

4

Signal Averaging

4.1 INTRODUCTION

Data analysis techniques are commonly subdivided into operations in the *time domain* (or *spatial domain*) and *frequency domain*. In this chapter we discuss processing techniques applied in the time (spatial) domain with a strong emphasis on *signal averaging*. Signal averaging is an important technique that allows estimation of small amplitude signals that are buried in noise. The technique usually assumes the following:

1. Signal and noise are uncorrelated.
2. The timing of the signal is known.
3. A consistent signal component exists when performing repeated measurements.
4. The noise is truly random with zero mean.

In the real world, all these assumptions may be violated to some degree; however the averaging technique has proven sufficiently robust to survive minor violations of these four basic assumptions. A brief overview of other frequently used time domain techniques can be found in Section 4.8.

4.2 TIME LOCKED SIGNALS

Averaging is applied to enhance a time-locked signal component in noisy measurements. One possible representation of such a signal is as measurement x consisting of a signal s and a noise component n , with the underlying assumption that the measurement can be repeated over N trials. In the case where each trial is digitized, the k th sample point in the j th trial (Fig. 4.1), can be written as

$$x_j(k) = s_j(k) + n_j(k) \quad (4.1)$$

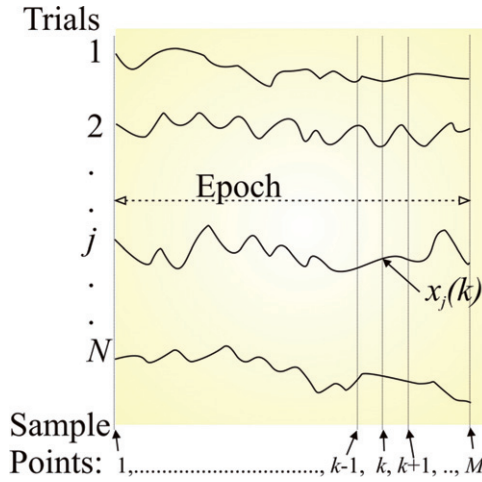


Figure 4.1 A set of N raw trials composed of a signal and significant noise component can be used to obtain an average with an enhanced signal-to-noise ratio. Each full epoch consists of a series of individual sample points $x_j(k)$, with $k = 1, 2, \dots, k, \dots, M$.

with k being the sample number ($k = 1, 2, \dots, M$). The *rms* value of s can be several orders of magnitude smaller than that of n , meaning that the signal component may be invisible in the raw traces. After completion of N repeated measurements, we can compute an average measurement $\overline{x(k)}_N$ for each of the k sample indices:

$$\overline{x(k)}_N = \frac{1}{N} \sum_{j=1}^N x_j(k) = \frac{1}{N} \sum_{j=1}^N [s_j(k) + n_j(k)] \quad (4.2)$$

The series of averaged points (for k from 1 to M) obtained from Equation (4.2) constitutes the average signal of the whole epoch. In the following we explore some of the properties of signal averaging in a simulation.

The following MATLAB routine pr4_1.m is a simulation of the averaging process.

```
% pr4_1
% averaging
clear

sz=256;
NOISE_TRIALS=randn(sz);    % a [sz x sz] matrix filled with noise

SZ=1:sz;                   % Create signal with a sine wave
SZ=SZ/(sz/2);              % Divide the array SZ by sz/2
```

```

S=sin(2*pi*SZ);

for i=1:sz;                                % create a noisy signal
    NOISE_TRIALS(i,:) = NOISE_TRIALS(i,:) + S;
end;

average=sum(NOISE_TRIALS)/sz;               % create the average
odd_average=sum(NOISE_TRIALS(1:2:sz,:))/(sz/2);
even_average=sum(NOISE_TRIALS(2:2:sz,:))/(sz/2);
noise_estimate=odd_average-even_average;

figure
hold
plot(NOISE_TRIALS(1,:), 'g')
plot(noise_estimate, 'k')
plot(average, 'r')
plot(S)
title('Average RED, Noise estimate BLACK; Single trial GREEN,
      Signal BLUE')

```

As shown in the simulation result depicted in Figure 4.2, the averaging process described by Equation (4.2) results in an estimate of the signal. As compared with the raw (signal + noise) trace in Figure 4.2, the averaged noise component is reduced in a signal average of 256 trials. When averaging real signals, the underlying component may not always be as clear as it is in the example provided in Figure 4.2. In these cases, the averages are often repeated in search of consistent components in two or three replicates (e.g., see the superimposed somatosensory-evoked potential (SEP) waveforms in Fig. 1.3). The idea here is that it is unlikely that two or more consistent averaged results will be produced by chance alone. A specific way of obtaining replicates is to average all *odd* and all *even* trials in separate buffers (see the superimposed *odd_average* and *even_average* in Fig. 4.2). This has the advantage of allowing for comparison of the even and odd results from interleaved trials. An average of the odd and even averages (i.e., addition of the odd and even results divided by 2) generates the complete averaged result, while the difference of the two constitutes an estimate of the noise (see Section 4.4 for details on such a noise estimate).

4.3 SIGNAL AVERAGING AND RANDOM NOISE

If the noise in Equation (4.2) is a 0-mean random process $\langle x(k) \rangle = \langle s(k) \rangle + 0$, where $\langle \dots \rangle$ indicates the true value of the enclosed variable, what would we get if averaged over a large number of trials (i.e., $N \rightarrow \infty$).

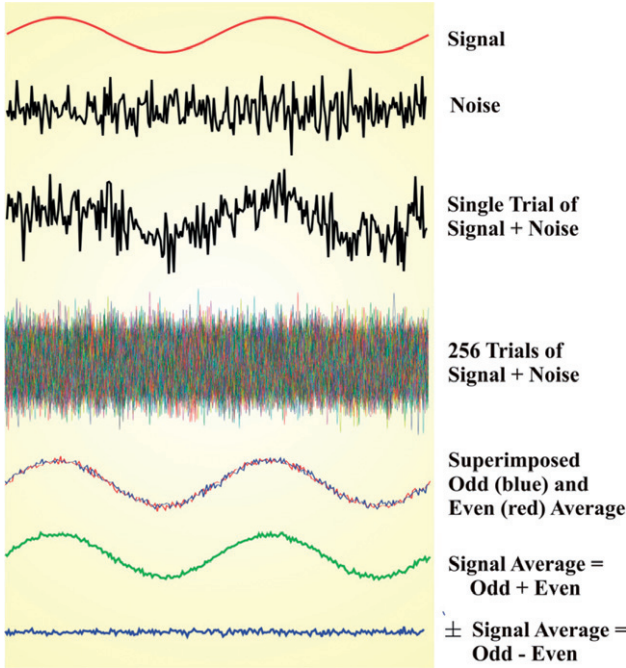


Figure 4.2 Signal averaging of a signal buried in noise (signal + noise). This example shows 256 superimposed trials (fourth trace) of such a measurement and the average thereof. The average results of the odd and even trials are shown separately (fifth trace). The sum of all trials divided by the number of trials (sixth trace, signal average) shows the signal with a small component of residual noise. A \pm average (bottom trace) is shown as an estimate of residual noise. The example traces are generated with MATLAB script pr4_1.

Therefore, the general idea of signal averaging is to reduce the noise term $\rightarrow 0$ for large N such that $x(k)_N \rightarrow s(k)_N$. Because in the real world $N \ll \infty$, we will not reach the ideal situation where the measurement x exactly equals the true signal s ; there is a residual noise component that we can characterize by the variance of the estimate $\bar{x}(k)_N$. To simplify notation, we will indicate $\text{Var}(\bar{x}(k)_N)$ as $\text{Var}(\bar{x})$. The square root of $\text{Var}(\bar{x})$ is the standard error of the mean (SEM). We can use Equation (3.11) to estimate $\text{Var}(\bar{x})$:

$$\text{Var}(\bar{x}) = E\{(\bar{x} - \langle x \rangle)^2\} = E\{\bar{x}^2 - 2\bar{x}\langle x \rangle + \langle x \rangle^2\}$$

Taking into account that $\langle x \rangle$ represents the true value of x (therefore $E\{\langle x \rangle\} = \langle x \rangle$ and $E\{\langle x \rangle^2\} = \langle x \rangle^2$), we may simplify

$$E\{\bar{x}^2 - 2\bar{x}\langle x \rangle + \langle x \rangle^2\} = E\{\bar{x}^2\} - 2\langle x \rangle E\{\bar{x}\} + \langle x \rangle^2$$

Further, we note that the expected value of the average of x (\bar{x}) is equivalent to $\langle x \rangle$ (i.e., $E\{\bar{x}\} = \langle x \rangle$); the expression can be simplified further leading to

$$Var(\bar{x}) = E\{\bar{x}^2\} - \langle x \rangle^2 \quad (4.3)$$

Combining Equations (4.3) and (4.2), we obtain

$$\begin{aligned} Var(\bar{x}) &= E\left\{\left[\frac{1}{N}\sum_{i=1}^N x_i\right]^2\right\} - \langle x \rangle^2 = E\left\{\left[\frac{1}{N}\sum_{j=1}^N x_j\right]\left[\frac{1}{N}\sum_{i=1}^N x_i\right]\right\} - \langle x \rangle^2 \\ &= \frac{1}{N^2} \sum_{j=1}^N \sum_{i=1}^N E\{x_j x_i\} - \langle x \rangle^2 \end{aligned} \quad (4.4)$$

The two summations in this expression represent all combinations of i and j , both going from 1 to N and therefore generating $N \times N$ combinations. This set of combinations can be separated into N terms for all $i = j$ and $N^2 - N = N(N - 1)$ terms for $i \neq j$:

$$Var(\bar{x}) = \frac{1}{N^2} \underbrace{\sum_{j=1}^N E\{x_j^2\}}_{\text{for } i=j} + \frac{1}{N^2} \underbrace{\sum_{j=1}^N \sum_{i=1, i \neq j}^N E\{x_j x_i\}}_{\text{for } i \neq j} - \langle x \rangle^2 \quad (4.5)$$

The separation in Equation (4.5) is useful because the properties of the terms for $i = j$ and for $i \neq j$ differ significantly. As we will see in the following, by exploring this product $x_i x_j$, we will be able to further simplify this expression as well as clarify the mathematical assumptions underlying the signal averaging technique. As these *assumptions surface in the text of this section, they will be presented underlined and italic*. Using Equation (4.1), we can rewrite a single contribution to the summation of N terms with $i = j$ in equation (4.5) as

$$E\{x_j^2\} = E\{[s_j + n_j]^2\} = E\{s_j^2\} + 2E\{s_j\}E\{n_j\} + E\{n_j^2\} \quad (4.6)$$

In the second term we simplified $E\{2s_j n_j\}$ to $2E\{s_j\}E\{n_j\}$ because we assume that *noise and signal are independent*. Assuming that the noise component n_j has zero mean and variance σ_n^2 —that is, $E\{n_j\} = 0$ and $E\{n_j^2\} = \sigma_n^2$,

$$E\{x_j^2\} = E\{s_j^2\} + \sigma_n^2 \quad (4.7)$$

The variance of the signal component (σ_s^2) is given by $\sigma_s^2 = E\{s_j^2\} - \langle s \rangle^2$, which we may substitute for the first term in Equation (4.7), producing

$$E\{x_j^2\} = \sigma_s^2 + \langle s \rangle^2 + \sigma_n^2 \quad (4.8)$$

Combining Equations (4.1) and (4.5), the expression for one of the $N(N - 1)$ cross terms can be written as

$$E\{x_j x_i\} = E\{[s_j + n_j][s_i + n_i]\} = E\{s_j s_i\} + E\{n_j n_i\} + E\{s_j n_i\} + E\{s_i n_j\} \quad (4.9)$$

If we assume that *the noise and the signal are statistically independent* within a given trial and also that the *noise and signal measurements are themselves independent across trials* (i.e., independent between trials i and j), then this independence assumption allows us to rewrite all combined expectations as the product of the individual expectations:

$$\begin{aligned} E\{s_j s_i\} &= E\{s_j\}E\{s_i\} = \langle s \rangle \times \langle s \rangle = \langle s \rangle^2 \\ E\{n_j n_i\} &= E\{n_j\}E\{n_i\} = 0 \times 0 = 0 \\ E\{s_j n_i\} &= E\{s_j\}E\{n_i\} = \langle s \rangle \times 0 = 0 \\ E\{s_i n_j\} &= E\{s_i\}E\{n_j\} = \langle s \rangle \times 0 = 0 \end{aligned} \quad (4.10)$$

Substituting from Equation (4.8) for the N $i = j$ terms and from Equations (4.9) and (4.10) for the $N(N - 1)$ $i \neq j$ terms into Equation (4.5), we obtain the following expression for the variance:

$$Var(\bar{x}) = \frac{1}{N^2} [N(\sigma_s^2 + \langle s \rangle^2 + \sigma_n^2) + (N^2 - N)\langle s \rangle^2] - \langle x \rangle^2 \quad (4.11)$$

Finally, again using the assumption that $\langle n \rangle = 0$, the true value of the measurement x is the averaged signal (i.e., $\langle x \rangle = \langle s \rangle$). This allows us to simplify Equation (4.11):

$$Var(\bar{x}) = \frac{1}{N^2} [N(\sigma_s^2 + \langle s \rangle^2 + \sigma_n^2) + (N^2 - N)\langle s \rangle^2] - \langle s \rangle^2 \quad (4.12)$$

This expression simplifies to

$$Var(\bar{x}) = \frac{\sigma_s^2 + \sigma_n^2}{N} \quad (4.13)$$

Equation (4.13) quantifies the variance of the average (\bar{x}), showing that the estimate of the mean improves with an increasing number of repetitions N . In our example, the variances σ_s^2 , σ_n^2 are generated by two independent sources. In this case, the compound effect of the two sources is obtained by adding the variances, similar to the combined effects of independent sources on v_{off} in Equation (3.15). The square root of the expression in Equation (4.13) gives us the standard error of the mean; therefore we conclude

that the estimate of s in the average \bar{x} improves with a factor of $\frac{1}{\sqrt{N}}$.

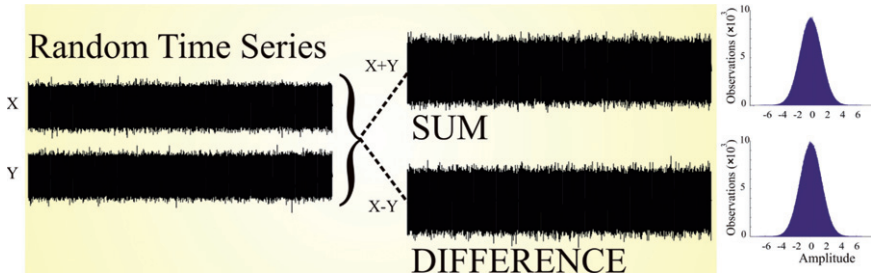


Figure 4.3 Random noise traces X and Y , their sum ($X + Y$), and difference ($X - Y$) waves. The two amplitude distributions on the right are similar for the sum and difference signals, suggesting that they can be characterized by the same PDF. For random noise, an addition or subtraction creates different time series (i.e., $X + Y \neq X - Y$) but does **not** create different statistical characteristics. This property of random noise is used when considering the \pm signal average (Fig. 4.2, bottom trace) as the basis for estimating the *rms* of the residual noise in the averaged result.

4.4 NOISE ESTIMATES AND THE \pm AVERAGE

The ultimate reason to perform signal averaging is to increase the signal-to-noise ratio (Chapter 3). The estimate of residual noise can easily be established in a theoretical example illustrated in the simulation in pr4_1 where all the components are known. In real measurements, the noise and signal components are unknown and the averaged result is certain to contain both signal and residual noise (as in Fig. 4.2). One way of establishing the amount of residual noise separately is by using so-called \pm averaging, a procedure in which measurements from every other trial are inverted prior to creating the averaged result. This technique removes any consistent signal component by the alternating addition and subtraction. However, the residual noise is maintained in the end result (Fig. 4.3). The *rms* value of the noise component estimated from the \pm average is the same as that produced by the standard average because random noise samples from the inverted traces have the same distribution as the ones from noninverted trials. A demonstration (not a proof) is provided in the example in Figure 4.3 where a pair of random signals X and Y are added and subtracted. The similarity of the amplitude distributions of $X + Y$ and $X - Y$ confirm that the sum and difference signals have the same statistical properties.

4.5 SIGNAL AVERAGING AND NONRANDOM NOISE

The result in the previous section depends heavily on a noise component being random, having zero mean, and being unrelated to the signal. A

Periodic Noise Source (e.g., Hum at 50 Hz)

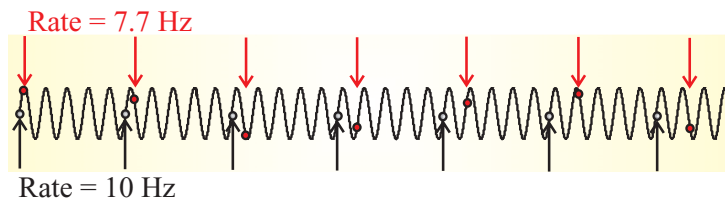


Figure 4.4 The stimulus rate and a periodic component (e.g., a 50-Hz or 60-Hz hum artifact) in the unaveraged signal can produce an undesired effect in the average. An average produced with a 10-Hz rate will contain a large 50-Hz signal. In contrast, an average produced with a 7.7-Hz rate will not contain such a strong 50-Hz artifact. This difference is due to the fact that a rate of 10 Hz results in a stimulus onset that coincides with the same phase in the 50-Hz sinusoidal noise source (black dots), whereas the non-integer rate of 7.7 Hz produces a train of stimuli for which the relative phase of the noise source changes with each stimulus (red dots).

special case occurs when the noise is not random. This situation may affect the performance of the average and even make it impossible to apply the technique without a few critical adaptations. The most common example of such a situation is the presence of hum (50- or 60-Hz noise originating from the power lines; Chapter 3 and Fig. 3.4). In typical physiological applications, an average is obtained by repeating a standard stimulus of a certain sensory modality and recording a time-locked epoch of neural (of neuromuscular) responses at each repetition. Usually these series of stimuli are triggered at a given stimulus rate dictated by the type of experiment. It is critical to understand that in this scenario not only are time-locked components evoked by each stimulus enhanced in the average result but also periodic components (Fig. 4.4) with a fixed relation to the stimulus rate! For example, if one happens to stimulate at a rate of exactly 50 Hz, one enhances any minor 50-cycle noise in the signal (the same example can be given for 60 Hz). The situation is worse, because any stimulus rate r that divides evenly into 50 will have a tendency to enhance a small 50-cycle noise signal (for example, the 10-Hz rate represented by the black dots in Fig. 4.4). This problem is often avoided by either *randomizing the stimulus interval* or by using a *noninteger stimulus rate* such as 3.1, 5.3, or 7.7 Hz (red in Fig. 4.4).

Although this consideration with respect to periodic noise sources seems trivial, averaging at a poorly chosen rate is a common mistake. I have seen examples where expensive Faraday cages and power supplies were installed to reduce the effects of hum, while with normal averaging procedures, a simple change of the stimulus rate from 5.0 to 5.1 would have been much cheaper and usually more effective.

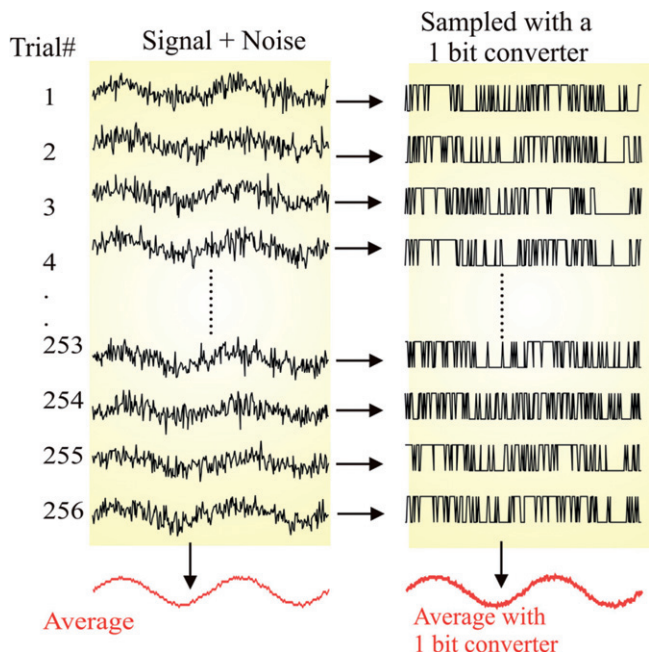


Figure 4.5 Signal averaging of the 256 traces generated by `pr4_1.m` is shown in the left column. The right column shows individual traces that were digitized with a 1-bit ADC using the MATLAB commands in `pr4_3`. The averaged result of the traces in the right column is surprisingly close to the average obtained from the signals in the left column. Note that the relative noise component of the 1-bit average is large compared to the standard result shown in the left column. Because the 1-bit converter only produces values between 0 and 1, all amplitudes are normalized to allow comparison.

The figure generated by the preceding commands/script shows a digitized representation of the signal on a scale from 0 to 1. Figure 4.5 compares the averaging result obtained in our original run of `pr4_1` and the result obtained here by simulating a 1-bit converter.

The example shows that reasonable averaging results can be obtained with a low-resolution ADC using the statistical properties of the noise component. This suggests that ADC resolution may not be a very critical component in the signal averaging technique. To explore this a bit further, let's compare two signal averagers that are identical with the exception of the ADC precision: *averager A* has a **4-bit resolution ADC**, and *averager B* has a **12-bit ADC**. *Let us say we want to know the number of trials N required to obtain an averaged result with signal-to-noise ratio of at least 3* (according to Equation (3.13) ≈ 9.5 dB) in both systems. Further, let's assume we have a ± 15 V range at the ADC input and an amplification of 100,000 \times . In this example, we consider an *rms* value for the signal of

5 μV and for the noise component of 50 μV . For simplicity, we assume a consistent signal (i.e., the variance in the signal component is zero). The signal-to-noise ratio at the amplifier input of both (Equation (3.13)) is $20 \log_{10} \frac{5}{50} = -20 \text{ dB}$. (Our target is therefore a $9.5 - (-20) = 29.5 \text{ dB}$ improvement in signal-to-noise ratio.) At the amplifier output (= the ADC input) of both systems, we have

$$\begin{aligned} rms_{\text{signal}} & 5 \mu\text{V} \times 100,000 = 0.5 \text{ V} \\ rms_{\text{noise}} & 50 \mu\text{V} \times 100,000 = 5 \text{ V} \end{aligned} \quad (4.14)$$

The quantization noise q_A and q_B in **systems A and B** is different due to the different resolution of their ADC components. At the output of the systems, the range of this added noise is

$$\begin{aligned} \text{Averager A: } q_A &= \pm 15 \text{ V} / 2^4 = 30/16 \approx 1.88 \text{ V} \\ \text{Averager B: } q_B &= \pm 15 \text{ V} / 2^{12} = 30/4096 \approx 7.3010^{-3} \text{ V} \end{aligned} \quad (4.15)$$

The variances $\sigma_{q_A}^2$ and $\sigma_{q_B}^2$ associated with these quantization ranges (applying Equation (3.26)) are

$$\begin{aligned} \text{Averager A: } \sigma_{q_A}^2 &= (30/16)^2 / 12 \text{ V}^2 \\ \text{Averager B: } \sigma_{q_B}^2 &= (30/4096)^2 / 12 \text{ V}^2 \end{aligned} \quad (4.16)$$

Combining the effect of the two noise sources in each system, we can determine the **total noise** at the input of the ADC as the combination of the original noise at the input (5^2 V^2) and that produced by quantization:

$$\begin{aligned} \text{Averager A: } 5^2 + \sigma_{q_A}^2 &= 5^2 + (30/16)^2 / 12 \text{ V}^2 \\ \text{Averager B: } 5^2 + \sigma_{q_B}^2 &= 5^2 + (30/4096)^2 / 12 \text{ V}^2 \end{aligned} \quad (4.17)$$

According to Equation (4.13), these noise figures will be attenuated by a factor N_A or N_B (number of trials in systems A and B) in the averaged result. Using the signal-to-noise ratio $rms_{\text{signal}} / rms_{\text{Total Noise}}$ and including our target (a ratio of 3 or better), we get

$$\begin{aligned} \text{Averager A: } \frac{0.5}{\sqrt{\frac{5^2 + \sigma_{q_A}^2}{N_A}}} &= \frac{0.5}{\sqrt{\frac{5^2 + (30/16)^2}{N_A}}} \geq 3 \\ \text{Averager B: } \frac{0.5}{\sqrt{\frac{5^2 + \sigma_{q_B}^2}{N_B}}} &= \frac{0.5}{\sqrt{\frac{5^2 + (30/4096)^2}{N_B}}} \geq 3 \end{aligned} \quad (4.18)$$

Solving for the number of trials required in both systems to get this signal-to-noise target, we find that $N_A = 901$ and $N_B = 1027$. From this example we conclude that in a high noise environment (i.e., with a high noise level relative to the quantization error q), the precision of the ADC does not influence the end result all that much; in our example, a huge difference in precision (4 versus 12 bit, which translates into a factor of 256) only resulted in a small difference in the number of trials required to reach the same signal-to-noise ratio (1027 versus 901, a factor of ~ 1.14). The example also shows that in a given setup, improvement of the signal-to-noise ratio with averaging is best obtained by increasing the number of trials; from Equation (4.18) we can determine that the signal-to-noise improvement is proportional to \sqrt{N} .

4.7 EVOKED POTENTIALS

Evoked potentials (EPs) are frequently used in the context of clinical diagnosis; these signals are good examples of the application of signal averaging in physiology (Chapter 1). The most commonly measured evoked potentials are recorded with an EEG electrode placement and represent neural activity in response to stimulation of the auditory, visual, or somatosensory system (AEP, VEP, or SEP, respectively). These examples represent activity associated with the primary perception process. More specialized evoked potentials also exist; these record the activity generated by subsequent or more complex tasks performed by the nervous system. One example is the so-called oddball paradigm, which consists of a set of frequent baseline stimuli, occasionally (usually at random) interrupted by a rare test stimulus. This paradigm usually evokes a centrally located positive wave at 300 ms latency in response to the rare stimulus (the P300). This peak is generally interpreted as representing a neural response to stimulus novelty.

An even more complex measurement is the contingent negative variation (CNV) paradigm. Here the subject receives a warning stimulus (usually a short tone burst) that a second stimulus is imminent. When the second stimulus (usually a continuous tone or a series of light flashes) is presented, the subject is required to turn it off with a button press. During the gap in between the first (warning) stimulus and second stimulus, one can observe a centrofrontal negative wave. Relative to the ongoing EEG the CNV signal is weak and must be obtained by averaging; an example of individual trials and the associated average is shown in Figure 4.6. Here it can be seen that the individual trials contain a significant amount of noise, whereas the average of only 32 trials clearly depicts the negative slope between the

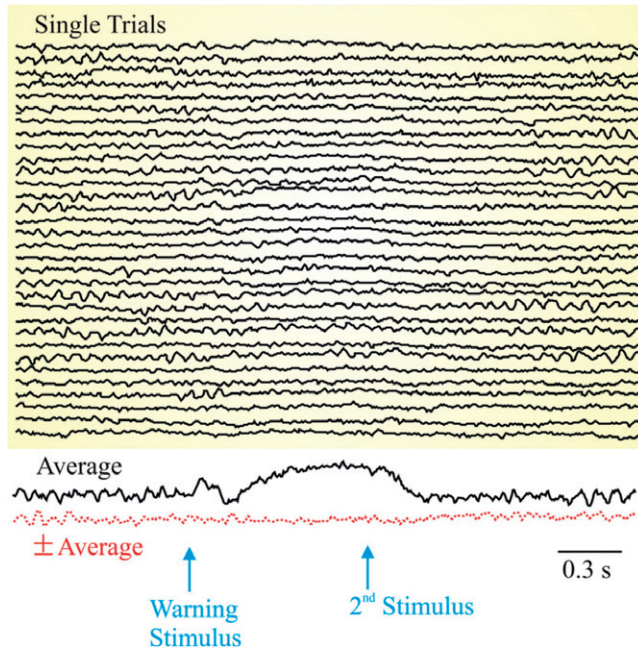


Figure 4.6 The contingent negative variation (CNV) measured from C_z (the apex of the scalp) is usually made visible in the average of individual trials in which a subject receives a warning stimulus that a second stimulation is imminent. The second stimulus must be turned off by a button press of the subject. The lower pair of traces shows the standard average revealing the underlying signal and the \pm average as an estimate of the residual noise.

stimuli (note that negative is up in Fig. 4.6). The \pm average provides an estimate for the residual noise in the averaged result. The original trials are included on the CD (single_trials_CNV.mat).

Typing the following MATLAB commands will display the superimposed 32 original traces as well as the average of those trials.

```
clear
load single_trials_CNV
figure
plot(single)
hold
plot(sum(single')/32,'k+')

```

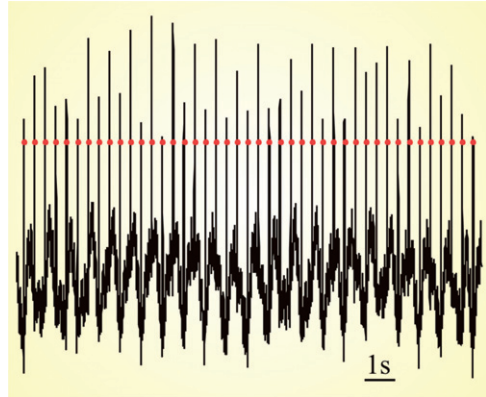
4.8 OVERVIEW OF COMMONLY APPLIED TIME DOMAIN ANALYSIS TECHNIQUES

1. **Power and related parameters (Chapter 3).** Biomedical applications often require some estimate of the overall strength of measured signals. For this purpose, the variance (σ^2) of the signal or the mean of the sum of squares $\left(\frac{1}{N} \sum_{n=1}^N x^2(n)\right)$ is frequently used. Time series are also frequently demeaned (baseline corrected) before further analysis, making the mean of the sum of squares and the variance equivalent. Another variant is the *rms* (root mean square; Chapter 3).

Hjorth (1970) described the signal variance σ^2 as the *activity index* in EEG analysis. In the frequency domain, activity can be interpreted as the area under the curve of the power spectrum. To this metric he added the standard deviations from the first and second derivatives of the time series, σ_d and σ_{dd} , respectively. On the basis of these parameters, Hjorth introduced *mobility* $\frac{\sigma_d}{\sigma}$ and *complexity* $\frac{\sigma_{dd}/\sigma_d}{\sigma_d/\sigma}$ parameters. In the frequency domain, mobility can be interpreted as the standard deviation of the power spectrum. The complexity metric quantifies the deviation from a pure sine wave as an increase from unity.

2. **Zero-crossings.** The 0-crossings in a demeaned signal can indicate the dominant frequency component in a signal. For example, if a signal is dominated by a 2-Hz sine wave, it will have four zero-crossings per second (i.e., the number of 0-crossings divided by 2 is the frequency of the dominant signal component). The lengths of epochs in between 0-crossings can also be used for *interval analysis*. Note that there are two types of 0-crossings, from positive to negative and vice versa. Zero-crossings in the derivative of a time series can also be used to find local *maxima* and *minima*.
3. **Peak detection.** Various methods to detect peaks are used to locate extrema within time series. If the amplitudes between subsequent local maxima and minima are measured, we can determine the amplitude distribution of the time series. In case of peak detection in signals consisting of a series of impulses, the peak detection procedures are used to calculate intervals between such events. This routine is frequently used to detect the events in signals containing spikes or in the ECG to detect the QRS complexes (Chapter 1, Fig. 1.4). An example of QRS complex detection in human neonates is shown in Figure 4.7. The general approach in these algorithms consists of two stages: first pretreat the signal in order to remove artifacts, and then detect extreme values above a set threshold.

Figure 4.7 An ECG signal from a human neonate and the detected QRS complexes (red dots).



The following part of a MATLAB script is an example of a peak detector used to create Figure 4.7. Note that this is a part of a script! The whole script `pr4_4.m` and an associated data file (`subecg`) are included on the CD.

```
% 1. preprocess the data
[c,d]=butter(2,[15/fN 45/fN],'bandpass');      % 2nd order 30–90 Hz
                                                % Chapter 13
subecgFF=filtfilt(c,d,subecg-mean(subecg));    % use filtfilt to prevent
                                                % phase-shift

% 2. detect peaks
% In this routine we only look for nearest neighbors (three
% subsequent points)
% adding additional points will make the algorithm more robust
threshold=level*max(subecgFF);                % detection threshold
for i=2:length(subecgFF)-1;
    if (subecgFF(i)>threshold);                % check if the level is
                                                % exceeded
        % is the point a relative maximum (Note the >= sign)?
        if((subecgFF(i)>=subecgFF(i-1))&(subecgFF(i)>=subecgFF(i+1)));
            % if yes, is it not a subsequent maximum in the same heartbeat
            if (i-i_prev > 50)
                D(n)=i;                        % Store the index in D
                i_prev=i;
                n=n+1;
            end;
        end;
    end;
end;
end;
```


4. **Level and window detection.** In some types of time series (such as in extracellular recordings of action potentials), one is interested in identifying epochs in which the signal is within a certain amplitude range. Analog- or digital-based window and level detectors are available to provide such data processing.
5. **Filtering** (Chapters 10 to 13). The filters we will consider in later chapters are both analog and digital implementations. For the analog filters, we will focus on circuits with a resistor (R) and capacitor (C) (RC circuits), the digital implementations will cover infinite impulse response (IIR) and finite impulse response (FIR) versions.
6. **Real convolution** (Chapter 8). Convolution plays an important role in relating input and output of linear time invariant systems.
7. **Cross-correlation** (Chapter 8). Cross-correlation is related to convolution and can be used to quantify the relationship between different signals or between different parts of the same signal (termed *auto-correlation*).
8. **Template matching.** In some applications, signal features are extracted by correlating a known template with a time series. Wavelet and scaling signals can be considered as a special type of template.
9. **Miscellaneous.** In some cases, the task at hand is highly specific (e.g., detection of epileptic spikes in the EEG). In these instances, a specially developed metric may provide a good solution. For example, in EEG spike detection, a “sharpness index” works reasonably well (Gotman and Gloor, 1976).