S.L. Hopp · M.J. Owren · C.S. Evans (Eds.)

Animal Acoustic Communication

Sound Analysis and Research Methods

With 115 Figures

Springer

CHAPTER 4

Application of Filters in Bioacoustics

P. K. STODDARD

1 Introduction

One afternoon in 1981, the screen went blank on the laboratory's real-time digital spectrograph, a recent technological advance at that time. The president and chief engineer of the company selling this spectrograph diagnosed the problem over the phone. "Oh, that's just the filter board. What are you using the analyzer for? Bird song? For the frequency ranges of your signals you don't really need the filters. Go ahead and bypass the filter board." I was worried that an integral component of the machine was not working, but was uncertain about the function of the filter board. If the machine's designer said the filter was unneccesary, then it must be. I removed the offending circuit board, bypassed the connection, and went back to work.

A year later, a colleague brought me a recording he had made of half-masked weaver song and we proceeded to scan the tape on the filterless analyzer. There on the screen appeared one of the most striking examples I had ever seen of the avian *two-voice* phenomenon. Two distinct voices, not harmonically related, appeared almost to mirror each other. A loud rising whistle was accompanied by a softer falling whistle. Sometimes the two voices even crossed each other, making a skewed "X" on the screen.

Six months later, while making publication-quality spectrograms on our antiquated, analog Kay Sonagraph, I found to my dismay and disappointment that the second voice was gone. I thought the upper voice must have been weaker than the lower one and had faded from the tape. Still, there should have been a trace of the second voice, which could in fact still be seen using the real-time spectrograph. I increased the gain on the analog spectrograph, but no second voice appeared. Only then did I remember the missing filter board in the real-time digital machine.

This incident impressed upon me the importance of filters in the daily life of a bioacoustician. Many times since then I have been able to trace peculiar signal artifacts to filters that were either not working or inappropriate to the task at hand. The appropriate use of filters is an essential part of any technical analysis of sound signals. This chapter therefore provides an intuitive explanation of filter theory, filter applications in bioacoustics, and the hazards of misapplying or omitting filtering. Readers may also wish to consult other tutorials on digital filter application that have been written specifically for non-engineers (e.g., Menne 1989; Cook and Miller 1992). For those who are facile in linear algebra and complex arithmetic, the presentation may be too simplistic — the chapter is written specifically for the bioacoustician who wishes to understand filters at a conceptual level, without taking a crash course in electrical engineering. More rigorous treatments can be found in books on signal processing and filter theory (e.g. Haydin 1986; Stearns and David 1988; Hamming 1989; Oppenheim and Schafer 1989).

2
General Uses of Filters and Some Cautions

Filters are frequency-selective resonant devices or algorithms that are used to remove noise from signals, to change the spectral balance or the phase composition of signals, to smooth the analog output of *digital-to-analog* (D/A) converters and as above, to prevent sampling artifacts from contaminating signals during the *analog-to-digital* (A/D) conversion. Filters can also play a critical role in automated signal detection and identification.

Most often, then, filters are used to change signals by damping or excluding certain components while allowing others to pass. However, they also have nonintuitive properties that can pose problems for the unwary. Rather than simply selectively removing or separating the components of a signal, filtering can destroy certain components and irreversibly transform others. For instance, while energy at unwanted frequencies might be attenuated by a filter, the filtering process might be simultaneously distorting the amplitude and phase characteristics of energy in spectral ranges meant to be left unaffected.

3
Anatomy and Performance of a Filter

Figure 1 shows the *gain functions* of four hypothetical filters demonstrating the most common graphical depiction of a filter's characteristics. In these diagrams, the abscissa represents frequency, and can be either discrete (showing individual frequency components) or continuous. The ordinate shows the amplitude of the filtered waveform (the filter output) relative to the unfiltered waveform (the filter input).

As shown in this figure, frequency-selective filters assume two basic attenuation patterns and two composite patterns. In each case, the frequency at which attenuation nominally begins or ends is a *corner frequency*. *Lowpass* filters (upper left panel) have a single corner frequency, allowing the passage of frequencies lower than that point while attenuating higher frequencies. *Highpass* filters (upper right panel) do the opposite, passing frequencies higher than the corner and attenuating lower frequencies. *Bandpass* filters (lower left panel) combine the first two forms to create a frequency band of *unity gain* (where energy passes without attenuation) surrounded by regions of attenuation. A *bandstop* filter (lower right panel) consists of both a highpass and a lowpass filter, and attenuates an intermediate frequency band. Whereas a *graphic equalizer* is an array of bandpass filters with adjacent frequency ranges, a *notch* filter is a sharp, narrow bandstop filter that has corner frequencies and attenuation functions selected to attenuate a narrow frequency band. An application of the latter might be removal of AC power transmission noise, which occurs at either 50 or 60 Hz (depending on the country).

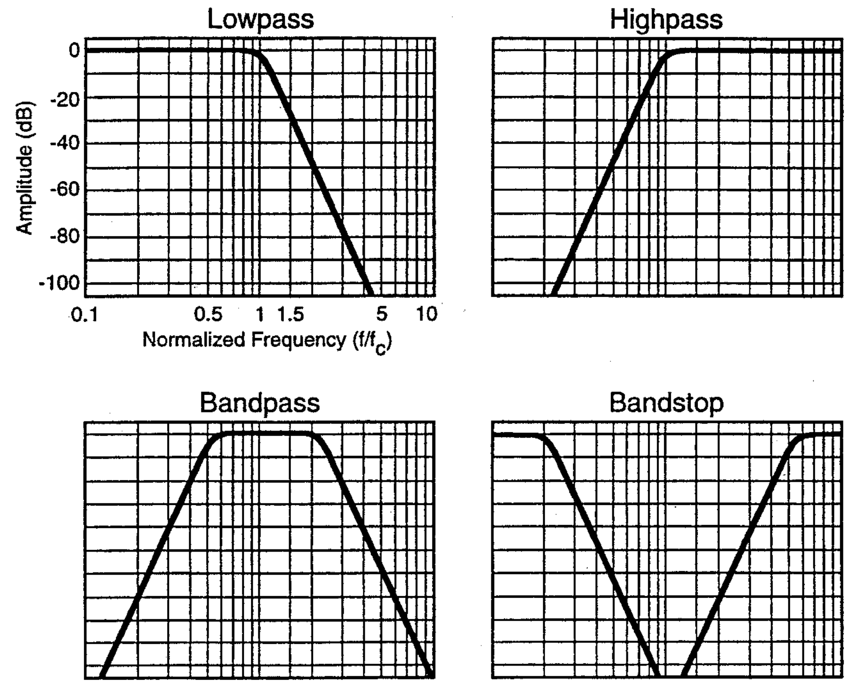

Fig 1. The two basic attenuation patterns, lowpass and highpass, can be combined to pass a particular frequency band (bandpass) or exclude a particular frequency band (bandstop)

Figure 2 illustrates a number of important terms in the context of a lowpass filter. The frequency identified as the corner frequency is the location on the gain function at which signal input is attenuated by 3 dB — the *half-power point*. If, for instance, a lowpass filter with a corner frequency of 10 kHz is applied to a signal that includes energy up to 10 kHz, the highest frequencies will be significantly attenuated. In general, a filter with a corner frequency that is somewhat beyond the frequencies of interest should be used in bioacoustics applications, in order to avoid attenuation of potentially important high-frequency signal components. Note, however, that in some filter designs the transition point has no associated attenuation and is known as the *cut-off frequency* rather than the corner frequency. The frequency range that is not attenuated (or is attenuated by less than 3 dB) is called the *passband*. The frequency range over which maximum attenuation occurs is the *stopband*. A *transition-band* is the frequency range of increasing or decreasing attenuation lying between a corner frequency and the beginning of stopband or passband, respectively.

A filter's *attenuation slope* is specified in decibels per octave (an octave is a doubling or halving of frequency). The greater the number of elements in the filter (its *order* or number of *poles*), the more sharply defined is its gain function. In analog equipment,

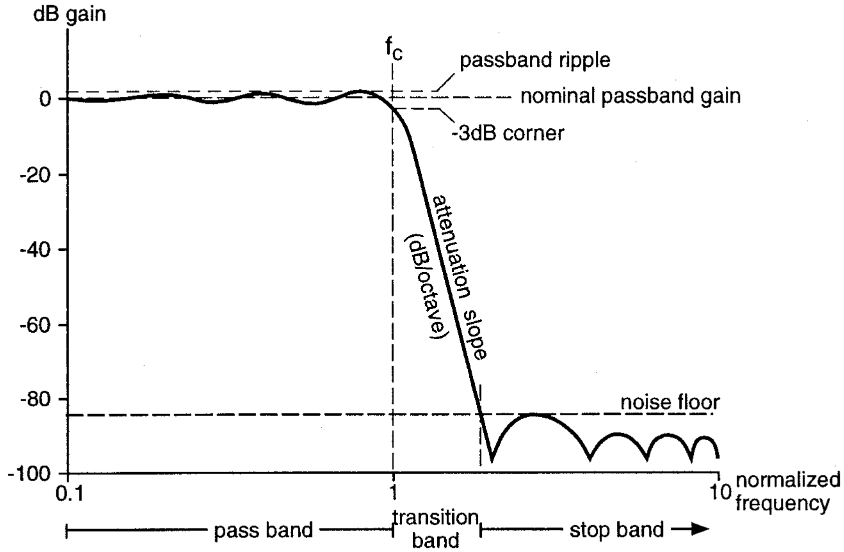

Fig 2. The Anatomy of a frequency-selective filter is illustrated here with a hypothetical lowpass filter. Frequency is normalized so that the corner appears at nominal frequency of 1.0. The attenuation throughout most of the passband is 0 dB (i.e., the gain is 1)

filter order is determined by the number of filter stages in the circuit, with two per stage. In a digital filter, the order is determined by the number of elements in the filter array. Thus, sharp attenuation requires a high-order filter. The optimal filter for attenuation of undesired frequencies is often one with a flat passband and precipitously sloping transition bands. For such purposes, one can obtain sharp filters with attenuation slopes exceeding 100 dB per octave. Engineers and scientists proudly call these "brickwall" filters, a term that was evidently coined to evoke images of powerful filters (and powerful engineers) that crush any undesired frequency components with deep stopbands and vertical transition slopes. In reality, however, even the sharpest filters show finite attenuation slopes and variable attenuation effects over their transition bands.

In addition, sharp attenuation generally comes at a price. For a given filter design, the more acute the attenuation, the greater the occurrence of undesirable artifacts in the passband. One such side-effect is variable gain, called *passband ripple*. The amount of passband ripple is generally included in the performance specifications provided with analog filters and may serve as a criterion for choosing among various designs. Likewise, the maximum amount of tolerable passband ripple can often be specified in the course of designing a digital filter. Increasing the order of a digital filter can reduce the trade-off between ripple and attenuation, although at the cost of more computation.

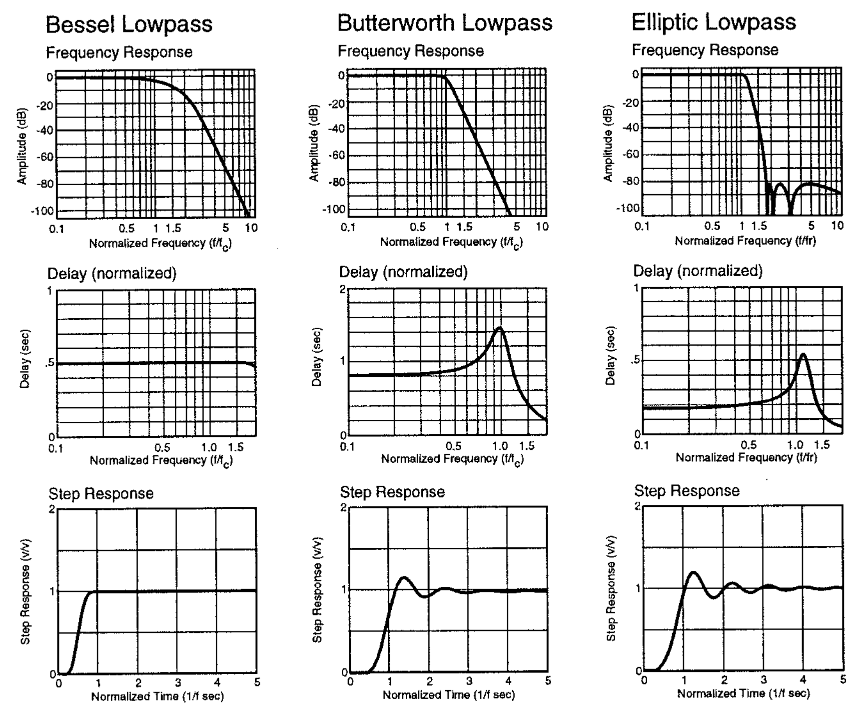

Fig 3. Performance graphs of three common eight-pole, lowpass, analog filter designs. The Bessel filter gives the cleanest step response and least phase distortion but takes an octave to reach its full attenuation slope. The Butterworth filter has the flattest passband and reaches its full attenuation slope over a smaller frequency range. The elliptic function is used when a steep transition band is needed and phase and step response is not important. The elliptic filter's cut-off frequency (or ripple frequency, f_r) has no attenuation. The other two filters attenuate 3 dB at the corner frequency, f_c.

In addition to their amplitude-related effects, filters also delay the output of a signal relative to the input. The *group delay* of a filter is the average time-delay produced in the output as a function of frequency, as shown in the middle row of panels Figure 3. If all frequencies are delayed by a constant interval, phase relations among these components are unchanged. *Constant group delay* is the same as *linear phase delay*, where phase delay is phase divided by frequency, multiplied by -1. Plotted as a function of phase angle and frequency, constant group delay yields a linearly increasing function, because identical delays require increasing proportions of a wavelength as the frequency increases and the period shortens.

Filters with nonunity group delay or nonlinear phase delay in the passband distort the filtered waveform by differentially altering the phases of various spectral components. The examples of filtered waveforms in Figure 4 illustrate linear and non-linear phase delay of different filters. It is clear that the differences are considerable. Scientists working with animal signals should consider the importance of phase information when choosing a filter for their work.

The response of a filter to very brief or *transient* signals is indicated by its *step response*, i.e., the response to a discrete amplitude change called a *square input step* (shown in the bottom panels of Figure 3). The step response reflects the degree of *damping* in a filter. If a filter absorbs a square input step with no overshoot or undershoot in the output, it is said to be critically damped. Critical-damping in a filter is analogous to the action of a shock absorber of a car that completely cushions the vehicle's passengers from any bounces due to bumps in the road. *Underdamping*, or *overshoot*, causes the filter to *ring*, producing ripple in the output. This effect is analogous to oscillation occurring in a heavily laden car when encountering a bump. *Overdamping* causes the filter to *undershoot*, which is analogous to the rough ride experienced in an unladen truck. Effective filtering of signals with sharp transients requires a good step response. Examples of such signals include the acoustic pulses and clicks of many diurnal insects, as well as the discharges of many electric fishes. A filter with an underdamped step response causes these transient signals to ring, smearing them in time.

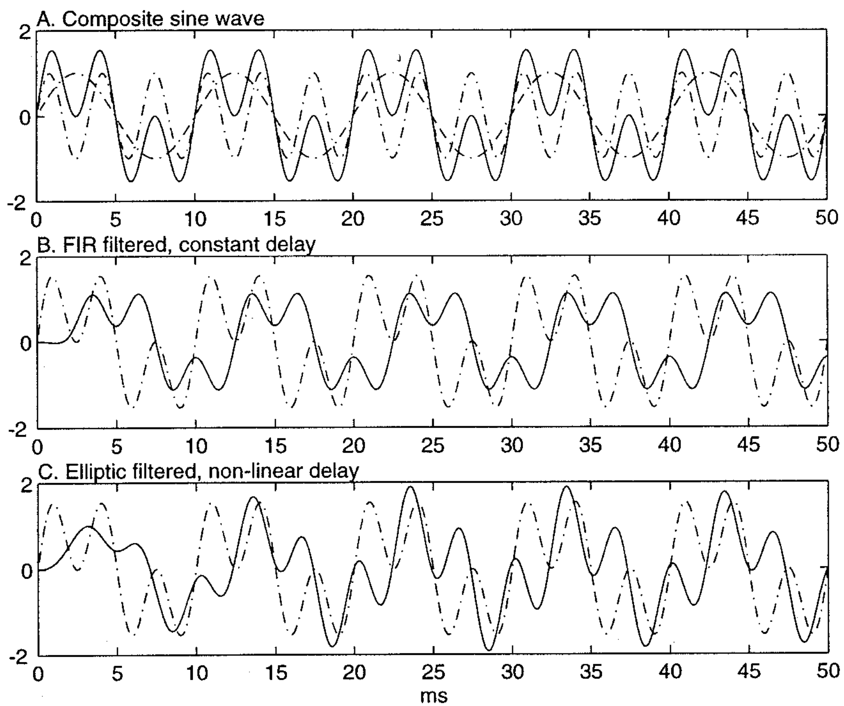

Fig 4. A-C. Examples of linear and nonlinear phase delay. **A** A composite waveform (solid line) is composed of two sine waves of 100 and 300 Hz (dashed lines) .**B-C** The composite waveform is lowpass filtered with a symmetric FIR filter and an elliptic IIR filter both with cut-off frequencies set to 330 Hz. The FIR filter (**B**)has linear phase filter delay; the output waveform (solid line) is delayed relative to the input waveform (dashed line) but the phase relationship between the component sine waves is preserved. **C.** The elliptic IIR filter has non-linear phase delay; the 300 Hz component is delayed more than the 100-Hz component (solid line). The filter startup delays are evident in the first 2 ms of both filtered outputs. The FIR filter used here also shows some attenuation at the upper edge of the passband as seen in the slight amplitude reduction of the filtered output

4 Properties of Various Analog Filters

An analog filter function can usually be classified as being one of five basic designs: Bessel, Butterworth, elliptic, and Tschebychev (or Chebyshev) types I and II. Figure 3 shows the first three of these types in corresponding order. Digital implementations often emulate these filter-types as well.

Bessel filters produce the least phase distortion, because their phase delays in the passband are linear with frequency. However, a Bessel filter has a gradual attenuation curve and its attenuating effect stretches back farther into the passband as the latter gradually slopes to -3 dB at the corner. Bessel filters require about an octave to reach their full attenuation slope of -6 dB per octave per pole, and its gradual attenuation makes this filter unsuitable to applications for which steep attenuation slopes and flat passbands are essential. Nonetheless, the linear phase delay and damped step response associated with this design are useful in some applications. Digital implementations of the Bessel filter do not have linear phase delay.

Butterworth filters are popular because they show neglible passband ripple. Their phase delay is moderately non-linear in the passband. Each pole contributes about 6 dB attenuation per octave beyond the corner frequency, amounting to a halving of signal magnitude with each doubling or halving of frequency. Thus, an eighth-order (or eight-pole) Butterworth filter produces an attenuation effect of 48 dB per octave.

Elliptic filters have extremely sharp attenuation curves. However, no simple rule relates the order of an elliptic filter to the attenuation slope of the transition band, as in the previous two filter designs. Elliptic filters tend to distort the signal in the outer 20 % of the passband, showing significant passband ripple, group delay, and overshoot. Elliptic notch filters that remove 50- or 60-Hz AC noise are notorious for contaminating the passband with nonlinear phase delay.

Tschebychev type I filters (not shown in the figure) have ripple in the passband, fairly sharp transition bands, and a flat responses in the stopband. Tschebychev type II filters have a flat passband, a less steep attenuation slope, and ripple in the stopband. Tschebychev filters are not used as commonly as the other three filter designs described.

5 Antialiasing and Antiimaging Filters

5.1 A/D Conversion Requires an Analog Lowpass Filter

A signal *alias* is a phantom signal pattern introduced into a waveform (also called a time series) by sampling at too low a rate during A/D conversion. Figure 5 shows how such aliasing occurs. The true signal in this example is a transient that sweeps from 0 to 10 Hz. When the analog waveform is digitized, the A/D converter provides numerical readings of the signal's amplitude at regular points in time (represented by the circled points in Figure 5). When connected, these sample points should approximate the original waveform. If the sampling interval is too long, however, the waveform is undersampled and the sampled frequency actually drops. In the example, the analog waveform is sampled ten times per second. The sampled waveform tracks the frequency of the original from 0 to 5 Hz, but cannot keep up thereafter. The digital version drops back

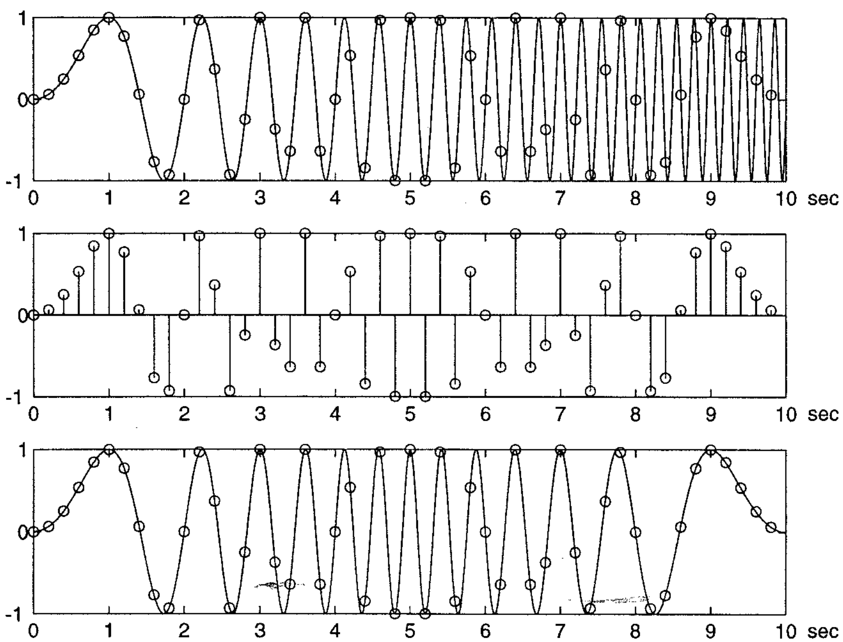

Fig 5. The true signal here (*upper panel*) is a transient that sweeps from 0 to 10 Hz. When the waveform is digitized at 10 samples/sec (*middle panel*), the signal's instantaneous amplitude is measured every tenth of a second, represented by the *circled points*. When connected, these sample points should approximate the original waveform (*lower panel*). However the sampling interval is inadequate and the waveform is undersampled. The waveform reconstructed from sampling actually drops in frequency where the original waveform exceeds half the sample frequency. The false signal that results from undersampling is called an alias

to 0 Hz as the analog signal sweeps upwards to 10 Hz.

According to the *Nyquist theorem*, digital sampling of an analog signal limits the bandwidth of the sample series to a frequency range between 0 Hz and half the sampling rate. Thus, the latter is the highest frequency that can be represented in a digital time series, known as the *Nyquist frequency* (also known as the *folding* frequency). The frequency of an aliased, low-frequency component mirrors that of the higher-frequency counterpart in the original signal, reflected (or folded) downward at the Nyquist frequency. Obviously then, we cannot obtain a digital representation of any signal component with a frequency higher than the Nyquist frequency. Of greater consequence, however, is the risk of obtaining false signals (as I did with my colleague's recordings of half-masked weaver birds).

The first microcomputers to arrive sound-capable from the manufacturer digitized sound at a rate of approximately 22 kHz (actually 22254.5 Hz). When a signal is digitized at 22 kHz, the sampled waveform must be restricted to a passband between 0 and 11 kHz. A Nyquist frequency of 11 kHz should be fine for most frogs and passerine birds, but would not be adequate for rats, bats, dolphins, or other species whose vocalizations contain higher frequencies. These higher frequencies reappear as aliases in the passband, where they become inseparable and often indistinguishable from true signal components. Other, more insidious sources of high-frequency noise can also produce aliases in the passband. In addition to the intended signal, the tape recording may contain a variety of extraneous noises such as wind, footsteps, or the like that will probably contain energy above 11 kHz. Harmonic distortion caused by imperfect electronic processing and especially by occasional overloading during recording and playback can also introduce energy above 11 kHz. When recording with analog equipment, one may not notice the presence of high-frequency energy, particularly high-frequency harmonics.

Alias contamination of digitized signals can be avoided by using an analog, lowpass filter to screen out any energy above the Nyquist frequency before it reaches the A/D converter. This antialiasing filter, if chosen correctly, attenuates frequencies above the Nyquist frequency to insignificant levels while allowing the lower frequencies to pass unaffected. If an inappropriate filter is used, however, the signal may not only be aliased, additional artifacts may be introduced by the filter itself! In practice, it is impossible to obtain a digitized waveform that perfectly represents the original analog signal between 0 Hz and the Nyquist frequency because analog lowpass filters do not have infinite attenuation slopes. If the stopband is set to begin exactly at the Nyquist frequency, some attenuation inevitably occurs in the passband. If the corner frequency is set at the Nyquist frequency, some aliasing will occur. For these practical reasons, sampling rates must be set higher than twice the highest frequency of interest.

5.2 Choosing an Antialiasing Filter

If an A/D converter does not include a built-in antialiasing filter, the user must add one. Similarly, if a digitizing system has a built-in, fixed lowpass filter designed for a sampling rate that is higher than the user would choose (for instance to save storage space), the signals must be prefiltered using a lowpass filter with a lower corner frequency.

Failure to do so will result in aliasing. An adjustable-frequency bandpass filter can be used to prevent aliasing, but often at a cost of several thousand dollars. Less than one hundred dollars will buy a fixed frequency antialiasing filter module that needs only a bipolar power supply and two BNC connectors. Either way, the user needs to select the filter function, the filter order, and the corner frequency.

The amount of attenuation needed at and above the Nyquist frequency depends on the dynamic ranges of both the system and of the signal. A 16-bit digitizing system, for instance, provides a total dynamic range of about 96 dB (each binary bit of quantization increases the dynamic range by 6 dB). A microphone might have a maximum signal-to-noise ratio of 78 dB. A *noise floor* of -78 dB, therefore, is appropriate and sufficient for recordings made under ideal circumstances in which the dynamic range of the microphone is a limiting factor in recording quality. Attempting to achieve a noise floor of -96 dB would be overkill. If recordings are made in the field, however, the signal-to-noise ratio is probably not much better than 34 dB (Wickstrom 1982). That is to say, the signal will be at most 34 dB louder than the average background noise level. A reasonable minimum noise floor in these"real-world" cases is therefore -36 dB.

Let us assume in two examples that one wishes to digitize field recordings of song sparrows (*Melospiza melodia*). The energy in their songs ranges from 1.6 to 10 kHz. Once these songs are in the computer, the plan is to make spectrograms and to conduct playback experiments with birds in the field. In the first example, signals are digitized at 48 kHz and in the second, data storage requirements are cut in half by digitizing at 24 kHz. In both cases, the task is to choose an appropriate analog, antialiasing filter.

Example 1. 48 kHz Sampling Rate. With a sampling rate of 48 kHz, the Nyquist frequency is 24 kHz, 14 kHz above the frequency of highest interest in these examples. The lowpass, antialiasing filter should therefore leave all frequencies below 10 kHz unaffected while reducing energy at frequencies above 24 kHz by at least 36 dB. The corner frequency should be 10 kHz, or perhaps a little higher to accommodate attenuation occurring at the corner.

Figure 3 illustrates the properties of a series of prototypical eight-pole lowpass antialiasing filters modules acquired from a well-known manufacturer. The analog filter functions available include Bessel, Butterworth, and elliptic versions. All the functions are normalized to f / f_c (the ratio of the frequency of interest to the corner frequency), meaning that the corner frequency has a value of 1.

The eight-pole Bessel lowpass filter will be considered first, as it has the best phase and step performance. The corner frequency of the Bessel filter is defined at -3 dB, so it is best to raise it somewhat above 10 kHz. Approximately -1 dB might be deemed acceptable attenuation or ripple in the passband. Inspection of the filter's gain function shows the amplitude is down by 1 dB at $f / f_c = 0.6$, which corresponds to a new corner frequency of about 17 kHz (i.e., 10 kHz / 0.6). The response curve requires all values to be normalized with respect to the corner frequency. The normalized frequency that corresponds to the Nyquist frequency is 1.4 (i.e., f_N / f_c = 24 kHz / 17 kHz). The gain at 1.4 is -6 dB, which is not even close to -36 dB. The shallow attenuation slope of the eight-pole Bessel filter makes it unacceptable for this situation.

Like the Bessel design, the corner frequency of the Butterworth filter is defined at -3 dB. Again, the corner frequency should be increased to preserve the integrity of the passband. Attenuation of -1 dB is found in the Butterworth filter gain function at a normalized frequency of 0.9. This more stringent criterion places the corner frequency at 11 kHz (i.e., 10 kHz / 0.9 = 11 kHz). Recalculation of the normalized Nyquist frequency, f_N / f_c, yields a value of 2.2 (i.e., 24 kHz / 11 kHz). The Butterworth filter function shows the gain at a normalized frequency of 2.2 to be about -55 dB, far below the citerion of -36 dB set above. The eight-pole Butterworth lowpass filter with an 11 kHz corner easily prevents aliasing of the digitized field recordings.

Example 2. 24-kHz Sampling Rate. In this example, the data are sampled at only 24 kHz in order to reduce storage requirements. The new Nyquist frequency of 12 kHz is now only 2 kHz above the highest frequency in these signals. This time, then, the task is to find a filter that can provide 36-dB attenuation in the narrow band between 10 kHz and 12 kHz.

The corner is adjusted to 11 kHz for the eight-pole Butterworth filter. At the 12 kHz point, the normalized Nyquist frequency, f / f_c, is 1.1 (i.e., 12 kHz / 11 kHz). Attenuation at this point is only -7.5 dB, far short of the -36 dB that is needed. The Butterworth filter therefore will not work. The eight-pole elliptic filter is well-suited to a 10-kHz cut-off frequency, but also does not provide 36 dB attenuation at 12 kHz. At this point, the normalized Nyquist frequency is 1.2 (i.e., 12 kHz / 10 kHz) and attenuation is only -8.3 dB.

The remaining options are either to sample at a higher rate and reduce the data with *digital decimation*, or to search for a sharper elliptic filter design. The first approach, also known as *downsampling*, produces a lower sampling-rate by discarding a fixed proportion of the digital samples. It is discussed in greater detail below. The second approach takes advantage of the potential for building elliptic filters with extremely sharp attenuation slopes.

It happens that the manufacturer whose eight-pole filters are illustrated in Figure 3 also makes a 7-pole elliptic filter module that provides 20 dB attenuation at a normalized Nyquist frequency of 1.2. This filter produces the desired 36 dB attenuation at f / f_c = 1.3, corresponding to a frequency of 13.4 kHz in this particular case. Using this filter, then, some energy above the Nyquist frequency is aliased, but only between 12 and 13.4 kHz. Frequencies above 13.4 kHz are attenuated by 36 dB and energy in the aliased band is attenuated by at least 20 dB. Energy remaining between 12 and 13.4 kHz folds back across the 12 kHz Nyquist frequency, but the aliases extend no lower than 10.6 kHz (12-(13.4 - 12) = 10.6).

If spectrographic analysis is restricted to the 0 to 10 kHz range, aliases in the signal will not even be visible. In using these songs for playback trials, however, the investigator needs to consider whether subjects will be affected by spurious signals in the 10.6- to 12 kHz range. One important consideration is that the bird's ear has its own "filters". The song sparrow audiogram shows this bird to be approximately 35 dB less sensitive to frequencies above 10 kHz than at its "best" frequency of 4 kHz (Okanoya and Dooling 1987). Song sparrows would therefore probably not notice faint aliases above 10.6 kHz.

The frequency response of the elliptic filter is therefore acceptable, allowing us to consider other performance characteristics. Its normalized group delay curve shows a large amount of nonlinear phase shift well before the cut-off frequency. If phase relations are critical, then, this filter should be rejected. In this particular example, however, signals would be recorded and played back in the *far sound field* (distances over which environmental transmission characteristics affect the signal) and phase relations are therefore probably not critical. Overall, even though this elliptic filter did not meet the initial criteria, the compromises involved in using it turned out to be acceptable.

5.3
D/A Conversion also Requires an Analog Lowpass Filter

When a digital signal is resynthesized using a D/A converter, the output is an analog step function that approximates the smooth analog wave (Figure 6). These tiny steps carry the high-frequency energy typical of square waves. A spectrum of the direct analog output therefore has *signal images*, reflections of the intended output signal that fold upwards accordion-style about multiples of the Nyquist frequency. As a result, a second analog lowpass filter is needed that *follows* D/A conversion to eliminate reconstruction

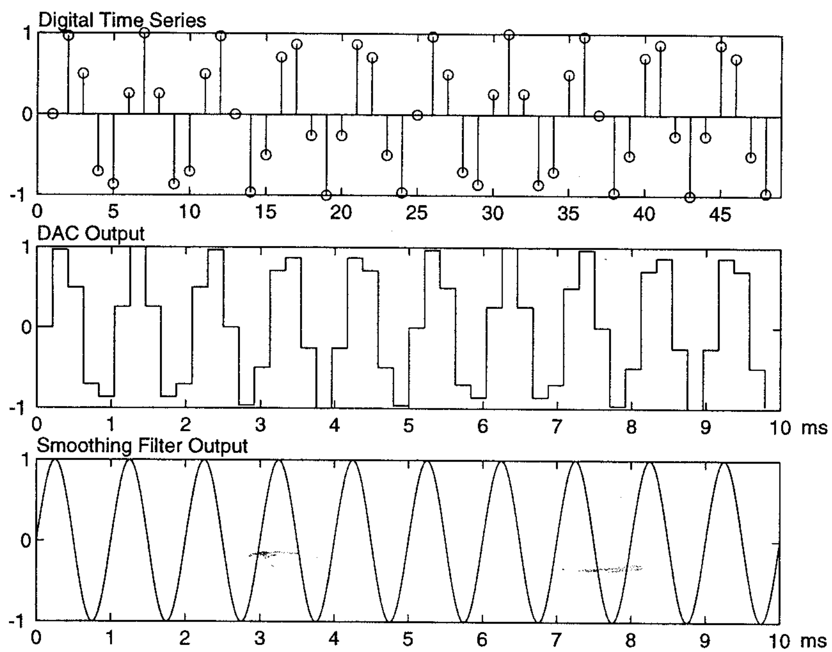

Fig 6. A digital-to-analog converter (DAC) converts a 1 kHz sine wave digitized at 48 kHz (*upper panel*) into an analog step function. Each step is similar to a square wave with energy present at the odd harmonics. An analog lowpass smoothing filter reverts this harmonic energy back to the fundamental (*lower panel*).

images. This *anti-imaging*, or *smoothing* filter recovers the energy in a high-frequency image and puts it back into the smoothed output signal.

If the smoothing filter is absent, inadequate, or incorrectly configured, the output signal sounds peculiar. When played through a speaker, it has an tinny, ringing quality. One can experience the unpleasant sound of imaging by halving the clock rate of the D/A converter without a commensurate halving of filter settings (e.g., playing a signal sampled at 48 kHz at 24 kHz). Many computer audio systems have smoothing filters that are preset in accordance with their highest sample rate. For better or worse, most audio software lets the user play back the signal at half speed with no regard for the smoothing filter. If no smoothing filter is present at all, high-frequency images can eventually damage the high frequency response of the playback speakers.

5.4
Analog Filters: Passive Versus Active Components

Analog filters can be either *passive*, requiring no additional power source, or *active*, requiring some nominal DC source to power their amplifier circuitry. Passive filters typically show some *insertion loss* in the passband, meaning they attenuate the passband by some constant amount and the stopband by a greater amount. The components of a passive filter are all *reactive* (i.e., either *resistive*, *capacitive*, or *inductive*), so the output has high *impedance* (resistance in an AC circuit) and may therefore not be well-suited for driving some kinds of circuitry. A/D converters usually have high input impedance and passive filters can therefore be used effectively for antialiasing. Independence from additional power supplies reduces the overall cost of passive filters and increases their appeal to those with an aversion to soldering. Active filters contain integrated circuit amplifiers and require a clean, bipolar power supply. Active filters have unity gain in the passband and no insertion loss. They are generally built with a low-impedance output stage and can therefore drive just about any kind of equipment except long cables and audio speakers.

6
Analog Versus Digital Filters

Analog filters are constructed from electronic components. An analog filter accepts a continuous voltage signal as input and produces real-time output. Digital filters, by contrast, use a mathematical algorithm to alter the composition of a digital time series such as digitized animal signal. Many frequency-filtering functions can be performed by either analog or digital technology. In these kinds of applications, much of the relevant theory applies to both types. However, digital filters can perform some functions better than analog filters, and can be designed with capabilities that analog filters cannot have.

For instance, digital filters can easily filter a segment of a digitized signal while leaving the rest untouched. With selective digital cutting, filtering, and pasting, a careful person can sometimes separate signals that overlap in time. Digital filters are quiet and do not introduce the noise that is inherent in using a comparable series of analog amplifier circuits. Extremely sharp digital filters can be constructed in a fraction of a second, whereas equivalent analog filters can only be built after testing boxes of identically labeled capacitors for units with particular values. The parameters of digital filter algorithms can be changed with no change in cost, whereas analog counterparts have fixed response functions and one pays a premium for frequency-selectable devices. Digital filters can be made to exhibit virtually any response pattern and can further be designed to detect or isolate particular signals, a feat that is difficult or impossible with analog circuitry. However, digital filters cannot be substituted for analog filters on the A/D input and D/A output because a digital filter cannot prevent aliasing or imaging during the conversion process.

6.1 Designs and Functions of Digital Filters

The most widely used digital filters compute a *weighted moving average* of a digital time series. Energy at frequencies outside the passband is averaged out. The filter characteristics are determined by the weighting function, a combination of a weighting algorithm, and an array of weights (the filter's impulse function). The *finite impulse response* (FIR) filter computes a moving-average of fixed length. Therefore, the contribution of any individual datum to the filtered output is of finite duration. In contrast, the *infinite impulse response* (IIR) filter factors in a proportion of the past moving average to future moving averages. Every datum is thus infinitely present, albeit in ever decreasing proportions, in all future moving averages of the input series.

FIR filters are especially useful because they can be designed to have a perfectly linear phase response in the passband. IIR filters, on the other hand, can be designed to implement a broad array of filter shapes, including close mimics of the traditional analog functions. FIR filters are conceptually and computationally simpler to implement than are IIR filters. IIR filters, however, can often meet a given set of specifications with a much lower-order filter than can a corresponding FIR filter and so may require fewer computations. FIR filters are inherently stable, which means that filters of virtually any steepness of attenuation can be constructed that will yield predictable results. IIR filters, in contrast, are not inherently stable and must be constructed within certain constraints or they may produce unpredictable results.

The key in digital filtering is the design of an appropriate impulse function, an array of weights. This array can be saved and entered in any number of commercial or user-written programs that will perform the filtering of digitized or synthetic signals. Numerous commercial software packages can help users design, test, and implement digital filters. Different filter design routines optimize and trade-off different parameters of the filter function, in particular passband ripple and attenuation slope. FIR filters work by *convolving* an array of weights (the impulse function) against the input waveform (this process is illustrated below in an example problem). In convolution, the weighting array is time-reversed, multiplied point for point against the input waveform, and the resulting product array is summed to produce a single value. These weighted sums are computed as the reversed impulse function is advanced one point at a time across the input waveform. The resulting array of weighted values is the filtered waveform. If the weighting array is symmetrical, the filter will have linear- phase delay, i.e., preserve the signal's phase relations.

FIR filtering is performed efficiently in the frequency domain using the *fast Fourier transform* (FFT) as a shortcut for convolution. The convolution program computes overlapping FFTs of both the input waveform and the filter waveform, then multiplies the Fourier coefficients of each, point for point. An *inverse fast Fourier transform* (IFFT) of the multiplied coefficients results in the filtered (convolved) waveform. In most cases, the shorter segment of the two must be padded with zeros to the length of the longer before the FFT can be computed. For longer signals, such as a birdsong, the digitized wave array can be chopped up into sections and each segment filtered separately by the FFT method (Oppenheim and Schafer 1989).

6.2 Cascaded Filters and Repeated Filtering

Combining filters with different functions is often useful. Two analog filters can be connected in series (*cascaded*) or a digital signal can be filtered successively with different functions. The analog lowpass Bessel filter, for instance, can be improved as an antialiasing filter by being cascaded with an elliptic lowpass filter with a significantly higher corner frequency. Positioning the cut-off frequency of the elliptic filter well beyond the corner of the Bessel filter keeps the phase delay to a minimum but still abbreviates the Bessel filter's gradual transition band.

Filter cascading can be misapplied as well. Suppose a digital filter function improves the signal-to-noise ratio of a digitized signal, but not as much as desired. One can then connect two identical analog filters in series or refilter digitally with the same function and take out more noise each time the signal passes through the filter. Cascading two Butterworth lowpass filters, each with 8 poles (48-dB attenuation per octave) and corner frequencies of 10 kHz, does indeed produce a filter with an attenuation of 96 dB per octave. The result, however, is not a 16-pole Butterworth filter with a 10 kHz corner. As described earlier, the passband is already attenuated by -3 dB at the corner frequency. The cascaded filter doubles this attenuation to -6 dB at 10 kHz corner frequency, effectively shifting the -3 dB corner down to about 9.5 kHz. Repeated filtering is easy and tempting when a software package can filter a waveform with the click of a button. However, the edge of the passband is nibbled away with every filtering.

7 Special Uses of and Considerations Regarding Digital Filters

7.1 Segment Filtering and Elimination of Edge Effects

An especially useful property of digital filters is the ease with which they can be applied to selected segments of waveforms. Digital filtering of individual segments is useful for creating playback stimuli, for instance by removing an extraneous sound from the original recording, or by attenuating a particular harmonic or spectral peak. Because FIR filters compute weighted averages of the waveform, the beginning (or *edge*) of a finite waveform segment does not receive the same treatment as does the rest of the signal. The result is a transient artifact occurring at the onset of the filtered signal. This effect can be eliminated either by filtering the edge with an IIR filter, by filtering the edge with FIR filters of successively shorter length (Menne 1989), or by eliminating the edge altogether. The latter can be achieved by inserting a time-reversed version of the beginning segment at the onset of the waveform, thereby making a new edge in the appended segment. This time-reversal technique usually assures a smooth transition into the original waveform. After filtering, the added segment is removed. However, each of the cutting and pasting operations must occur at transition points where the waveform is at or near zero, to avoid introducing a broadband click.

7.2 Zero-Delay Filtering

Double-filtering provides a way to eliminate the phase delay of a digital FIR or IIR filter. The time-series waveform is filtered in the normal fashion, which produces phase delay in the output. The filtered series is then time-reversed and filtered again with the same function, shifting the delay back to zero. The double-filtered series is again time- reversed and thereby set right. Double-filtering shifts the cut-off frequency as explained in Section 6.2.

7.3 Decimation

In the second of the two anti-alias filter examples considered earlier, the sample rate was halved to reduce data storage requirements. Rather than halving the initial sampling rate, however, one can instead save only every other sample point of a signal that has already been digitized. Simply discarding alternate samples (downsampling by a factor of two) results in an aliased signal identical to that obtained by halving the sampling rate of the A/D converter without changing the corner of the antialiasing filter. Downsampling effectively lowers the Nyquist frequency. In fact, downsampling was used to produce the alias in the undersampled time series of Figure 5. Aliasing can be avoided if, prior to downsampling, the digitized series is refiltered with a sharp digital lowpass filter set to achieve the desired noise floor at the new Nyquist frequency. This process of sample rate reduction, *oversampling* followed by decimation, takes advantage of the ease and low cost of designing a sharp lowpass digital filter.

Decimation is implemented in real-time by commercial digital audio devices. If the device oversamples by a factor of 5 or more, a very clean analog antialiasing filter such as a Bessel design can provide adequate attenuation at the elevated Nyquist frequency while providing linear phase delay and a good step response. The 16-bit "delta-sigma" A/D converter on my current computer oversamples by a factor of 64. Real-time digital filters are also available that accept functions supplied by the user's software. Such a hybrid filter contains an analog antialiasing filter that is applied to the input, an A/D converter to digitize the analog signal, a programmable digital signal processor (DSP chip) that rapidly executes a digital filtering algorithm, a D/A converter that resynthesizes the filtered series as an analog signal, and an analog anti-imaging filter that is applied to the output.

7.4 Simulated Echo

Convolution of digital signals can also be exploited to create echoes and artificial reverberation in either a natural or a synthetic manner. As anything multiplied by one is itself, a digital signal is unchanged by convolution against the unit impulse array {1,0,0,0,0,0,0,0}. If the single unity value appears four points later in the array, i.e., {0,0,0,0,1,0,0,0}, convolution reproduces the original signal while delaying it by four sample points. For example, convolution of a signal against an array of {0.9,0,0,0,0.1,0,0,0} reproduces the signal at nine-tenths the original intensity, followed by an echo occurring four sample points later at one-tenth the original intensity. Digital echo simulation finds use in experimental exploration of echolocation and sound localization (e.g., Keller and Takahashi 1996).

7.5 Recreating Environmental Attenuation and Reverberation

An FIR filter can reproduce the lowpass filtering and reverberation of an environmental sound transmission pathway. For instance, a click might be broadcast and rerecorded some distance away in an environment characterized by multiple irregular, reflective surfaces. Compared to the original, the rerecorded click will show spectral changes and temporal smearing due to attenuation and reverberation effects. Digitizing the rerecorded click then yields an array that can be used to impose similar effects on any digitized signal by convolution (Menne 1989).

7.6 Matched Filters and Signal Detection

The frequency-selective filters described thus far can be used to improve the signal-to-noise ratio of a signal when at least part of the noise band exceeds the signal band. However, frequency-selective filters do not help much when the signal and noise occupy the same frequency band. In some such cases, matched-filter techniques can be used to detect the signal in the noise, provided that consistent differences exist between the two in basic attributes like waveform structure, amplitude envelope, or time-by-frequency characteristics. Matched filters search for particular signal events, providing a quantified measure of the degree to which segments of the sample match the characteristics of a particular target event. A simple technique for matched-filtering is to mathematically compare a relatively noise-free exemplar of the desired signal to a sample consisting of signals that are imbedded in noise. Comparisons based on convolution, for instance, in which a signal is reversed in time and correlated with the original version can be useful in at least partially extracting the signal from a noisy background noise.

This technique depends on *cross-correlation*, in which correlation values for two digital signals are repeatedly computed as one moves in step-wise, point-by-point fashion along the other. The result is a *detection function*, in which the similarity of the two signals is shown by the correlation value calculated at each point in time. If a signal is cross-correlated with a copy of itself (i.e., *autocorrelated*), the detection function reaches as maximum value of 1.0 at the point the two copies are perfectly aligned. Detection functions for non-identical signals vary between 0 and 1.0.

Although cross-correlation is most often applied to time-series waveforms, this technique can be especially useful for detection and comparison of animal vocalizations when applied to spectrographic representations (e.g., Clark et al. 1987). This approach is particularly advantageous when comparing complex acoustic signals. Whereas a time-series waveform has only two dimensions (i.e., time and amplitude), a time spectrogram separates signals into three dimensions (i.e., time, frequency, and amplitude). Adding a third analysis dimension provides additional signal separation, an advantage evident to anyone who has compared a time spectrogram of a birdsong to the time-by-amplitude oscillogram-based waveform display. Cross-correlation of time spectrograms works best when the spectrographic analysis involved represents an optimal combination of frequency versus time resolution (see Beecher 1988, as well as chapters by Clements, Beeman, and Owren and Bernacki in this Volume, for detailed discussion of issues related to spectral and temporal resolution in Fourier-transform-based spectrographic analysis).

8
Reducing Environmental Noise: An Example Using an FIR Filter and a Simple Matched Filter

A few years ago, I wondered if the interpulse interval within a field cricket's chirps vary with temperature, as do the intervals between the chirps themselves. Crisp autumn weather made trips to the front yard more appealing than a trip to the library. Precise measurement of interpulse interval was easily accomplished by extracting measurements from tape-recorded waveforms digitized directly into the soundport of a Macintosh computer. I stepped outside every couple of hours to record the song of an amorous male field cricket (*Gryllus pennsylvanicus*) as the temperature changed throughout the day. A problem arose in the late afternoon when a male ground cricket (*Allonembius* sp.) moved next to the singing field cricket and dominated my recordings with its persistent trill. Rush-hour traffic had begun and a steady stream of cars whooshed by the yard. The field cricket chirps that were initially so obvious in the waveform became invisible in the noise. I probably could have found field cricket chirps by looking at a spectrogram instead of the waveform, but rejected this option because I wanted better time resolution than a digital spectrogram can provide. Instead of losing the sample, I used digital filtering techniques to extract the signal from the noise. The procedure used is described below.

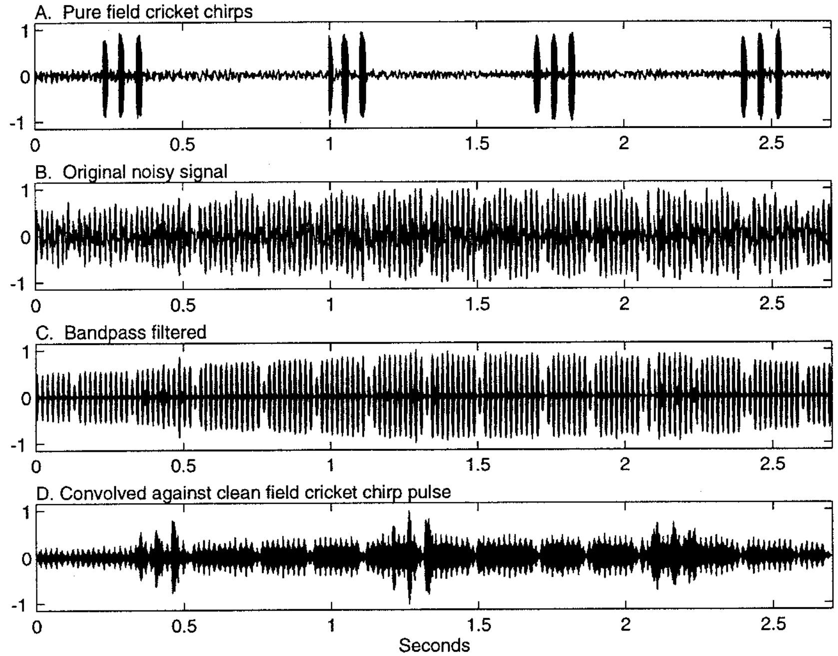

Fig 7 A-D. Cleaning a noisy signal with digital filtering. A. The first waveform is a reasonably pure series of four chirps of a male field cricket. **B.** The second waveform includes three chirps of a field cricket obscured by the trill of a ground cricket and the low frequency noise of passing automobiles. **C.** The noisy signal (B) is bandpass filtered with a 100th order FIR filter. **D.** The result of convolving the bandpass filtered signal **C.** against a single chirp pulse from signal **A.** The three chirps are now evident

I began by digitizing two 2.75 s sound segments from the audio tape. The first contained a fairly pure series of four fieldcricket chirps (illustrated in Figure 7A), while the second included a series of 3 fieldcricket chirps contaminated with both the ground cricket's trill and car noise (Figure 7B). The field cricket's chirps were nowhere evident in this noisy segment.

Before attempting to filter out the noise, I needed to better understand the spectral properties both of the target signal and of the noise obscuring it. I therefore plotted power spectrum density estimates of a pure chirp and a section of noise with no chirp (Figure 7A and 7B, respectively), and used Welch's averaged periodogram method to average the squared FFT magnitudes of overlapping 256-point segments. The "pure" chirp turned out to be not so pure – its spectrum is shown as the solid line in Figure 8. As can be seen, the low-frequency noise extended up to about 250 Hz, the chirp's fundamental frequency (basic rate of vibration) was about 4.1 kHz, and its second harmonic was at 8.2 kHz. The noise segment (the dashed line in Figure 8) had a similar, but louder, low-frequency band and was dominated by the trill of the field cricket. This call showed a fundamental of 4.9 kHz and a second harmonic of 9.8 kHz. As these were noisy field

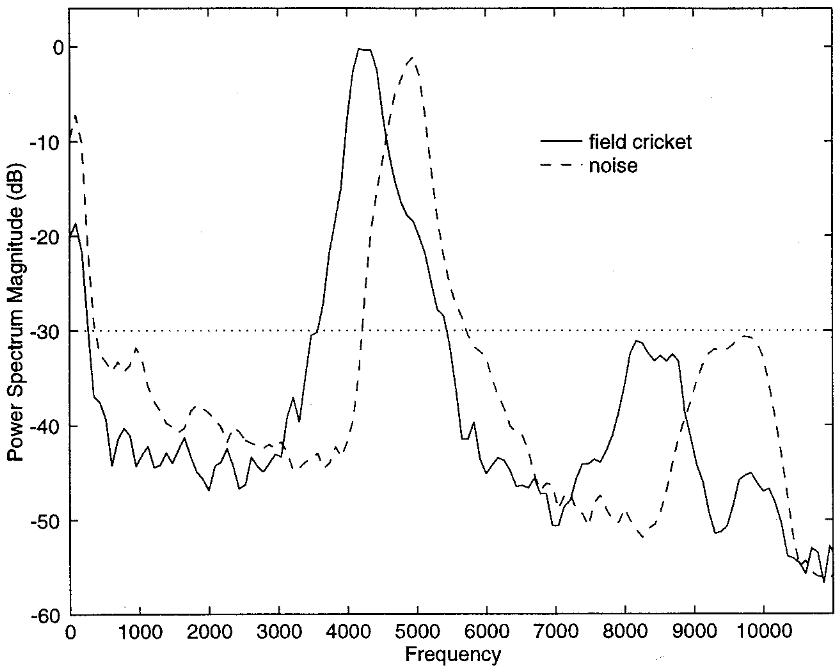

Fig. 8. Normalized spectral density estimates of the clean cricket chirps in Fig. 7A and a section of noise from waveform in Fig. 7B. The field cricket signal (*solid line*) is dominated by the fundamental frequency of its song. The noise spectrum (*dashed line*) is dominated by the loud trill of a ground cricket. The second harmonics are evident an octave above the fundamentals. Low-frequency noise from automobiles is seen at the lower edge of the spectrum

recordings, I decided to set -30 dB as the retention threshold. That is, I did not attempt to preserve components that were lower than -30 dB in relative amplitude (shown by the dotted line in Figure 8).

The low-frequency automobile noise could be removed easily with a digital highpass filter set to a cut-off frequency of 3.5 kHz. Unfortunately, the ground cricket signal overlapped the field cricket signal in time and partially overlapped it in frequency. A digital lowpass filter set to a cut-off of 5.4 kHz could therefore help but would not be sufficient. It was, worth a try. Using a program that designs linear delay FIR filters of arbitrary shape, I requested a 100th-order bandpass filter with the following characteristics:

Frequency (Hz)	0	2500	3500	5400	5800	Nyquist
Relative amplitude	0	1	1	0	0	0

Figure 9 shows the "requested" filter function and the frequency response of the impulse function my program returned. The impulse function itself (shown in Figure 10A) looked nothing like the frequency response curve. Instead its waveform resembles coarsely the waveform structure of the clean field cricket chirp pulse, as is evident in

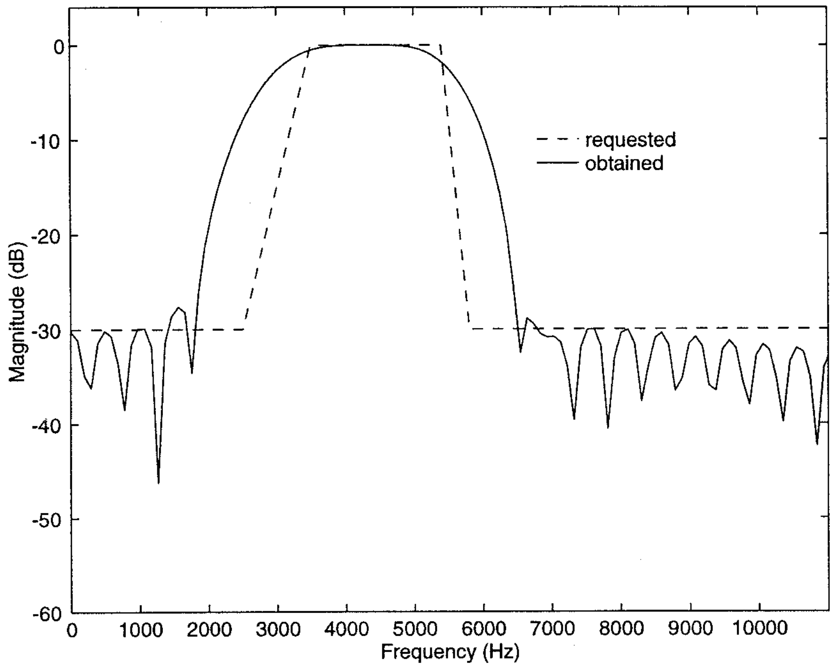

Fig 9. Digital FIR bandpass filter frequency x amplitude response patterns requested (*dashed line*) and obtained (*solid line*) from a digital filter design program

comparing Figures 10A,B. Convolving the FIR impulse function against the noisy signal produced the filtered output shown in Figure 7C. The field-cricket chirps were still obscured by the ground cricket trill – something more was needed.

The repetitive and stereotypical nature of the cricket chirp waveforms lend themselves to matched filtering. On this basis, I felt confident extracting the field cricket chirps from the combined signal by convolving a clean chirp against the combined signal. If an entire chirp were to be used as the model, its interpulse timing would affect the timing of the output. Therefore I selected as my filter function a single pulse from a pure fieldcricket chirp (shown in Figure 10c), processing it using an FIR bandpass filter, as described above, to remove extraneous low-frequency noise. I convolved this filtered pulse (see Figure 10B) against the bandpass-filtered chirp series (illustrated in Figure 7C) and the resulting waveform showed three distinct field-cricket chirps (as in Figure 7D) from which I could easily extract the interpulse intervals.

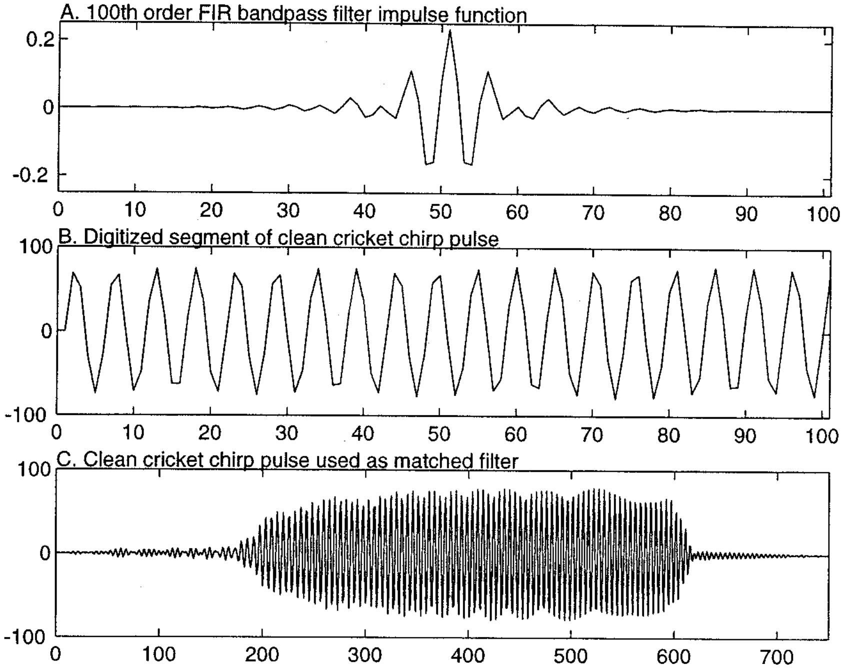

Fig 10A. Impulse function for the digital FIR bandpass filter described in Fig 9. Front-to-back symmetry of the impulse function insures linear phase delay. This impulse function is convolved against a waveform (Fig. 7B) to produce the filtered output (Fig. 7C). B. Section of a clean chirp. Notice the similarity in waveform between the impulse function and this chirp segment. C. The bandpass filtered chirp pulse that is used as a matched filter. This chirp pulse is convolved with the waveform in Fig. 7C to extract the chirp structure seen in Fig. 7D

9 Endnote: Solution to the Puzzle of the Disappearing Half-Masked Weaver Bird Voice

In recording the half-masked weaver song, my colleague had overloaded the analog audio recorder's circuitry, thereby clipping the waveform's amplitude peaks and squaring their tops. The Fourier series of a square wave has energy at all odd-numbered harmonics (h_n, where n = 1, 3, 5...) with the relative power of each being equal to 1 / n. As a result, the original signal, a modulated whistle with a fundamental that swept from about 2 kHz to 3 kHz, was now accompanied by a loud third harmonic that ranged from about 6 kHz to 9 kHz. I had set the real-time analyzer to a frequency range of 0–5 kHz and in so doing apparently had lowered the machine's Nyquist frequency to about 5 kHz. Ordinarily, most of the energy in the third harmonic would have been removed by a lowpass filter, but with no filter board in place, this component had folded downward across the Nyquist frequency. There, it appeared as a downward sweep from about 4 kHz to 1 kHz, crossing the fundamental and bearing no obvious resemblance to a harmonic. Only the antiquated, analog technology of the old Kay Sonagraph had exposed my ignorance and saved me from attempting to publish a description of a recording artifact as an exciting new finding. Had I that day in 1982 access to the digital printout capabilities that are now available, I might have submitted for publication a textbook example of signal aliasing. I wonder how many reviewers would have noticed the error.

Acknowledgments I thank Ben Clopton for introducing me to sampling and filter theory in his "Sound and Math Seminar" at the University of Washington. My exploration of digital filters was facilitated by the DADiSP and DADiSP Filters software packages, provided generously by the DSP Development Corporation for assistance with this chapter. Example signals and figures were prepared with MATLAB and the MATLAB Signal Processing Toolbox (The Mathworks, Inc.) on a Power Macintosh 7100 (Apple Computer) equipped with an Audiomedia II A/D card (Digidesign, Inc.). Support was provided by a summer grant from the Florida International University Foundation.

References

Beecher MD (1988) Spectrographic analysis of animal vocalizations: implications of the "uncertainty principle". Bioacoustics 1:187–208
Clark CW, Marler P, Beeman K (1987) Quantitative analysis of animal vocal phonology: An application to swamp sparrow song. Ethology 76:101–115
Cook EW, Miller GA (1992) Digital filtering: background and tutorial for psychophysiologists. Psychophysiology 29:350–367
Hamming RW (1989) Digital filters. Prentice Hall, Englewood Cliffs, New Jersey
Haydin S (1986) Adaptive filter theory. Prentice Hall, Englewood Cliffs, New Jersey
Keller CH, Takahashi TT (1996) Responses to simulated echoes by neurons in the barn owl's auditory map. J Comp Physiol A 178:499–512
Menne D (1989) Digital filters in auditory physiology. Bioacoustics 2:87–115
Okanoya K, Dooling RJ (1987) Hearing in passerine and psittacine birds: a comparative study of absolute and masked auditory thresholds. J Comp Physiol 101:7–15
Oppenheim AV, Schafer RW (1989) Discrete-time signal processing. Prentice-Hall, Englewood Cliffs,
Stearns S, David R (1988) Signal processing algorithms. Prentice Hall, Englewood Cliffs,
Wickstrom DC (1982) Factors to consider in recording avian sounds. In: Kroodsma DE, Miller EH, Ouellet H (eds) Acoustic communication in birds. vol 1. Academic Press, New York, p 1