at all frequencies. This is the type of noise that is often assumed or hoped for when designing or implementing signal processing methods, but exceptions are very common. The NPS for the white noise in Figure 2(a) is shown in Figure 2(d). Note that because of the stochastic nature of noise, the NPS is not perfectly flat.

Often purely white noise is corrupted by pink noise or $1/f$ noise. This noise, also known as drift in the time domain and arising from flicker noise, has a NPS which varies as the reciprocal of the intensity. The dominance at low frequencies is indicative of correlated noise in the time domain. The third type of noise readily identified in the NPS is interference noise, which appears as spikes at the corresponding frequencies and often higher harmonics. Figure 3 illustrates a typical correlated noise sequence and its representative autocorrelation function and NPS. Note the slow decay of the autocorrelation function indicating that the noise is not white. Both $1/f$ noise and interference noise are apparent in the NPS. It should be noted that, in order to obtain a clear representation of the autocorrelation function and the NPS in this case, it was necessary to average results from 20 noise sequences, because a single noise sampling does not usually give a clear indication of its characteristics.

3.3 Signal Averaging

One of the most effective ways to separate the pure signal from the noise is through the use of signal averaging. True signal averaging, as opposed to boxcar averaging or smoothing (section 5), requires a repetitive signal sequence. A common example is the fluorescence or phosphorescence decay curve generated by pulsing a laser or flash lamp repetitively. Each experiment (pulse) results in a signal that can be added to the next, and the total sum can be averaged over the number of experiments. Unlike most of the methods described here, which take advantage of signal correlation within a single experiment, signal averaging exploits signal correlation at a given time channel for repeated experiments. For this reason, it is discussed here, separately from the other techniques.

In order for signal averaging to be effective and useful, a number of conditions need to be met. First, the noise should be uncorrelated between corresponding time channels for successive experiments or else it will not be effectively removed. (Note that correlation of noise within an experiment is not important.) A second requirement is that the shape of the signal needs to be truly repetitive in the time domain. Translation of the signal or changes in its profile (other than scaling) due to poor synchronization or changes in the process will lead to an unrepresentative average, although noise should be reduced. Finally, the duration of the experiment needs to be short enough to make signal averaging practical.

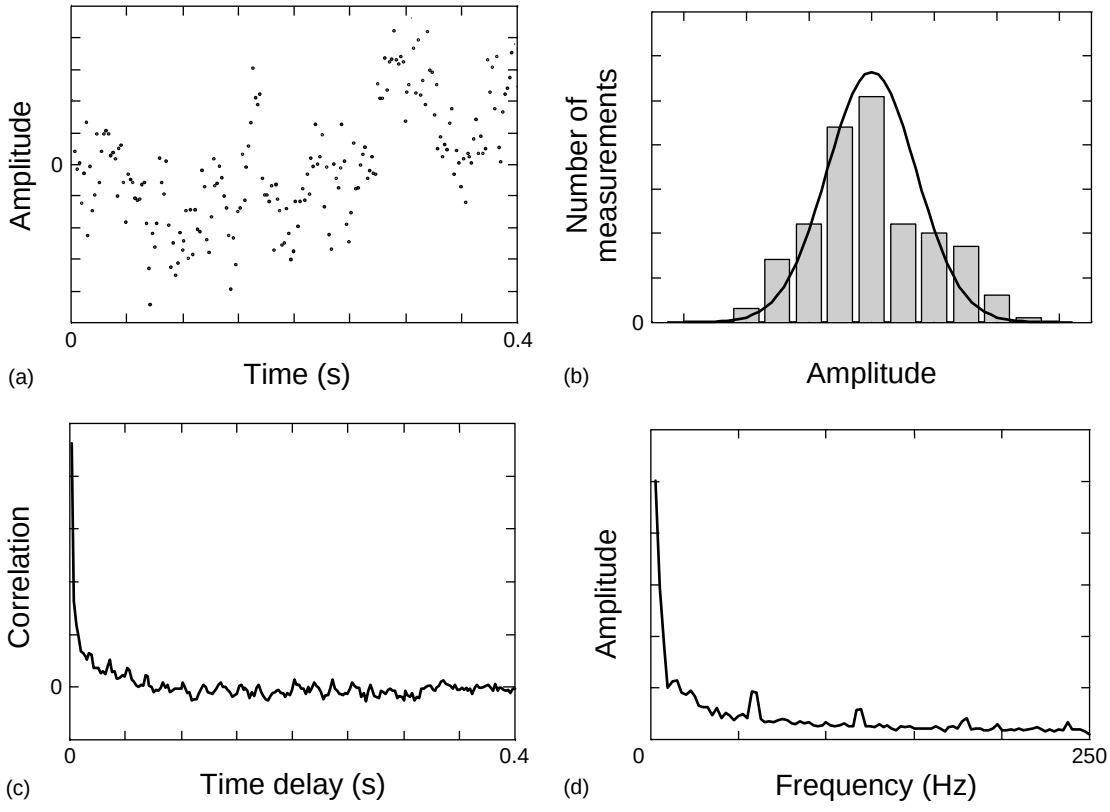

Figure 3 Representations of correlated noise: (a) noise sequence in the time domain, (b) distribution, (c) autocorrelation function, (d) FT.

When these conditions are met, signal averaging can be particularly advantageous because: (a) the signal-to-noise ratio (SNR) improves by a factor of $\sqrt{n}$, where n is the number of repetitions, and (b) there is no distortion in the profile of the signal. In contrast, most of the methods discussed in the sections that follow have the potential to distort the shape of the signal.

4 ANALOG SIGNAL PROCESSING

4.1 Overview

As virtually all instrumental methods involve some form of electrical transduction of a particular phenomenon into a continuous signal, analog signal processing is as universal as it is diverse. Such processing begins the moment the quantity being measured is converted into some electrical property such as current or voltage (assuming that it did not originate in that form) and continues until the final measurements are recorded in digital or analog form. This obviously opens up a tremendous range of topics, the detailed coverage of which is beyond the scope of this article. However, there has been a gradual shift over the years which has placed an increasing emphasis on digital signal processing over analog signal processing. One reason for this is that access to the analog signal in modern instruments has become more restricted and more often the chemist is presented with data that have already been digitized, as evidenced by the demise of the chart recorder from most analytical labs. A second reason is that improvements in the speed and storage capacity of digital components have removed many of the limitations of early devices. Finally, digital processing of results has the advantage that it can be carried out any time after the data are required, whereas this is not true for analog signal processing.

On this basis, it could be argued that a detailed comprehension of analog signal processing is less important now than it once was, although it is still essential to understand the basic capabilities and limitations of this stage of processing, because it will always precede the generation of digital information and can be the 'weak link in the chain' if care is not taken. For this reason, a brief coverage of electronic signal processing methods is presented here, with a special emphasis on two aspects which are closely related to digital processing, analog filters and A/D conversion.

The electronic manipulation of signals can be divided into methods which are based on passive or active

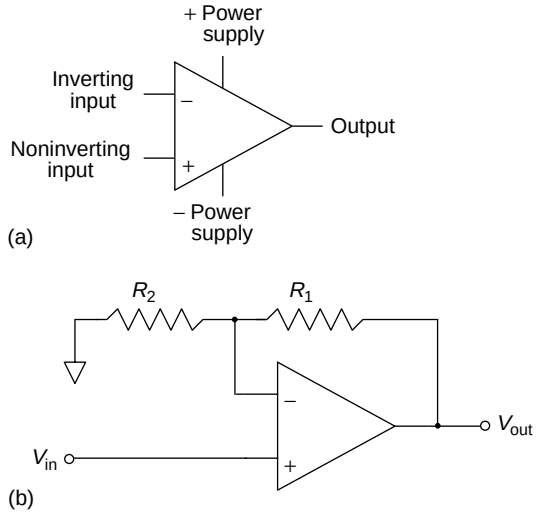

Figure 4 (a) Symbolic representation of an operational amplifier. (b) A noninverting amplifier with gain.

circuits. Passive circuits (those consisting only of simple components such as resistors, capacitors or diodes) do not require an external power source (other than the signal itself) whereas active circuits do. Although passive circuits are normally much simpler than active circuits, their capabilities are severely limited by comparison. Most complex manipulations of electronic signals are based on operational amplifiers, active circuit elements employed for a wide variety of linear and nonlinear operations. A basic operational amplifier, represented symbolically in Figure 4(a), is essentially a high gain (typically 10^5) differential amplifier. By taking advantage of the high gain in negative feedback, the two inputs (referred to as the inverting (−) and noninverting (+) inputs) and the output can be configured into a wide range of useful circuits using passive components. As an example, the configuration for a simple fixed-gain voltage amplifier (gain $= V_{out}/V_{in} = (R_1 + R_2)/R_2$) is shown in Figure 4(b).

The number of electronic operations that can be carried out on electrical signals is too extensive to cover here, but a brief summary of some important operations is given in Table 1. More details on analog signal processing can be found in appropriate references on the subject.[3–5] Analog filters and A/D conversion are also covered in more detail in sections 4.2 and 4.3.

4.2 Analog Filters

One of the simplest operations that can be carried out to improve signal quality is the application of an electronic filter. As the object of signal processing is to distinguish the pure signal from the noise, a means

Table 1 Summary of some common analog signal processing applications

Operation or circuit	Description
Domain conversion	Conversion of an analog signal between domains, such as current-to-voltage, resistance-to-voltage, time-to-amplitude, A/D.
Amplification	Multiplication of an analog signal by a constant factor called the gain. Sometimes coupled with domain conversion.
Inversion	Changes the sign of an analog signal.
Addition/subtraction	Two or more signals are added/subtracted.
Multiplication/division	Two signals are multiplied/divided by one another. Often used with modulation (see below).
Other mathematical operations	Includes logarithm, antilogarithm, absolute value, reciprocal, etc.
Integration/differentiation	Integrals or derivatives of analog signals.
Modulation	Process by which the property (e.g. amplitude or frequency) of a carrier wave, typically a high frequency sinusoid or square wave, is modified to convey information about an analog signal of interest. Demodulation is used to recover the original analog signal.
Comparator	Circuit that compares two voltages and produces one of two outputs, depending on which signal is larger.
Pulse height discriminator	Circuit that detects the presence of pulses with a peak amplitude within a certain threshold region.
Peak detector	Circuit that follows an analog signal until it reaches a peak value and then holds that value for a certain time period.
Sample-and-hold amplifier	Circuit that samples an analog voltage at a particular point in time and holds the value until further processing, normally A/D conversion, is completed.
Boxcar integrator	Circuit used to measure a rapidly changing but repetitive signal by sampling it after a particular delay time and holding the sampled measurement. Often the delay time is scanned to obtain the time profile of the signal.
Lock-in amplifier	One type of circuit used to demodulate analog signals (see above).

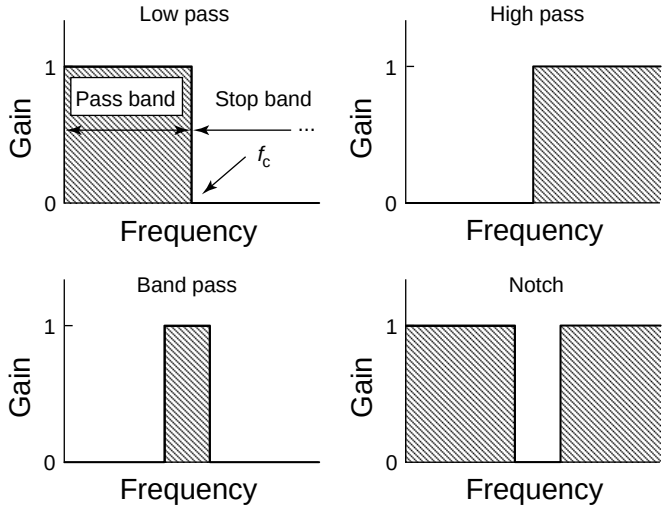

Figure 5 Transfer functions for ideal filters of various types.

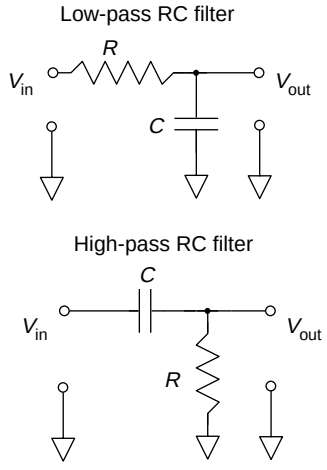

Figure 6 Simple passive RC low-pass and high-pass filters.

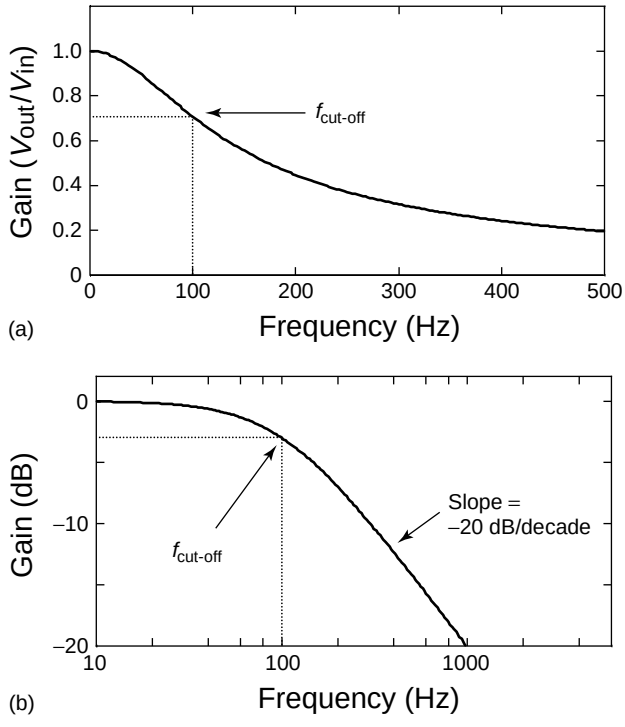

Figure 7 (a) Transfer function and (b) Bode plot for a simple low-pass RC filter with a cut-off frequency of 100 Hz.

is generally sought to distinguish the two. As noted in section 3, this distinction can often be made in the frequency domain. Typically, white noise has a flat NPS, whereas a slowly varying signal will have most of the information at low frequencies. Therefore, a filter that removes high-frequency components from a noisy signal, called a low-pass filter, will retain most of the information about the pure signal while eliminating much of the noise. However, drift and offset noise (low frequency) may be dominant features in the noise, and a high-pass filter may be more effective, provided there are sufficient high-frequency components in the signal. In another scenario, the signal may be modulated at a particular frequency and it may be necessary to use a band pass filter to isolate the components of interest. Finally, if interference noise, such as 60 Hz noise, is a problem, it may be removed using a notch filter. The ideal transfer functions for each of these types of filter is shown in Figure 5. The transfer function gives the amplification or gain of the filter as a function of the frequency of the input and ideally should be unity within the pass band and zero in the stop band.

As shown in section 5, analog filters bear many similarities to digital filters, but it is important to understand the former for several reasons. First, analog filters evolved before digital filters and there is substantial overlap of terminology. Second, although digital filters are becoming more widely used, analog filters are often a more effective way to eliminate noise near the source and are essential to limit the noise bandwidth in any digital data acquisition system. Finally, although they share similarities in their characteristics, transfer functions for digital and analog filters are significantly different.

Analog filters can be classified as either active or passive. Passive filters use only resistive (i.e. resistors) and reactive (i.e. capacitors and inductors) components, whereas active filters employ operational amplifiers as well. Figure 6 shows examples of simple low-pass and high-pass resistor–capacitor (RC) filters. In principle, filters could also be constructed using resistors and inductors, but inductors tend to be more bulky, expensive and less ideal, so in practice capacitors are more commonly used. The transfer function for a simple low-pass filter is shown in Figure 7(a). The gain is given by the ratio of the capacitive reactance to the total impedance

(Equation 5):

$$\text{gain} = \frac{V_{\text{out}}}{V_{\text{in}}} = \frac{X_{\text{C}}}{Z} = \frac{1/(2\pi fC)}{\sqrt{R^2 + 1/(2\pi fC)^2}}$$
$$= \frac{1}{\sqrt{(2\pi fRC)^2 + 1}} \quad (5)$$

Here X_{C} is the capacitive reactance, Z is the total impedance of the circuit, f is the frequency in hertz, R is the resistance in ohms, and C is the capacitance in farads. Note that the voltages referred to in this equation are not instantaneous voltages, but rather peak or rms values for inputs of a fixed frequency, as analog filters usually impose a phase shift on the original signal. More commonly, the frequency response of a filter is shown with a Bode plot, in which logarithmic scales are used on both axes. The voltage gain on the vertical axis is normally expressed in decibels (dB), given by Equation (6):

$$\text{gain(dB)} = 20 \log \left(\frac{V_{\text{out}}}{V_{\text{in}}} \right) \quad (6)$$

A Bode plot for a simple low-pass filter is shown in Figure 7(b). The frequency cut-off is usually taken to be the point at which the capacitive reactance equals the resistance, i.e. $V_{\text{out}}/V_{\text{in}} = 1/\sqrt{2}$. This corresponds to a gain of -3.01 dB, and so the operational cut-off frequency is usually referred to as the '3 dB point'. The cut-off frequency depends on the product RC and, for the simple low pass filter shown in Figure 6, this is given by Equation (7):

$$f_{\text{cut-off}} = \frac{1}{2\pi RC} \quad (7)$$

After the cut-off frequency, the Bode plot shows a linear region which has a slope of -20 dB per decade, more gradual than the ideal case shown in Figure 5. In order to provide a steeper slope, the order of a filter has to be increased. The filter order, also referred to as the number of poles, is the number of reactive components required for each cut-off frequency. Therefore, a first-order low- or high-pass filter such as the ones above requires only one capacitor, whereas a band-pass filter would require two. The higher the order of the filter, the more one can approach the ideal filter with a sharp transition between the pass band and the stop band. Practical limitations related to loading and other factors restrict the number of poles that can be used with passive filters, however, and active filters are normally used when more sophisticated filters are required.

The use of operational amplifiers in active filters allows greater flexibility in filter design. The design of active filters is well beyond the scope of this article, but fortunately, in those cases where a chemist may wish to incorporate an active filter, standard designs have been developed.[3–7] An active filter is classified not only according to its order, but also according to the way it optimizes a variety of other parameters, such as the steepness of the transition region, the flatness of the pass band, and its phase characteristics. Two of the most common types are the Butterworth, or maximally flat, filter and the Chebyshev, or equal ripple, filter. As shown in Figure 8, the frequency response of the Butterworth filter is very flat in the pass band, but its transition region is not as steep as the Chebyshev filter, which exhibits ripples in the pass band. The design for this active filter is shown in Figure 9. Note that the difference between the two filter types lies simply in the selection of the components and that the ripple can always be decreased at the expense of sharpness. The implementation of these filters is made even simpler through the use of integrated circuits incorporating switched capacitor filters that allow the cut-off frequency to be set simply with a reference clock signal.

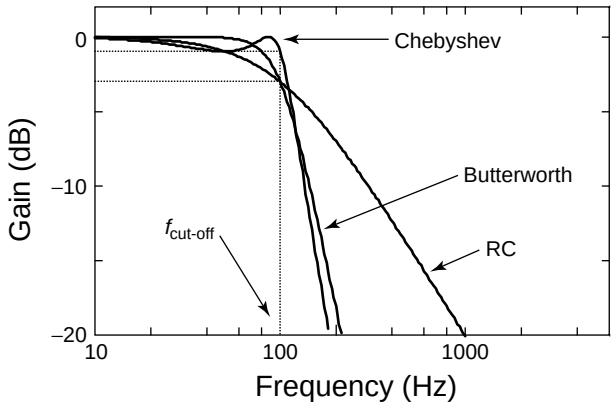

Figure 8 Frequency response (Bode plot) for passive (RC) and active (Butterworth and Chebyshev) filters.

From a digital signal acquisition perspective, perhaps the most important kind of filter is the anti-aliasing filter. When a signal is converted from the analog domain into the digital domain, it must be sampled at discrete and normally equally spaced time intervals, Δt. The sampling frequency is given by $f_{\text{s}} = 1/\Delta t$, and restricts the

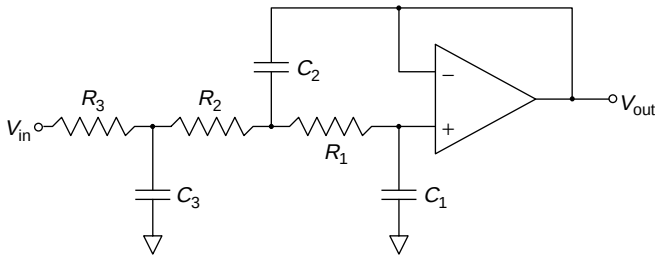

Figure 9 A third-order active filter.

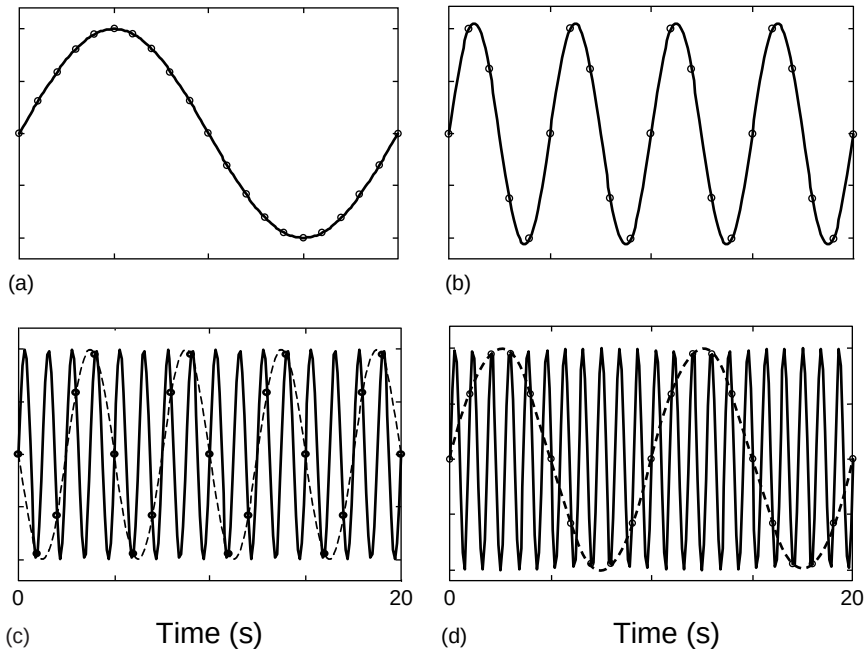

Figure 10 An illustration of aliasing in which open circles indicate sampled points. Parts (a) and (b) represent signals below the Nyquist frequency which are not aliased. The solid lines in (c) and (d) represent signals above the Nyquist frequency which are aliased to lower frequencies (dashed lines).

upper limit of signal frequencies that can be accurately represented in the transition from A/D information. More specifically, any components of the signal with frequencies above the Nyquist frequency will have their components aliased to lower frequencies. The Nyquist frequency, f_N, is given by Equation (8):

$$f_N = \frac{f_s}{2} = \frac{1}{2\Delta t} \tag{8}$$

The phenomenon of aliasing is shown in Figure 10, where four different sinusoids are sampled at a frequency of 1 Hz (i.e. $f_N = 0.5$ Hz). The first two signals shown are below the Nyquist frequency and the sampling reflects their variations accurately. However, the last two signals are above the Nyquist frequency and aliased back to lower frequencies (Figure 10).

The importance of an anti-aliasing filter, which is simply a low-pass filter with a cut-off near the Nyquist frequency, has to do with the noise bandwidth of the analog signal. If noise components are present above the Nyquist frequency and no filtering is applied, this noise will be aliased back to lower frequencies and appear in the acquired signal. As no information at frequencies higher than the Nyquist frequency can be accurately extracted anyway, it is wise to always use an anti-aliasing filter to reduce noise in a digitally acquired signal. Also note that this noise can only be removed in the analog domain and digital filtering cannot help once the noise is aliased to lower frequencies.

4.3 Analog-to-digital Conversion

A/D conversion refers to the process by which a continuously variable analog signal, usually a voltage, is converted into a discrete numerical value with a fixed precision. In most modern analytical instruments, A/D conversion is a key step in the signal processing sequence. Although this process is generally transparent to the user, an understanding of the principles involved can be useful in practice.

As digital logic circuits are based on binary states, the digital representation of a measurement is made in the binary, or base 2, number system, consisting of a series of binary digits, or bits. Thus the number 27 in base 10 would be represented by the following 8 bits, or byte, in base 2:

$$\begin{aligned} 27_{10} &= 0001\,1011_2 \\ &= 1(2^4) + 1(2^3) + 1(2^1) + 1(2^0) \end{aligned}$$

Alternatively, binary coded decimal (BCD) can be used in which each series of 4 bits (sometimes called a nibble) represents a decimal digit:

$$27_{10} = 0010\,0111\ (\text{BCD})$$

For n bits, a binary representation gives the numbers from 0 to $2^n - 1$, whereas a BCD representation gives a smaller range, from 0 to $10^{n/4} - 1$. To include the sign with a binary number several strategies can be employed.

In the offset binary notation, a value of 2^{n-1} is subtracted from the binary number to give the signed result. In the sign magnitude notation, the most significant bit (MSB) (the left-most bit) is used to represent the sign (0 = positive, 1 = negative). Finally, the most practical representation from a mathematical point of view is the 2's complement representation. In this case, a number is negated by inverting the original bits and adding 1. The 8-bit representations of the number −27 using each of these notations is given below.

$$-27_{10} = 0110\,0101 \text{ (offset binary)}$$
$$= 1001\,1011 \text{ (sign magnitude binary)}$$
$$= 1110\,0101 \text{ (2's complement binary)}$$

A/D conversion is facilitated using a largely self-contained circuit, the ADC. In addition to binary coding issues, important ADC parameters from a signal-processing perspective include precision, accuracy, linearity, monotonicity, and speed. The precision of an ADC is directly related to the number of bits in the digital output, but increased precision also means decreased conversion speed for a given type of ADC. Typically, 8-bit converters are used in applications where speed is more critical than precision, whereas precision applications can use as many as 16 or 20 bits. For many scientific applications, a 12-bit ADC is a good compromise, with a precision of 1 part in 4095, or 0.02%. It is important that the precision of the ADC is better than the standard deviation of the noise in the measurement, or else the dominant source of noise will be the quantization (or digitization) noise arising from rounding of the result. As the quantization noise is always fixed at a value corresponding to the least significant bit (LSB), its relative contribution increases for small values. In some cases, this problem can be addressed through autoranging, in which small signals are amplified in the analog domain prior to conversion. This is most important when the amplitude of the noise increases with the signal.

The accuracy of an ADC refers to the closeness of the converted value to the expected value based on the range of the ADC and the reference voltage. Linearity is an indication of the constancy of the proportionality of the digital output to the analog voltage over the full range of conversion, and in some cases may be more important than the actual accuracy. In a plot of the digital output versus the analog input, linearity is often specified as the maximum deviation from the straight line drawn between zero and the full scale output, or alternatively the best-fit straight line. Monotonicity is a specification that requires an increasing analog input to give an increasing digital output with no missing codes over the full range, and likewise for decreasing inputs. These specifications (accuracy, linearity, and monotonicity) are primarily a function of the ADC design and the quality of the components used.

In addition to being discrete in the measurement domain, analog signals are also discrete in the time domain, and the required conversion speed, or sampling frequency, will depend on the application. It is essential that all of the important signal characteristics in the frequency domain fall below the Nyquist frequency, f_N (section 4.2). The speed of an ADC is primarily a function of the type of converter used, although it is also dependent on factors such as the number of bits and clock frequency. Although the design of new ADCs is an ongoing process driven by consumer electronics, most are variations on five basic types: (1) parallel, (2) tracking, (3) successive approximation, (4) integration, and (5) voltage-to-frequency (V/F) conversion.

The fastest and conceptually simplest ADC is the parallel ADC, often referred to as a flash converter. In this circuit, the input voltage is simultaneously compared with 2^n reference values and logic circuits use the closest match to produce the digital output. This brute-force approach can provide conversion speeds under 10 ns, but is not component efficient, requiring 2^n voltage comparators. As a consequence, this type of ADC is usually expensive and limited to 8 bits, although hybrid circuits referred to as half-flash converters, can increase the precision.

Other relatively fast circuits are based on the use of a digital-to-analog converter (DAC) to provide an analog voltage for comparison with the input voltage. The digital value passed to the DAC is systematically changed until its analog output matches as closely as possible the input voltage. At that point, the digital input to the DAC is taken to be the digital output of the ADC. Differences exist in the way that these devices change the digital values for comparison. In a tracking ADC, a simple counter is used to increment or decrement the digital value until a match is found. A block diagram for a simple tracking ADC is shown in Figure 11. Depending on the change in the voltage, the conversion time for this type of ADC could range from 0 to 2^n clock cycles for an n-bit converter. The successive approximation ADC operates on the same basic principle as the tracking ADC, but has a fixed conversion time of only n clock cycles. This is accomplished by replacing the counter with circuitry that uses an efficient binary search algorithm to split the digital range in two for each comparison. Successive approximation ADCs have typical conversion times ranging from 1 to 100 μs and are among the most common converters in use.

The remaining two types of ADCs are relatively slow by comparison. Integrating ADCs are based on the use of a

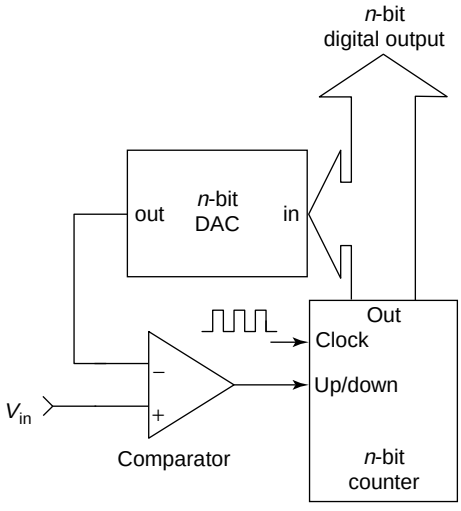

Figure 11 Block diagram of a simple tracking ADC.

fixed current to charge a capacitor through a resistor. The digital result is obtained by timing the period necessary for the capacitor to reach the input voltage. This basic strategy, termed a single slope integrating ADC, suffers from a dependence on accurate and stable circuit components and has been replaced by the dual slope (or even quad slope) integrating ADC. The dual slope ADC is also based on charge integration, but uses a charge–discharge cycle to cancel the effects of component variations. This type of ADC is known for its accuracy, stability, monotonicity, low cost and noise rejection characteristics and has been used in a large number of precision applications where speed is not critical. Typical conversion rates are around 10 conversions per second. In many applications, the dual slope ADC has been replaced by the V/F converter. This device simply produces a series of pulses whose frequency is proportional to the input voltage applied, with typical maximum frequencies in the range from 10 kHz to 1 MHz. To complete the conversion to a binary number, a counter is attached to the pulsed output and counted for a fixed period of time. Precision can be improved by counting for a longer period. In addition to low cost, simplicity, and good linearity, the V/F converter has the advantage that its output can be transmitted in serial over a single line, simplifying remote data acquisition. Like the integrating ADCs, however, this type of ADC is relatively slow.

Although the analyst typically has very little control over the type of A/D conversion that is used in a particular instrument, this is the first stage of all subsequent digital signal processing, so it is important to recognize the strengths and weaknesses of these devices. Further details on the design and application can be found in a number of references.[3–5,7–8]

5 DIGITAL FILTERING

5.1 Introduction

Because digital filters are among the most widely used methods for signal processing in analytical chemistry, much of this article is dedicated to describing their implementation and operation. The term 'filter' is a reference to the similarities they share with their electronic counterparts. In both cases, the data are presented to the filter in a sequential fashion and the distinction between pure signal and noise is often made on the basis of differences in power spectra. Strictly speaking, however, a filter uses only information from the past up to and including the current point in obtaining an estimate of the current point. Although this is true for electronic filters, it does not hold for most digital filters, which use points before and after the measurement of interest to form an estimate. Thus, they may be more properly classified as smoothers or smoothing filters, but all three terms are used in the literature. The term 'filter' is used in a general way throughout this section, incorporating both smoothers and other types of filters.

Digital filtering can be performed either in real-time or in a post-acquisition mode. The advantage of the former is that it can be transparent to the user and optimized at the time of instrument design. Fast digital signal processors (DSPs) are available for performing common signal processing operations and can be built into the instrument itself. A disadvantage of this approach is that it removes some flexibility and may obscure some of the features of the original data. For these reasons, real-time digital filtering is often kept to a minimum and much signal processing is carried out in post-acquisition mode. Therefore, it is important to appreciate the advantages and limitations of different types of digital filters. A complete coverage of the subject of digital filters is well beyond the scope of this article. The objective here is to present some of the terminology and describe some of the digital filters commonly used in analytical chemistry. For readers seeking more information on the subject, there is abundant literature available.[9–19]

5.2 Filter Types

In general terms, a digital filter could be defined simply as an operation that is carried out on a contiguous subset of the original signal sequence to produce an estimate of a value in the filtered signal sequence. This is illustrated in Figure 12. For conventional digital filters, the filtering operation consists of the convolution of a series of filter coefficients with the signal, but this approach is by no means universal. Digital filters are therefore classified

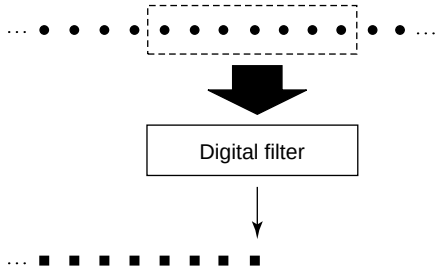

Figure 12 General operation of a digital filter.

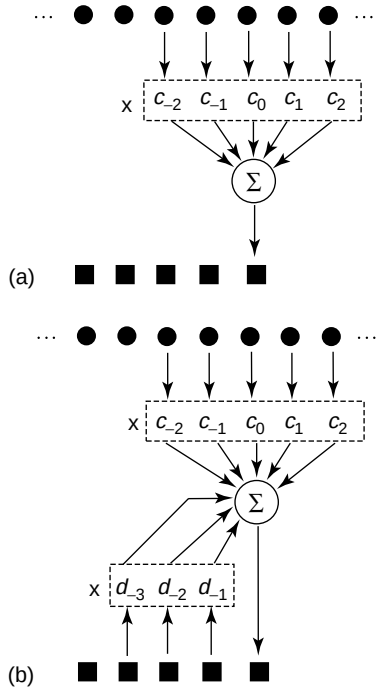

Figure 13 Operation of (a) nonrecursive and (b) recursive digital filters.

in a number of ways according to the manner of their operation.

The most common type of digital filter employed in analytical chemistry is the nonrecursive filter, also referred to as the finite impulse response (FIR) filter because its response to an impulse (delta) function will always fall to zero at some point in time. As illustrated in Figure 13(a), nonrecursive filters use the conventional approach of convoluting a set of filter coefficients with the sequence of measurements to produce the filtered signal. If z_i represents the filtered value for measurement i, it is determined mathematically by Equation (9):

$$z_i = \sum_{j=p}^{q} c_j y_{i+j} \tag{9}$$

where the c terms are the filter coefficients and the y terms are the original measurements. Most often, the filter coefficients are arranged symmetrically around the point to be estimated so that $p = -q$, but this is not a requirement. Nonrecursive filters have good stability and are relatively easy to design. A simple example of a nonrecursive filter is a five-point moving average filter, which averages the five points around a central value (two on either side plus the point itself) to obtain its estimate. The coefficients in this case would be $c_{-2} = c_{-1} = c_0 = c_1 = c_2 = 0.2$.

Recursive filters differ from nonrecursive filters in that they make use of previously filtered values to estimate the current measurement, as shown in Figure 13(b). Mathematically, this can be represented by Equation (10),

$$z_i = \sum_{j=p}^{q} c_j y_{i+j} + \sum_{k=r}^{s} d_k z_{i+k} \tag{10}$$

where z represents the filtered measurement, y is the unfiltered measurement, and c and d are the filter coefficients. Note that the indices r and s must be less than zero, as the filter coefficients d_k can only be applied to previously filtered values. In practice, many nonrecursive filters are designed to function in real time and so also limit the maximum value of q to be zero to make them physically realizable (i.e. they do not make use of future values). A simple example of a recursive digital filter is integration using the trapezoid rule, which gives (Equation 11):

$$z_i = \tfrac{1}{2}(y_{i-1} + y_i) + z_{i-1} \tag{11}$$

This corresponds to coefficients $c_{-1} = 0.5$, $c_0 = 0.5$ and $d_{-1} = 1$. Recursive digital filters are also known as infinite impulse response (IIR) filters because it is possible for a single impulse input to influence filter output values indefinitely, as illustrated in this example. These filters can be used for normal smoothing operations in addition to integration, and have the advantage that they provide more efficient filters with fewer coefficients than nonrecursive filters. However, recursive filters are more difficult to design and have more complex properties than their nonrecursive counterparts, so they have not been widely used in analytical chemistry.

It is not required that digital filter coefficients remain constant throughout a sequence of measurements. In some cases, the coefficients may change in response to variations in another external input or the signal itself. These kinds of filters are called adaptive filters. Such filters are used, for example, in removing noise from audio signals using external measurements of noise characteristics. Chemical applications of this type of filter are rare, although digital processing of modulated signals, such as modulated sources in atomic absorption

spectroscopy, could be considered to be a simple form of adaptive filtering.

Filters can also be classified according to the types of results they are intended to produce. Smoothing filters are intended to reproduce the pure signal and suppress the noise, whereas derivative filters are intended to estimate the derivative (first, second, or higher) of the pure signal. In other cases, one may wish to increase the apparent sampling frequency of a set of data, filling in points for esthetic or practical reasons (e.g. locating a peak maximum). This can be done using an interpolation filter, although care should be taken when applying such approaches. In contrast, in cases where the signal is highly oversampled, one may wish to reduce the volume of data while improving its quality by using a decimation filter.

Some types of digital filters do not follow the usual pattern of smooth convolution of a set of filter coefficients with the data, but are considered digital filters nevertheless. In boxcar averaging, for example, the signal is divided into subsets of n measurements which are averaged to produce a single result, as shown in Figure 14. As this reduces the total number of points, it is one type of decimation filter. Another rarely used but very useful filter is the median filter. Although this filter should be used with care because of its nonlinear transformation of the data, it is particularly effective at removing spikes or outliers from a measurement sequence. These outliers may arise, for example, from cosmic rays striking a photodetector or bubbles passing through a detector flow-cell. The median filter works by sorting the data within a window of length n and choosing the median value as the filtered estimate, thus automatically eliminating outliers unless they occur in clusters. An example of the application of the median filter is shown in Figure 15.

Because of their dominance in chemical applications the main emphasis of this article is on nonrecursive digital filters, particularly polynomial least-squares filters, although some discussion of Kalman filters is also presented.

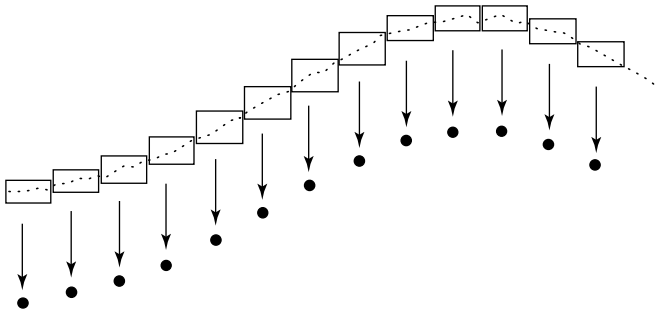

Figure 14 Illustration of boxcar averaging.

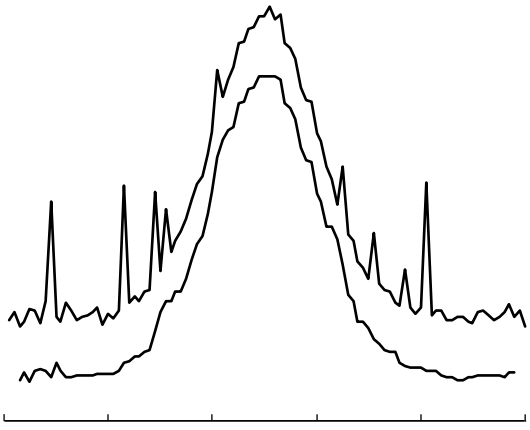

Figure 15 A noisy peak (upper curve) before and (lower curve) after filtering with a median filter. Note the removal of spikes.

5.3 Polynomial Least-squares Smoothing Filters

By far the most widely applied digital filter in analytical chemistry is the polynomial least-squares smoothing filter. These filters are more commonly known to analytical chemists as SG filters, a reference to their introduction into the analytical chemistry literature by Savitzky and Golay in 1964.[1] Although these filters were known in the field of signal processing prior to this, the SG paper made their utility known to chemists at a time when the digital acquisition of signals was becoming more commonplace. In addition, the paper presented tables of precalculated coefficients for different types of filters. At the time, computational efficiency was poor, so the authors presented the coefficients as integers with a normalization factor rather than as a series of floating point numbers.

Among the advantages of the SG filters are their simplicity and versatility. Application of these filters assumes that a local region of the data set (i.e. the filter window) can be fit to a low-order polynomial, and the central point within that window is estimated by performing such a fit. This is illustrated in Figure 16 for first- and second-order polynomials. Fitting the data in this way should model the correlations in the pure signal while reducing the influence of random noise fluctuations. Furthermore, when the measurements are evenly spaced in the time domain, the fitted estimate of the central point can be obtained simply by multiplying the points in the window by a set of precalculated coefficients, thus making the fit equivalent to a nonrecursive digital filter.

The original tables published by SG had a number of errors that were later corrected in the literature,[20] but it is now just as simple to calculate the coefficients for a given application, so the tables are seldom used. To illustrate how this is done, consider the design of a second-order polynomial smoothing filter using a five-point window.

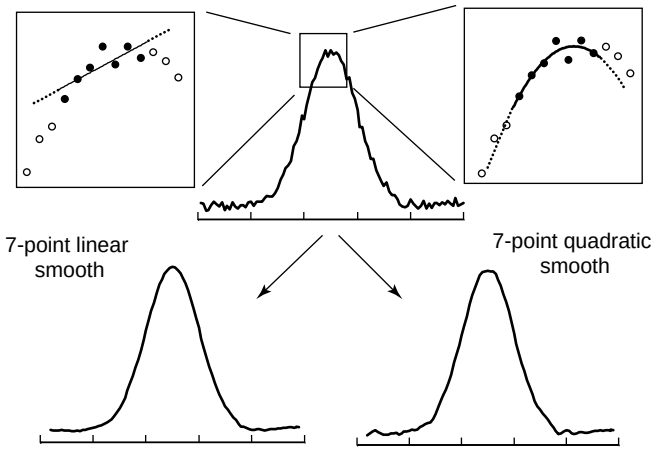

Figure 16 Illustration of polynomial smoothing as a least-squares fitting procedure.

The model to be fit is Equation (12):

$$y = b_0 + b_1 x + b_2 x^2 \tag{12}$$

Equation (13) is the equivalent matrix form:

$$\begin{bmatrix} y_1 \\ y_2 \\ y_3 \\ y_4 \\ y_5 \end{bmatrix} = \begin{bmatrix} 1 & x_1 & x_1^2 \\ 1 & x_2 & x_2^2 \\ 1 & x_3 & x_3^2 \\ 1 & x_4 & x_4^2 \\ 1 & x_5 & x_5^2 \end{bmatrix} \begin{bmatrix} b_0 \\ b_1 \\ b_2 \end{bmatrix} \tag{13}$$

which can be expressed as (Equation 14):

$$\mathbf{y = Xb} \tag{14}$$

In these expressions, x represents the time or other ordinal variable, while $\mathbf{X}$ is the matrix containing the basis functions for the polynomial fit. It is important to note that the fitted values obtained for y are independent of the scale of x and, if the time interval between each measurement is equal (as is usually the case), we can arbitrarily set $\mathbf{x} = [-2 \quad -1 \quad 0 \quad 1 \quad 2]$, giving Equation (15):

$$\mathbf{X} = \begin{bmatrix} 1 & -2 & 4 \\ 1 & -1 & 1 \\ 1 & 0 & 0 \\ 1 & 1 & 1 \\ 1 & 2 & 4 \end{bmatrix} \tag{15}$$

The least-squares solution for the vector of regression coefficients, $\mathbf{b}$, is well known from linear algebra to be (Equation 16):

$$\mathbf{b} = (\mathbf{X}^{\mathrm{T}}\mathbf{X})^{-1}\mathbf{X}^{\mathrm{T}}\mathbf{y} = \mathbf{A}\mathbf{y} \tag{16}$$

The matrix $\mathbf{A}$ is a 3×5 matrix which can be regarded as being composed of three row vectors, $\mathbf{a}_1$, $\mathbf{a}_2$, and $\mathbf{a}_3$; as in Equation (17):

$$\mathbf{A} = \begin{bmatrix} a_{11} & a_{12} & a_{13} & a_{14} & a_{15} \\ a_{21} & a_{22} & a_{23} & a_{24} & a_{25} \\ a_{31} & a_{32} & a_{33} & a_{34} & a_{35} \end{bmatrix} = \begin{bmatrix} \leftarrow \mathbf{a}_1 \rightarrow \\ \leftarrow \mathbf{a}_2 \rightarrow \\ \leftarrow \mathbf{a}_3 \rightarrow \end{bmatrix} \tag{17}$$

Note that the intercept coefficient for the fit, b_0, is obtained from Equation (18),

$$b_0 = \mathbf{a}_1\mathbf{y} = a_{11}y_1 + a_{12}y_2 + \cdots + a_{15}y_5 \tag{18}$$

Also note that, as $x = 0$ for the central point in the five point sequence, Equation (19) holds:

$$\hat{y}_3 = b_0 + b_1(0) + b_2(0)^2 = b_0 = \mathbf{a}_1\mathbf{y} \tag{19}$$

Therefore, because of the way the problem has been set-up, the estimate of the central point in the sequence is obtained simply by multiplying each measurement by the corresponding element in $\mathbf{a}_1$. In other words, the digital filter coefficients are simply the first row of the matrix $(\mathbf{X}^{\mathrm{T}}\mathbf{X})^{-1}\mathbf{X}^{\mathrm{T}}$, i.e. $\mathbf{c} = \mathbf{a}_1$.

The above reasoning holds for polynomial smoothing filters of any length and any order. All that is required to determine the filter coefficients is to set up the matrix of basis functions, $\mathbf{X}$, perform the calculation in Equation (16) using a spreadsheet or other software, and extract the first row of the resulting matrix. Polynomial smoothing filters are convenient for improving the appearance and SNR of many signals and have the added advantage over electronic filters that different types of filters can be applied after the signal has been recorded.

One of the drawbacks of polynomial smoothing filters is sometimes referred to as the edge effect. As the filters are designed to obtain an estimate of the central point in a window, there will be points at the beginning and end of a measurement vector that cannot be estimated with the symmetric filter. For example, with the five-point filter described above, two points could not be filtered at each end of the data sequence. Several options are available to deal with this problem. The simplest is either to drop these points from the data set, or leave them in the data set unfiltered. Another possibility, if only baseline data occurs at the limits of the measurement vector, is to use the points at one end of the data set to filter those at the other. For example, to estimate y_1 with the five-point filter, we could use the sequence (y_{n-2}, y_{n-1}, y_1, y_2, y_3). Finally, we could employ what are sometimes referred to as initial point filters or extended sliding window filters,[21,22] designed to estimate values other than the central point of a sequence. The coefficients for these filters are easily obtained by simply shifting the $\mathbf{X}$ matrix accordingly. For example, to obtain coefficients to estimate the first point of a five point sequence with a

second-order smooth, we would use Equation (20):

$$\mathbf{X} = \begin{bmatrix} 1 & 0 & 0 \\ 1 & 1 & 1 \\ 1 & 2 & 4 \\ 1 & 3 & 9 \\ 1 & 4 & 16 \end{bmatrix} \tag{20}$$

All other aspects of the problem are the same. Although this is an elegant way to solve the problem, it requires several sets of filter coefficients to handle the points at the edges of the data and the noise rejection and signal distortion characteristics of these filters are not identical to the symmetric filters.[23]

The selection of smoothing filter parameters (order, number of points) for a given application is often a matter of trial and error and intuition. Obviously, one would like to obtain the maximum noise reduction with the minimum amount of distortion. Although the best noise reduction occurs with wider filters (more points), wider filters also limit the ability of the chosen function to obtain a good local model for a changing signal. In general, noise rejection improves and signal distortion increases as the width of the signal increases and the order of the filter decreases. It should be noted, however, that as a consequence of the mathematics, smoothing filters for orders 0 and 1 are identical, as are those for orders 2 and 3, and so on. To determine the amount of noise reduction a filter will provide, there is a very simple relationship (Equation 21):

$$\frac{\sigma^2_{\text{filtered}}}{\sigma^2_{\text{unfiltered}}} = \sum_{j=p}^{q} c_j^2 \tag{21}$$

Thus the ratio of the variance of the noise in the filtered signal to that in the unfiltered signal is simply the sum of the squared filter coefficients. This equation applies to any type of nonrecursive digital filter, but is only valid in cases where the signal exhibits white noise.

Although Equation (21) is useful for describing the amount of noise reduction, it provides no information about the extent of signal distortion. To this end, it is often more useful to examine the response of a digital filter in the frequency domain in much the same way as for an electronic filter. For a symmetric smoothing filter, the frequency response is given by Equation (22),

$$H(f) = \left| \sum_{k=-m}^{m} c_k \cos\left(\frac{k\pi f}{f_{\text{N}}}\right) \right| \tag{22}$$

where $H(f)$ is the amplitude gain of the filter at frequency f, c_k represents the filter coefficients, and f_{N} is the Nyquist frequency. Note that this does not apply to unsymmetric smoothing filters, such as the initial point filters, because these filters also involve a phase shift,

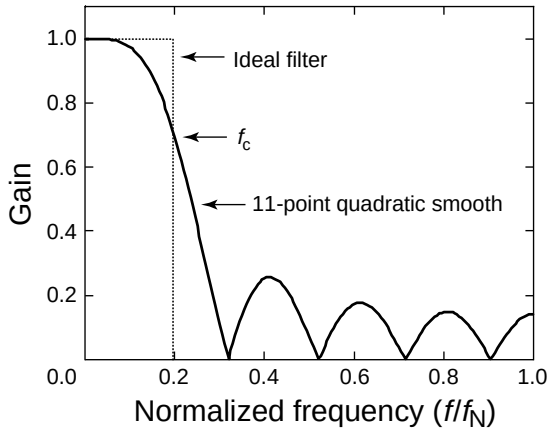

Figure 17 Frequency response (transfer function) for an 11-point quadratic smoothing filter.

but alternative expressions are available.[23] Figure 17 shows the frequency response for an 11-point quadratic smoothing filter, plotted as a function of f/f_{N} to make it universal. As expected, the amplitude gain is unity at low frequencies and drops off at higher frequencies. Unlike simple electronic filters, however, the gain of these filters does not smoothly approach zero, but instead oscillates around a number of nodes that are related to the size and order of the filter. Furthermore, if the frequency response were plotted beyond the Nyquist frequency, the function would simply reflect itself as signals were aliased to lower frequencies, reaching a gain of unity once again at the sampling frequency. If one has some idea of the amplitude spectrum of the signal to be filtered, plots such as Figure 17 can be very useful in assessing the degree of signal distortion that will result from filter application. The amount of noise rejected can be also ascertained from the ratio of the area under the NPS before and after multiplication by the filter frequency response. Clearly, this type of filter will be effective when white noise is present, but will be less effective for $1/f$ (drift) noise, as the noise exists predominantly at low frequencies. It should also be noted that even if white noise was present before filtering, measurement noise will become correlated after filtering.

Of course, whenever a digital filter is applied to experimental data, there will be changes in the shape of the signal. In certain cases where parameters of the signal such as peak height, area or width are important in themselves, consideration must be given to the consequences of applying a digital filter. Generally, for signals which exhibit the same shape, the effects in the time domain (e.g. width at half-maximum) will be the same, whereas effects on the amplitude (e.g. peak height/area) will be linear with the magnitude of

the signal. The reader is referred to a useful, although somewhat empirical, study by Enke and Nieman for more details on these effects.[16]

5.4 Derivative Filters

The numerical differentiation of signals with respect to time (or other ordinal variable) is a common practice in analytical chemistry. This procedure can be used, for example, to locate the position of a peak maximum, to determine the end-point of a titration, or to highlight poorly defined features in a signal sequence (e.g. the shoulder on a peak). These can be regarded as qualitative applications in the sense that one is looking for the location of specific points in the signal derivative (e.g. the maximum or zero values) but not using the derivative sequence for further calculations. Quantitative applications, in which the differentiated signal is used for purposes such as calibration, have become more common in recent years. Because the derivative of a function is unaffected by the addition of a constant, differentiation proves valuable for methods which exhibit a baseline shift (offset) between samples. Likewise, if sample measurements are plagued with a baseline that changes linearly with the ordinal variable (drift), calculation of the second derivative can solve the problem.

One serious problem with signal differentiation is the selective amplification of high-frequency noise. Because the derivative of a signal is, by definition, its rate of change, the more rapidly varying components of a signal, including noise, are amplified to a greater extent than the more slowly changing features typically associated with the pure signal. Because of this, derivative filters (first, second or higher) are most useful for signals which exhibit relatively small amounts of high-frequency noise compared to the low-frequency contributions, i.e. cases where $1/f$ or drift noise dominates. A classic example of this is near infrared (NIR) spectroscopy, where second derivatives are routinely calculated prior to quantitative analysis. Although NIR measurements are widely characterized in the literature as having very high SNRs, these measurements suffer from serious noise problems in the form of baseline offset and drift, but traditional SNR calculations normally do not incorporate these components. Fortunately, the characteristics of NIR spectra make them almost ideal benefactors of derivative filtering.

The noise amplification characteristics of derivative filters can be better understood by examining the transfer function in the frequency domain. As shown in Figure 17, the gain of a smoothing filter is typically unity at low frequencies and falls off at high frequencies. In contrast, Figure 18 shows that the gain of a true derivative filter increases linearly with frequency. This can be easily

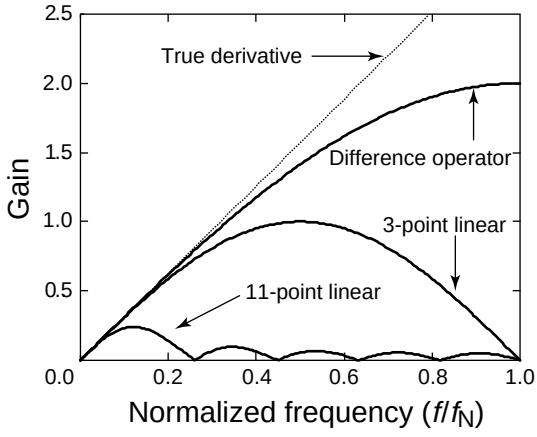

Figure 18 Frequency response (transfer function) for various types of derivative filters.

confirmed by recognizing that (Equation 23),

$$\frac{\mathrm{d}(\sin wt)}{\mathrm{d}t} = w\cos wt \tag{23}$$

where $w = 2\pi f$ is the angular frequency. As a signal in the time domain can be represented as the sum of a series of sines and cosines (section 6.2) it is clear the calculation of the derivative amplifies a signal component by a factor of w and also gives rise to a 90° phase shift. The consequence of this is that the excessive amplification of high-frequency noise components generally makes the calculation of the true derivative for practical measurements useless (and is the reason why the term 'true' is used instead of 'ideal'). In practice, most derivative filters implicitly combine the derivative calculation with a low-pass filter to reduce the contribution of high frequency components.

The simplest method for calculating a signal derivative, and one which can be described as a simple digital filter, is the difference operator, defined by Equation (24),

$$\hat{y}'_i = \frac{y_{i+1} - y_i}{\Delta t} \tag{24}$$

where $\hat{y}'_i$ represents the estimate of the derivative of the function at point i, and Δt is the sampling interval. The transfer function for this type of filter is shown in Figure 18. It is clear that the frequency response for a simple difference operator matches that of the true derivative closely except at very high frequencies. This type of filter provides no low-pass filtering, however.

An alternative way of calculating derivatives is to use a slight modification of the polynomial least-squares filters described in section 5.2. If we carry out the least-squares fit in the same manner as the previous example for a five-point quadratic model, the estimate of the derivative for the central point is given by

Equation (25),

$$\hat{y}'_0 = \frac{\mathrm{d}}{\mathrm{d}x}(b_0 + b_1 x_0 + b_2 x_0^2) = b_1 + 2b_2 x_0$$
$$= b_1 + 2b_2(0) = b_1 \qquad (25)$$

Therefore, the derivative estimate of the central point is simply the first-order coefficient of the fit. In a manner analogous to Equation (18), this is obtained by simply multiplying the second row of the **A** matrix by the windowed measurement vector **y**. This means that the coefficients of the derivative filter are given by the second row of **A** ($\mathbf{c} = \mathbf{a}_2$) as opposed to the first row of **A** for the smoothing filter. Likewise, the filter coefficients for the second derivative are given by the third row of **A**, and so on. It is clear then, that there is a simple, common path to the calculation of polynomial filters of various types.

A number of characteristics of derivative filters calculated in this way should be noted. First, unlike smoothing filters, the calculation of numerically correct derivatives requires consideration of the sampling interval. The adjusted coefficients necessary to obtain the correct scale are given by Equation (26),

$$\mathbf{c}_{\text{adjusted}} = \frac{\mathbf{c}_{\text{original}}\, p!}{(\Delta t)^p} \qquad (26)$$

where p is the order of the derivative. In many cases, this scaling is ignored, because it is only the relative changes in the derivative that are important. A second characteristic of these derivative filters that should be apparent from the mathematics is that the determination of coefficients for a pth order derivative requires at least a pth order polynomial. Also, as with smoothing filters, there is a duplication of coefficients for adjacent polynomial orders, although the pairing shifts with each higher derivative. For example, for the first derivative, the filter coefficients for the linear and quadratic polynomials are the same, whereas the quadratic and cubic coefficients are the same for the second derivative.

The polynomial filters described here are symmetric in the sense that there are an equal number of coefficients on either side of the central point (in fact, a first-derivative filter is better described as antisymmetric, as $c_{-i} = -c_i$). As with smoothing filters, it is possible to develop derivative filters for the edges of a window, but the usual precautions regarding the quality of estimates apply. For symmetric first-derivative filters, the frequency response is given by,

$$H(f) = \left| \sum_{k=-m}^{m} c_k \sin\left(\frac{k\pi f}{f_{\mathrm{N}}}\right) \right| \qquad (27)$$

Equation (27) is a slight modification of Equation (22), where the substitution of sine for cosine results from the 90° phase shift brought about by the derivative filter. The frequency responses for 3-point and 11-point first-order derivative filters are shown in Figure 18 for comparison with the true derivative and difference operator. Note that although the transfer functions for the polynomial filters match the true derivative at low frequencies, there is significant attenuation at high frequencies due to the low-pass filtering. The effects of this low-pass filtering are clearly seen in Figure 19, which shows the application of three types of derivative filters to a noisy signal. Because of the effect of high-frequency noise on derivative filter response, it is a common practice by some to first apply a smoothing filter to the data, but Figures 18 and 19 demonstrate that if the derivative filter is properly designed, this practice is redundant.

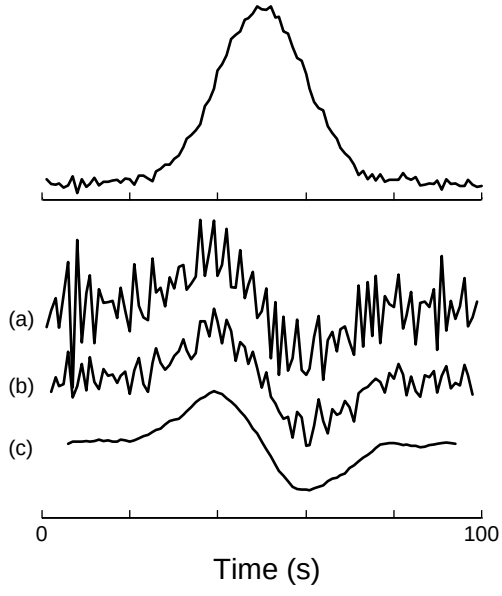

Figure 19 Result of the application of various types of derivative filters to a noisy peak, shown above: (a) difference operator, (b) 3-point linear derivative filter, (c) 11-point linear derivative filter.

5.5 Kalman Filters

The Kalman filter is a recursive linear least-squares estimator with the capability of estimating the parameters associated with a system model in real time.[17–19] It is not so much a filter in the conventional sense as it is a means for carrying out linear least-squares in a recursive fashion. The estimates it provides are not the smoothed measurements, but rather the parameters associated with the linear model, or the state parameters. These parameters are sometimes considered to represent a state vector in an n-dimensional state space (n = number of parameters). Once the state parameters have been estimated, it is possible to generate a smooth curve for

the measurements from the model, but this is usually a secondary objective.

A simple example of recursive estimation in a manner similar to the Kalman filter is the calculation of a mean from a series of measurements as the measurements are being acquired. It is usual to start with an estimate of the mean equal to the first measurement, i.e. $\hat{m}_1 = x_1$. Once a second measurement was acquired, the estimate could be improved by using $\hat{m}_2 = 1/2\hat{m}_1 + 1/2x_2$. In general, after the ith measurement, Equation (28) would hold:

$$\hat{m}_i = \frac{i-1}{i}\hat{m}_{i-1} + \frac{1}{i}x_i \tag{28}$$

As each new measurement is assimilated into the estimation, the quality of the parameter estimate improves. It is apparent that Equation (28) has the form of a recursive filter whose coefficients are changing with each measurement. An advantage of recursive estimation is that continuous updates of the parameter(s) of interest are obtained with each new measurement. Although this could also be done in batch mode, the recursive formulation is computationally more efficient.

The general model which is covered by the Kalman filter can be described by Equations (29) and (30):

$$\mathbf{x}_{k+1} = \mathbf{F}_k\mathbf{x}_k + \mathbf{w}_k \tag{29}$$

$$\mathbf{z}_k = \mathbf{H}_k\mathbf{x}_k + \mathbf{v}_k \tag{30}$$

In these equations, $\mathbf{x}_k$ represents the $n \times 1$ vector of parameters to be estimated (the state vector) at measurement interval k and $\mathbf{z}_k$ represents the $m \times 1$ vector of measurements at interval k. The first equation describes how the state vector is expected to change from one measurement interval to the next and contains both systematic and stochastic terms. The $n \times n$ state transition matrix, $\mathbf{F}$, describes the systematic linear transformation, whereas the vector of random variables, $\mathbf{w}$, represents the stochastic change. Each element of $\mathbf{w}$ is assumed to be derived from a zero-mean white-noise sequence and $\mathbf{w}$ is characterized by an $n \times n$ covariance matrix $\mathbf{Q}$.

Equation (30) describes how the state parameters are translated into a measurement or observation vector. The linear relationship is described by the $m \times n$ observation matrix, $\mathbf{H}$. There is also a random noise component assumed for the observations, represented by the vector $\mathbf{v}$. The elements of this vector are also assumed to comprise a white-noise sequence and the covariance of noise in the measurement vector is described by the $m \times m$ covariance matrix, $\mathbf{R}$.

The basic algorithm for the Kalman filter is shown in Figure 20, although several variants exist. The application of this algorithm is best described through a simple example. Suppose, for the purposes of illustration, we are using an absorption spectrometer to monitor a reaction

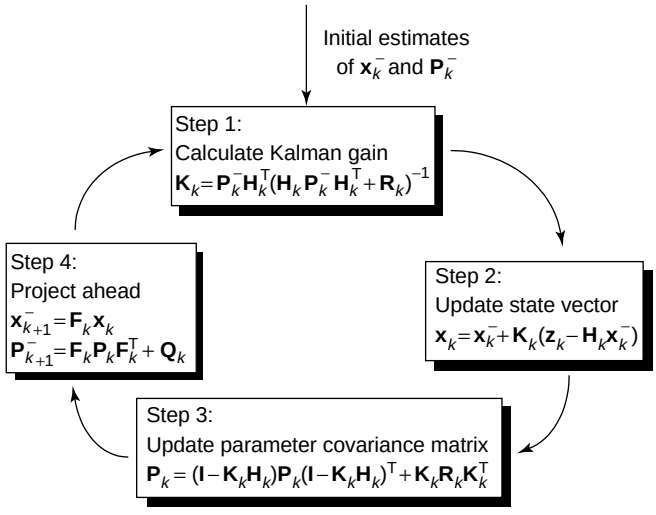

Figure 20 Basic algorithm for the Kalman filter.

in which two absorbing species, A and B, are reacting independently to form nonabsorbing products by first-order kinetics. The defining Equation (31) is:

$$\mathbf{z}(t) = A(t) = C_A\varepsilon_A e^{-k_A t} + C_B\varepsilon_B e^{-k_B t} + v(t) \tag{31}$$

where $A(t)$ is the absorbance at time t, C_A and C_B represent the initial concentrations of the two species, ε_A and ε_B are their molar absorptivities, and k_A and k_B are their first-order decay constants. Assuming that all quantities are known except the initial concentrations which are to be estimated, this is a linear problem (Equations 32):

$$\mathbf{x}(t) = \begin{bmatrix} C_A \\ C_B \end{bmatrix}; \quad \mathbf{H}(t) = \begin{bmatrix} \varepsilon_A e^{-k_A t} & \varepsilon_B e^{-k_B t} \end{bmatrix} \tag{32}$$

A simulated data set was generated using $C_A = 0.2\,\text{mM}$, $C_B = 0.5\,\text{mM}$, $\varepsilon_A = 1000\,\text{M}^{-1}\,\text{cm}^{-1}$, $\varepsilon_B = 1500\,\text{M}^{-1}\,\text{cm}^{-1}$, $k_A = 0.1\,\text{s}^{-1}$, $k_B = 0.3\,\text{s}^{-1}$, and a noise level of 0.02 absorbance units (AU). These data, showing the measured points and the curve with no error, are presented in Figure 21(a). The objective of the Kalman filter is to provide estimates of C_A and C_B as each new measurement is made.

To initiate the Kalman filter prior to step 1, we need an estimate of the state parameters and the error covariance matrix, $\mathbf{P}$, describing the uncertainty in those parameters. Assuming we have no prior knowledge, we use Equations (33):

$$\mathbf{x}_1^- = \begin{bmatrix} 0 \\ 0 \end{bmatrix}; \quad \mathbf{P}_1^- = \begin{bmatrix} 10^{10} & 0 \\ 0 & 10^{10} \end{bmatrix} \tag{33}$$

The superscript '$-$' indicates that these are the estimates before we have assimilated the first measurement. As we have no prior knowledge of the parameters, the diagonal

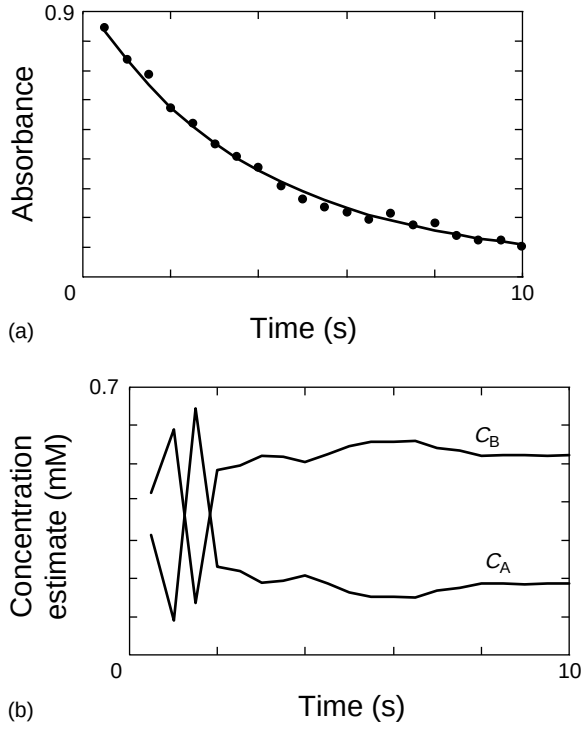

Figure 21 Results from the application of the Kalman filter to the first-order kinetic data described in the text. (a) The solid line shows the true absorbance decay curve and the points indicate the noisy measurements made. (b) The evolution of estimates for the initial concentrations of components A and B by the Kalman filter.

elements (variances) of the covariance matrix are set to very large values. In order to estimate the Kalman gain, **K**, in step 1, the observation matrix, **H**, can be calculated from Equation (32). As the measurement, **z**, is a scalar in this case, the measurement error covariance matrix, **R**, is simply equal to the variance of the measurements. Thus, at $t = 0.5$ s, Equations (34) hold:

$$\mathbf{H}_1 = [951.2 \quad 1291.1]; \quad \mathbf{R}_1 = (0.02)^2 = 0.0004 \qquad (34)$$

With these values, the $n \times 1$ Kalman gain vector for this iteration is Equation (35):

$$\mathbf{K}_1 = \begin{bmatrix} 3.699 \times 10^{-4} \\ 5.020 \times 10^{-4} \end{bmatrix} \qquad (35)$$

In step 2 of the algorithm, the difference between the actual observation, $\mathbf{z}_k$, and the observation predicted by the current state parameters, $\mathbf{H}_k\mathbf{x}_k$, is calculated. This difference is sometimes called the innovation and is like an ordinary residual except that it is calculated using the current rather than the final parameter estimates. The Kalman gain vector, **K**, determines how much the innovation is weighted in updating the state parameter estimates. It is also used in the third step of the algorithm to update the error parameter error covariance matrix, **P**, following integration of the new measurement. Using the first observation of 0.846 AU gives Equations (36):

$$\mathbf{x}_1 = \begin{bmatrix} 0.313 \\ 0.425 \end{bmatrix} \times 10^{-3}; \quad \mathbf{P}_1 = \begin{bmatrix} 6.5 & -4.8 \\ -4.8 & 3.5 \end{bmatrix} \times 10^{9} \qquad (36)$$

Note that because only one measurement has been processed and there are two parameters to be estimated, neither **x** nor **P** can be regarded as reliable at this point.

In step 4, the state vector and its covariance matrix are projected ahead to the next measurement interval. This requires a knowledge of **F** and **Q**, which are trivial in this example. As the state parameters here are static (i.e. $\mathbf{x} \neq \mathbf{f}(t)$), the state transition matrix, **F**, is simply the identity matrix. This would not be the case, for example, if the state parameters were the concentrations at time t, rather than the initial concentrations, but the modifications to **H** and **F** would be straightforward in that case. Likewise, we are assuming no random variation in the initial concentrations, so the state vector covariance matrix, **Q**, is equal to zeros. The role of **Q** in a more complex application is to allow for random variation in the state parameters over time. As an example, suppose that instead of absorbance, we were measuring total pressure in a gas phase reaction which was subject to random temperature fluctuations between measurements. This would effectively change the initial pressures we were trying to estimate.

The iterations of the Kalman filter continue in this way until all of the measurements have been processed. For the example presented here, Figure 21(b) shows the concentration estimates as a function of time with the final estimates:

$$C_A = 0.186 \pm 0.019\,\text{mM}$$

$$C_B = 0.522 \pm 0.023\,\text{mM}$$

The uncertainties are the standard deviations of the parameters from **P**. Note that these converge to values close to the true concentrations. It should also be noted that these are essentially the same estimates that would have been obtained by linear least-squares implemented in batch mode.

Although Kalman first introduced this filter in 1960,[24] applications in chemistry were not abundant until the late 1970s and early 1980s. However, many of these applications employed the Kalman filter mainly as a recursive implementation of simple least-squares, such as the example above, and did not exploit its full capabilities. For instance, given spectra of mixture components, the Kalman filter can be used to estimate component concentrations as a spectrum was being scanned. Although this offered certain advantages such as speed and the ability to terminate an experiment

when the desired precision was achieved, developments in instrumental and computational efficiency have made these benefits less significant. The potential exists for more effective utilization of the algorithm, however.

At least two modifications of the basic Kalman filter have also appeared in the analytical chemistry literature. The extended Kalman filter[25] has been used to model nonlinear systems (e.g. the estimation of rate constants in the example above) through a linearization of the equations but, like most nonlinear methods, convergence can be slow and subject to initial estimates. Often, several passes are needed, defeating the advantages of recursion. A more successful application has been the adaptive Kalman filter,[26] which examines the innovations sequence to detect model errors and effectively turns the filter off in those regions by using an inflated measurement variance estimate. This allows the filter to be applied in situations where strict adherence to the model is not a certainty.

5.6 Other Filters

This section has only scratched the surface of digital filter design, focusing on those filters which are most commonly implemented in analytical chemistry. The reader should be aware that nonrecursive filters with more desirable transfer characteristics, such as a flatter stop band, can be designed with relatively little additional effort, and is referred to appropriate texts on the subject.[9–11] Even more flexibility can be achieved with recursive filters, with characteristics analogous to the Butterworth and Chebyshev designs described earlier for analog signal processing. The popularity of polynomial least-squares filters appears to be a consequence of their intuitive simplicity and the fact that, although not necessarily optimal, they are sufficient for many applications.

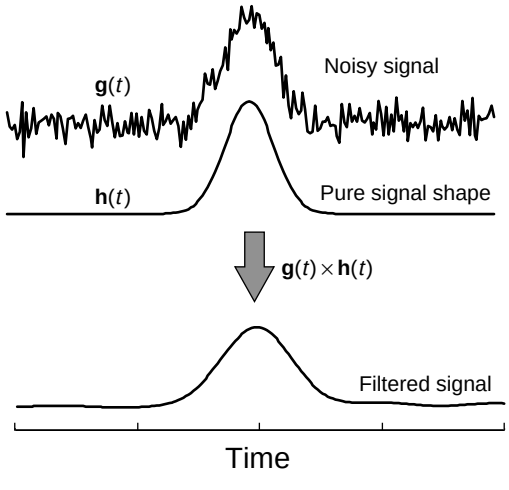

Figure 22 Illustration of the application of a matched filter ($\mathbf{h}(t)$) to a noisy signal ($\mathbf{g}(t)$).

The subject of optimal filtering is revisited in the next section with the Weiner filter in the Fourier domain. In terms of optimal filtering in the time domain, however, one additional filter, the matched filter, deserves mention because it often appears in the analytical literature. With a matched filter, the filter coefficients are obtained simply by normalizing the shape of the pure signal. This is illustrated in Figure 22 with a noisy Gaussian. For white noise, the matched filter is optimal in the sense that it produces the largest SNR, interpreted as the maximum value divided by the baseline noise. Unfortunately, it requires an advance knowledge of the signal shape and has the undesirable consequence of broadening the peak. The optimality of the matched filter derives from its connection to regression. This connection, as well as the relationship between Kalman filtering and regression, has been described by Erickson et al.[27]

6 DOMAIN TRANSFORMATIONS

6.1 Introduction

In the context of signal processing, a domain transformation can be defined as a mathematical or physical process that converts a sequence of measurements into an alternative representation which retains all of the information in the original sequence. A domain transformation is distinguished from a simple domain conversion, such as scaling or current-to-voltage conversion, in that it involves a redefinition of the ordinal variable. As such, the procedures are comparatively complex.

There are two principal reasons why domain transformations are used in chemistry. The first is so that information can be represented in a form commonly used for interpretation. A familiar example is Fourier transform infrared (FTIR) spectroscopy in which the signal, collected by means of an interferometer in the time domain, must undergo a transformation in order to represent it as the familiar plot of transmittance versus wavenumber. The second use of domain transformations is to allow certain operations to be carried out on signals with greater ease. As the objective of signal processing is to separate the pure signal from the noise, transformations which provide a better distinction between these two elements of the signal are useful.

Although there are a large number of possible domain transformations that can be employed, this section will focus on three which have been particularly useful in analytical chemistry: the FT, the WT and the HT.

6.2 Fourier Transforms

Without a doubt, the most widely encountered domain transformation in chemistry is the FT. In addition to being

a useful stand-alone signal processing tool, the FT has become an integral part of many instrumental methods (FTIR, FT/NMR (nuclear magnetic resonance), FTRS (Fourier transform Raman Spectroscopy) and FTMS (Fourier transform mass spectrometry)).[28,29] Although the FT can be applied to both continuous and discretely sampled functions, it is the latter which dominates instrumental applications and will be the focus of this section. The section begins with a basic description and simple illustration of the principles of the FT and concludes with some examples of its application to signal processing. Abundant supplementary information can be found in the literature.[13,14,28–34]

The fundamental principle behind the discrete FT is that any signal sampled at equal intervals in the time domain can have the sampled points reproduced by the addition of a finite number of sinusoids at defined frequency intervals with variable amplitude and phase. This is illustrated in Figure 23 with the simple example of a sampled square wave. Figure 23(a) shows the square wave with the sampled points and the reconstruction using the combination of sinusoids. Although the reconstruction does not match the square wave exactly, it does reproduce the sampled points exactly, which is its only requirement. If the square wave were sampled more frequently, a larger number of sinusoids would be required for reconstruction. Figure 23(b) shows the individual sinusoids added to give the reconstruction, including the DC (direct current) offset (sine wave with frequency of zero).

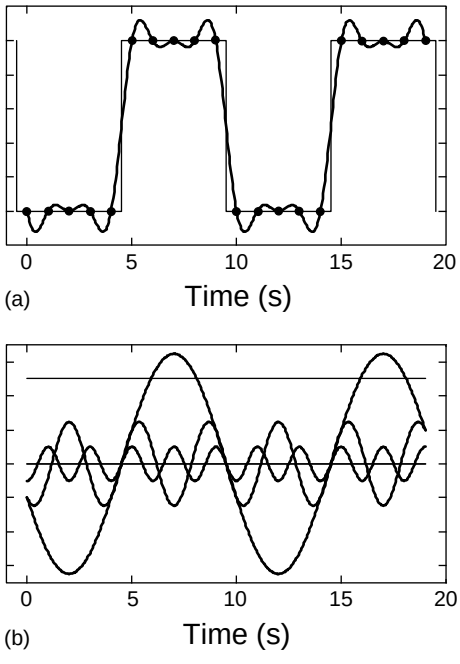

Figure 23 (a) Illustration of the reconstruction of a sampled square wave with a sum of sinusoids. (b) The individual sinusoids used for the reconstruction.

Although two cycles of the square wave are shown in Figure 23 for clarity, the FT is based on a single cycle of 10 points. The reconstruction of the sampled points from the sinusoid terms can be represented as

$$h(t) = \sum_{n=0}^{N/2} C_n \cos\left(\frac{2\pi n f_s t}{N} + \phi_n\right) \qquad (37)$$

In Equation (37), $h(t)$ represents the reconstructed signal at time t, f_s is the sampling frequency, N is the number of points sampled, and C_n and ϕ_n represent the amplitude and phase of the nth sinusoid, which has a frequency of $f_n = (n/N)f_s$. Note that an equivalent representation using sines rather than cosines could have been written simply by adding 90° to ϕ_n, but a cosine expansion is more consistent with FT calculations. For the example in Figure 23, the coefficients C_n and phase angles ϕ_n are given in Table 2 (a sampling interval of 1 s was assumed). Several points should be noted here. First, two sets of amplitudes and angles are given in the table to illustrate an ambiguity in this type of representation – the same result can be obtained by changing the sign of any of the coefficients and shifting the corresponding phase angle by 180°. A second important point is that, whichever set of values is used, there are 12 parameters provided to describe the sinusoids (6 amplitudes and 6 phase angles). Given that we are representing 10 points in the time domain, it would seem that an excessive number of parameters is needed to describe the signal in the Fourier (frequency) domain. However, this is misleading, because the mathematical restrictions can always fix the phase angles for 0 Hz and the Nyquist frequency (0.5 Hz in this case) to be 0°. Therefore, an equal number of values can be used to represent the signal in both domains. Finally, it should be noted that the periodic nature of the sinusoidal basis functions will give rise to a periodic reconstruction even if the original signal is not periodic. This does not mean that nonperiodic signals cannot be transformed, but it should be kept in mind that the FT will treat them as if they are periodic. Discontinuities in amplitude between the beginning and the end of a signal sequence will be reflected in the high frequency components of the FT.

The ambiguity which arose in the amplitude/phase representation of the FT can be resolved by exploiting

Table 2 Amplitudes and phase angles for simple FT example

n	0	1	2	3	4	5
f_n (Hz)	0	0.1	0.2	0.3	0.4	0.5
C_n	0.5	−0.647	0	−0.247	0	−0.1
ϕ_n (°)	0	−72	0	−36	0	0
C_n	0.5	0.647	0	0.247	0	0.1
ϕ_n (°)	0	108	0	144	0	180
A_n	0.5	−0.2	0	−0.2	0	−0.1
B_n	0	0.616	0	0.145	0	0

the fact that a phase shifted sinusoid can be represented as a linear combination of sine and cosine terms. Therefore, an equivalent form of Equation (37) is Equation (38):

$$h(t) = \sum_{n=0}^{N/2} A_n \cos\left(\frac{2\pi n f_s t}{N}\right) + B_n \sin\left(\frac{2\pi n f_s t}{N}\right) \quad (38)$$

For the square wave example, the coefficients A_n and B_n are also given in Table 2. Although the phase angle has been removed, the same number of parameters as before is required to describe the signal, but there is no ambiguity. The basic objective of the FT is to obtain the coefficients A_n and B_n. Mathematically, this is done by separating the sine and cosine terms through complex arithmetic, recalling Euler's relationship (Equation 39):

$$e^{-i\theta} = \cos\theta - i\sin\theta \quad (39)$$

where $i = \sqrt{-1}$.

The discrete FT is mathematically defined as follows. Given a series of N measurements in the time domain, h_k, where $k = 0 \ldots N-1$, the N complex coefficients of the FT, H_n, where $n = 0 \ldots N-1$, are given by Equation (40):

$$H_n = \sum_{k=0}^{N-1} h_k e^{2\pi i k n/N} \quad (40)$$

This calculation results in N real coefficients and N imaginary coefficients which are related to the coefficients A_n and B_n in Equation (38). However, the total number of coefficients here is $2N$, whereas the total number in Equation (38) was $N+2$, which suggests that there is some redundancy. This arises because the FT produces coefficients at both positive and negative frequencies. This is illustrated in Figure 24 which shows the actual FT of the sampled square wave in Figure 23. Both the real and imaginary parts of the FT are shown and the labels on the x axis indicate the correspondence between the coefficient number, n, and the frequency. Note that 0 Hz (DC) and the Nyquist frequency (f_N) are represented only once in the mapping, whereas both positive and negative values are shown for other frequencies. Furthermore, there is a symmetry between the positive and negative frequencies such that Equation (41) holds:

$$H(-f) = H(f)^* \quad (41)$$

where the asterisk indicates the complex conjugate. This symmetry arises from the fact that h_k is a real function and has no imaginary components.

The representation of the real and imaginary parts of the FT as shown in Figure 24 is an unambiguous presentation of the transform and is the one used in calculations, but it is not normally the one shown in practice. Typically, figures show an amplitude spectrum or a power spectrum and (less frequently) a phase spectrum. Unfortunately, there is a great variability in the scaling and presentation of these spectra, so caution needs to be employed in their interpretation. One way to calculate the amplitude spectrum is to calculate the modulus of H directly (Equation 42):

$$\begin{aligned} \mathrm{Amp}(f) &= \sqrt{\mathrm{real}(H(f))^2 + \mathrm{imag}(H(f))^2} \\ &= \sqrt{H(f)H(f)^*} = |H(f)| \end{aligned} \quad (42)$$

The results of this calculation for the square wave example are shown in Figure 25(a), where the frequency values have also been properly ordered. Because of the symmetry of the figure for real data, the amplitude spectrum is often represented as simply the right-hand side, excluding negative frequencies. If this were done here, it is clear that the amplitude spectrum would not be consistent with the values given in Table 2. There are three reasons for this. First, the amplitude spectrum calculated in this way will always give positive values because of the ambiguity in the sign of the square root. Second, there is a scaling factor of $1/N$ needed to go between Figure 25(a) and Table 2. This scaling factor normally appears in the *inverse Fourier transform* (IFT), defined by Equation (43):

$$h_k = \frac{1}{N} \sum_{n=0}^{N-1} H_n e^{-2\pi i k n/N} \quad (43)$$

(Note that the IFT is essentially the same as the forward FT except for the sign change and the scaling factor.)

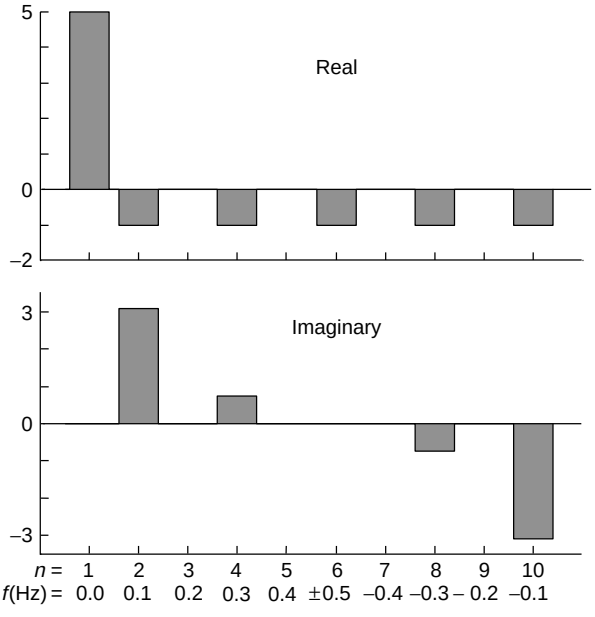

Figure 24 FT of the sampled square wave in Figure 23(a) (one cycle of 10 points only) showing the real and imaginary components and the mapping of points to the frequency domain.

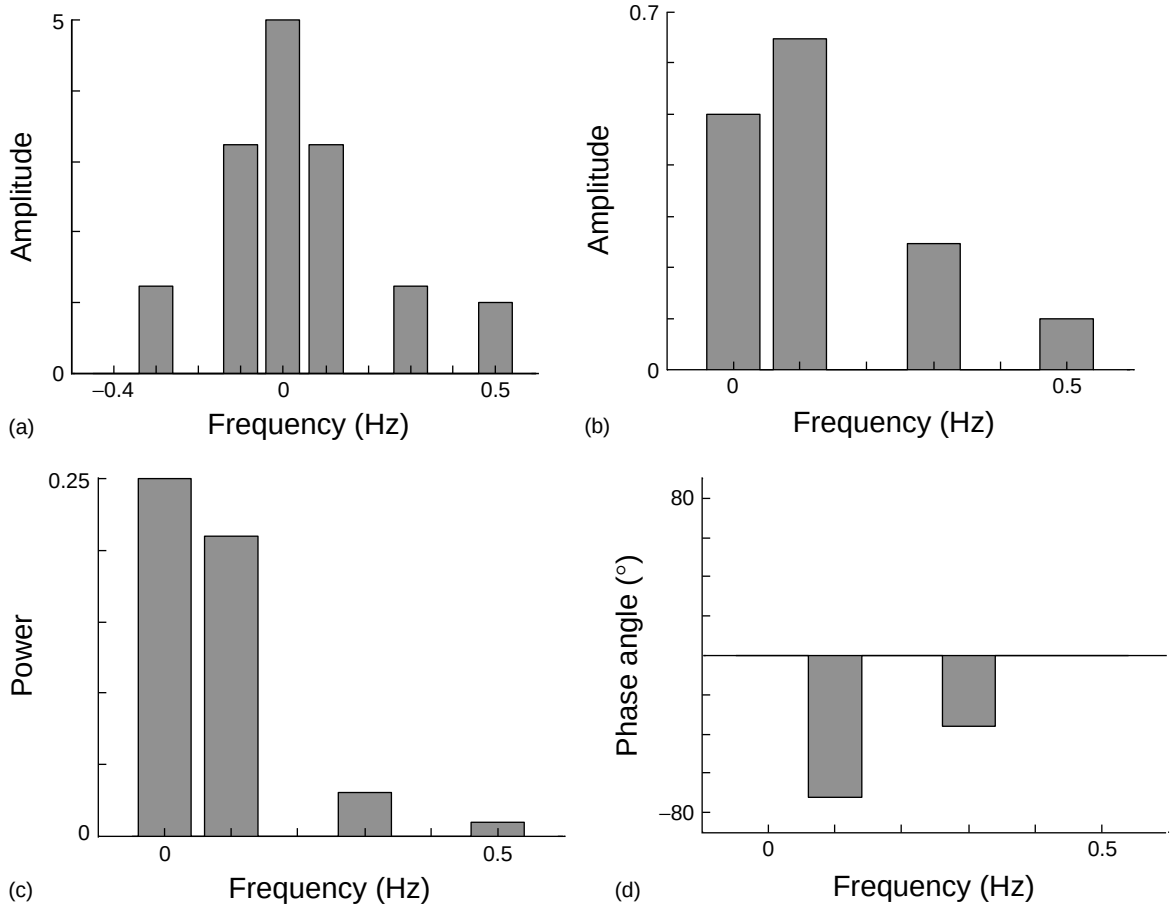

Figure 25 Alternative representations of the FT of the sampled square wave in Figure 23(a). Parts (a) and (b) are different representations of the amplitude spectrum sometimes used; (c) is the power spectrum and (d) is the phase spectrum.

Finally, in order to arrive at an amplitude consistent with Table 2, it is necessary to combine positive and negative frequencies (except for DC and f_N). Thus an alternative definition of the amplitude spectrum (nonnegative frequencies only) is Equations (44):

$$\begin{aligned} \text{Amp}(0) &= \frac{1}{N}|H(0)| \\ \text{Amp}(f) &= \frac{1}{N}(|H(f)| + |H(-f)|) = \frac{2}{N}|H(f)| \\ \text{Amp}(f_N) &= \frac{1}{N}|H(f_N)| \end{aligned} \tag{44}$$

This representation of the amplitude spectrum is shown in Figure 25(b) and, except for the signs, is consistent with the data in Table 2.

The power spectrum (or power spectral density function) is often used in place of the amplitude spectrum. As its calculation involves squaring the amplitudes, the same scaling inconsistencies exist here as for the amplitudes, so care should be taken. The power of the signal in the two domains is related through Parseval's theorem (Equation 45):

$$\sum_{k=0}^{N-1} |h_k|^2 = \frac{1}{N} \sum_{n=0}^{N-1} |H_n|^2 \tag{45}$$

One method to arrive at a valid power spectrum is to use a set of equations similar to those given in Equation (44) for the amplitude, replacing amplitude with power by squaring each of the modulus terms and each N. This gives the power spectrum in Figure 25(c). It can be verified that the sum of the elements is equal to 0.5, which is equal to the mean squared value of the signal in the time domain. There are other aspects to the calculation of power spectra, such as windowing methods to prevent leakage among frequencies and improve the quality of the spectral estimation, but these will not be described here.

Although there are some variations in the manner of calculation of amplitude and power spectra, these are not especially serious when the FT is used for descriptive purposes. The phase spectrum, which plots the phase angle as a function of frequency, is less useful than the amplitude spectrum in most instances.

The phase spectrum for the current example is shown in Figure 25(d). As the negative frequencies contain redundant information by symmetry, only the positive half of the phase spectrum is shown. The phase angles are calculated from Equation (46):

$$\phi(f) = \tan^{-1}\left(\frac{\text{imag}(H(f))}{\text{real}(H(f))}\right) = \arg(H(f)) \qquad (46)$$

As in the amplitude calculation, this equation will have an ambiguity due to the fact that the angle calculated will always be between -90° and $+90^\circ$. Table 2 indicates that, in this case, positive amplitudes should produce phase angles outside this range, but this is clearly not reflected in the phase spectrum. Therefore, although the amplitude/power and phase spectra are the most common descriptive forms of the FT, the indeterminacy of the resultant parameters means that they cannot actually be used to regenerate the original signal in most cases. It should also be noted that Equation (46) involves the ratio of two numbers and this can cause problems in the phase angle calculation when both terms are very close to zero, since round-off error leads to an arbitrary phase angle. Although this is of no importance in the final result (the amplitude of the frequency component is zero), it can complicate the interpretation of the phase spectrum.

As already noted, the calculation of the discrete FT can be carried out using Equation (40), but for most real applications involving a substantial number of data points, the application of this equation is impractical due to the large number of operations required (on the order of N^2). For this reason, few applications employed the FT until the mid-1960s, when the FFT algorithm was popularized by Cooley and Tukey.[2] The FFT greatly reduced the number of operations required (of the order of $N \log N$) and made transformations practical for a wide variety of problems. Although beautifully elegant in its partitioning of the problem, a somewhat annoying requirement of the original algorithm was that it required the number of points to be equal to a power of two. Improvements on the original algorithm have largely removed this restriction (although they are not quite as efficient), and FTs are now calculated with ease for most signals of moderate length using a variety of software packages.

One of the principal applications of the FT is the conversion of data recorded by instruments such as FTIR spectrometers, where it is needed to transform the measurements from the time domain to the frequency domain (or vice versa) before it can be interpreted in the conventional way. In addition to this, the FT is used for a great many signal manipulation purposes, such as smoothing, deconvolution, and interpolation. Some of these applications are now illustrated.

A great deal of the utility of the FT in signal processing derives from the convolution theorem, which states that the convolution of two signals in the time domain is equivalent to the element-by-element multiplication of the functions in the frequency domain. Mathematically, if g and h are functions in the time domain and G and H are the corresponding functions in the frequency domain, Equation (47) holds:

$$g(t) * h(t) = \text{IFT}[G(f)H(f)] \qquad (47)$$

where the asterisk indicates the convolution of the two functions and IFT indicates the inverse transform. As digital filtering is the convolution of filter coefficients with a noisy signal, this immediately leads to an application in Fourier smoothing. The difference here is that we can specify the transfer function of the filter exactly through its FT. This is illustrated in Figure 26 with the smoothing of a noisy Gaussian. The FT of the noisy signal is first calculated and then both the real and imaginary parts are multiplied by the ideal transfer function which sets all of the high frequencies where there is no significant signal

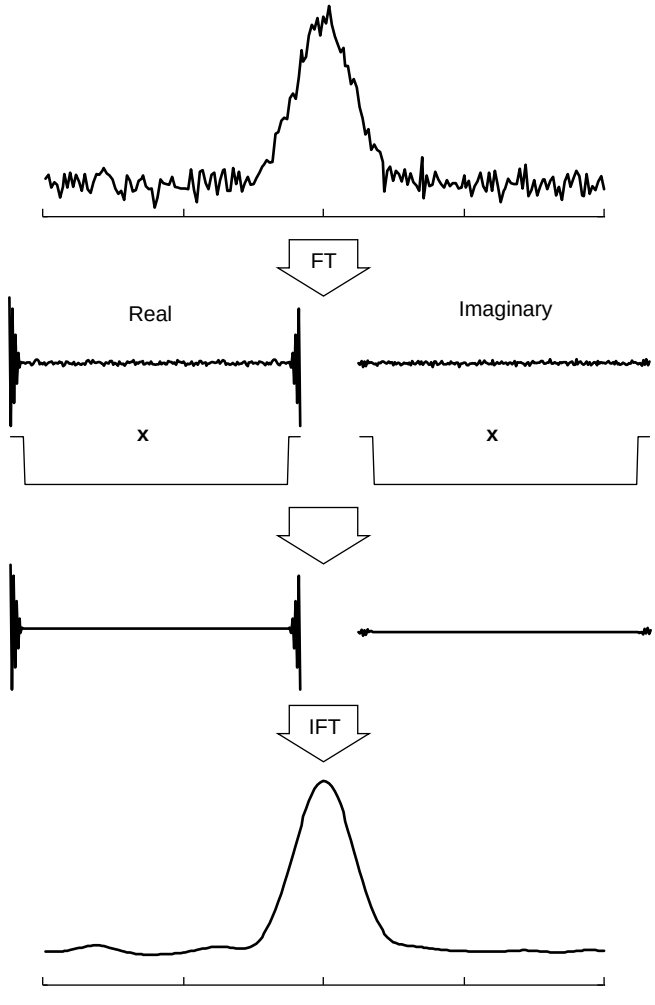

Figure 26 Fourier filtering of a noisy signal.

contribution to zero. Note that both positive and negative frequencies must be included in this multiplication, which is why the transfer function looks somewhat different than that shown earlier. After the multiplication is carried out, an IFT is applied to the result to give the smoothed signal in the time domain.

Although this procedure works very well, it has some drawbacks. First, it is slower than a digital filter and cannot be done in real time because the entire signal is required. Second, artifacts such as the oscillations near the tails of the peak are often observed due to the sharp transition of the transfer function. More severe distortion can result if the cut-off frequency is moved closer to the signal components, but there will be less noise reduction if it is moved to higher frequencies. To avoid this characteristic of the ideal filter, a more gradual decrease in the transfer function is often employed. Such a function is sometimes referred to as an apodization function. If the FT of the pure signal is designated as S and that of the noise as N, it can be shown that the transfer function of the optimal, or Weiner, filter Φ is given by Equation (48):

$$\Phi(f) = \frac{|S(f)|^2}{|S(f)|^2 + |N(f)|^2} \tag{48}$$

This is the transfer function that will give the optimal reproduction of the true signal in the least-squares sense. The difficulty with applying this filter is in the estimation of $S(f)$ and $N(f)$ for the pure signal and noise, but that is beyond the scope of this article.

In addition to convolution, the FT can aid in the deconvolution of two signals. If h represents the convolution of two signals, f and g in the time domain ($h = f^*g$), where g is known, the deconvoluted signal f can be obtained through an element-by-element division in the frequency domain (Equation 49):

$$f(t) = \text{IFT}\left[\frac{H(f)}{G(f)}\right] \tag{49}$$

This is illustrated in Figure 27 where a simulated spectral doublet has been convoluted with the slit function of the spectrometer which smears the two peaks. Through Fourier deconvolution, it is seen that the original line shape can be recovered. Practically speaking, there are a number of difficulties associated with this procedure. First, one of the convolution functions needs to be known in advance. Second, the division presents problems when the denominator in Equation (49) is close to zero and adjustments need to be made in this case. Depending on how this is done, artifacts such as those apparent in the baseline of the deconvoluted spectrum of Figure 27 can result. Finally, the presence of noise can lead to additional complications in the deconvolution. In a best-case scenario, however, this type of deconvolution can be

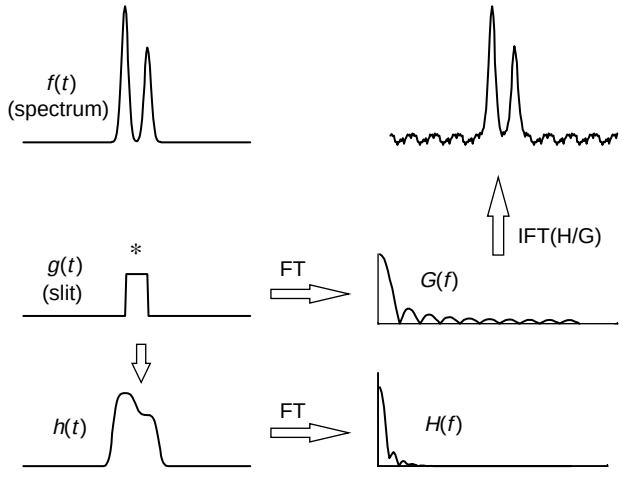

Figure 27 Convolution of a spectral doublet with a slit function followed by Fourier deconvolution to recover the original line shapes.

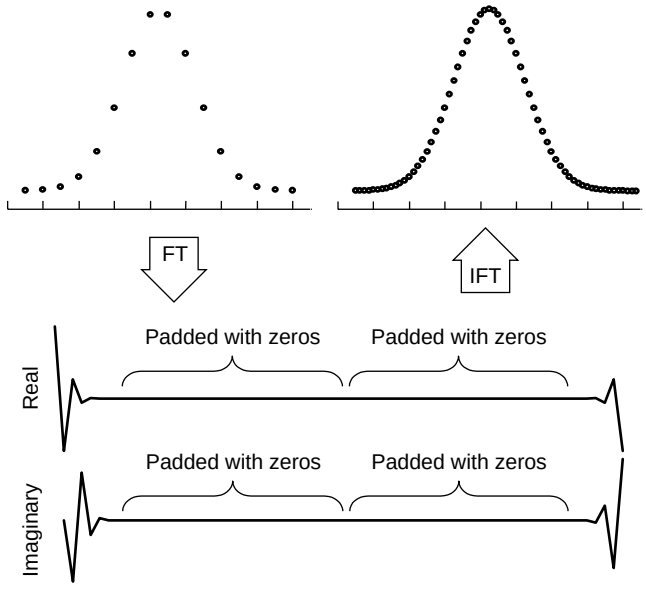

Figure 28 Fourier interpolation of an undersampled signal by a factor of four. Note that the FT has been padded with 24 zeros in both the positive and negative frequency regions to give a total of 64 points.

used to improve the resolution of the instrument after the measurements have been obtained.

As a final example of the use of the FT in signal processing, Figure 28 shows an example of function interpolation. This procedure, which has been referred to as the zoom FT, is based on the fact that the Nyquist frequency is directly related to the sampling frequency. If the FT of a signal is padded with zeros at the frequency limit where there is little signal contribution, the Nyquist frequency can be increased,

and consequently the sampling interval decreased. This essentially interpolates the function between existing measurements. Although this approach has been used in certain applications, such as locating the peak maxima in undersampled mass spectrometry peaks, it should be used with caution because the zero-padding makes implicit assumptions about the form of the function between the original points which may be erroneous.

6.3 Wavelet Transforms

Although a more recent development than the FT, the WT approach is gaining increased acceptance as a signal processing tool for the analytical chemist. The first applications in the chemical literature of the WT as a denoising, smoothing and data compression procedure appeared in the early 1990s, and their frequency of mention has increased steadily. The utility of the WT for noise reduction purposes rests largely on its decomposition of the signal into successive levels of high- and low-frequency components. In data compression applications the decimation filter property of the WT is useful, effectively reducing the number of elements needed to represent the signal with minimal loss of information. Numerous algorithms for the WT have been devised, with the most popular being the recursive form of the discrete wavelet transform (DWT) attributable to Mallat,[35] the generalization of which is known as the wavelet packet transform (WPT).[30,36] An increasing number of software packages are now available for performing DWTs and WPTs as well as related functions in a relatively straightforward manner.

Like the FT, the WT converts the data into a more useful domain for signal processing by projecting the observed signal onto a set of orthogonal basis functions. In the FT, the signal is projected into the frequency domain using sinusoids as the basis functions. In the frequency domain, the basis functions are localized, but when transformed to the time domain the functions extend globally along the time axis. In contrast, the WT uses basis functions that are both localized in the frequency and time domains to project the data into the wavelet domain. The WT, therefore, has perhaps a more intuitive appeal for some who routinely deal with signals that are time localized, such as chromatographic or spectroscopic measurements. A very readable introduction to denoising and compression of chemical signals using wavelets has been written in tutorial fashion by Walczak and Massart.[37]

In the popular DWT pyramid algorithm of Mallat,[35] a recursive decomposition of the signal is performed using both high- (**H**) and low-pass (**L**) filter matrices which are rectangular (each $n/2 \times n$, if the observed signal vector **x** has n elements). The coefficients of these decimation filters depend on the family of wavelets that are used; the Daubechies family appears to be the most popular in chemical applications. The portion of the signal that passes through the low-pass filter is typically called the mth-level approximation to **x**, $\mathbf{a}_m$, and the portion of the signal that is rejected by **L** (passes through the high-pass filter) is referred to as the mth-level detail in **x**, $\mathbf{d}_m$ ($\mathbf{d}_m$ can be considered to be the information in $\mathbf{a}_{m-1}$ not included in the approximation $\mathbf{a}_m$). As **H** and **L** are decimation matrices with a down-sampling rate of two, the number of elements in both **a** and **d** drops by a half at each level of approximation (an effective doubling of the sampling interval). The algorithm requires the number of measured channels in the signal vector to be a power of two, although this requirement can be side-stepped with zero-padding. Edge effects can often result from the filter convolutions; however, signals that are not of a length 2^J (J is an integer) can be zero-padded on both ends of the signal vector to sterilize the distortion.

The DWT algorithm is logistically summarized in Figure 29(a). Clearly, the detail at a given level is not used in subsequent approximation steps of the pyramid algorithm, because $\mathbf{a}_m$ is only approximated from $\mathbf{a}_{m-1}$, and the possibility exists that important information was rejected by the low-pass filter and resides in the detail vector. When the detail vectors are also incorporated into the decomposition, the WPT algorithm results (illustrated in Figure 29b). The structure of the **L** and **H** filters used in the decomposition is shown in Figure 30 for the Daubechies-4 wavelet (four coefficients). Note that the coefficients for **H** are the same as for **L** except that the order is reversed and alternate signs are changed. Normally, the **L** and **H** matrices are concatenated and multiplied by the concatenated basis vectors at each level. With this approach, the arrangement of **L** and **H** matrices will change at each level, as shown in Figure 31.

With the decomposition of the original signal accomplished, it is possible to examine the coefficients of the approximation and detail levels. This is analogous to examining the coefficients of a Fourier decomposition, except that the wavelet decomposition is arranged in two dimensions – time and frequency. As in Fourier denoising applications, it is desirable to eliminate or reduce coefficients believed to be associated with noise, and retain coefficients reflecting information in the original signal. In the WPT, however, not only must coefficients be selected in the frequency realm, but also in the level of approximation. In order to determine the appropriate basis vectors for reconstruction of the signal, it is useful to examine the approximation and detail vectors resulting from the transform. A simple example of denoising using the WPT is presented in Figure 32. In this example, the WPT has only been carried out to two levels for purposes of illustration. At level 2, it is apparent that three of the

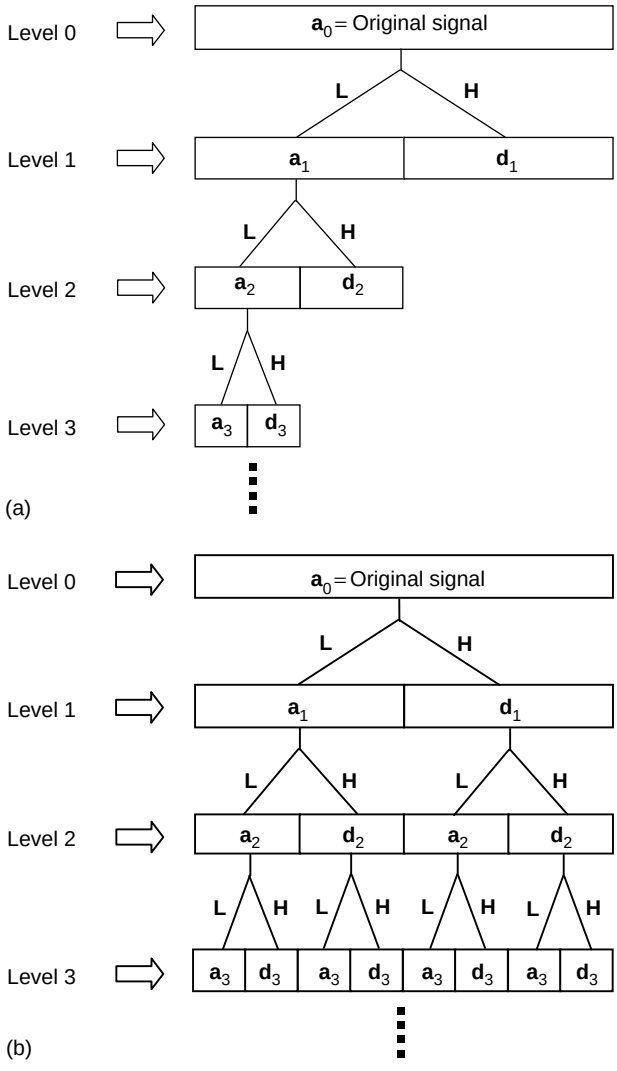

Figure 29 Illustration of (a) the DWT and (b) the WPT.

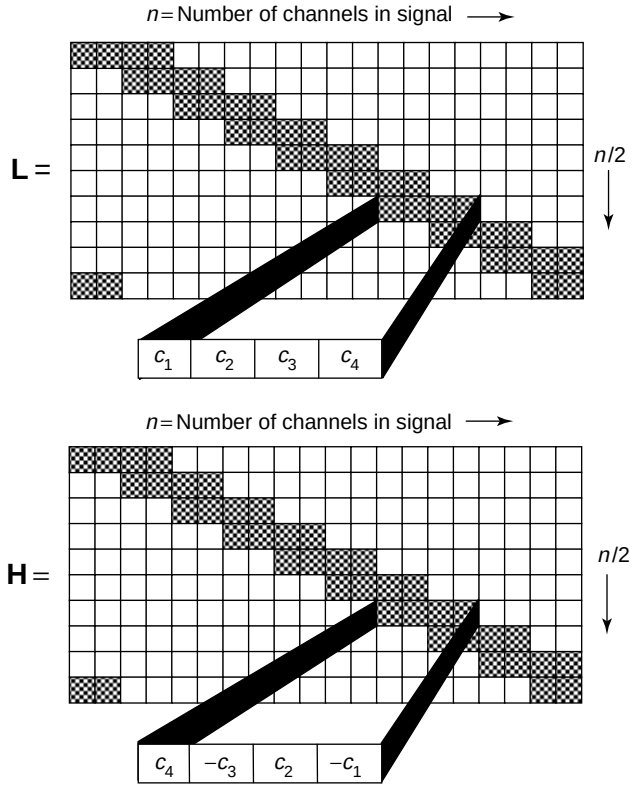

Figure 30 Illustration of the **L** and **H** matrices (four coefficients) used in the WPT to generate the approximation (**a**) and detail (**d**) vectors, respectively, at successive levels of resolution. For the Daubechies-4 wavelet, the coefficients are $c_1 = 0.4830$, $c_2 = 0.8365$, $c_3 = 0.2241$, and $c_4 = -0.1294$. Note that the coefficients will not change for different levels of resolution, but the size of the filter matrix will depend on n, the number of channels in the signal at the previous level of resolution.

four vectors contain little useful information, so these are set to zero before the inverse wavelet transform (IWT) is carried out, resulting in a reduction in the noise. It should be noted that no attempt was made to optimize the denoising in this example, and decompositions to additional levels may have allowed further improvement. Denoising with the WT involves more options than that for the FT, such as the selection of a set of basis functions, the level of decomposition used, and the choice of basis vectors to set to zero.

Because of the wide range of possibilities, algorithmic methods of basis selection have been proposed, with the most intuitive and successful using the minimum entropy (or maximum information) condition of Coifman and Wickerhauser.[38] Best-basis selection according to the minimum entropy condition proceeds on the principle that the most useful basis vectors will be those that contain the most information, and informative vectors will tend to be those which have some large coefficients and some small ones. The coefficients of uninformative basis vectors will largely be the same. The minimum entropy condition in best-basis selection is typically applied by seeking the basis vectors which have the greatest number of coefficients above a preset threshold value.[39] Another method of selecting the best basis vectors is the minimum description length (MDL) method.[40,41] The MDL is primarily used when data compression is desired, and proceeds on a version of the principle of parsimony, seeking the basis vectors which contain the most information in the fewest coefficients.

With the best set of basis vectors selected from the full decomposition, additional signal adjustment can be made by using hard or soft thresholding measures on the remaining coefficients. In hard thresholding, coefficients above a preset threshold are retained, whereas coefficients below this level are discarded. Soft thresholding can also entail zeroing of coefficients below the threshold,

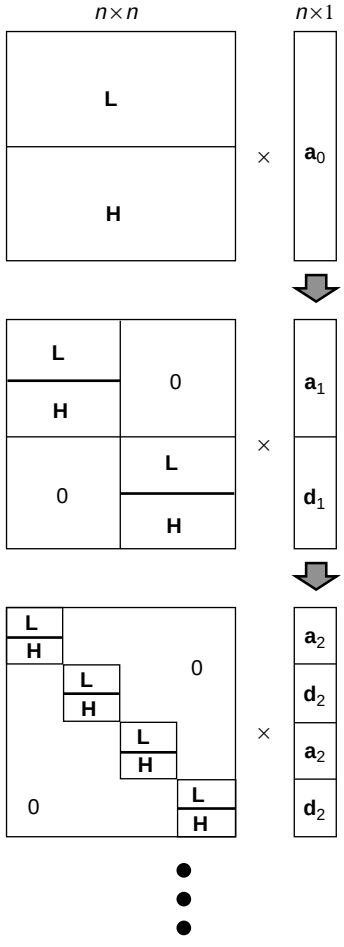

Figure 31 Arrangement of the **L** and **H** filter matrices through the first three levels of the WPT.

but coefficients above the threshold are also typically shrunk towards zero by an amount inversely proportional to their magnitude. Evidently the selection of the threshold value is crucial in these procedures, and several methods exist for estimating the optimal threshold value, including estimating the threshold based on the level of noise,[42] and setting the threshold as a percentage of the largest coefficient.[43] Typical hard, and soft thresholding functions are shown in Equations (50) and (51):

hard thresholding:

$$c_{\text{new}} = \begin{cases} 0, & \text{if } |c_{\text{old}}| < T \\ c_{\text{old}}, & \text{if } |c_{\text{old}}| \geq T \end{cases} \tag{50}$$

soft thresholding:

$$c_{\text{new}} = \begin{cases} 0, & \text{if } |c_{\text{old}}| \leq T \\ \text{sign}(c_{\text{old}})(|c_{\text{old}}| - T) & \text{if } |c_{\text{old}}| > T \end{cases} \tag{51}$$

As an alternative to using threshold values to select relevant coefficients, wavelet smoothing can be achieved by simply discarding detail vectors and performing the inverse WPT from the desired approximation vectors. Although this technique has the potential to achieve greater compression ratios, it is a perilous operation when one lacks knowledge of the location of relevant information in the wavelet decomposition – some useful information may well be contained in some detail vectors. With this possibility looming, it is generally recommended that wavelet smoothing by discarding detail vectors be reserved for situations in which extensive knowledge of the signal allows for educated detail removal.

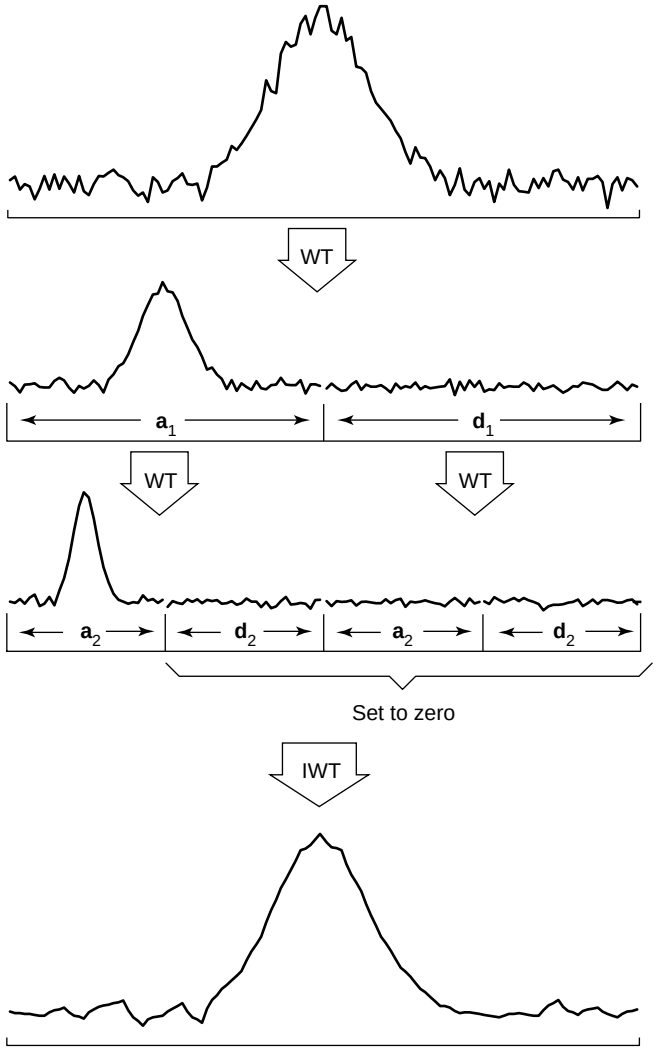

Figure 32 Simple illustration of signal denoising using the WPT.

With basis selection, and coefficient adjustment complete, it is possible to approximate the original signal in the original resolution. This domain is revisited by passing the selected basis vectors back through the high- and low-pass filters. To make the filters interpolation rather than decimation filters, the conjugates of **H**, and **L** are employed. Based on the orthonormality of the

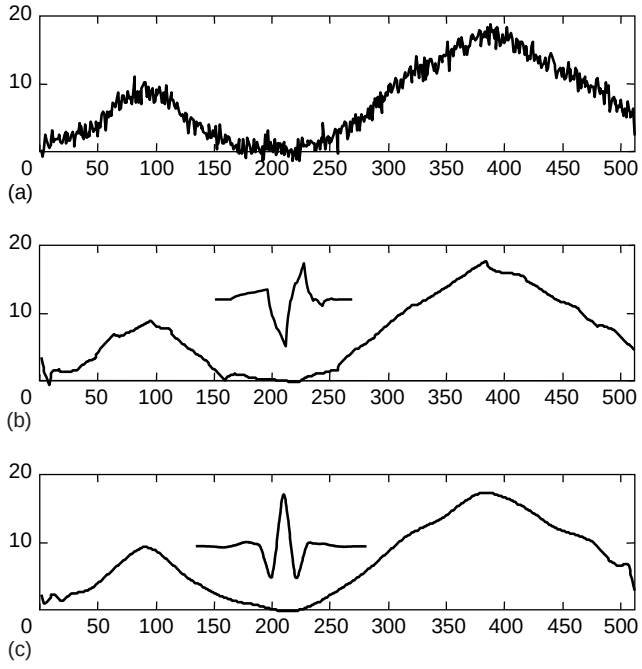

Figure 33 Illustration of signal denoising using Coifman's best-basis selection algorithm and coefficient adjustments using hard thresholding: (a) original simulated noisy spectrum, (b) denoised spectrum using the Daubechies-4 wavelet (inset), and (c) denoised spectrum using the Coiflet-3 wavelet (inset).

two matrices, the conjugates of **H** and **L** are equivalent to the transposes. Therefore the IWT proceeds straightforwardly through the up-sampling filters $\mathbf{H}^{\mathrm{T}}$ and $\mathbf{L}^{\mathrm{T}}$ (transposes of the matrices in Figure 31) until the desired resolution is achieved, with the denoised signal resulting.

Figure 33 illustrates the utility of the WT in signal denoising using a simulated spectrum. White noise was added at a level corresponding to $\sigma_{\mathrm{noise}} = 0.1$, giving the noisy spectrum in Figure 33(a). The first wavelet chosen for the transform was the Daubechies-4 ($c_1 = 0.4829$, $c_2 = 0.8365$, $c_3 = 0.2241$, $c_4 = -0.1294$). Best-basis selection was performed using the minimum entropy procedure of Coifman, and hard thresholding was used on the chosen basis vectors. The results of the overall wavelet denoising with the Daubechies-4 wavelet are shown in Figure 33(b). To illustrate the effect that different wavelet families can have, the Coiflet-3 wavelet was also used, and the results shown in Figure 33(c). In practice, a comparison would more probably be made between different members of a family to find the best result.

6.4 Hadamard Transforms

Like other transform methods, the HT can be thought of as a transformation from one space to another, with a Hadamard matrix acting as the transformation matrix.[44–47] HTs are one method of gaining the Felgett advantage, or multiplex advantage, as it is often called. The multiplex advantage is a statistical gain in SNR as a result of simultaneously measuring multiple spectral resolution elements. In contrast to dispersive methods, in which a single spectral element is measured at a time, multiplex methods measure several coincident spectral elements simultaneously. In order for the multiplex design to prove beneficial from an SNR perspective, the noise in the signal must be considered to be independent of the strength of the incident radiation (i.e. detector noise is the overwhelming noise source). If this condition is met, simple propagation of error reveals that the SNR of a multiplex instrument, relative to a dispersive instrument is $\sqrt{N}$, where N is the number of spectral elements that impinge on the detector at any one time.

Although FT spectrometers are perhaps the best known instruments to utilize the multiplex advantage, the HT spectrometer is also a valuable option. The principles of the HT are based on the concept of Hadamard matrices. As noted above, the benefit of the HT stems from propagation of measurement error into the estimated spectral values. If one observation is made with an inherent detector error of e, then the error in the estimated value is e. However when we wish to estimate several unknowns we can reduce the error associated with a particular estimate by measuring groups of unknowns together in a well-designed fashion. We can subsequently use systems of linear equations to solve for the estimates, and decrease the error in those estimates in the process. The classic analogy is to a weighing scheme for several unknown objects. In the example sketched here, four objects are weighed in experiment A one at a time, with a detector noise level of 0.1. (Experiment A is analogous to a dispersive spectrometer.) In experiment B, two or three objects are weighed together at any one time, although we still only have four total measurements to estimate each individual object's mass. (Experiment B is analogous to a single detector HT spectrometer.)

Experiment A

$$\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} m_1 \\ m_2 \\ m_3 \\ m_4 \end{bmatrix} = \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix} + \begin{bmatrix} e_1 \\ e_2 \\ e_3 \\ e_4 \end{bmatrix}$$

$$\mathbf{W}_{\mathrm{A}}\mathbf{m} = \mathbf{x} + \mathbf{e}$$

$$\mathbf{m} = \mathbf{W}_{\mathrm{A}}^{-1}\mathbf{x} + \mathbf{W}_{\mathrm{A}}^{-1}\mathbf{e}$$

Experiment B

$$\begin{bmatrix} 0 & 1 & 1 & 1 \\ 1 & 1 & 0 & 0 \\ 1 & 0 & 1 & 0 \\ 1 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} m_1 \\ m_2 \\ m_3 \\ m_4 \end{bmatrix} = \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix} + \begin{bmatrix} e_1 \\ e_2 \\ e_3 \\ e_4 \end{bmatrix}$$

$$\mathbf{W}_{\mathrm{B}}\mathbf{m} = \mathbf{x} + \mathbf{e}$$

$$\mathbf{m} = \mathbf{W}_{\mathrm{B}}^{-1}\mathbf{x} + \mathbf{W}_{\mathrm{B}}^{-1}\mathbf{e}$$

Here the x_i terms represent the observed reading on the scales, and the m_i terms represent the estimated mass of the ith object. Through propagation of error, it is relatively easy to show that if the measurement uncertainties for the x_i terms are independent and given by σ_x, then the uncertainties in the masses will be given by Equation (52),

$$\begin{bmatrix} \sigma_{m_1}^2 \\ \sigma_{m_2}^2 \\ \vdots \end{bmatrix} = \sigma_x^2 \text{diag}\,[(\mathbf{W}\mathbf{W}^{\mathrm{T}})^{-1}] \tag{52}$$

where 'diag' indicates extraction of the diagonal elements. Solution of this equation using $\sigma_x = 0.1$ gives $\sigma_m = 0.1$ for all masses in experiment A, whereas the values for experiment B are 0.067, 0.088, 0.088, 0.088 for m_1–m_4, respectively. Clearly, noise reduction in the estimates has occurred via the multiplex advantage.

In HT spectrometers, the weighing design matrix as shown above is embodied by a mask (Hadamard mask) that physically impedes the incidence of some spectral elements while letting others pass through to the detector. Whereas early HT instruments used a moving mask, the current inclination is toward stationary masks whose codes are changed using electrooptical devices. In true HT spectrometers, light is not only blocked from the coagulating detector, but it is also reflected back to a subtracting detector, such that the measured total intensity is the difference of the adding and subtracting detectors. The weighing matrix in these scenarios, **H**, is a series of 1 and (-1) values representing which elements are subtracted and which are added. These matrices are designed based on Hadamard mathematics. When this arrangement is used, the SNR enhancements observed in FT instruments can be achieved. In practice the HT instruments are difficult to construct to the required specifications and thus single-detector instruments are principally used. The weighing matrix used in these systems is the **S** matrix, and the elements are similar to the weighing matrices shown above (zeros and ones). **S** matrices can be easily constructed from Hadamard matrices by removing the first row and column of **H** and changing all -1 elements in **H** to zeros in the **S** matrix. Although closely related to the Hadamard matrices, **S**-matrix methods do not afford the same enhancement in the SNR as **H**-matrix methods because, with N spectral elements, only $(N+1)/2$ may be measured at any one time.

Like the interferogram resulting from the Michelson interferometer, the encodegram is the resulting signal output from a Hadamard mask experiment. The encodegram relates the radiative flux reaching the detector with the position of the Hadamard mask. To convert this signal in the Hadamard domain to the desired frequency domain the inverse HT is used. Given the properties of **S** (orthonormal rows/columns and square) this is easily accomplished by convolution of the encodegram with the inverse of **S**, i.e. $\mathbf{S}^{-1} = \mathbf{S}^{\mathrm{T}}$.

With the use of electrooptic Hadamard masks come new problems with the standard HT. Although these stationary masks remove the problem associated with the continuously moving parts of the FT instruments, nonidealities in the opacity or transmissiveness of the mask require adjustments to the weighing matrices.[47]

When noise is independent of the signal intensity, as is the case when detector noise dominates, Hadamard multiplexing can prove a useful method of improving the SNR of the spectral estimates. In true Hadamard multiplexing noise reduction follows the general formula of Equation (53):

$$\sigma_{\mathrm{HT}} = \frac{\sigma}{\sqrt{N}} \tag{53}$$

where σ_{HT} is the standard deviation of the estimated elements using HT methods, σ is the standard deviation of the detector output (equivalent to the noise level in the same experiment using a monochromator), and N is the number of spectral elements to be estimated. However, most HT instruments employ **S** matrix methods which, at best, allow reduction of the uncertainty of the estimate according to Equation (54):

$$\sigma_{\mathrm{HT}} = \frac{2\sqrt{N}}{N+1}\sigma \approx \frac{2\sigma}{\sqrt{N}} \tag{54}$$

Although the HT has found some utility in analytical applications and is likely to continue to do so in situations for which multiple channel detection or FT methods are unfeasible, its implementation has not been extensive. With the increasing prominence and quality of multichannel detection systems, the multiplex approach of the HT to signal processing is likely to have limited future utility.

7 HIGHER-ORDER SIGNAL PROCESSING

The bulk of this article centers on signal processing methods for first-order data sets – those cases where the signal can be represented as a vector of measurements. In recent years, however, there has been an increased emphasis on the use of higher order data in analytical chemistry. This is particularly true for second-order data sets (matrices of measurements), but increased use of third-order data is also apparent in the literature. This phenomenon can be attributed to three main factors: (a) the demand for new kinds of analytical information and more efficient analytical methods; (b) the increased availability of multichannel detectors, such as photodiode arrays and charge-coupled

devices, as well as rapid scanning instruments; and (c) the development of chemometric methods capable of dealing with multidimensional data. Techniques such as multivariate calibration and pattern recognition are now used routinely, and their application has led to the increased need for signal processing for higher-order data.

In discussing signal processing for higher-order data, it is necessary to make the distinction between the order of the data and the order of the signal. A defining characteristic of a signal is that it exhibits correlation in the ordinal variable for some domain, so a higher-order signal should exhibit correlation in the ordinal variable for each dimension. For example, a collection of spectra from different samples for a multivariate calibration or pattern recognition study would not normally exhibit correlations among the samples, and so this second-order data set can be regarded as a collection of first-order signals. In contrast, spectra obtained during a chromatography or kinetics experiment would result in second-order signals in a second-order data set, because there would be a relationship among the spectra in the time domain. Other combinations are also possible. For instance, if fluorescence emission–excitation spectra were collected for an arbitrary series of samples, we would have second-order signals composing a third-order data set.

For data sets that are composed of first-order signals, signal processing is generally restricted to first-order methods such as those already described. Nevertheless, such signal processing can still have effects across multiple orders and for that reason may be regarded as even more important for higher-order data than for the first-order case. For example, the presence of a variable baseline offset or drift between sample spectra can be detrimental to multivariate calibration methods, but this effect can be minimized by derivative filtering in the spectral domain. The application of such techniques prior to data analysis falls under the subject area of data preprocessing, and includes such methods as mean-centering, baseline subtraction, scaling, smoothing, differentiation, and domain transformation. Choice of an appropriate preprocessing method can be critical and often determines the success or failure of a multivariate analysis application. A complete discussion of these methods is beyond the scope of this article; however, many appropriate texts on chemometrics give more information.[13,48–50]

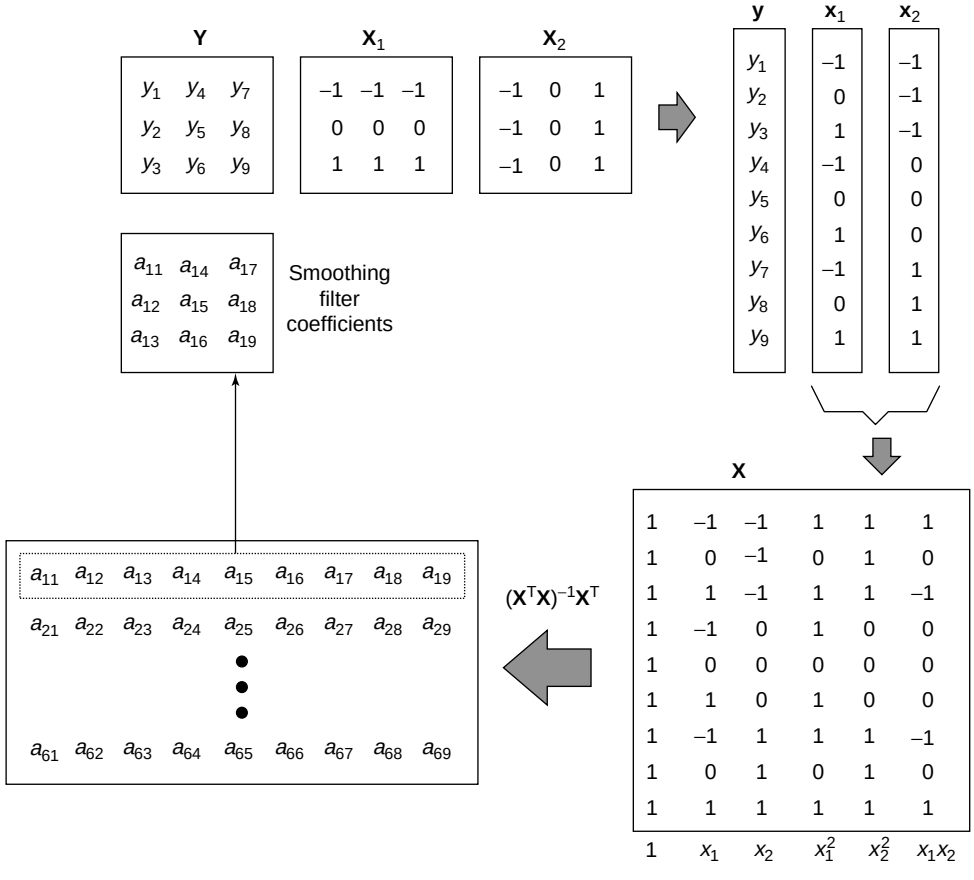

Figure 34 Illustration of the calculation of filter coefficients for a 3 × 3 quadratic filter with an interaction term.

For signals that are truly higher order, first-order signal processing methods can still be used, but other options are also available. In large part, these are extensions of the first-order methods which have already been discussed. In the case of second-order signals, for example, there are two-dimensional (2D) smoothing methods, 2D FTs and 2D WTs. The application of these techniques can offer greater power and flexibility since the characteristics of the signals in both dimensions can be exploited. For example, in the case of spectra collected during a chromatography experiment (a spectrochromatogram), filtering using a nine-point moving average filter in either the time or spectral dimension requires convoluting each signal vector with a 1×9 smoothing vector. However, 2D smoothing could use a convolution of the full matrix with a 3×3 smoothing matrix and the same level of noise reduction would be achieved (in the case of a moving average filter) with less distortion. Understandably, the use of 2D techniques introduces greater complexities in terms of computation, implementation, interpretation and optimization than their one-dimensional (1D) counterparts, but these can be overcome.

As most higher-order signal processing methods are extensions of their 1D counterparts, a detailed discussion is not presented here. However, one example of 2D smoothing is presented as an illustration. In this example, a 3×3 polynomial smoothing filter is used. To demonstrate the design, a quadratic filter with an interaction term was chosen, with the corresponding Equation (55):

$$\hat{y} = b_0 + b_1 x_1 + b_2 x_2 + b_3 x_1^2 + b_4 x_2^2 + b_5 x_1 x_2 \quad (55)$$

where x_1 and x_2 represent the two ordinal variables. The generation of filter coefficients requires the unfolding of matrices representing the ordinal variables and the process is illustrated in Figure 34. The resulting smoothing coefficients are (Equation 56)

$$\mathbf{C} = \begin{bmatrix} -0.111 & 0.222 & -0.111 \\ 0.222 & 0.556 & 0.222 \\ -0.111 & 0.222 & -0.111 \end{bmatrix} \quad (56)$$

The result of the application of this filter to the noisy fluorochromatogram of a mixture of pyrenes in Figure 35(a) is shown in Figure 35(b). Although some noise reduction results, it is not as great as that obtained with a simple 3×3 moving average filter, as shown in Figure 35(c). It is clear that the 2D filter involves the same trade-off between noise reduction and distortion as the 1D filters, but the optimization in the 2D case involves a greater number of options, such as the size and order in each dimension and the inclusion of interaction terms. So far, unlike first-order methods, there have not been extensive studies on the relationship between second-order signal processing methods and the signals they are applied to in chemistry, but this is likely to change as higher-order data become more prevalent.

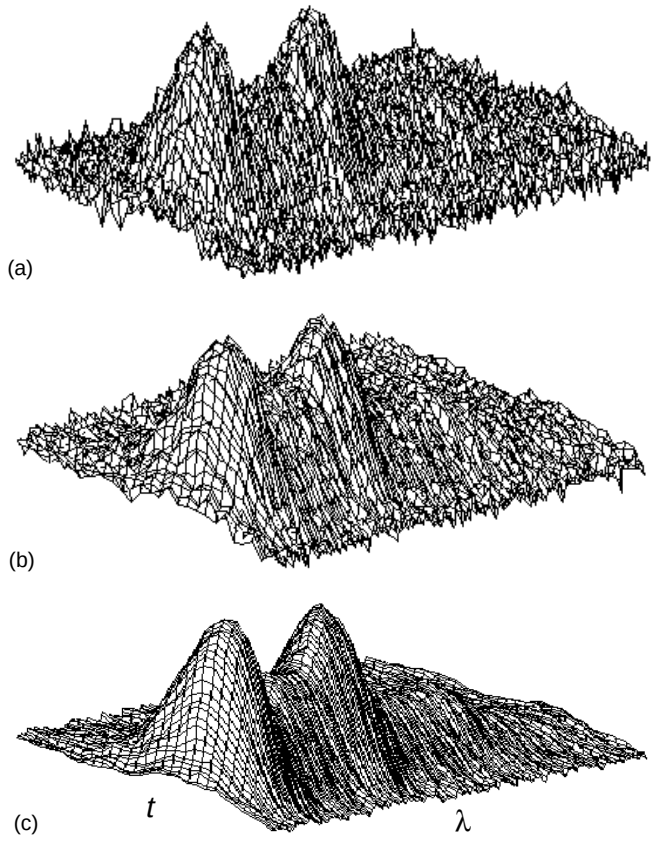

Figure 35 Application of a 2D smoothing filter to noisy data from a chromatogram with multiwavelength fluorescence detection: (a) original data, (b) data filtered with a 3×3 quadratic filter, (c) data filtered with a 3×3 moving average filter.

8 CONCLUSIONS

This article provides a general description of some of the signal processing tools commonly employed in analytical chemistry. As a general principle, it is apparent that all signal processing methods make assumptions about the models for signals and for noise in order to distinguish the two. The power of a particular method in a given application depends on the nature of the assumptions made (very general or very restrictive) and the extent to which they are valid. It is also true that the use of signal processing methods is a double-edged sword. Although the quality of information may be enhanced, it is also possible to distort the signal to the point where results become unreliable. Clearly, a knowledge of the nature of signals, noise and

the capabilities of signal processing methods is essential. For this reason, a significant portion of this article is dedicated to the practical aspects of implementing different methods and their effects on signals.

Developments in signal processing applications to analytical measurements will no doubt continue, particularly for digital signals. Although some methods, such as polynomial smoothing and FT-related techniques, will continue to permeate all areas of analytical chemistry, other methods, such as Kalman filtering and HTs, have found more specialized niches. The impact of WTs is evidence of the ongoing research in signal processing applications. Undoubtedly, future developments will exploit greater computational abilities and present new challenges in application and interpretation. As noted in the previous section, applications to higher-order methods will be a focus of research. In any case, it is apparent that, whether it is the enhancement of fuzzy images from atomic microscopes or the removal of background signals in remote sensing from space, signal processing methods will continue to play a key role in all aspects of analytical chemistry.

ACKNOWLEDGMENTS

The authors gratefully acknowledge the research support of the Natural Sciences and Engineering Research Council of Canada (NSERC) and the Dow Chemical Company. Prof. L. Ramaley is thanked for his helpful comments.

ABBREVIATIONS AND ACRONYMS

A/D	Analog-to-digital
ADC	Analog-to-digital Converter
AU	Absorbance Units
BCD	Binary Coded Decimal
DAC	Digital-to-analog Converter
DC	Direct Current
DSP	Digital Signal Processor
DWT	Discrete Wavelet Transform
FFT	Fast Fourier Transform
FIR	Finite Impulse Response
FT	Fourier Transform
FTIR	Fourier Transform Infrared
FTMS	Fourier Transform Mass Spectrometry
FTRS	Fourier Transform Raman Spectroscopy
HT	Hadamard Transform
IFT	Inverse Fourier Transform
IIR	Infinite Impulse Response
IWT	Inverse Wavelet Transform
LSB	Least Significant Bit
MDL	Minimum Description Length
MSB	Most Significant Bit
NIR	Near Infrared
NMR	Nuclear Magnetic Resonance
NPS	Noise Power Spectrum
RC	Resistor–Capacitor
rms	root-mean-square
SG	Savitzky–Golay
SNR	Signal-to-noise Ratio
V/F	Voltage-to-frequency
WPT	Wavelet Packet Transform
WT	Wavelet Transform
1D	One-dimensional
2D	Two-dimensional

RELATED ARTICLES

Process Instrumental Methods **(Volume 9)**
Chemometric Methods in Process Analysis

Chemometrics **(Volume 11)**
Chemometrics • Multivariate Calibration of Analytical Data • Second-order Calibration and Higher

Electronic Absorption and Luminescence **(Volume 12)**
Ultraviolet and Visible Molecular Absorption and Fluorescence Data Analysis

Gas Chromatography **(Volume 12)**
Data Reduction in Gas Chromatography

Infrared Spectroscopy **(Volume 12)**
Spectral Data, Modern Classification Methods for

Kinetic Determinations **(Volume 12)**
Data Treatment and Error Analysis in Kinetics

General Articles **(Volume 15)**
Multivariate Image Analysis

REFERENCES

1. A. Savitzky, M.J.E. Golay, 'Smoothing and Differentiation of Data by Simplified Least Square Procedures', *Anal. Chem.*, **36**, 1627–1639 (1964).
2. J.W. Cooley, J.W. Tukey, 'An Algorithm for the Machine Calculation of Complex Fourier Series', *Math. Comput.*, **19**, 297–301 (1965).

3. A.J. Diefenderfer, B.E. Holton, *Principles of Electronic Instrumentation*, 3rd edition, Saunder's College Publishing, Philadelphia, 1994.
4. P. Horowitz, W. Hill, *The Art of Electronics*, 2nd edition, Cambridge University Press, New York, 1989.
5. H.V. Malmstadt, C.G. Enke, S.R. Crouch, *Electronics and Instrumentation for Scientists*, Benjamin/Cummings, Menlo Park, CA, 1981.
6. A.B. Williams, *Active Filter Design Handbook: LC, Active and Digital Filters*, 2nd edition, McGraw-Hill, New York, 1988.
7. Z.H. Meiksin, P.C. Thackray, *Electronic Design with Off-the-shelf Integrated Circuits*, 2nd edition, Prentice-Hall, Englewood Cliffs, NJ, 1984.
8. K.M. Daugherty, *Analog-to-digital Conversion: A Practical Approach*, McGraw-Hill, New York, 1995.
9. R.W. Hamming, *Digital Filters*, 2nd edition, Prentice-Hall, Englewood Cliffs, NJ, 1983.
10. T.J. Terrell, *Introduction to Digital Filters*, 2nd edition, Wiley, New York, 1988.
11. L.R. Rabiner, B. Gold, *Theory and Application of Digital Signal Processing*, Prentice-Hall, Englewood Cliffs, NJ, 1975.
12. A.V. Oppenheim, R.W. Schafer, *Digital Signal Processing*, Prentice-Hall, Englewood Cliffs, NJ, 1975.
13. D.L. Massart, B.G.M. Vandeginste, S.N. Deming, Y. Michotte, L. Kaufman, *Chemometrics: A Textbook*, Elsevier, Amsterdam, 1988.
14. S.D. Brown, 'Signal Processing and Data Enhancement', in *Practical Guide to Chemometrics*, ed. S.J. Haswell, Marcel Dekker, New York, 1992.
15. M.U.A. Bromba, H. Ziegler, 'Application Hints for Savitzky–Golay Digital Smoothing Filters', *Anal. Chem.*, **53**, 1583–1586 (1981).
16. C.G. Enke, T.A. Nieman, 'Signal-to-noise Ratio Enhancement by Least-squares Polynomial Soothing', *Anal. Chem.*, **48**, 705A–712A (1976).
17. R.G. Brown, *Introduction to Random Signal Analysis and Kalman Filtering*, Wiley, New York, 1983.
18. S.D. Brown, 'The Kalman Filter in Analytical Chemistry', *Anal. Chim. Acta*, **181**, 1–26 (1986).
19. H.N.J. Poulisse, 'Multicomponent Analysis Computations Based on Kalman Filtering', *Anal. Chim. Acta*, 361–374 (1979).
20. J. Steinier, Y. Termonia, J. Deltour, 'Comments on Smoothing and Differentiation of Data by Simplified Least Square Procedures', *Anal. Chem.*, **44**, 1906–1909 (1972).
21. A. Proctor, P.M.A. Sherwood, 'Smoothing of Digital X-ray Photoelectron Spectra by an Extended Sliding Least-squares Approach', *Anal. Chem.*, **52**, 2315–2321 (1980).
22. R.A. Leach, C.A. Carter, J.M. Harris, 'Least-squares Polynomial Filters for Initial Point and Slope Estimation', *Anal. Chem.*, **56**, 2304–2307 (1984).
23. P.D. Wentzell, T.P. Doherty, S.R. Crouch, 'Frequency Response of Initial Point Least Squares Polynomial Filters', *Anal. Chem.*, **59**, 367–371 (1987).
24. R.E. Kalman, 'A New Approach to Linear Filtering and Prediction Problems', *Trans. ASME, Ser. D*, **82**(Mar.), 35–45 (1960).
25. S.C. Rutan, S.D. Brown, 'Estimation of First-order Kinetic Parameters by Using the Extended Kalman Filter', *Anal. Chim. Acta*, **167**, 23–37 (1985).
26. S.C. Rutan, S.D. Brown, 'Adaptive Kalman Filtering Used to Compensate for Model Errors in Multicomponent Methods', *Anal. Chim. Acta*, **160**, 99–119 (1984).
27. C.L. Erickson, M.J. Lysaght, J.B. Callis, 'Relationship Between Digital Filtering and Multivariate Regression in Quantitative Analysis', *Anal. Chem.*, **64**, 1155A–1163A (1992).
28. A.G. Marshall, F.R. Verdun, *Fourier Transforms in NMR, Optical and Mass Spectrometry: A User's Handbook*, Elsevier, Amsterdam, 1990.
29. M. Cartwright, *Fourier Methods for Mathematicians, Scientists and Engineers*, Ellis Horwood Ltd, Chichester, 1990.
30. W.H. Press, B.P. Flannery, S.A. Teukolsky, W.T. Vetterling, *Numerical Recipes in C*, Cambridge University Press, New York, 1992.
31. R.P. Wayne, 'Fourier Transformed', *Chem. Britain*, **23**, 440–446 (1987).
32. R.B. Lam, R.C. Wieboldt, T.L. Isenhour, 'Practical Computations with Fourier Transforms for Data Analysis', *Anal. Chem.*, **53**, 889A–901A (1981).
33. G. Horlick, 'Digital Data Handling of Spectra Utilizing Fourier Transforms', *Anal. Chem.*, **44**, 943–947 (1972).
34. G. Horlick, 'Fourier Transform Approaches to Spectroscopy', *Anal. Chem.*, 61A–66A (1971).
35. S. Mallat, 'A Theory for Multiresolution Signal Decomposition: The Wavelet Representation', *IEEE Trans. Pattern Anal. Machine Intell.*, **11**, 674–693 (1989).
36. C.K. Chui, *Introduction to Wavelets*, Academic Press, Boston, 1991.
37. B. Walczak, D.L. Massart, 'Noise Suppression and Signal Compression Using the Wavelet Packet Transform', *Chemom. Intell. Lab. Syst.*, **36**, 81–94 (1997).
38. R.R. Coifman, M.V. Wickerhauser, 'Entropy-based Algorithms for Best Basis Selection', *IEEE Trans. Inform. Theory*, **38**, 713–719 (1992).
39. M.V. Wickenhauser, *Adapted Wavelet Analysis from Theory to Software*, A.K. Peters, Ltd, Wellesley, MA, 1994.
40. J. Rissanen, 'A Universal Prior for Integers and Estimation by Minimum Description Length', *Ann. Stat.*, **11**, 416–431 (1983).
41. J. Rissanen, *Stochastic Complexity in Statistical Inquiry*, World Scientific Publishing, Singapore, 1989.
42. D.L. Donoho, 'De-noising by Soft Thresholding', *IEEE Trans. Inform. Theory*, **41**, 613–627 (1995).

43. V.J. Barclay, R.F. Bonner, I.P. Hamilton, 'Application of Wavelet Transforms to Experimental Spectra: Smoothing, Denoising, and Data Set Compression', *Anal. Chem.*, **69**, 78–90 (1997).
44. A.G. Marshall, *Fourier, Hadamard, and Hilbert Transforms in Chemistry*, Plenum Press, New York, 1982.
45. D.K. Graff, 'Fourier and Hadamard: Transforms in Spectroscopy', *J. Chem. Ed.*, **73**, 304–309 (1995).
46. P.J. Treado, M.D. Morris, 'A Thousand Points of Light: The Hadamard Transform in Chemical Analysis and Instrumentation', *Anal. Chem.*, **61**, 723A–734A (1989).
47. M. Harwit, N.J.A. Sloane, *Hadamard Transform Optics*, Academic Press, New York, 1979.
48. K.R. Beebe, R.J. Pell, M.B. Seasholtz, *Chemometrics: A Practical Guide*, Wiley, New York, 1998.
49. H. Martens, T. Næs, *Multivariate Calibration*, Wiley, New York, 1989.
50. M.A. Sharaf, D.L. Illman, B.R. Kowalski, *Chemometrics*, Wiley, New York, 1986.