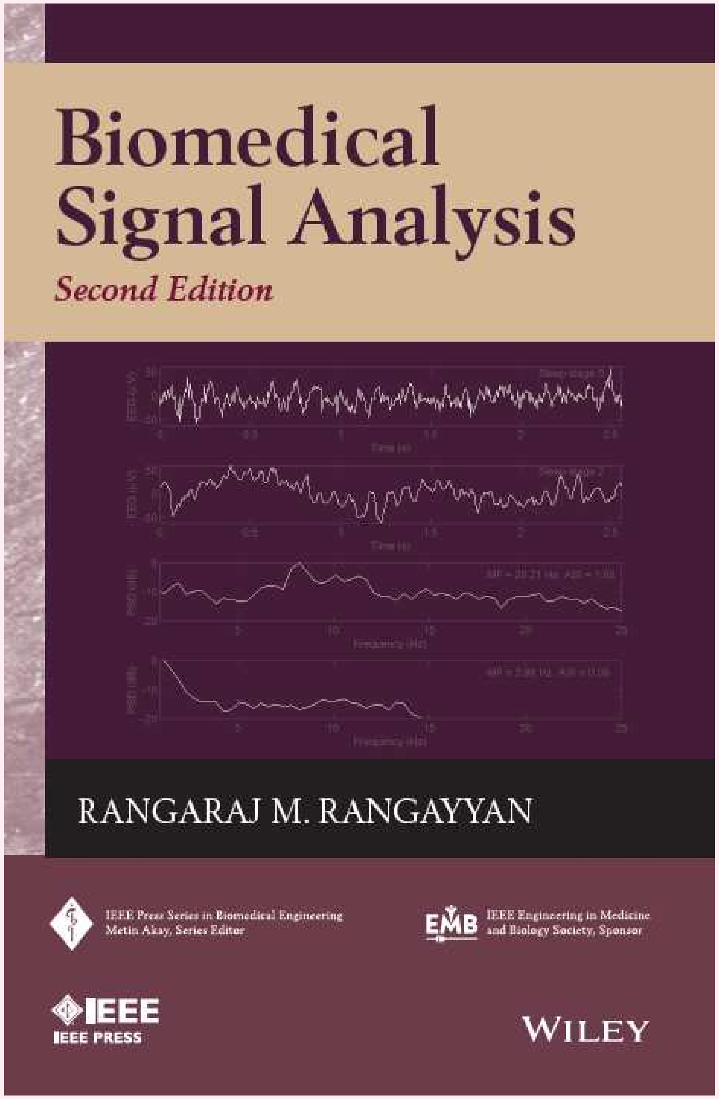
BIOMEDICAL SIGNAL ANALYSIS

Rangaraj M. Rangayyan

Professor
Department of Electrical and Computer Engineering
Schulich School of Engineering
Adjunct Professor, Departments of Surgery and Radiology
University of Calgary
Calgary, Alberta, Canada T2N 1N4

Phone: +1 (403) 220-6745
e-mail: ranga@ucalgary.ca

Web: http://people.ucalgary.ca/~ranga/enel563



Biomedical Signal Analysis

Second Edition

RANGARAJ M. RANGAYYAN

IEEE Press Series in Biomedical Engineering
Metin Akay, Series Editor

EMB IEEE Engineering in Medicine
and Biology Society, Sponsor

IEEE
IEEE PRESS

WILEY

IEEE/ Wiley, New York, NY, 2nd Edition, 2015

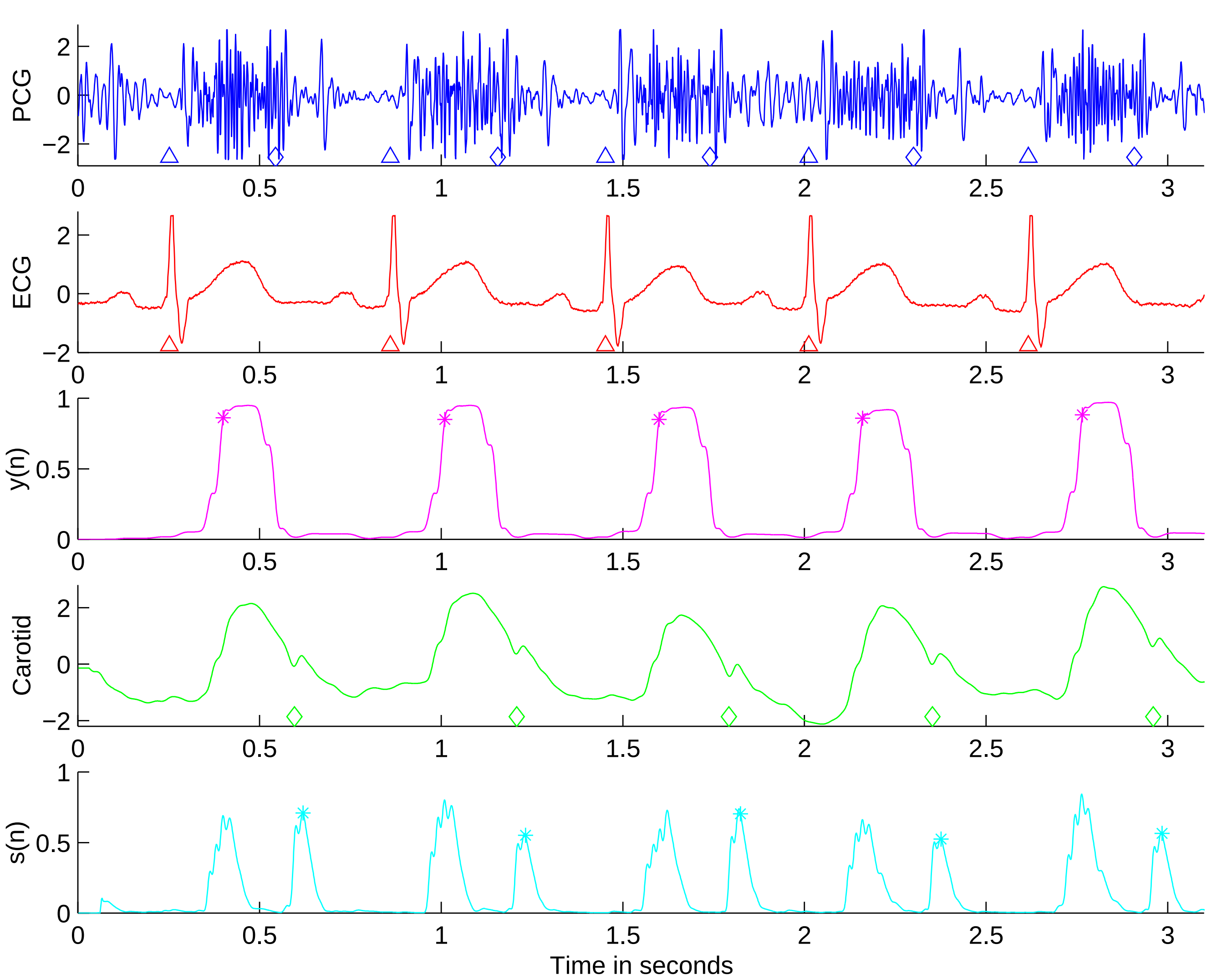
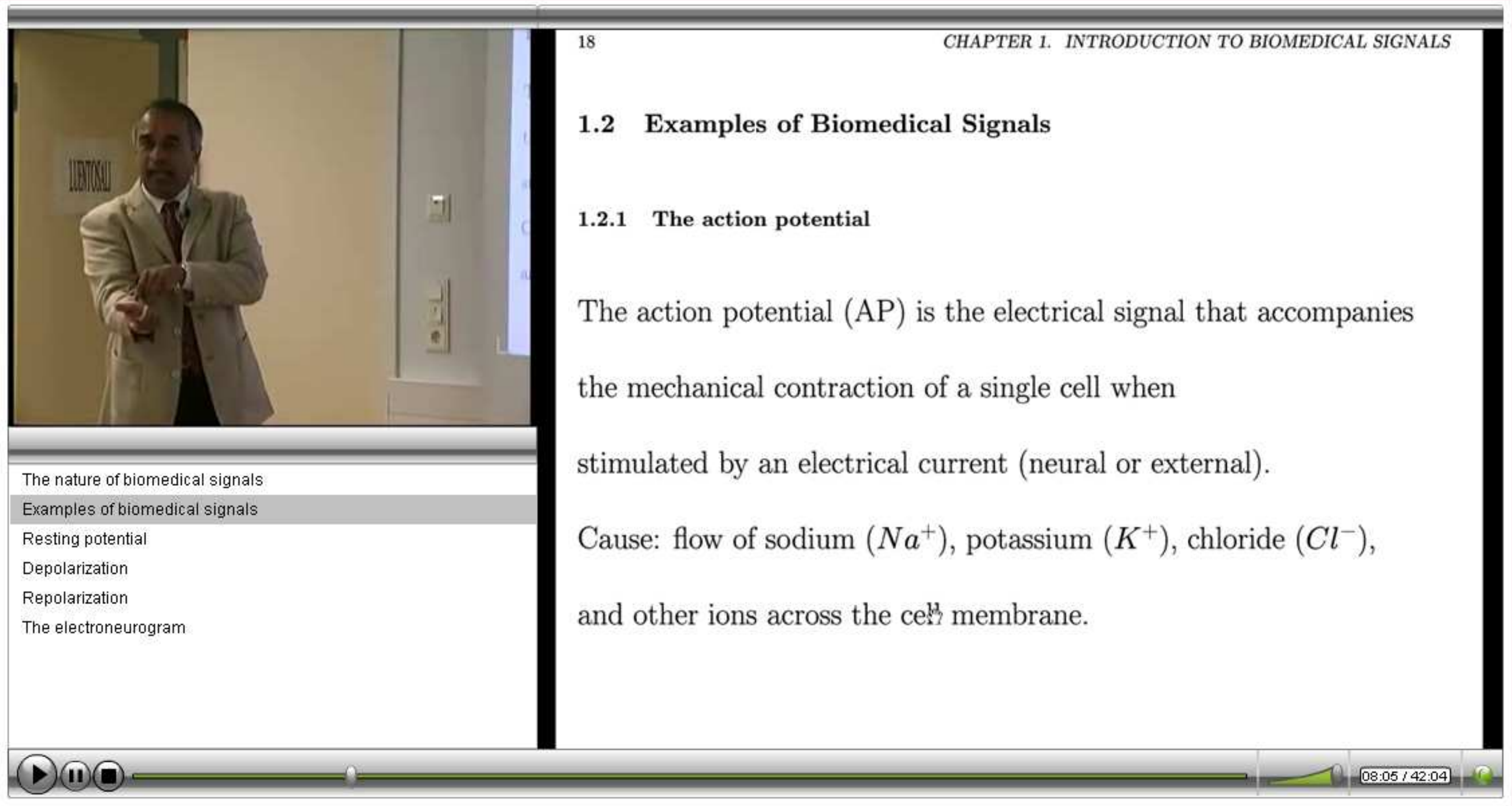


Illustration of various stages of biomedical signal processing and analysis



18 CHAPTER 1. INTRODUCTION TO BIOMEDICAL SIGNALS

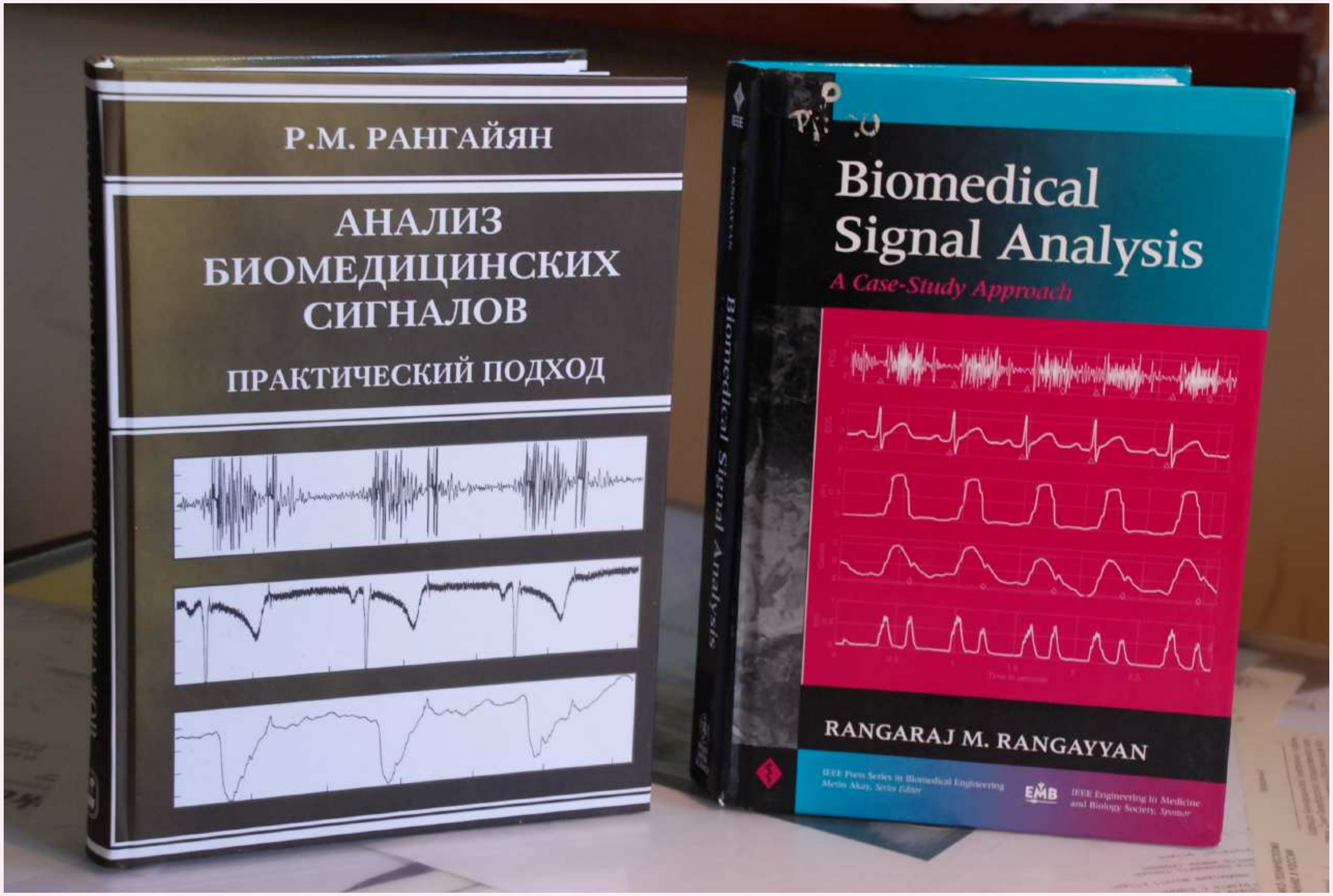
1.2 Examples of Biomedical Signals

1.2.1 The action potential

The action potential (AP) is the electrical signal that accompanies the mechanical contraction of a single cell when stimulated by an electrical current (neural or external).

Cause: flow of sodium (Na^+), potassium (K^+), chloride (Cl^-), and other ions across the cell membrane.

Video of course given at
Ragnar Granit Institute of Biomedical Engineering
Tampere University of Technology, Tampere, Finland
www.evicab.eu


Р.М. РАНГАЙЯН
АНАЛИЗ БИОМЕДИЦИНСКИХ СИГНАЛОВ
ПРАКТИЧЕСКИЙ ПОДХОД
Biomedical Signal Analysis
Biomedical Signal Analysis
A Case-Study Approach
RANGARAJ M. RANGAYYAN
IEEE Press Series in Biomedical Engineering
Metin Akay, Series Editor
EMB
IEEE Engineering in Medicine and Biology Society, Sponsor

Russian translation of 1st Edition by A. Kalinichenko, 2007
Physmathlit, Moscow, Russia

3

Filtering for Removal of Artifacts

Biomedical signals are weak signals

in an environment that is teeming

with many other signals of various origins.

Any signal other than that of interest is

interference, artifact, or *noise.*

Sources of noise: physiological, instrumentation,

or the environment of the experiment.

3.1 Problem Statement

Noise is omnipresent!

Analyze the various types of artifacts
that corrupt biomedical signals
and explore filtering techniques to remove them
without degrading the signal of interest.

3.2 Random, structured, and physiological noise

3.2.1 Random noise

Deterministic signal: value at a given instant of time may be computed using a mathematical function of time,

or predicted from a few past values of the signal.

A signal that does not meet this condition:

nondeterministic or random signal.

Test for randomness:

Based upon the number of peaks or troughs in the signal: collectively known as *turning points*.

Test for a turning point: sign of the first-order difference or derivative at the current sample not equal to that at the preceding sample.

Given a signal of N samples, signal is random if the number of turning points $> \frac{2}{3}(N-2)$.

Nonstationary signal: conduct the test using a running window of N samples.

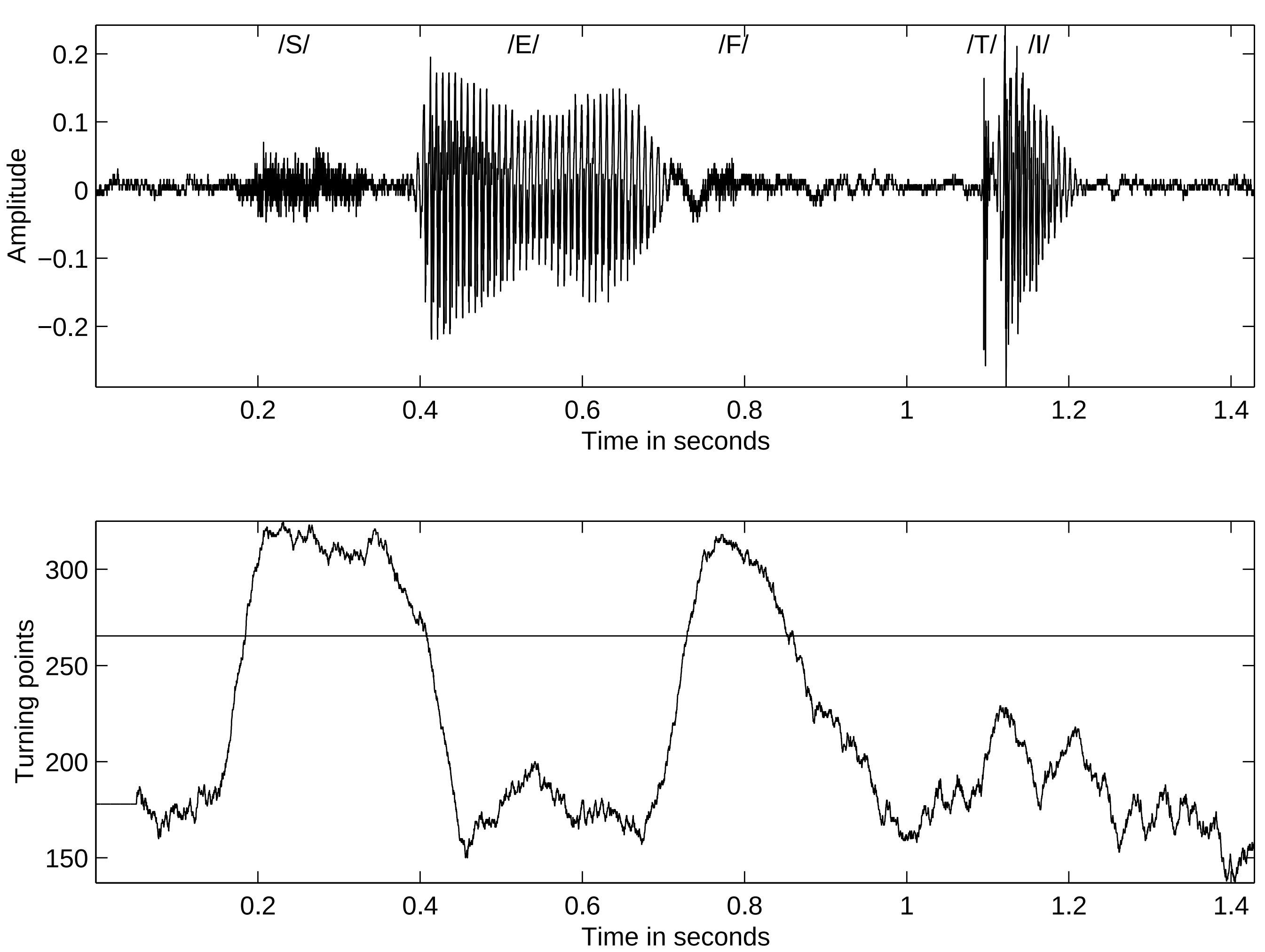


Figure 3.1: Top: Speech signal of the word "safety" uttered by a male speaker. Bottom: Count of turning points in a moving window of 50 ms (400 samples with $f_s = 8\ kHz$). The threshold for randomness for $N = 400$ is 265.

Interference that arises from thermal noise

in electronic devices is a random process.

A random process is characterized by the

probability density function (PDF)

representing the probabilities of occurrence

of all possible values of a random variable.

Statistical analysis of random processes:

Random process η characterized by PDF $p_\eta(\eta)$.

Mean μ_η = first-order moment of the PDF:

$$\mu_\eta = E[\eta] = \int_{-\infty}^{\infty} \eta \, p_\eta(\eta) \, d\eta, \tag{3.1}$$

where $E[\ ]$ represents the *statistical expectation operator*.

Common to assume mean of a random noise process = 0.

Mean-squared (MS) value = second-order moment:

$$E[\eta^2] = \int_{-\infty}^{\infty} \eta^2 \, p_\eta(\eta) \, d\eta. \qquad (3.2)$$

Variance σ_η^2 = second central moment:

$$\sigma_\eta^2 = E[(\eta - \mu_\eta)^2] = \int_{-\infty}^{\infty} (\eta - \mu_\eta)^2 \, p_\eta(\eta) \, d\eta. \qquad (3.3)$$

Square root of variance = standard deviation SD = σ_η.

$\sigma_\eta^2 = E[\eta^2] - \mu_\eta^2$.

If $\mu_\eta = 0$, then $\sigma_\eta^2 = E[\eta^2]$, variance = MS.

Coefficient of variation $CV = \sigma/\mu$.

As $\mu \rightarrow 0$, $CV \rightarrow \infty$.

CV is applicable in the analysis of nonnegative quantities.

Skewness: normalized third central moment

$$S_\eta = \frac{1}{\sigma_\eta^3} \int_{-\infty}^{\infty} (\eta - \mu_\eta)^3 \, p_\eta(\eta) \, d\eta. \tag{3.4}$$

Skewness characterizes the lack of symmetry of a PDF.

Symmetric PDFs such as the Gaussian: skewness = zero.

Higher probabilities of occurrence of larger values of the associated variable: negative skewness.

Higher probabilities of occurrence of smaller values of the associated variable: positive skewness.

Kurtosis: normalized fourth central moment

$$K_\eta = \frac{1}{\sigma_\eta^4} \int_{-\infty}^{\infty} (\eta - \mu_\eta)^4 \, p_\eta(\eta) \, d\eta. \tag{3.5}$$

Kurtosis characterizes the presence of a long tail in the PDF.

Gaussian PDF: kurtosis $= 3$.

Kurtosis excess: $K' = K - 3$ = difference in the kurtosis of a PDF with respect to that of a Gaussian.

A positive K' indicates a PDF with a strong peak near the mean that declines with a heavy tail.

A PDF that is nearly flat has a negative K' value.

The entropy of a random process is a statistical measure of the information conveyed by the process.

It is also a measure of the extent of disorder in the process.

$$H_\eta = - \int_{-\infty}^{\infty} p_\eta(\eta)\ \log_2[p_\eta(\eta)]\, d\eta. \tag{3.6}$$

The unit of entropy, defined as above, is $bits$ (b).

The entropy of a process is at its maximum when all associated values or events occur with equal probability: uniform PDF.

The definitions of the moments of a random process given above refer to a continuous variable with a known PDF.

In practice, when we have only a certain number, N, of the values of the process observed as $\eta(n)$, $n = 0, 1, 2, \ldots, N-1$, and no knowledge of its PDF, we estimate the statistical measures from the samples as

$$\mu_\eta = \frac{1}{N} \sum_{n=0}^{N-1} \eta(n) \,, \tag{3.7}$$

$$MS_\eta = \frac{1}{N} \sum_{n=0}^{N-1} [\eta(n)]^2 \,, \tag{3.8}$$

$$RMS_\eta = \sqrt{\frac{1}{N} \sum_{n=0}^{N-1} [\eta(n)]^2} \, , \tag{3.9}$$

$$\sigma_\eta = \sqrt{\frac{1}{N} \sum_{n=0}^{N-1} [\eta(n) - \mu_\eta]^2} \, . \tag{3.10}$$

Some authors use the notation (nT),
$T = 1/f_s =$ sampling interval,
$f_s =$ sampling frequency,
to denote the index of a sampled signal;
we shall use just (n), the sample number.

For a process with a finite number L of discrete (quantized) values η_l, $l = 0, 1, 2, \ldots, L-1$, entropy

$$H_\eta = -\sum_{l=0}^{L-1} p_\eta(\eta_l)\, \log_2[p_\eta(\eta_l)]\,, \tag{3.11}$$

where $p_\eta(\eta_l)$ is the probability of occurrence of the l^{th} quantized value of η.

Each probability value may be estimated from a large number of observations of the values of the process.

This is known as Shannon entropy.

When the values of a random process η form a

time series or a function of time,

we have a random signal or a stochastic process $\eta(t)$.

Then, the statistical measures have physical meanings:

mean = DC component;

MS = average power;

square root of MS = root mean-squared or RMS value

= average noise magnitude.

Signal-to-noise ratio:

SNR in $dB =$

$$20\ \log_{10} \frac{peak - to - peak\ amplitude\ range\ of\ signal}{RMS\ value\ of\ noise}$$

or

$$10\ \log_{10} \frac{average\ power\ of\ signal}{average\ power\ of\ noise}.$$

Any biomedical signal of interest $x(t)$ may also,

for the sake of generality, be considered to be

a realization of a random process x.

When a signal $x(t)$ is observed with random noise,

the measured signal $y(t)$ may be treated as

a realization of another random process y.

In most cases the noise is additive:

$$y(t) = x(t) + \eta(t). \tag{3.12}$$

Each of the random processes x and y is characterized

by its own PDF $p_x(x)$ and $p_y(y)$, respectively.

In most practical applications, the random processes

representing a signal of interest and the noise

may be assumed to be *statistically independent processes.*

Two random processes x and η are said to be

statistically independent if their joint PDF

$$p_{x,\eta}(x,\eta) = p_x(x)\; p_\eta(\eta).$$

$$E[y] = \mu_y = \mu_x + \mu_\eta. \tag{3.13}$$

If $\mu_\eta = 0$, then $\mu_y = \mu_x$.

$$E[(y - \mu_y)^2] = \sigma_y^2 = \sigma_x^2 + \sigma_\eta^2. \tag{3.14}$$

Ensemble averages:

When the PDFs of the random processes of concern are not known, approximate the statistical expectation operation by averages of a collection or *ensemble* of sample observations of the random process: *ensemble averages*.

Suppose we have M observations of the random process

x as functions of time: $x_1(t), x_2(t), \ldots, x_M(t)$.

Estimate of the mean of the process at a particular

instant of time t_1:

$$\mu_x(t_1) = \lim_{M\to\infty} \frac{1}{M} \sum_{k=1}^{M} x_k(t_1). \tag{3.15}$$

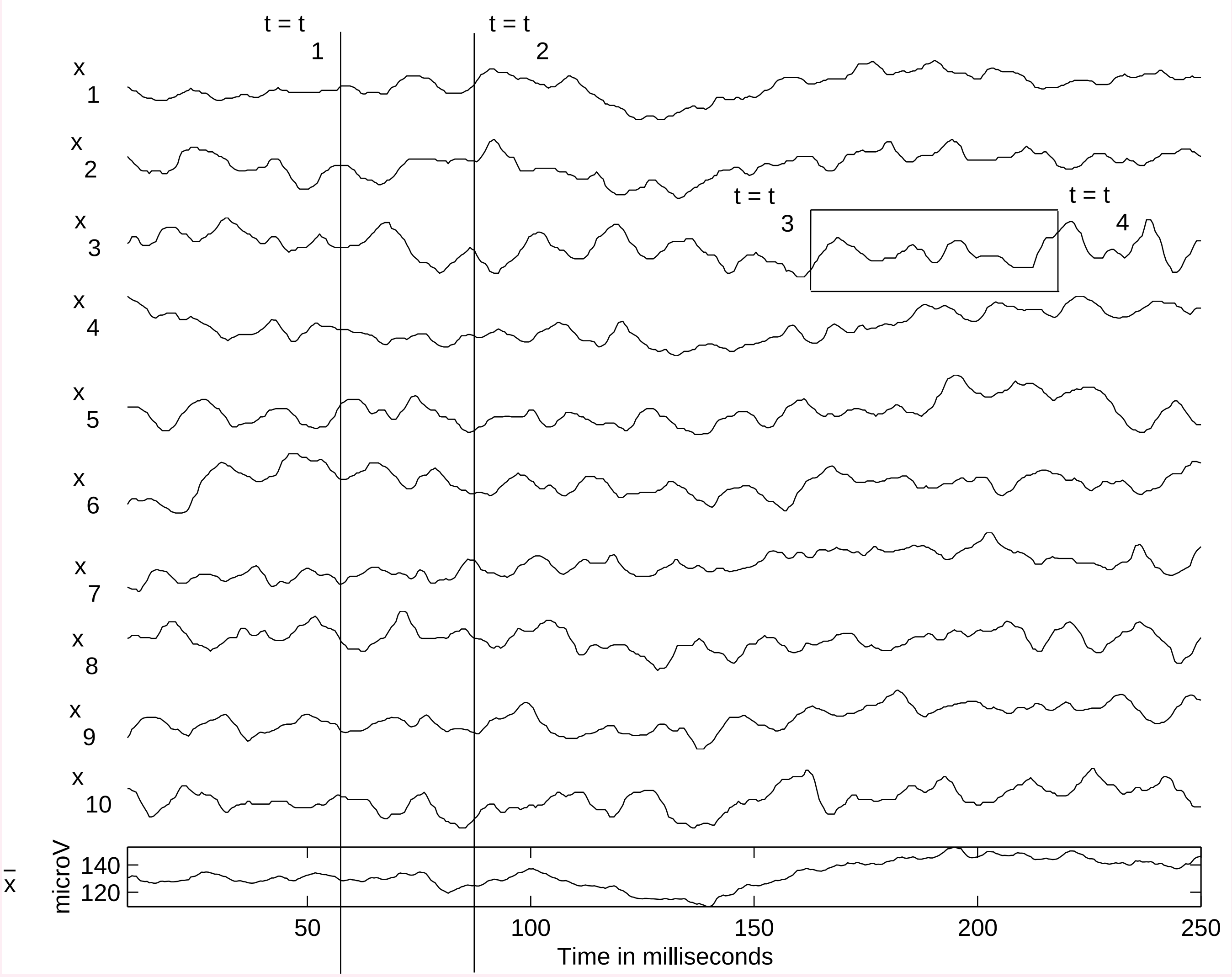


Figure 3.2: Ten sample acquisitions (x_1 to x_{10}) of individual flash visual ERPs from the occipital midline (oz) position of a normal adult male (the author of this book!). The ear lobes were used to form the reference lead (a1a2), and the left forehead was used as the reference (see Figure 1.37). The signals may be treated as ten realizations of a random process in the form of time series or signals. The vertical lines at $t = t_1$ and $t = t_2 = t_1 + \tau$ represent the ensemble averaging process at two different instants of time. The last plot (framed) gives the ensemble average or prototype $\bar{x}(t)$ of the ten individual signals. The horizontal box superimposed on the third trace represents the process of computing temporal statistics over the duration $t = t_3$ to $t = t_4$ of the sample ERP $x_3(t)$. See also Figure 3.41. Data courtesy of L. Alfaro and H. Darwish, Alberta Children's Hospital, Calgary.

Autocorrelation function (ACF) $\phi_{xx}(t_1, t_1 + \tau)$:

$$\phi_{xx}(t_1, t_1 + \tau) = E[x(t_1)x(t_1 + \tau)] \qquad (3.16)$$

$$= \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} x(t_1)\, x(t_1 + \tau)\, p_{x_1,x_2}(x_1, x_2)\, dx_1\, dx_2,$$

x_1 and x_2: random variables corresponding to

$x(t_1)$ and $x(t_1 + \tau)$;

$p_{x_1,x_2}(x_1, x_2)$: joint PDF of the two processes.

The ACF may be estimated as

$$\phi_{xx}(t_1, t_1 + \tau) = \lim_{M \to \infty} \frac{1}{M} \sum_{k=1}^{M} x_k(t_1)\, x_k(t_1 + \tau), \quad (3.17)$$

where τ is the delay parameter.

If the signals are complex: take the conjugate of one.

The two vertical lines at $t = t_1$ and $t = t_2 = t_1 + \tau$

in Figure 3.2: ensemble averaging process

to compute $\phi_{xx}(t_1, t_2)$.

ACF indicates how the values of a signal

at a particular instant of time

are statistically related to values at another instant of time.

Random processes as functions of time:

stochastic processes.

Compute ensemble averages at every point of time.

Averaged function of time $\bar{x}(t)$:

$$\bar{x}(t) = \mu_x(t) = \frac{1}{M} \sum_{k=1}^{M} x_k(t) \tag{3.18}$$

for all time t.

Signal $\bar{x}(t)$ may be used to represent

the random process x as a *prototype.*

Time averages:

Sample observation of a random process $x_k(t)$:

compute *time averages* or *temporal statistics*

by integrating over time:

$$\mu_x(k) = \lim_{T\to\infty} \frac{1}{T} \int_{-T/2}^{T/2} x_k(t)\, dt. \tag{3.19}$$

Integral replaced by summation for sampled signals.

Time-averaged ACF $\phi_{xx}(\tau, k)$:

$$\phi_{xx}(\tau, k) = \lim_{T\to\infty} \frac{1}{T} \int_{-T/2}^{T/2} x_k(t)\, x_k(t+\tau)\, dt. \qquad (3.20)$$

The horizontal box superimposed on the third trace in Figure 3.2: computing temporal statistics over the duration $t = t_3$ to $t = t_4$ of the sample ERP $x_3(t)$ selected from the ensemble of ERPs.

Thus, random noise may be characterized in terms of

ensemble and/or *temporal* statistics.

Mean: not important; assumed to be zero, or subtracted out.

ACF: plays an important role in the

characterization of random processes.

Fourier transform of ACF = power spectral density (PSD).

Covariance and cross-correlation:

When two random processes x and y need to be compared:

$$\begin{aligned} C_{xy} &= E[(x-\mu_x)(y-\mu_y)] \qquad (3.21) \\ &= \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} (x-\mu_x)(y-\mu_y)\, p_{x,y}(x,y)\, dx\, dy. \end{aligned}$$

$p_{x,y}(x,y)$: joint PDF of the two processes.

Covariance normalized to get the correlation coefficient:

$$\rho_{xy} = \frac{C_{xy}}{\sigma_x \, \sigma_y}, \tag{3.22}$$

with $-1 \leq \rho_{xy} \leq +1$.

High covariance: the two processes have similar statistical variability or behavior.

The processes x and y are uncorrelated if $\rho_{xy} = 0$.

Two statistically independent processes are also uncorrelated; the converse of this property is, in general, not true.

Cross-correlation function (CCF) of random processes

x and y that are functions of time:

$$\begin{aligned}\theta_{xy}(t_1, t_1 + \tau) &= E[x(t_1) y(t_1 + \tau)] \\ &= \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} x(t_1)\, y(t_1 + \tau)\, p_{x,y}(x, y)\, dx\, dy.\end{aligned} \tag{3.23}$$

CCF useful in analyzing the nature of variability,

spectral bandwidth of signals, and

detection of events by template matching.

3.2.2 Structured noise

Power-line interference at $50\ Hz$ or $60\ Hz$:

typical waveform of the interference known in advance.

Phase of the interfering waveform not known.

Interfering waveform may not be an exact sinusoid:

presence of harmonics of the fundamental $50\ Hz$ or $60\ Hz$.

3.2.3 Physiological interference

Human body: complex conglomeration of several systems.

Several physiological processes active at a given time,

each one producing many signals of different types.

Patient or subject not able to exercise control on all

physiological processes and systems.

Appearance of signals from systems or processes other than those of interest: *physiological interference.*

- EMG related to coughing, breathing, squirming in ECG.
- EGG interfering with precordial ECG.
- Maternal ECG getting added to fetal ECG.
- ECG interfering with the EEG.
- Ongoing EEG in ERPs and SEPs.
- Breath, lung, bowel sounds in heart sounds (PCG).
- Heart sounds in breath or lung sounds.
- Muscle sound (VMG) in joint sounds (VAG).

Physiological interference not characterized by any

specific waveform or spectral content —

typically dynamic and nonstationary:

linear bandpass filters will not be applicable.

Need adaptive filters with reference inputs.

3.2.4 Stationary, nonstationary, and cyclostationary processes

A random process is *stationary in the strict sense* or *strongly stationary* if its statistics are not affected by a shift in the origin of time.

In practice, only first-order and second-order averages used.

A random process is *weakly stationary*

or *stationary in the wide sense*

if its mean is a constant and its

ACF depends only upon the difference (or shift) in time.

Then, we have

$\mu_x(t_1) = \mu_x$ and

$\phi_{xx}(t_1, t_1 + \tau) = \phi_{xx}(\tau)$.

ACF function of the delay parameter τ only;

the PSD of the process does not vary with time.

A stationary process is *ergodic*

if the temporal statistics are

independent of the sample observed;

same result obtained for any sample observation $x_k(t)$.

Time averages are then independent of k:

$\mu_x(k) = \mu_x$ and $\phi_{xx}(\tau, k) = \phi_{xx}(\tau)$.

All ensemble statistics may be replaced by

temporal statistics when analyzing ergodic processes.

Ergodic processes:

important type of stationary random processes

because their statistics may be computed from

a single observation as a function of time.

Signals or processes that do not meet the conditions

described above: *nonstationary processes.*

A nonstationary process possesses

statistics that vary with time.

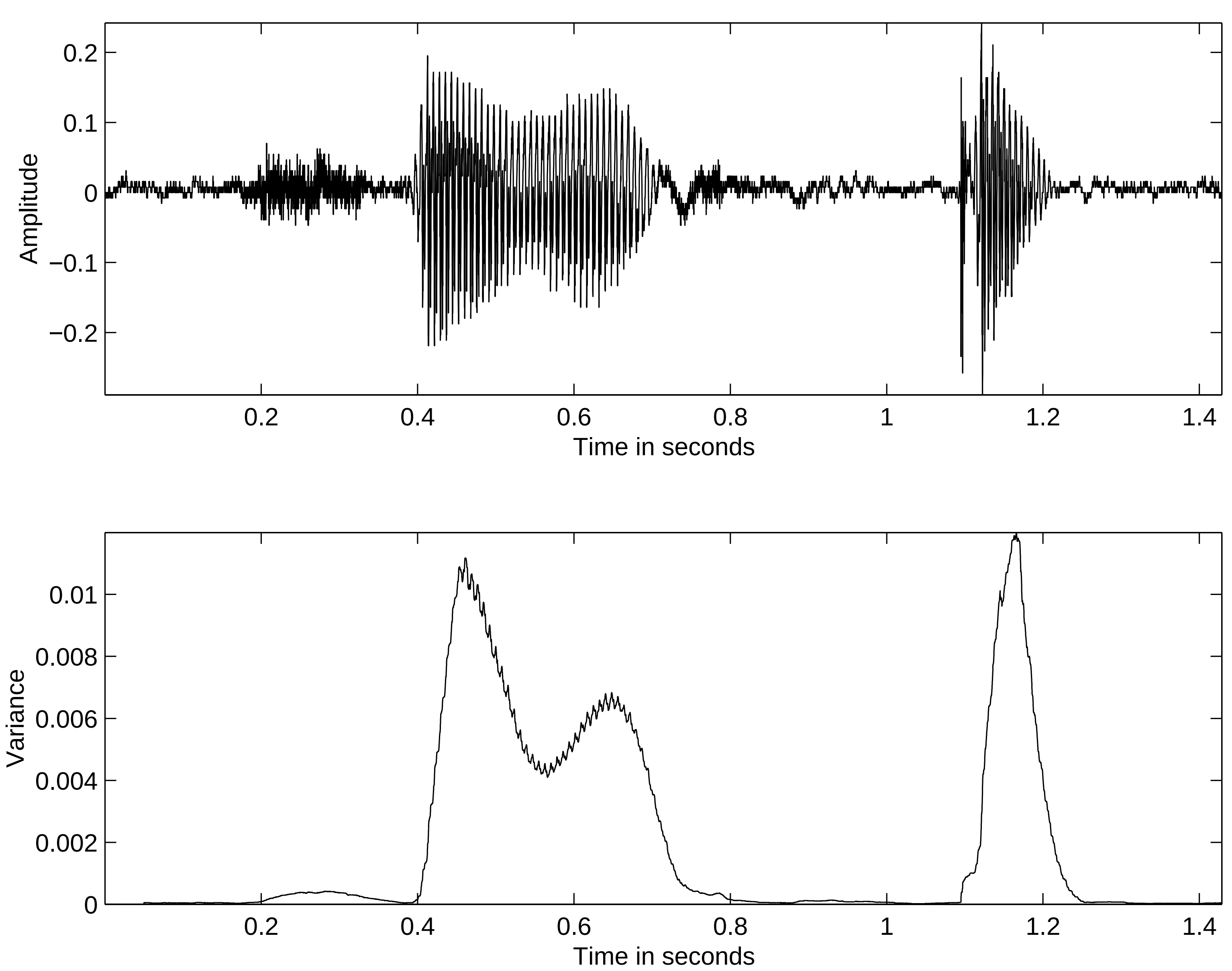


Figure 3.3: Top: Speech signal of the word “safety” uttered by a male speaker. Bottom: Variance computed in a moving window of $50\ ms$ (400 samples with $f_s = 8\ kHz$).

Most biomedical systems are dynamic:

produce nonstationary signals.

Examples: EMG, EEG, VMG, PCG, VAG, speech.

However, a physical or physiological system has

limitations in the rate of change of its characteristics.

This facilitates breaking the signal into

segments of short duration

over which the statistics of interest are not varying,

or may be assumed to remain the same.

Quasistationary process: short-time analysis.

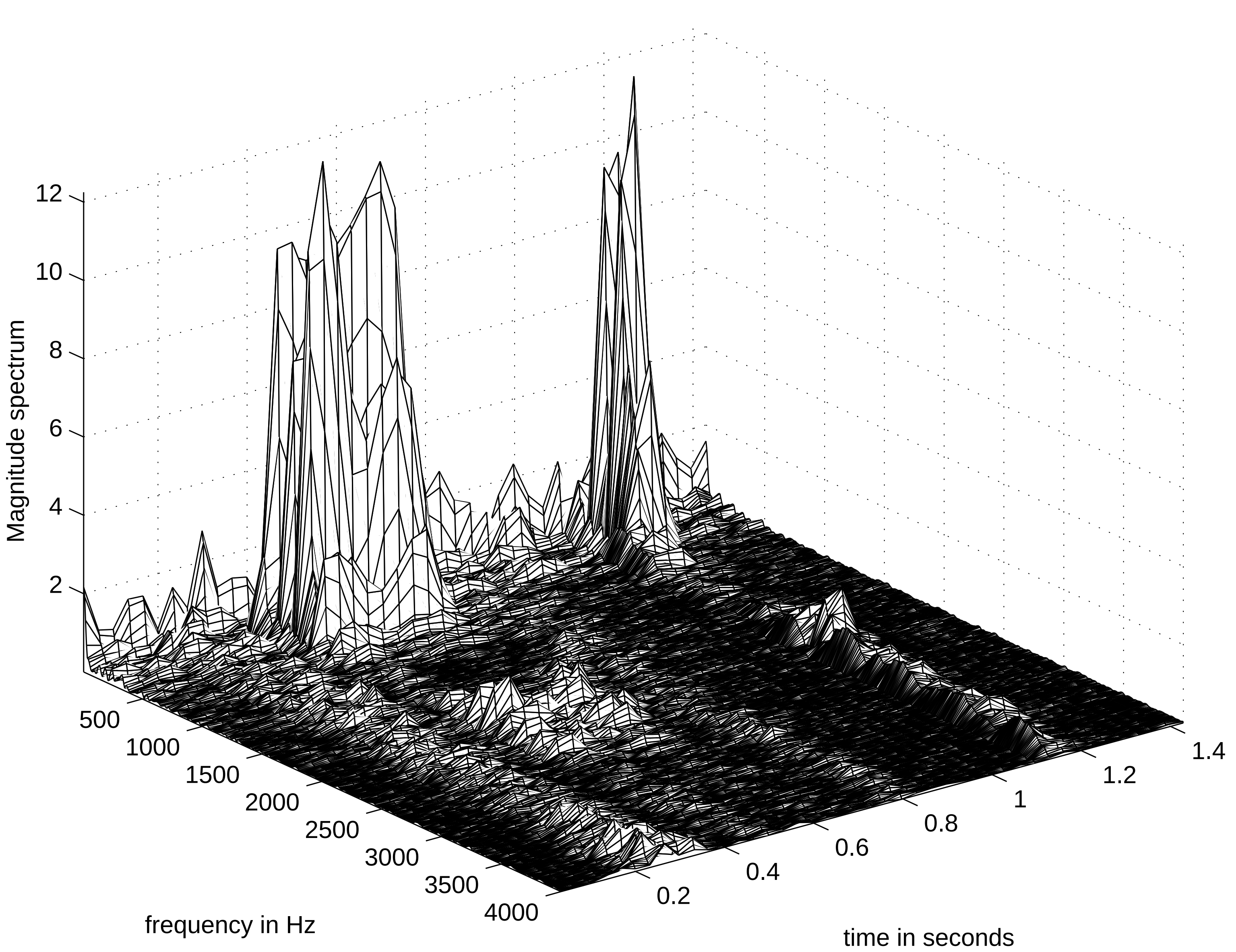


Figure 3.4: Spectrogram of the speech signal of the word “safety” uttered by a male speaker. (The signal is also illustrated in Figures 1.50, 3.1, and 3.3.) Each curve represents the magnitude spectrum of the signal in a moving window of duration 64 ms (512 samples with $f_s = 8\ kHz$), with the window advance interval being 32 ms. The spectrogram is plotted on a linear scale to display better the major differences between the voiced and unvoiced sounds.

Certain systems, such as the cardiac system,

normally perform rhythmic operations.

ECG, PCG, carotid pulse: almost periodic —

cyclostationary signals.

Statistics of PCG vary within the duration of a cardiac cycle,

especially when murmurs are present,

but repeat themselves at regular intervals.

Cyclic repetition facilitates ensemble averaging,

using epochs or events extracted from

an observation of the signal over many cycles, which is

strictly speaking, a single function of time.

3.3 Illustration of the Problem with Case Studies

3.3.1 Noise in event-related potentials

ERP: signal obtained in response to a stimulus.

Response of small amplitude ($\sim 10\ \mu V$),

buried in ambient EEG activity and noise.

A single response may not be recognizable.

Figure 3.2: ten individual flash visual ERP signals.

Recorded at occipital midline (oz) position

against left and right ear lobes combined (a1 – a2).

Left forehead used as the reference.

ERP buried in ongoing EEG and

power-line ($60\ Hz$) interference.

3.3.2 High-frequency noise in the ECG

Figure 3.5: ECG with high-frequency noise.

Noise could be due to instrumentation amplifiers,

the recording system,

pickup of ambient EM signals by cables, etc.

Also power-line interference at $60\ Hz$ and harmonics.

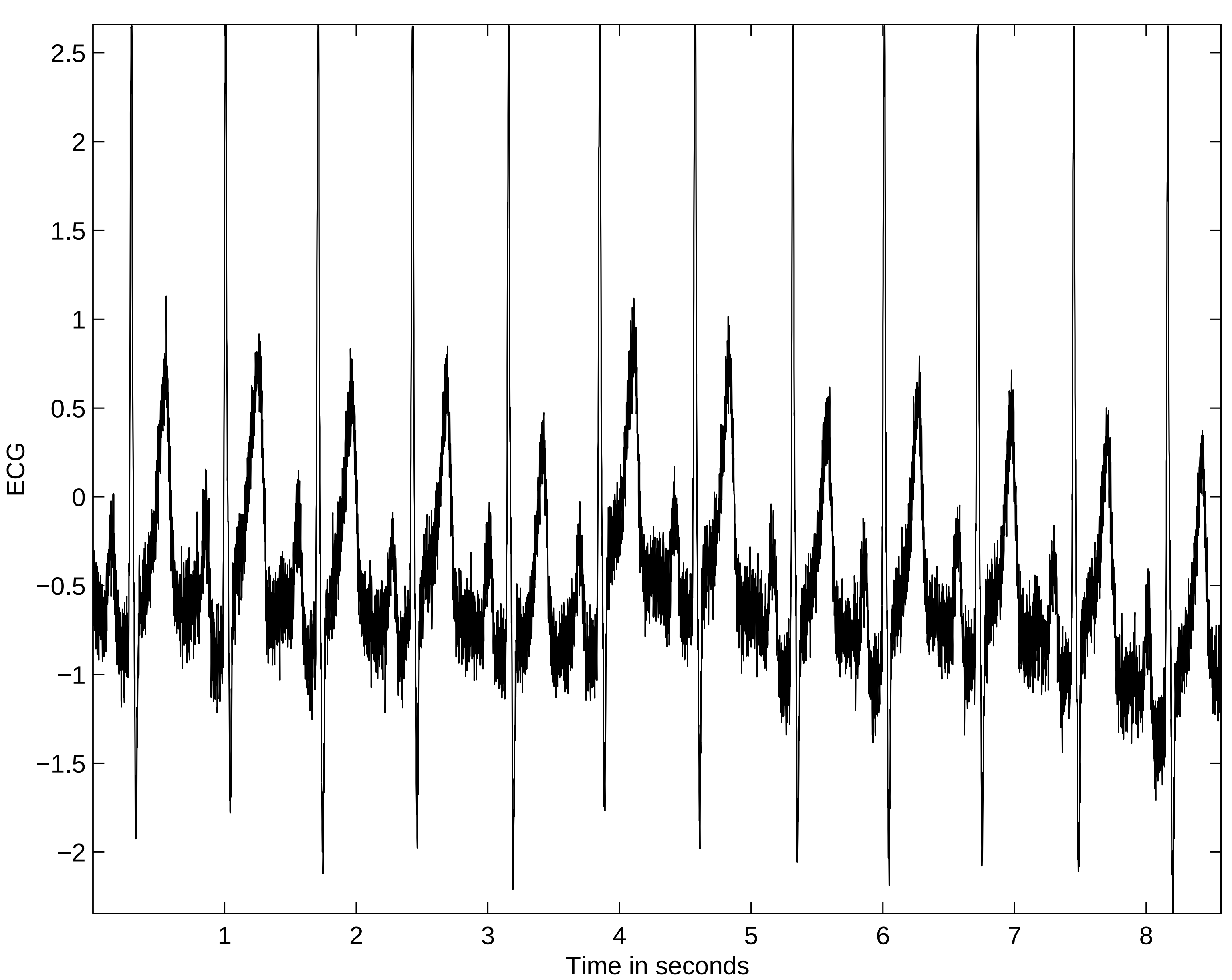


Figure 3.5: ECG signal with high-frequency noise.

3.3.3 Motion artifact in the ECG

Low-frequency artifacts and baseline drift in chest-lead ECG due to coughing or breathing with large movement of the chest, or when an arm or leg is moved in the case of limb-lead ECG.

EGG common source of artifact in chest-lead ECG.

Poor contact, polarization of electrodes: low-frequency artifacts.

Baseline drift: variations in temperature

and bias in the instrumentation and amplifiers.

Figure 3.6: ECG with low-frequency artifact.

Baseline drift: affects analysis of the

isoelectric nature PQ and ST segments.

Large baseline drift: positive or negative peaks in the

ECG clipped by the amplifiers or the ADC.

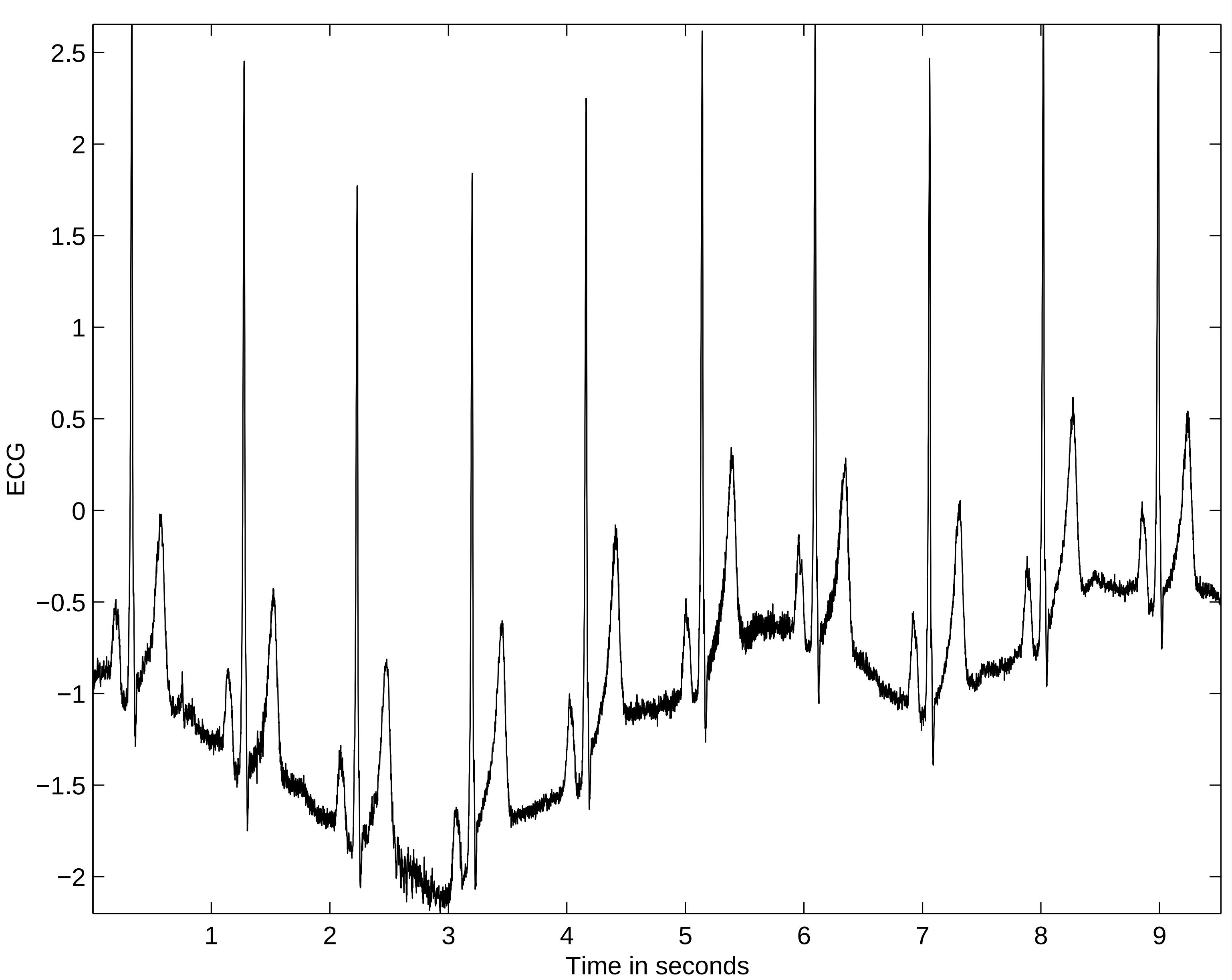


Figure 3.6: ECG signal with low-frequency artifact.

3.3.4 Power-line interference in ECG signals

Most common periodic artifact in biomedical signals:

power-line interference at $50\ Hz$ or $60\ Hz$ + harmonics.

Bandwidth of interest of ECG: $0.05 - 100\ Hz$.

Lowpass filtering not appropriate for removal of

power-line interference:

will smooth and blur QRS, and affect PQ and ST segments.

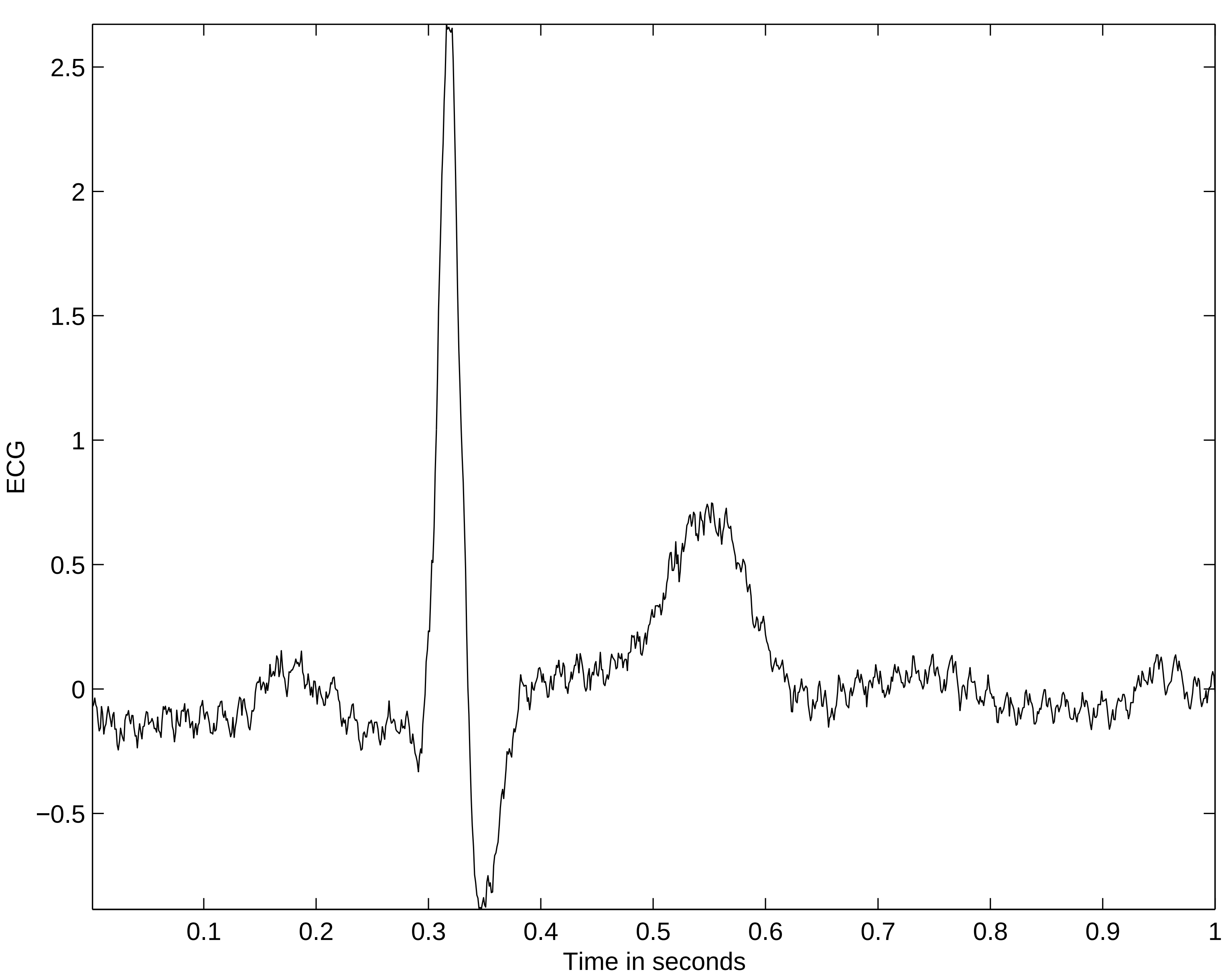


Figure 3.7: ECG signal with power-line (60 Hz) interference.

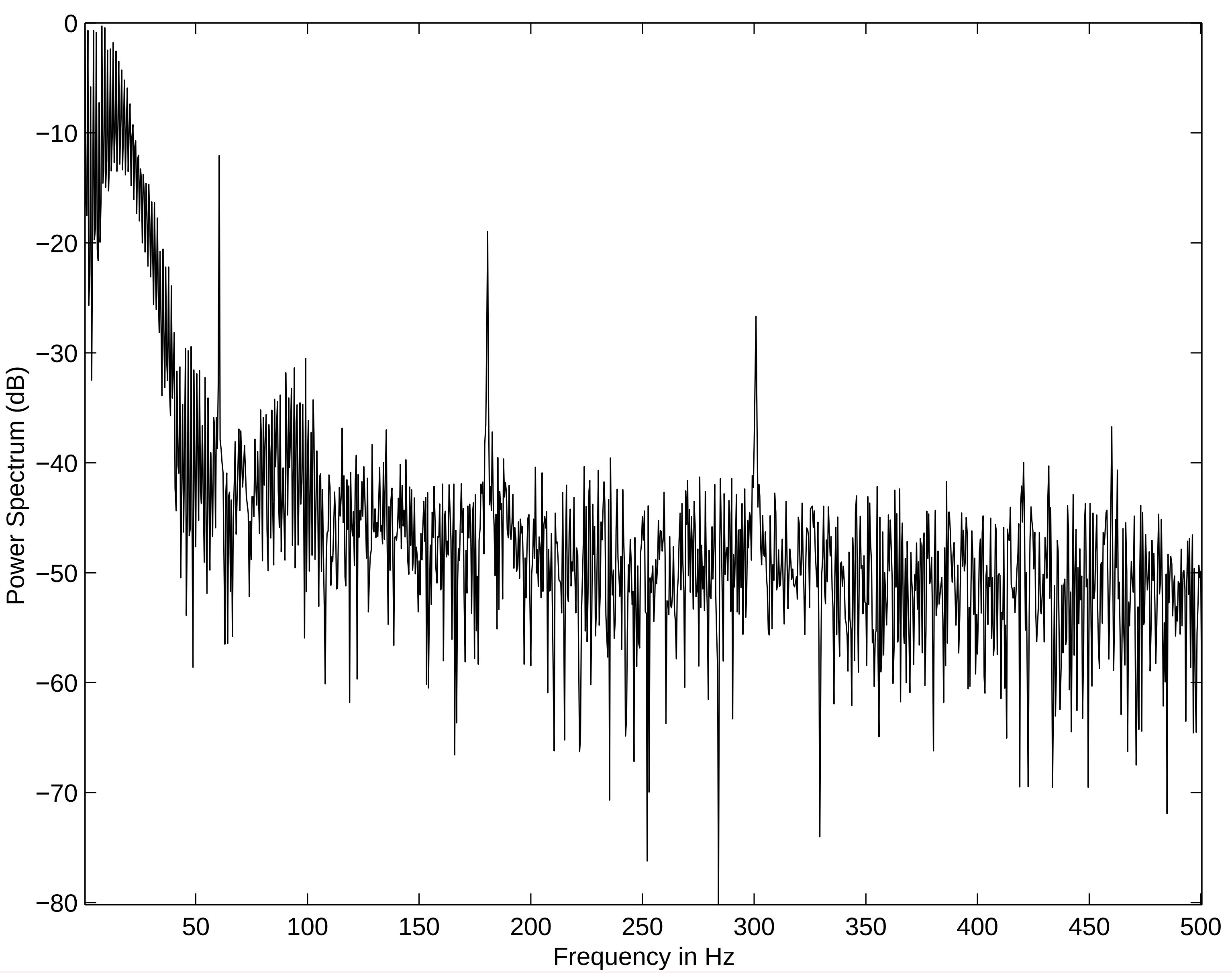


Figure 3.8: Power spectrum of the ECG signal in Figure 3.7 with power-line interference. The spectrum illustrates peaks at the fundamental frequency of 60 Hz as well as the third and fifth harmonics at 180 Hz and 300 Hz, respectively.

3.3.5 Maternal interference in fetal ECG

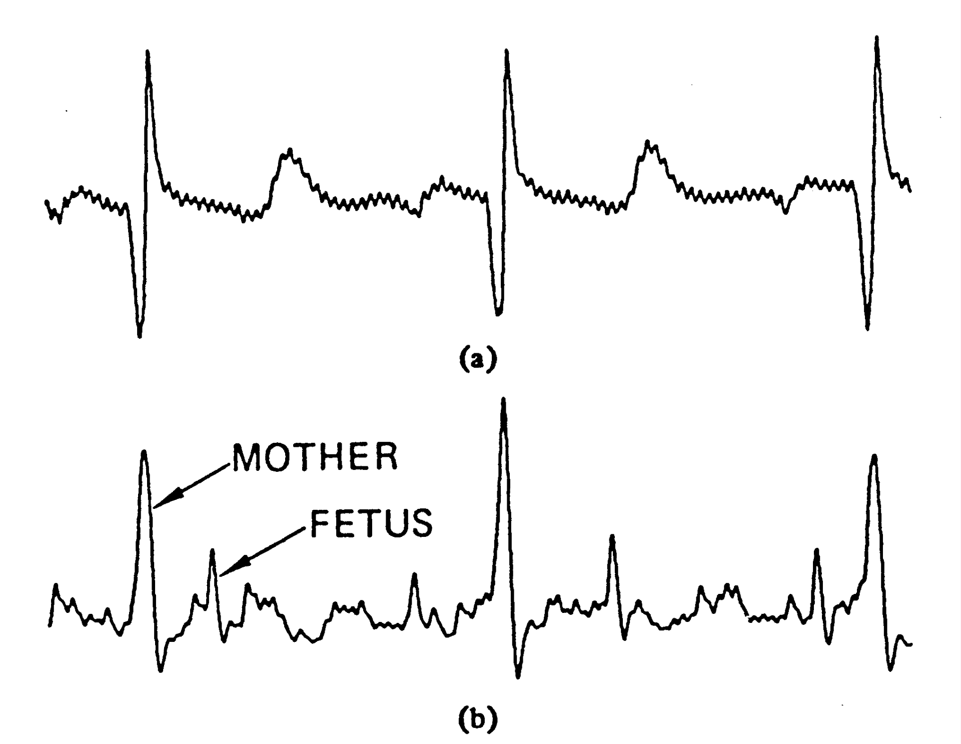


Figure 3.9: ECG signals of a pregnant woman from abdominal and chest leads: (a) chest-lead ECG, and (b) abdominal-lead ECG; the former presents the maternal ECG whereas the latter is a combination of the maternal and fetal ECG signals. (See also Figure 3.101.) Reproduced with permission from B. Widrow, J.R. Glover, Jr., J.M. McCool, J. Kaunitz, C.S. Williams, R.H. Hearn, J.R. Zeidler, E. Dong, Jr., R.C. Goodlin, Adaptive noise cancelling: Principles and applications, *Proceedings of the IEEE*, 63(12):1692–1716, 1975. ©IEEE.

3.3.6 Muscle-contraction interference in VAG signals

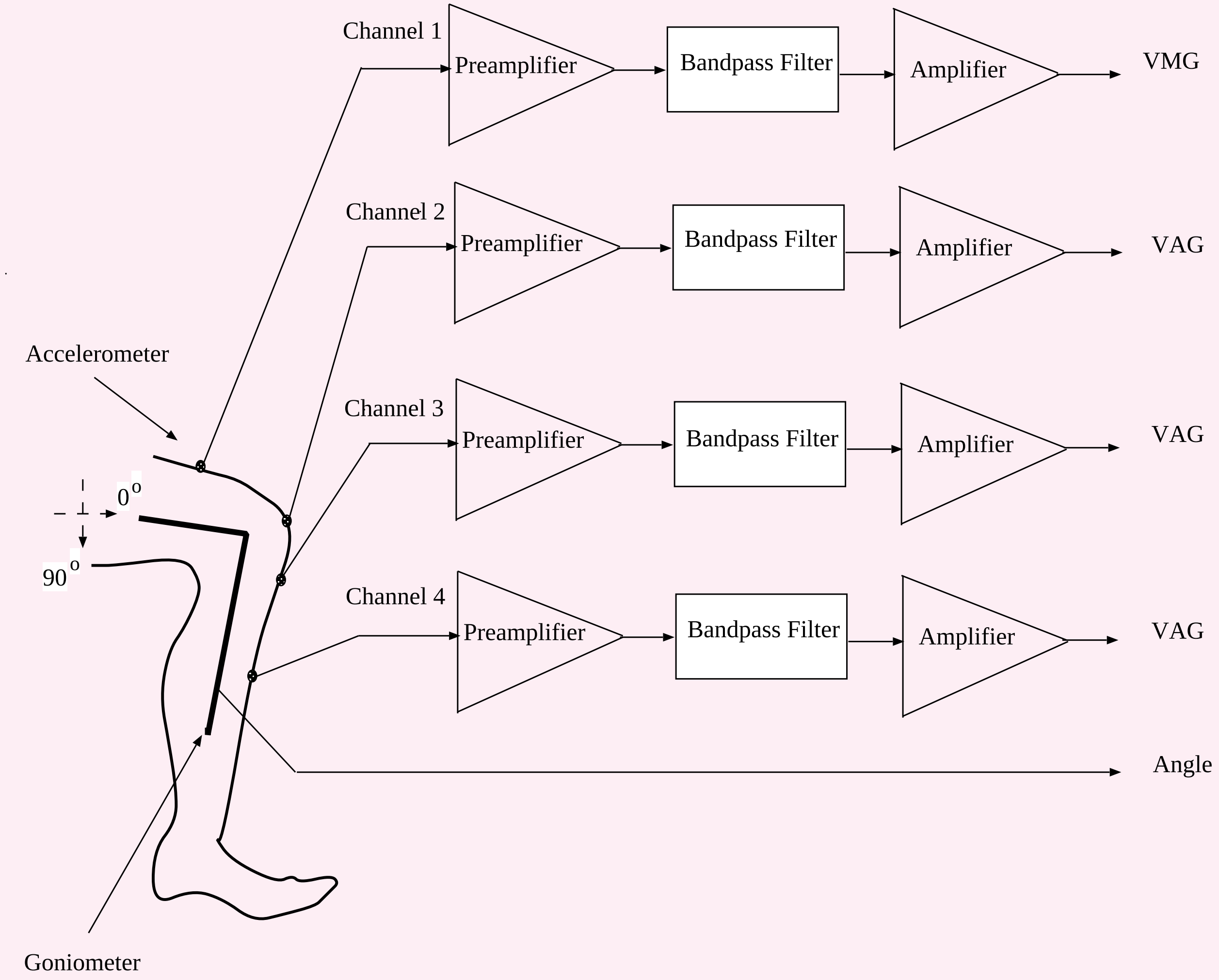
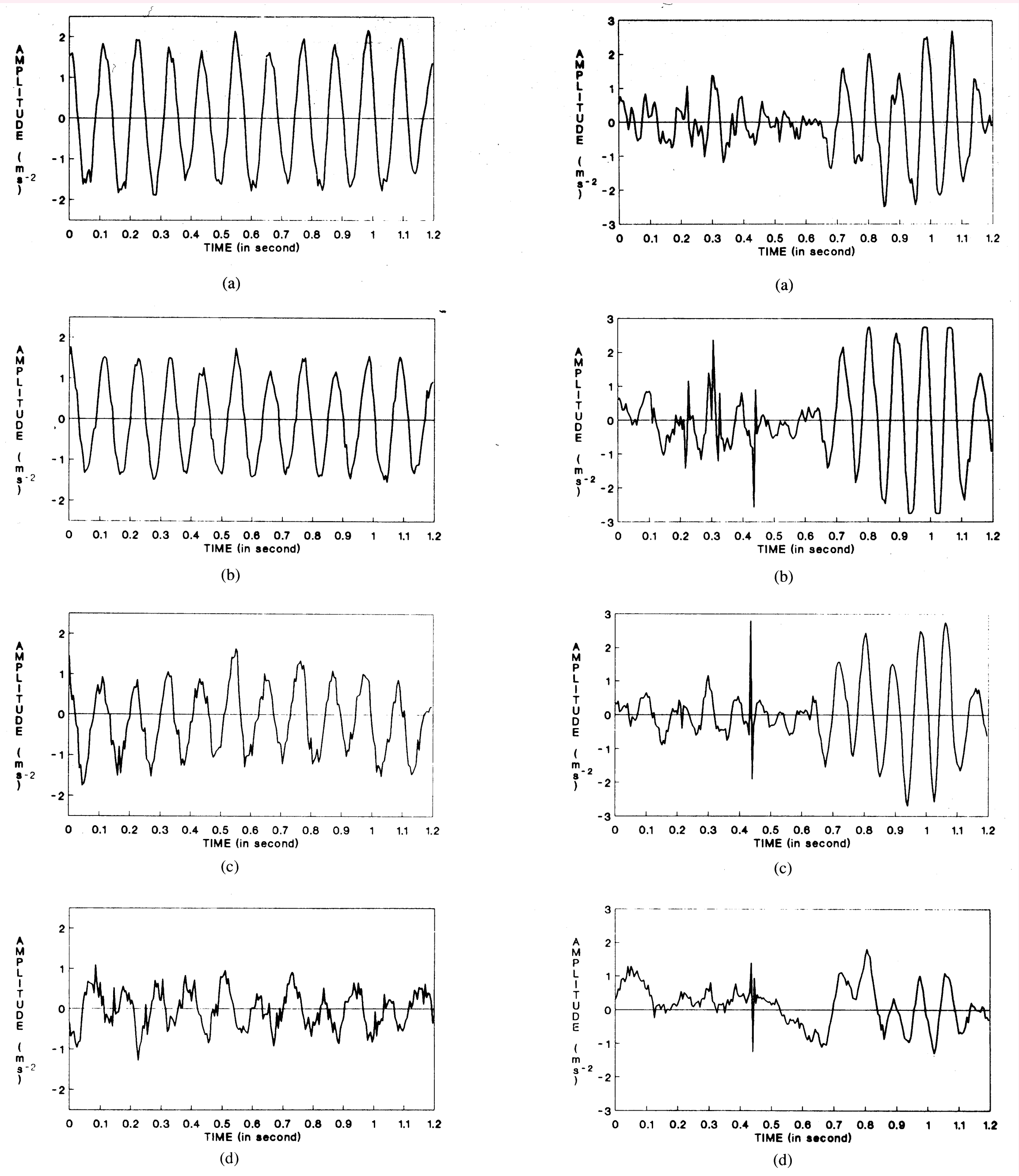


Figure 3.10: Experimental setup to measure VMG and VAG signals at different positions along the leg.


AMPLITUDE (m s-2)
2
1
0
-1
-2
3
-3
0 0.1 0.2 0.3 0.4 0.5 0.6 0.7 0.8 0.9 1 1.1 1.2
TIME (in second)

(a) (a)

(b) (b)

(c) (c)

(d) (d)

Figure 3.11: Left-hand column: VMG signals recorded simultaneously at (top-to-bottom) (a) the distal rectus femoris, (b) mid-patella, (c) tibial tuberosity, and (d) mid-tibial shaft positions during isometric contraction (no leg or knee movement). Right-hand column: Vibration signals recorded simultaneously at the same positions as above during isotonic contraction (swinging movement of the leg). Observe the muscle-contraction interference appearing in the extension parts (second halves) of each of the VAG signals (plots (b) – (d)) in the right-hand column. The recording setup is shown in Figure 3.10. Reproduced with permission from Y.T. Zhang, R.M. Rangayyan, C.B. Frank, and G.D. Bell, Adaptive cancellation of muscle-contraction interference from knee joint vibration signals, *IEEE Transactions on Biomedical Engineering,* 41(2):181–191, 1994. ©IEEE.

3.3.7 Potential solutions to the problem

A practical problem encountered by an investigator

in the field may not precisely match

a specific textbook problem.

However, the knowledge of several techniques

and appreciation of the results of their application

could help in designing innovative and appropriate

solutions to new problems.

3.4 Fundamental Concepts of Filtering

A filter is a signal processing system, algorithm, or method, in hardware or software, used to modify a signal.

A signal may be filtered to remove undesired components, noise, or artifacts, and to enhance desired components.

Filters may be categorized as

- linear or nonlinear,
- stationary or nonstationary,
- fixed (time-invariant) or adaptive (time-variant),
- active or passive, or
- statistical or deterministic.

3.4.1 Linear shift-invariant filters

A fundamental characteristic of a linear time-invariant (LTI) or shift-invariant (LSI) filter is its impulse response:

output of the system when the input is a Dirac delta or impulse function.

In continuous time, the Dirac delta function is defined as

$$\delta(t) = \begin{cases} undefined\ at\ t = 0, \\ 0 \qquad\qquad\qquad otherwise, \end{cases} \tag{3.24}$$

with

$$\int_{t=-\infty}^{\infty} \delta(t)\, dt = 1. \tag{3.25}$$

The delta function may be visualized as the limit of a function whose integral over the full extent of time or an independent variable is maintained equal to unity while its duration or extent is compressed toward zero.

Figure 3.12 demonstrates this definition graphically using a rectangular pulse function.

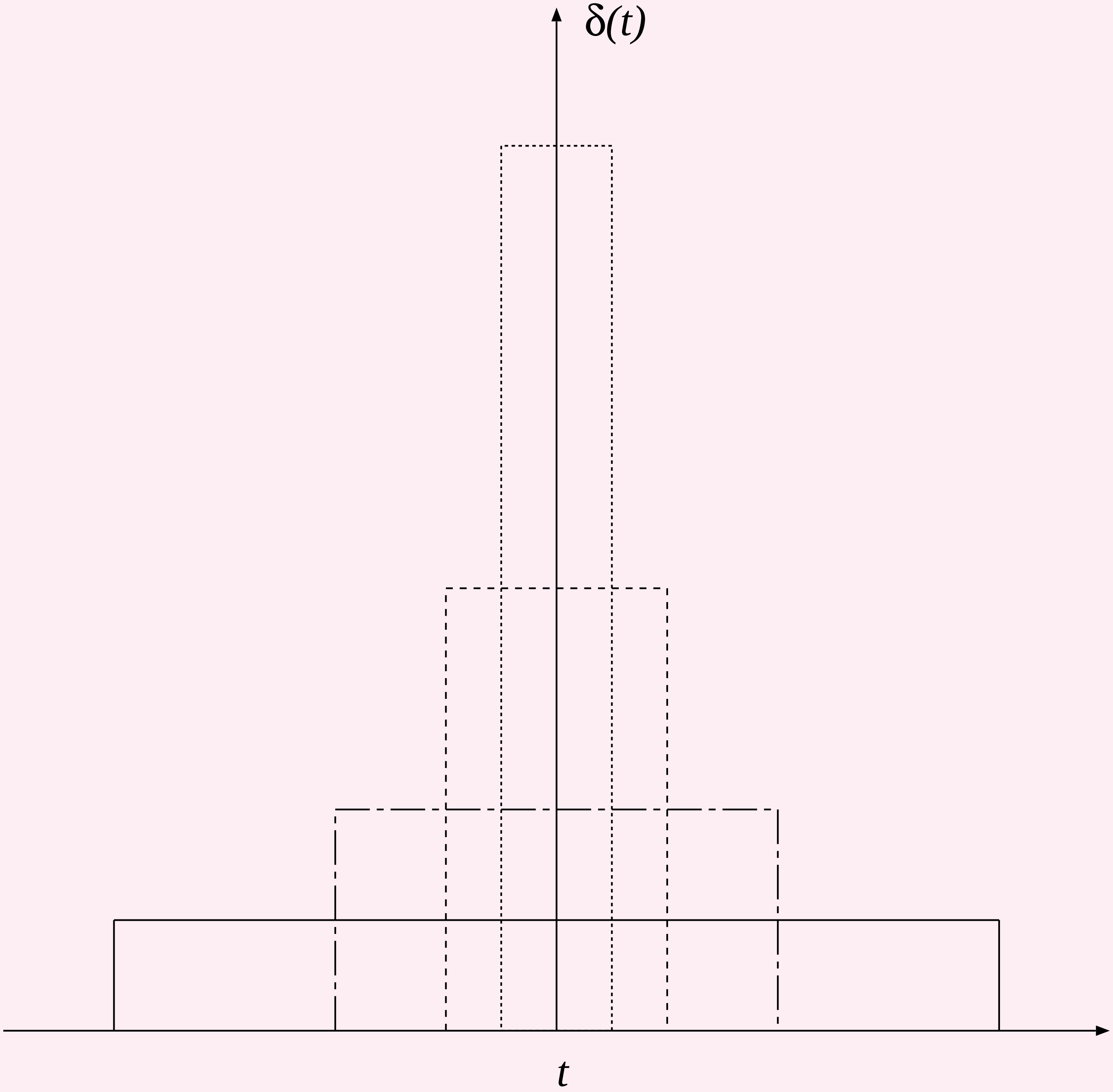


Figure 3.12: Schematic representation of the delta function as a limit of a rectangular pulse function. The time duration of the pulse is reduced while maintaining unit area under the function.

Another example:

$$\delta(t) = \frac{1}{2} \lim_{a \to 0} a|t|^{(a-1)}. \tag{3.26}$$

Figure 3.13 illustrates three plots of the function defined above for $a = 0.8, 0.4,$ and 0.2.

The emergence of the delta function as $a \longrightarrow 0$ is evident.

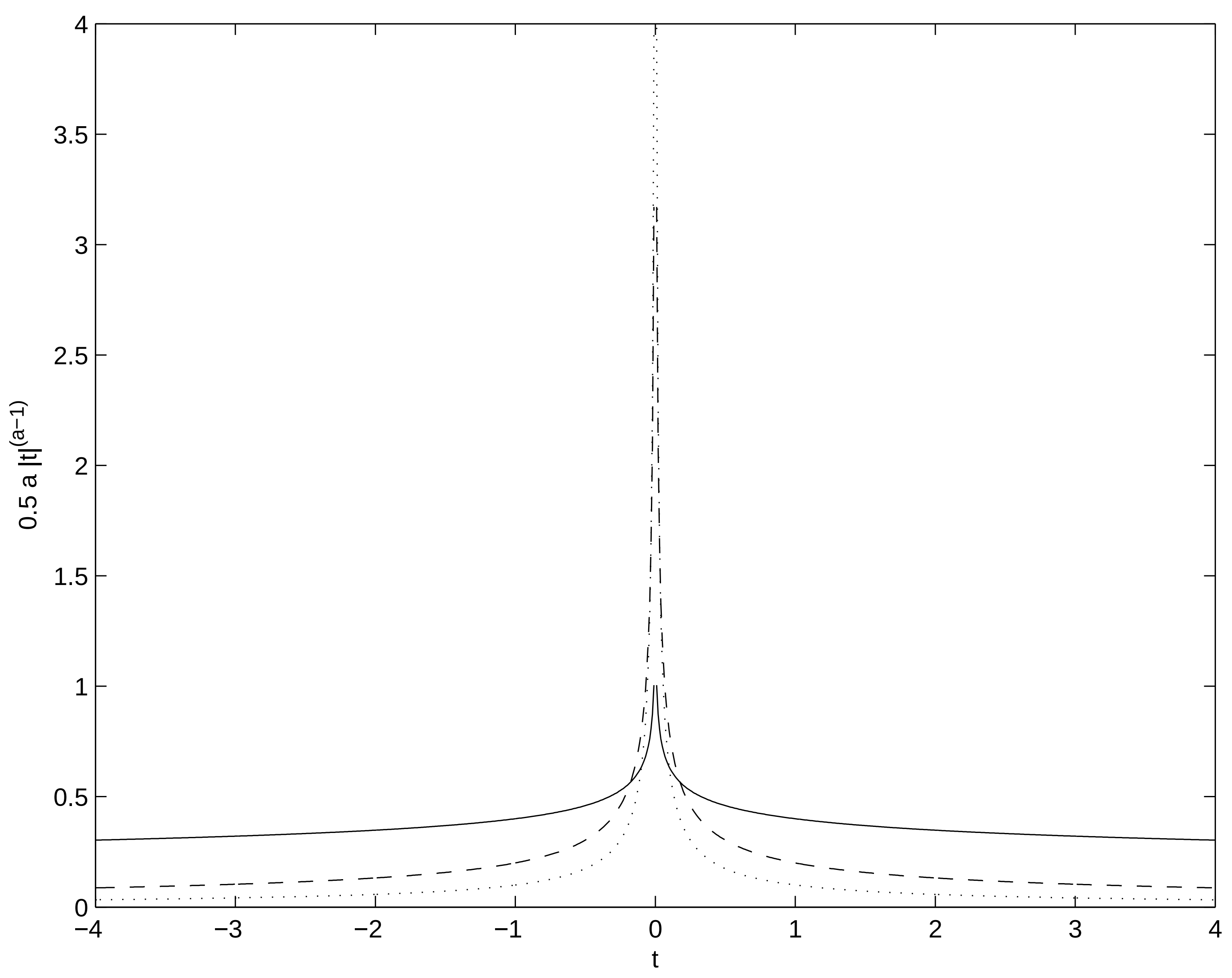


Figure 3.13: The delta function as the limit of $0.5\,a|t|^{(a-1)}$ as $a \rightarrow 0$. The function is plotted for $a = 0.8$ (solid line), $a = 0.4$ (dashed line), and $a = 0.2$ (dotted line).

The delta function is the derivative of the unit step function

$$u(t) = \begin{cases} 1 \; for \; t > 0, \\ 0 \; otherwise. \end{cases} \tag{3.27}$$

$$\delta(t) = \frac{d}{dt} u(t).$$

The delta function is also defined in terms of its action within an integral over a certain interval $[T_1, T_2]$:

$$\int_{T_1}^{T_2} x(t)\,\delta(t-t_o)\,dt = \begin{cases} x(t_o) & if\ T_1 < t_o < T_2, \\ 0 & otherwise, \end{cases} \tag{3.28}$$

where $x(t)$ is a function that is continuous at t_o.

This is known as the *sifting property* of the delta function, because the value of the function $x(t)$ at the location t_o of the delta function is sifted or selected from all of its values.

The expression above may be extended to all t as

$$x(t) = \int_{\alpha=-\infty}^{\infty} x(\alpha)\,\delta(t-\alpha)\,d\,\alpha, \tag{3.29}$$

where α is a temporary variable:

this represents resolving the arbitrary signal $x(t)$ into a weighted combination of mutually orthogonal delta functions.

Consider a continuous-time signal, $x(t)$, processed by an LSI system, as in Figure 3.14.

An LSI system is completely characterized or specified by its impulse response, $h(t)$, which is the output of the system when the input is a delta function.

The output of the system, $y(t)$, is given by the convolution of the input, $x(t)$, with the impulse response, $h(t)$:

$$y(t) = \int_{\tau=-\infty}^{\infty} x(\tau)\, h(t-\tau)\, d\tau, \tag{3.30}$$

where τ is a temporary variable; equivalently

$$y(t) = \int_{\tau=-\infty}^{\infty} h(\tau)\, x(t-\tau)\, d\tau. \tag{3.31}$$

The convolution operation given above is *linear*;

periodic or circular convolution is defined in Equation 3.85.

For a causal system, the lower limit of the integrals for convolution may be changed to zero and the upper limit changed to t, the current instant of time:

$$y(t) = \int_{\tau=0}^{t} x(\tau)\, h(t-\tau)\, d\tau \tag{3.32}$$

or

$$y(t) = \int_{\tau=0}^{t} h(\tau)\, x(t-\tau)\, d\tau. \tag{3.33}$$

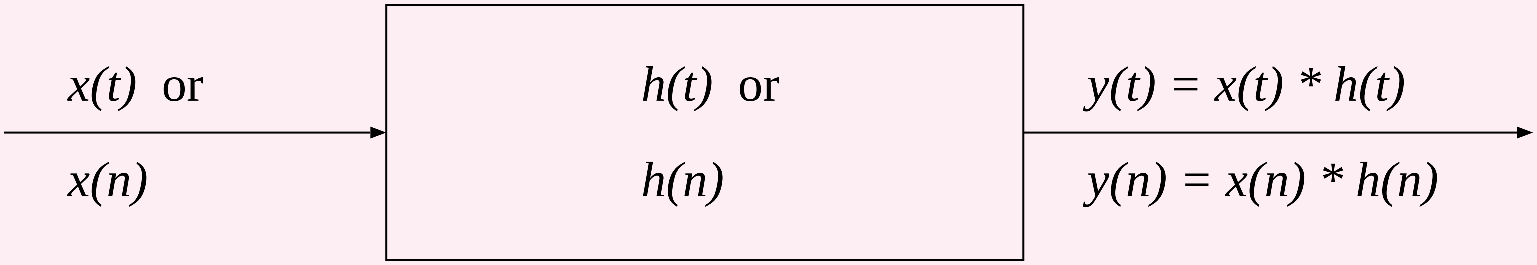


Figure 3.14: A schematic representation of a continuous-time or discrete-time LSI filter.

Discrete-time unit impulse function or delta function:

$$\delta(n) = \begin{cases} 1 \;\; if \; n = 0, \\ 0 \;\; otherwise. \end{cases} \tag{3.34}$$

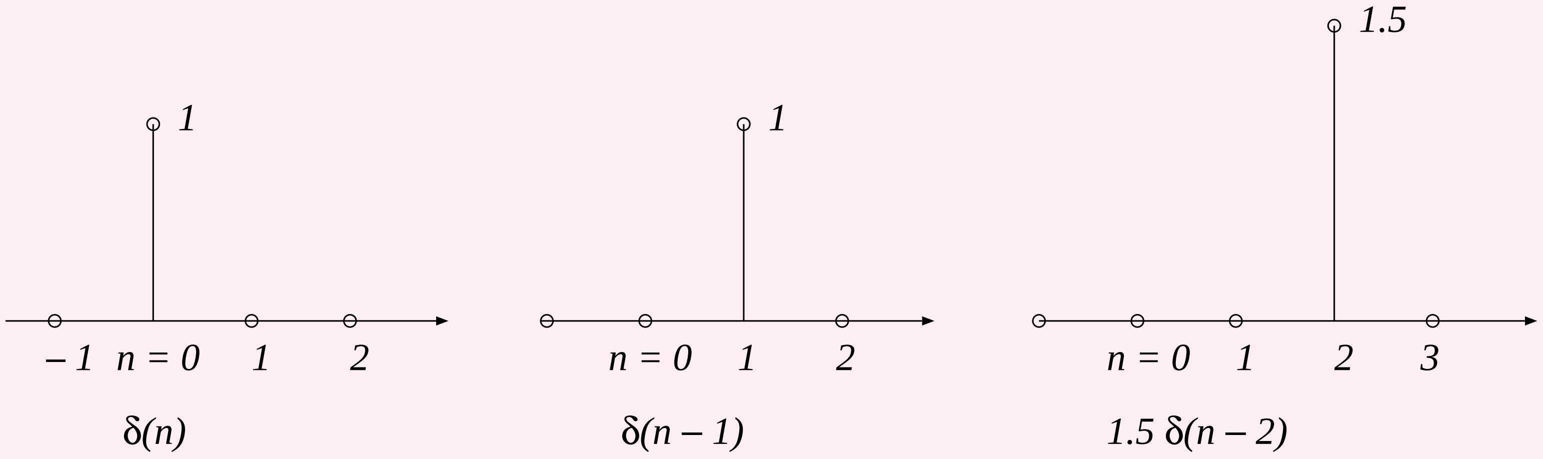


Figure 3.15: A schematic representation of the discrete-time unit impulse function or delta function as well as shifted and scaled versions of the same.

Discrete-time unit step function:

$$u(n) = \begin{cases} 1 & for\ n \geq 0, \\ 0 & otherwise. \end{cases} \tag{3.35}$$

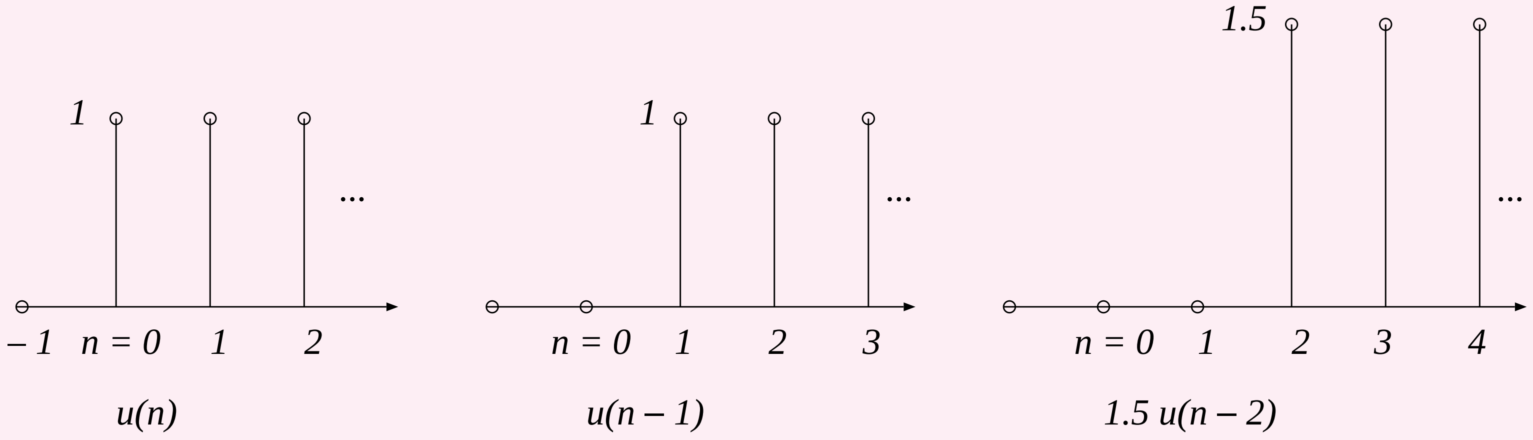


Figure 3.16: A schematic representation of the discrete-time unit step function as well as shifted and scaled versions of the same.

A discrete-time LSI system is shown schematically in Figure 3.17, displaying its impulse response, $h(n)$.

The output, $y(n)$, is given by the linear convolution of the input, $x(n)$, with the impulse response, $h(n)$:

$$y(n) = \sum_{k=0}^{n} x(k)\, h(n-k), \tag{3.36}$$

where k is a temporary variable of summation; equivalently

$$y(n) = \sum_{k=0}^{n} h(k)\, x(n-k). \tag{3.37}$$

Causality has been assumed.

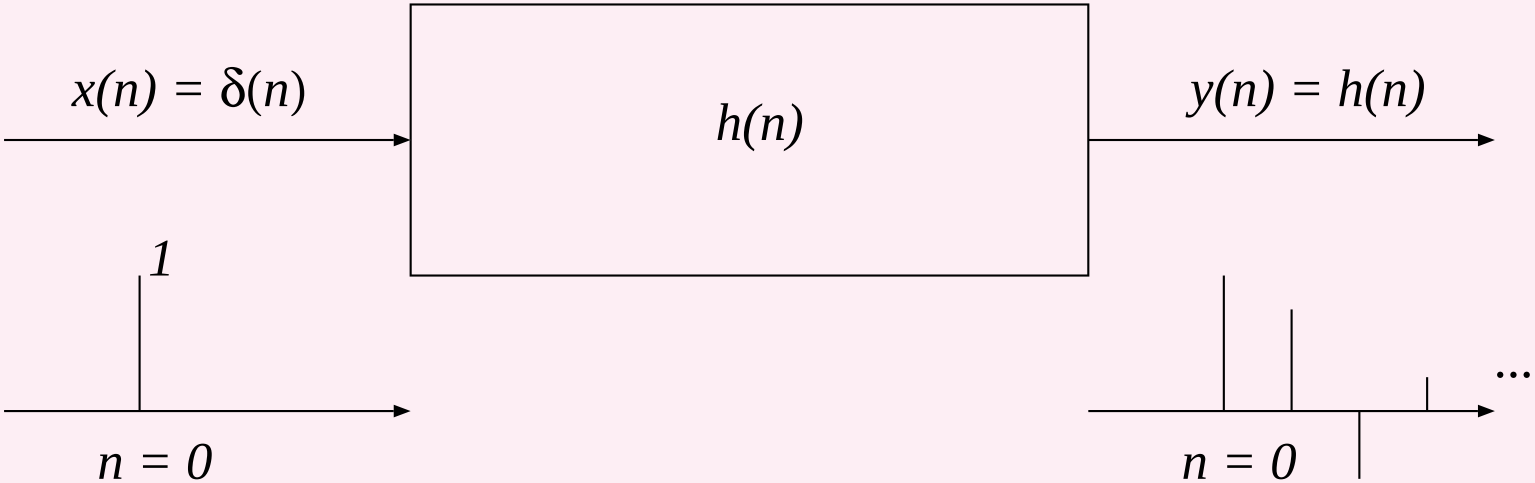


Figure 3.17: A schematic representation of the impulse response, $h(n)$, of a discrete-time LSI filter.

Consider the discrete-time version of convolution in Equation 3.36.

In addition to the change of the independent (time) variable from n to k, there are two important points to note:

- $h(-k)$ represents a reversal in time of $h(k)$;
- $h(n-k)$ represents a shift of the reversed signal $h(-k)$ by n samples.

Multiplication of $h(n-k)$ by $x(k)$ can be viewed as scaling.

The summation represents accumulation of the results or integration of $x(k)\,h(n-k)$ over the interval from $k=0$, the origin of time, to n, the present instant of time.

n:	0	1	2	3	4	5	6	7
$x(n)$:	4	5	3	1				
$x(n-1)$:		4	5	3	1			
$x(n-2)$:			4	5	3	1		
$x(n-3)$:				4	5	3	1	

Figure 3.18: A numerical illustration of shifted versions of signal. Blank spaces indicate samples of the signal that are undefined or zero in value. This type of shifting is known as linear shifting. See Figure 3.37 for an example of circular or periodic shifting.

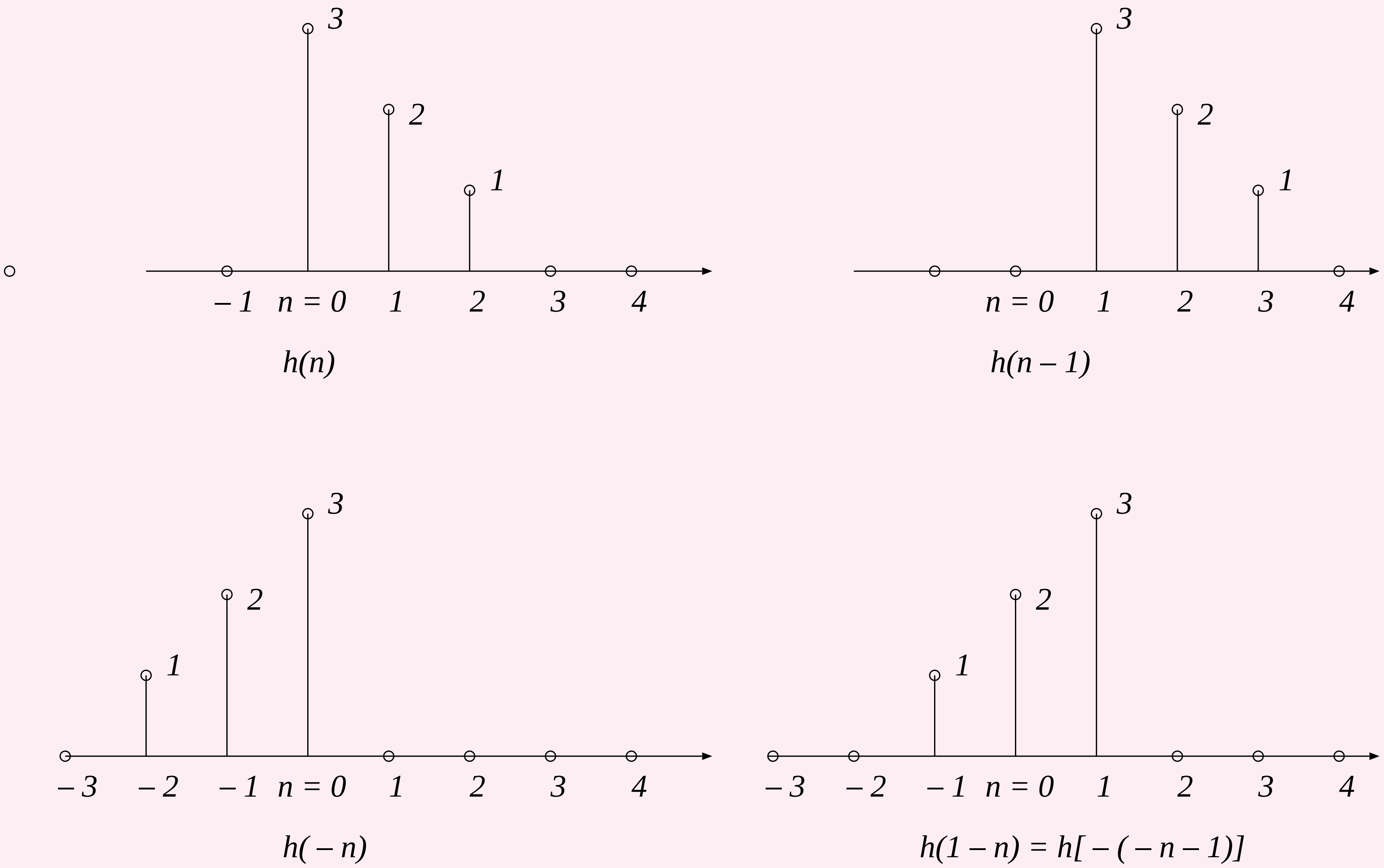


Figure 3.19: A schematic illustration of reversing and shifting of a signal.

We could expand and modify Equation 3.36 as follows:

$$\begin{aligned}
y(n) &= \sum_{k=0}^{n} x(k)\,h(n-k), \\
y(0) &= \sum_{k=0}^{0} x(k)\,h(0-k) \\
&= x(0)h(0). \\
y(1) &= \sum_{k=0}^{1} x(k)\,h(1-k) \\
&= x(0)h(1) + x(1)h(0). \\
y(2) &= \sum_{k=0}^{2} x(k)\,h(2-k) \\
&= x(0)h(2) + x(1)h(1) + x(2)h(0). \\
y(3) &= \sum_{k=0}^{3} x(k)\,h(3-k) \\
&= x(0)h(3) + x(1)h(2) + x(2)h(1) + x(3)h(0).
\end{aligned} \tag{3.38}$$

The output ceases to exist (or has only zero values) when the shift exceeds a certain amount such that $x(k)$ and $h(n-k)$ do not overlap in time any more.

Linear convolution of two discrete-time signals with N_1 and N_2 samples leads to a result with $N_1 + N_2 - 1$ samples.

n:			0	1	2	3	4	5	6	7
$x(n)$:			4	1	3	1				
$h(n)$:			3	2	1					
k:			0	1	2	3	4	5	6	7
$x(k)$:			4	1	3	1	0	0	0	0
$h(0-k)$:	1	2	3							
$h(1-k)$:		1	2	3						
$h(2-k)$:			1	2	3					
$h(3-k)$:				1	2	3				
$h(4-k)$:					1	2	3			
$h(5-k)$:						1	2	3		
$h(6-k)$:							1	2	3	
$y(n)$:			12	11	15	10	5	1	0	0
n:			0	1	2	3	4	5	6	7

Figure 3.20: A numerical illustration of the convolution of two discrete-time signals. The notations used in the figure agree with Equation 3.36. For each shift, the corresponding output sample value is obtained by multiplying each pair of samples of $x(k)$ and $h(n-k)$ that overlap (exist at the same instant of time k) and then adding the results.

Different approach: expand Equation 3.36 as

$$\begin{aligned} y(n) &= \sum_{k=0}^{n} x(k)\, h(n-k), \\ &= x(0)\, h(n) + x(1)\, h(n-1) + x(2)\, h(n-2) \\ &+ x(3)\, h(n-3) + \ldots . \end{aligned} \tag{3.39}$$

This may be interpreted as the sum of several delayed and weighted versions of the impulse response of the system, the weights being provided by the samples of the input signal.

The system produces the impulse response, $h(n)$, when the input is $\delta(n)$;

when the system is triggered by the input $x(0)$ at $n = 0$, it produces the output $x(0)h(n)$, which lasts for the duration of $h(n)$.

When the input is $x(1)$, occurring at $n = 1$, the output is $x(1)h(n - 1)$, recognizing the fact that the system is causal and the corresponding output can only start from $n = 1$.

Note that the part of the output signal that started at $n = 0$ will, in general, continue past $n = 1$, and hence the parts of the output that start at $n = 0$ and $n = 1$ will overlap.

The process continues, as illustrated in Figure 3.21.

n:	0	1	2	3	4	5	6	7
$x(n)$:	4	1	3	1				
$h(n)$:	3	2	1					
$x(0)\,h(n-0)$:	12	8	4	0	0	0	0	0
$x(1)\,h(n-1)$:	0	3	2	1	0	0	0	0
$x(2)\,h(n-2)$:	0	0	9	6	3	0	0	0
$x(3)\,h(n-3)$:	0	0	0	3	2	1	0	0
$y(n)$:	12	11	15	10	5	1	0	0
n:	0	1	2	3	4	5	6	7

Figure 3.21: A numerical illustration of the convolution of two discrete-time signals. The output, $y(n)$, is obtained by adding the corresponding values in the four rows labeled as $x(0)h(n-0)$ through $x(3)h(n-3)$. The notations used in the figure agree with Equation 3.39. Compare this procedure with that shown in Figure 3.20. Although the view is different, the end result is the same.

Illustrations of application: effects of random noise being added to a deterministic signal.

$$x(t) = 5\sin(2\pi\, 2t) + 2\cos(2\pi\, 3t). \tag{3.40}$$

Sampling frequency $= 2\, kHz$.

Random noise with a Gaussian PDF was simulated and added to the signal x for an effective SNR of $10\, dB$.

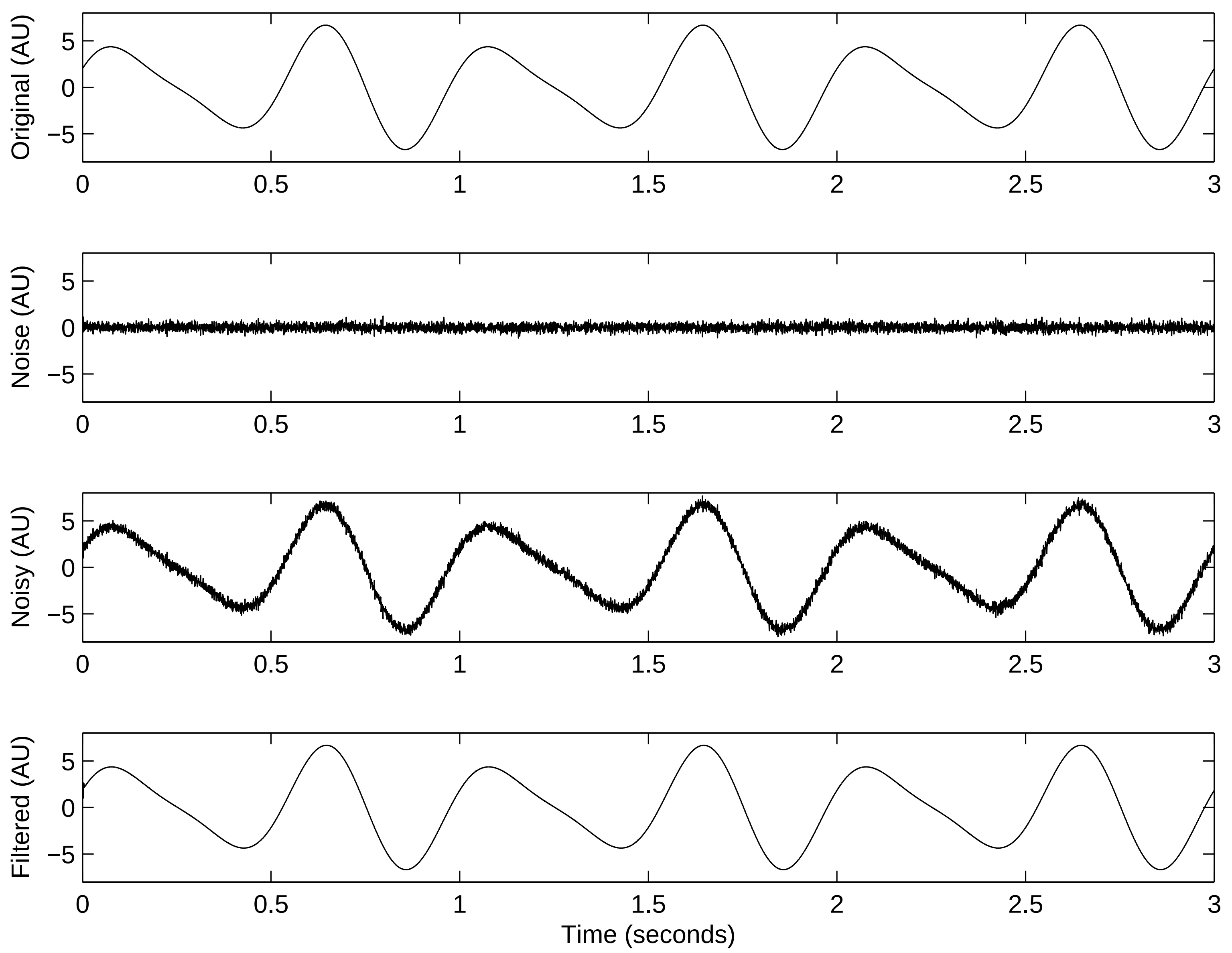


Figure 3.22: Top to bottom: original signal $x(t)$ as in Equation 3.40; Gaussian-distributed random noise; noisy signal; result of filtering using the mean of the signal values in a sliding window of width 11 samples or 5.5 ms. See Figure 3.23 for the histogram of the noise process.

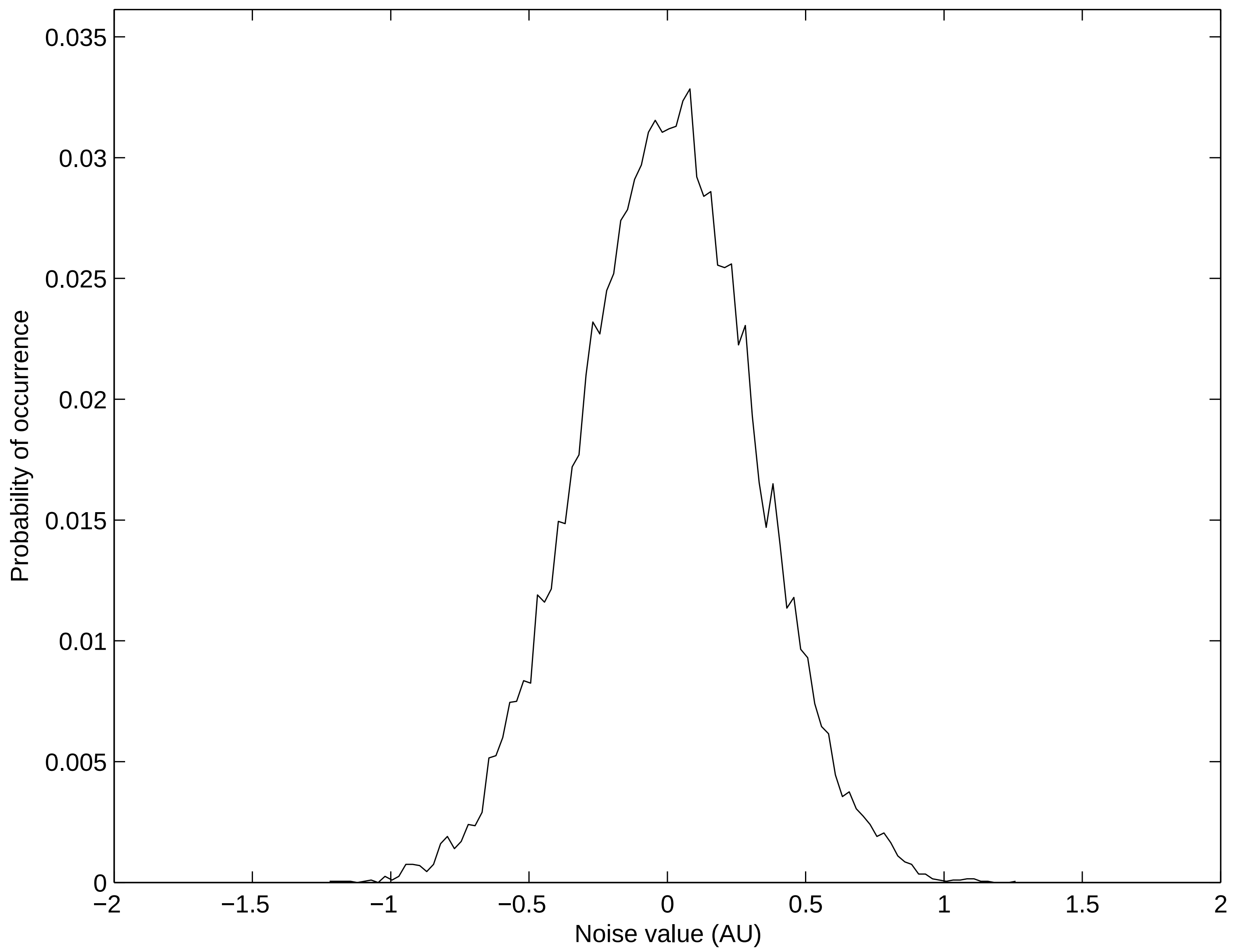


Figure 3.23: Histogram of a realization of the noise process used in the example in Figure 3.22.

The noisy signal was filtered by computing the mean of each sample and the preceding 10 samples:

$$y(n) = \frac{1}{11} \sum_{k=0}^{10} x(n-k), \tag{3.41}$$

for $n = 10, 11, \ldots, N-1$, where N is the number of samples in the signal.

A filter as above is known as a moving-average (MA) filter:

average values of the input signal are computed in a moving temporal window and used to define the output signal.

The operation in Equation 3.41 cannot be started until the first 11 samples are available in the input data stream.

The operation may also be viewed as convolution of $x(n)$ with the impulse response of the filter, as in Equation 3.37, with $h(k) = 1/11$, $k = 0, 1, \ldots, 10$.

Due to the random nature of the noise, the mean of a number of samples of the process tends to zero as the number of samples used increases.

Figure 3.24 shows another noisy signal.

Filter impulse response: linearly decreasing function or a ramp of duration $0.25\ s$,

$$h(t) = 10\,(0.25 - t),\ \ 0 \leq t \leq 0.25\ s, \tag{3.42}$$

sampling frequency $= 2\ kHz$.

Output divided by the sum of all of the values of $h(n)$.

The result is a weighted average of the corresponding values of the input signal.

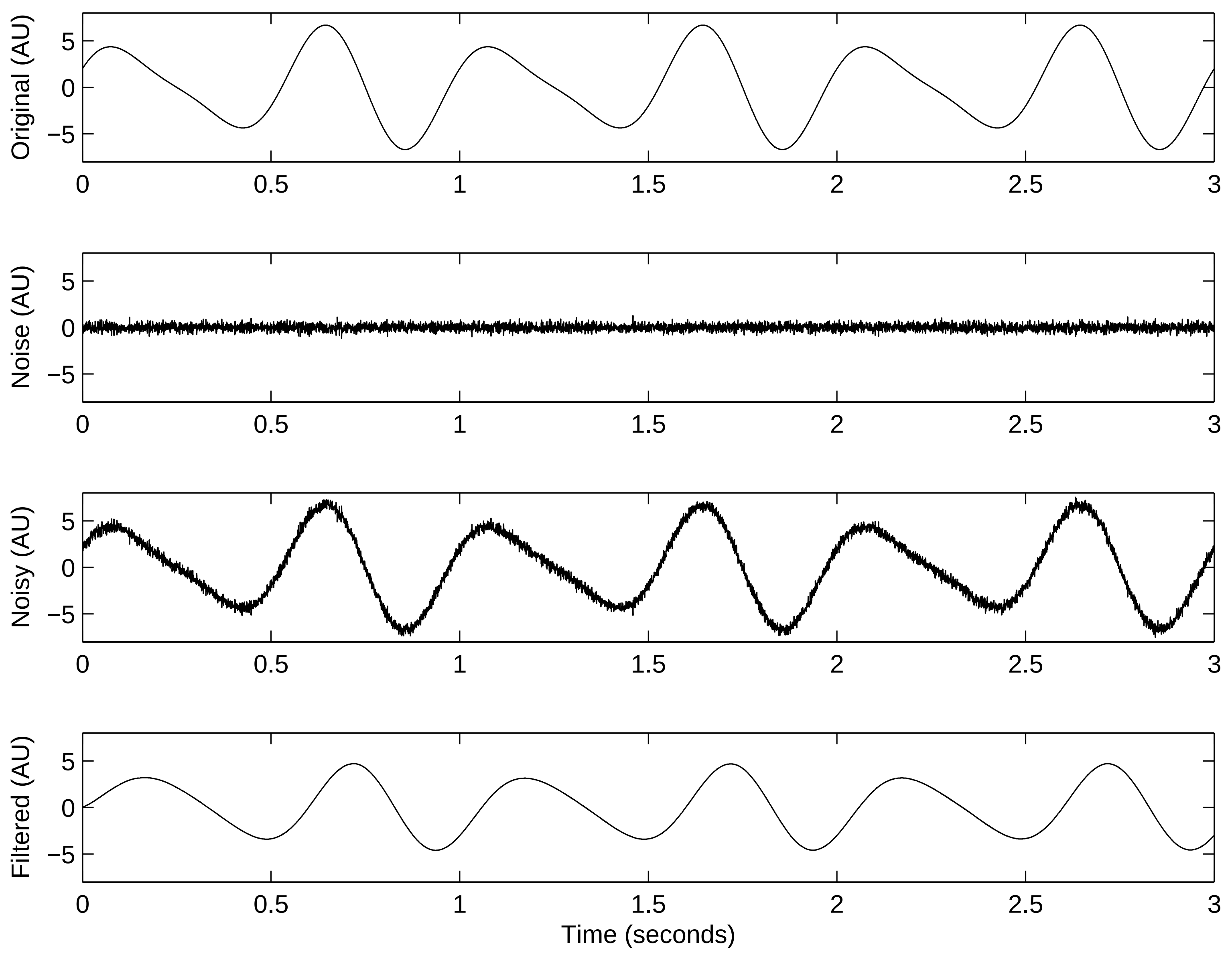


Figure 3.24: Top to bottom: original signal $x(t)$ as in Equation 3.40; Gaussian-distributed random noise; noisy signal; result of filtering using the ramp function in Equation 3.42. See Figure 3.23 for the histogram of the noise process. See also Figure 3.25.

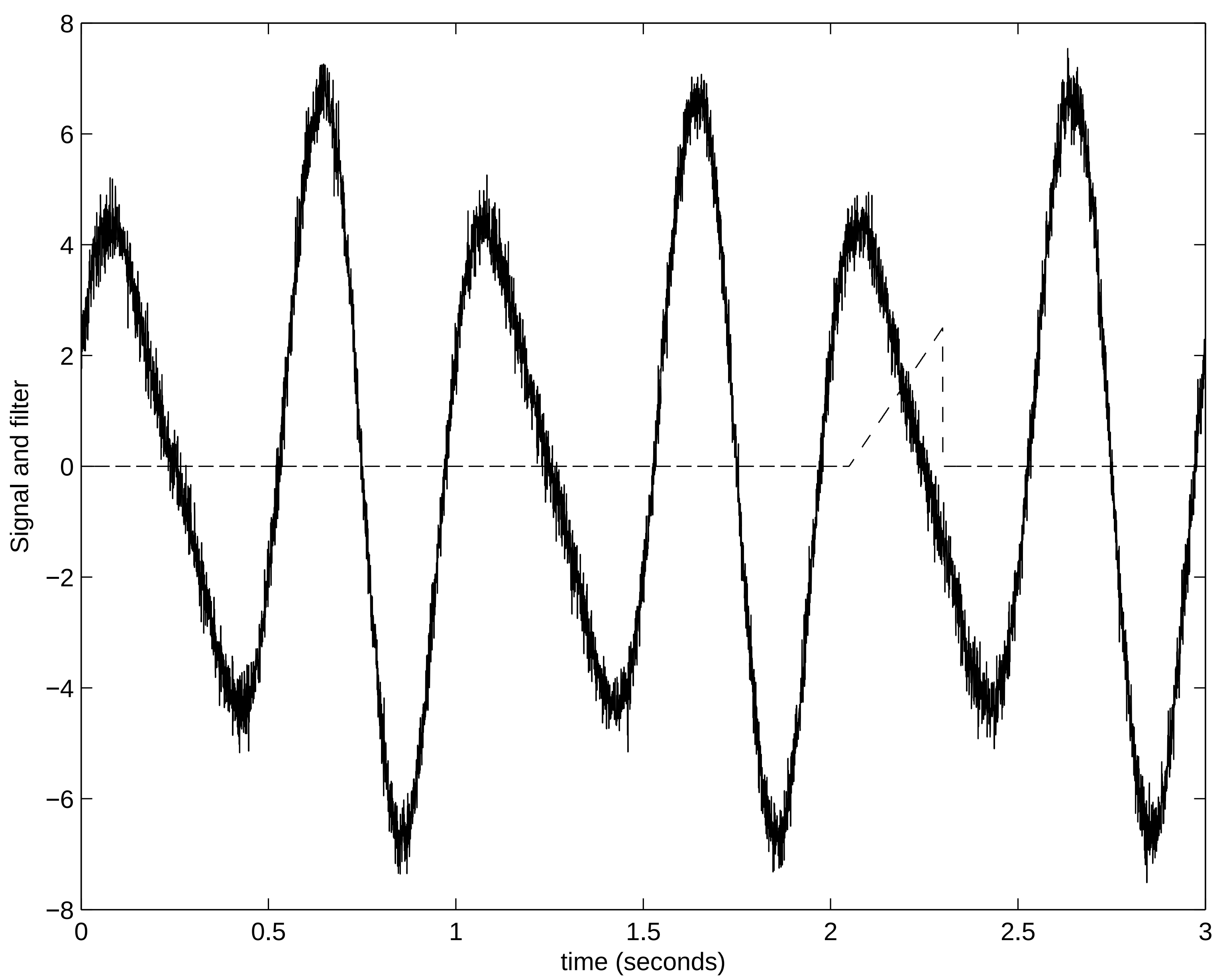


Figure 3.25: The linear ramp filter in Equation 3.42 is shown superimposed (in dashed line) on the noisy signal in the example shown in Figure 3.24. The impulse response of the filter has been reversed in time and placed at $t = 2.3\ s$. The output at $t = 2.3\ s$ is given by the area under the product of the signal and the impulse response; equivalently, it is given by the sum of the products of all samples of the signal and the impulse response that overlap.

LSI systems in series or parallel:

When a number of LSI systems are used in series or in parallel, their effects may be combined into a single equivalent system or operation.

Figure 3.26 shows two LSI systems in series (cascade).

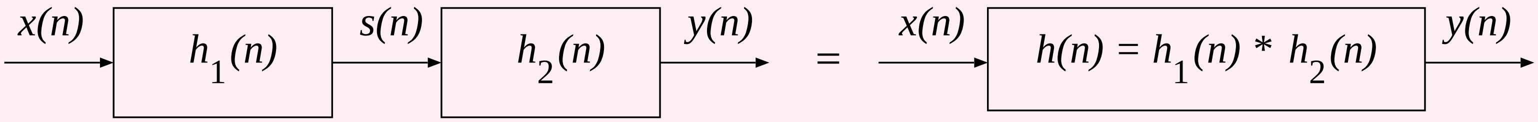


Figure 3.26: Two LSI systems in series and the equivalent system.

The first system, with the impulse response $h_1(n)$, operates upon the input, $x(n)$, to give the output as

$$s(n) = x(n) * h_1(n). \tag{3.43}$$

The second system, with the impulse response $h_2(n)$, operates upon $s(n)$ to produce the output

$$\begin{aligned} y(n) &= s(n) * h_2(n) \\ &= x(n) * h_1(n) * h_2(n) \\ &= x(n) * h(n), \end{aligned} \tag{3.44}$$

where

$$h(n) = h_1(n) * h_2(n) \tag{3.45}$$

is the impulse response of the two LSI systems in series.

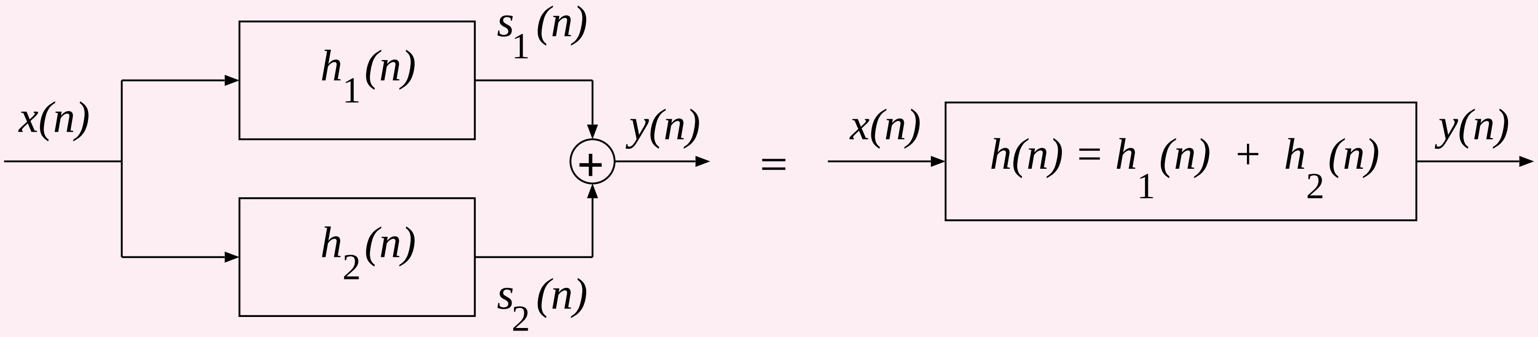


Figure 3.27: Two LSI systems in parallel and the equivalent system.

$$s_1(n) = x(n) * h_1(n). \tag{3.46}$$

$$s_2(n) = x(n) * h_2(n). \tag{3.47}$$

The combined result is

$$\begin{aligned} y(n) &= s_1(n) + s_2(n) \\ &= x(n) * h_1(n) + x(n) * h_2(n) \\ &= x(n) * [h_1(n) + h_2(n)] \\ &= x(n) * h(n), \end{aligned} \tag{3.48}$$

where

$$h(n) = h_1(n) + h_2(n) \tag{3.49}$$

is the impulse response of the two LSI systems in parallel.

3.4.2 Transform-domain analysis of signals and systems

It is convenient to analyze the behavior, characteristics, and performance of an LSI system in a transform domain.

The Laplace and Fourier transforms are useful for the analysis of continuous-time systems and signals.

The Laplace transform, $H(s)$, of the impulse response, $h(t)$, of an LTI system is defined as

$$H(s) = \int_{-\infty}^{\infty} h(t)\, \exp(-st)\, dt. \tag{3.50}$$

$s = \sigma + j\omega$

is a complex variable representing the transform domain;

$\omega = 2\,\pi\,f$ is the frequency variable in radians per second;

f is the frequency variable in Hz;

the unit for σ is neper or Napier.

If the signal $h(t)$ is causal and of finite duration, existing only over the interval of time $[0, T]$, we have

$$H(s) = \int_0^T h(t) \exp(-st)\, dt. \tag{3.51}$$

$H(s)$ is known as the transfer function of the filter.

Figure 3.28 shows a schematic representation of the s-plane.

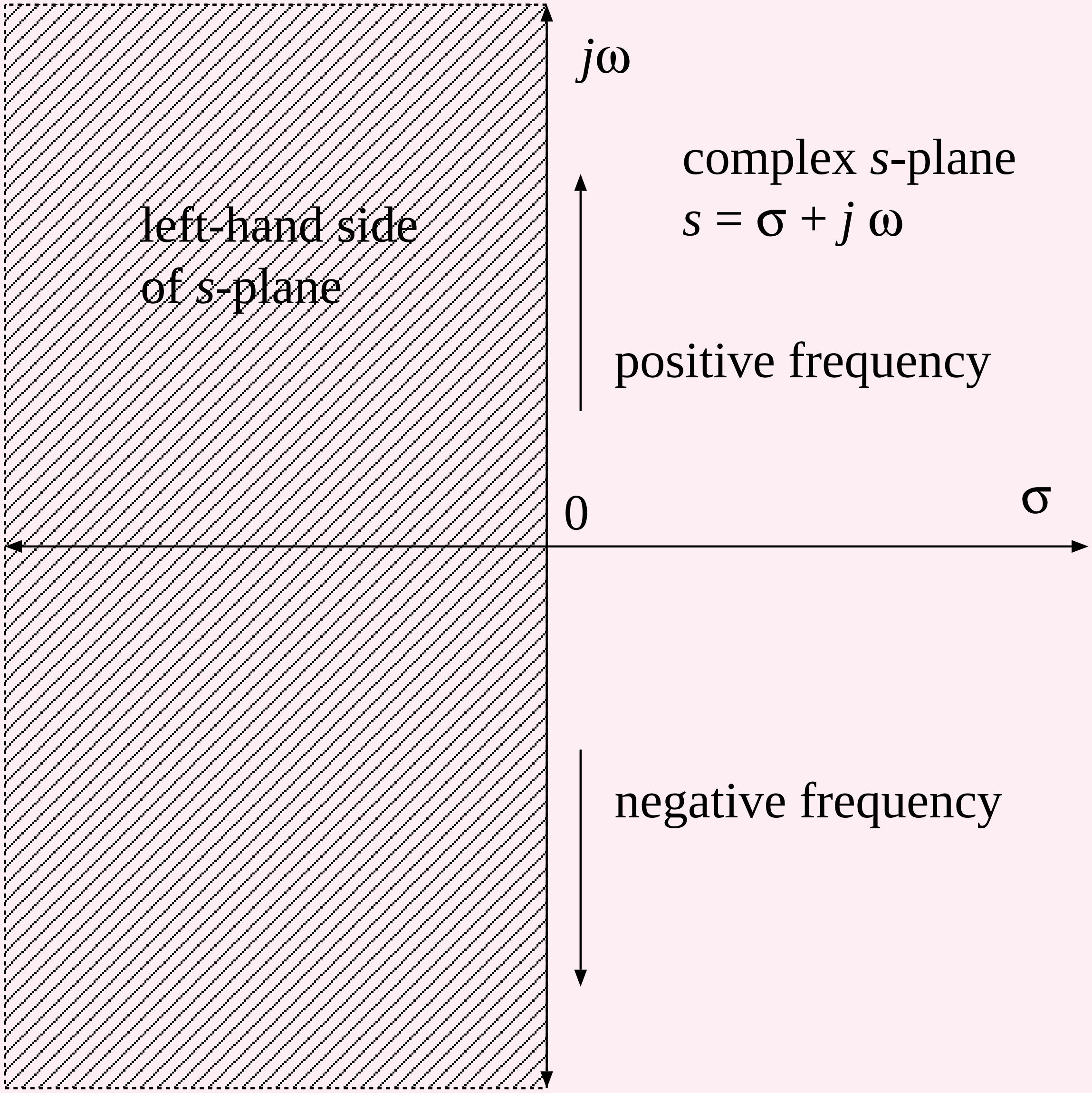


Figure 3.28: Schematic representation of the *s*-plane or Laplace transform domain.

When $H(s)$ is evaluated on the imaginary axis in the s-plane, we get the frequency response of the system:

$$H(\omega) = H(s)|_{s=j\omega} = \int_0^T \; h(t) \, \exp(-j\omega t) \, dt, \qquad (3.52)$$

which is the Fourier transform of the impulse response, $h(t)$.

Some authors express $H(\omega)$ as $H(e^{j\omega})$ or $H(j\omega)$.

In general, $H(\omega)$ is complex even if $h(t)$ is real.

Magnitude response of the system or filter $= |H(\omega)|$.

Phase response $= \angle H(\omega)$.

It is common to express the transfer function, $H(s)$, of an LTI system as a ratio of two polynomials in s.

The roots of the polynomial in the numerator give the zeros;

the roots of the denominator give the poles of the system.

For a stable LTI system, all of the poles should be located on the left-hand side (LHS) of the s-plane with $\sigma < 0$.

For a real-valued impulse response or signal, poles and zeros should occur in complex-conjugate pairs or with real values.

The pole–zero plot of a system is adequate to derive its impulse response, transfer function, and output (for a given input signal) except for a scale factor or gain.

The convolution of two signals in the time domain is converted to the product of their individual Laplace transforms in the s-domain:

$$\begin{aligned} if \;\; y(t) &= x(t) * h(t), \\ then \;\; Y(s) &= X(s)\, H(s), \\ Y(\omega) &= X(\omega)\, H(\omega). \end{aligned} \tag{3.53}$$

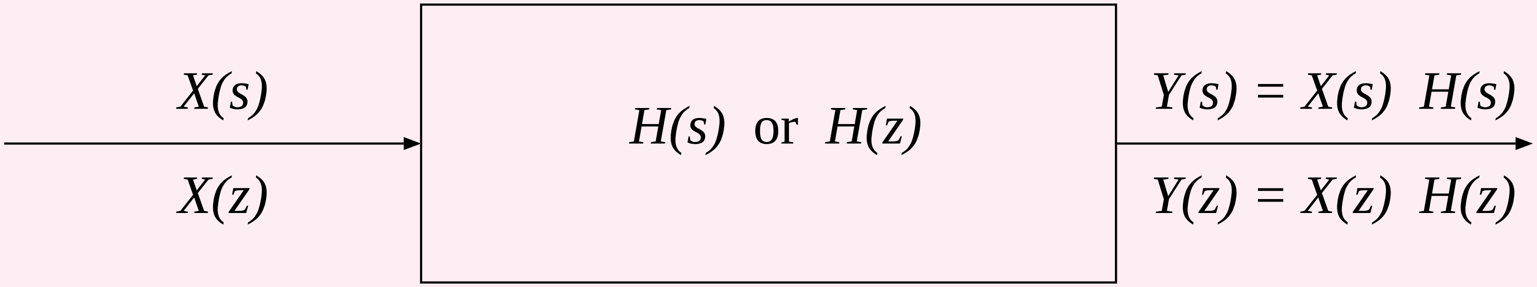


Figure 3.29: Input–output relationship for an LTI system in the s-domain or an LSI system in the z-domain.

For two LTI systems in parallel, we have the output

$$Y(s) = [H_1(s) + H_2(s)]\, X(s),$$

with the equivalent transfer function

$$H(s) = H_1(s) + H_2(s).$$

For two LTI systems in series, we have the output

$$Y(s) = H_1(s)\, H_2(s)\, X(s),$$

with the equivalent transfer function

$$H(s) = H_1(s)\, H_2(s).$$

The discrete-time counterpart of the Laplace transform is the z-transform, which is useful for the analysis of discrete-time LSI systems and signals.

The z-transform of a signal $x(n)$ is defined as

$$X(z) = \sum_{n=-\infty}^{\infty} x(n)\, z^{-n}, \tag{3.54}$$

where z is a complex variable.

The z-transform of the impulse response, $h(n)$, of a causal, LSI, finite-impulse response (FIR) system, with $h(n)$ existing only for $n = 0, 1, 2, \ldots, N-1$, is

$$H(z) = \sum_{n=0}^{N-1} h(n)\; z^{-n}, \tag{3.55}$$

which is the transfer function of the system.

The convolution of two signals in the time domain is converted to the product of their individual z-transforms:

$$\begin{aligned} if\ \ y(n) &= x(n) * h(n), \\ then\ \ Y(z) &= X(z)\ H(z). \end{aligned} \tag{3.56}$$

$$\begin{aligned} Y(z) &= \sum_{n=-\infty}^{\infty} y(n)\ z^{-n} \\ &= \sum_{n=-\infty}^{\infty} \sum_{k=-\infty}^{\infty} x(k)\ h(n-k)\ z^{-n} \\ &= \sum_{k=-\infty}^{\infty} x(k) \sum_{n=-\infty}^{\infty} h(n-k)\ z^{-n}. \end{aligned} \tag{3.57}$$

Now, let $m = n - k$. Then, $n = m + k$, and we have

$$\begin{aligned}
Y(z) &= \sum_{k=-\infty}^{\infty} x(k) \sum_{m=-\infty}^{\infty} h(m)\, z^{-(m+k)} \\
&= \sum_{k=-\infty}^{\infty} x(k) \sum_{m=-\infty}^{\infty} h(m)\, z^{-m}\, z^{-k} \\
&= \sum_{k=-\infty}^{\infty} x(k)\, z^{-k} \sum_{m=-\infty}^{\infty} h(m)\, z^{-m} \\
&= X(z)\, H(z). \qquad (3.58)
\end{aligned}$$

For two LSI systems in series, we have

$$S(z) = X(z)\,H_1(z), \tag{3.59}$$

$$\begin{aligned} Y(z) &= S(z)\,H_2(z) \\ &= X(z)\,H_1(z)\,H_2(z) \\ &= X(z)\,H(z), \end{aligned} \tag{3.60}$$

$$H(z) = H_1(z)\,H_2(z). \tag{3.61}$$

For two LSI systems in parallel, we have

$$S_1(z) = X(z)\, H_1(z), \tag{3.62}$$

$$S_2(z) = X(z)\, H_2(z), \tag{3.63}$$

$$\begin{aligned} Y(z) &= S_1(z) + S_2(z) \\ &= X(z)\, H_1(z) + X(z)\, H_2(z) \\ &= X(z)\, [H_1(z) + H_2(z)] \\ &= X(z)\, H(z), \end{aligned} \tag{3.64}$$

$$H(z) = H_1(z) + H_2(z). \tag{3.65}$$

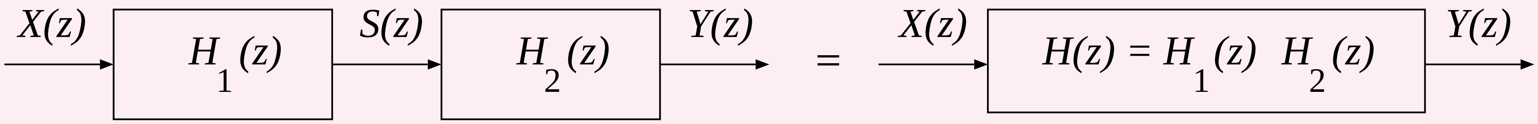


Figure 3.30: Two LSI systems in series and the equivalent system in the z-domain.

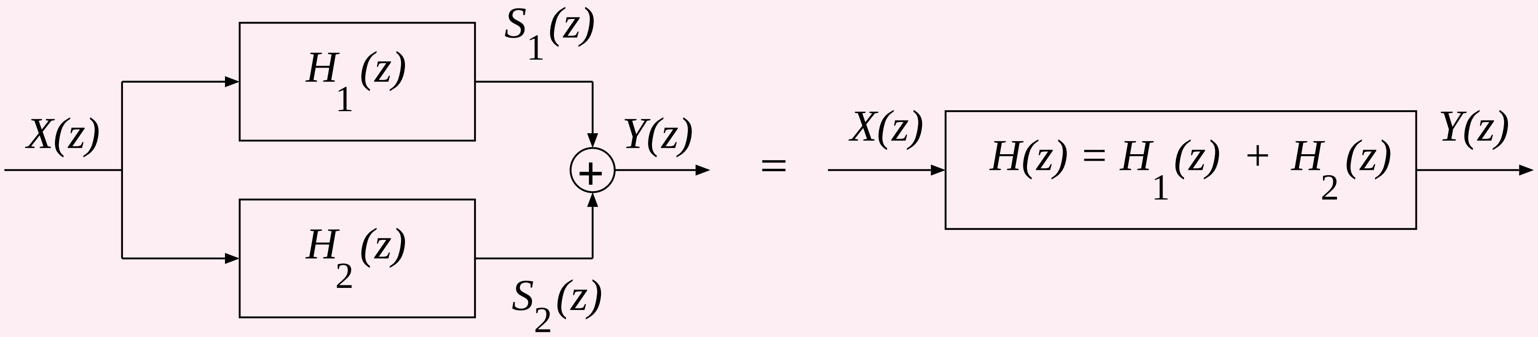


Figure 3.31: Two LSI systems in parallel and the equivalent system in the z-domain.

A few of the important properties of and examples related to the z-transform are:

- The z-transform of $\delta(n)$ is 1. The ROC is all z.
- The z-transform of the unit step function, $u(n)$, is $\frac{1}{1-z^{-1}}$. The ROC is $|z| > 1$.
- The z-transform of $a^n\, u(n)$ is $\frac{1}{1-a\, z^{-1}}$. The ROC is $|z| > |a|$.
- The z-transform of a causal signal $x(n)$ shifted by k samples, that is, $x(n-k)$, is $z^{-k}\, X(z)$, where $X(z)$ is the z-transform of $x(n)$.
- If $y(n) = x(n) * h(n)$, then $Y(z) = X(z)\, H(z)$.

Figure 3.32 illustrates the relationship between the Laplace domain and the z-domain with $f_s =$ the sampling frequency:

$z = \exp(sT)$, where $T = 1/f_s$ is the sampling interval.

The maximum frequency permitted in a signal without aliasing errors is $f_m = f_s/2$:
folding frequency or Nyquist frequency.

The entire LHS of the s-plane is mapped to the region within the unit circle in the z-plane.

All poles of a stable system should lie within the unit circle.

The bilinear transformation is a method for mapping from the s-plane to the z-plane.

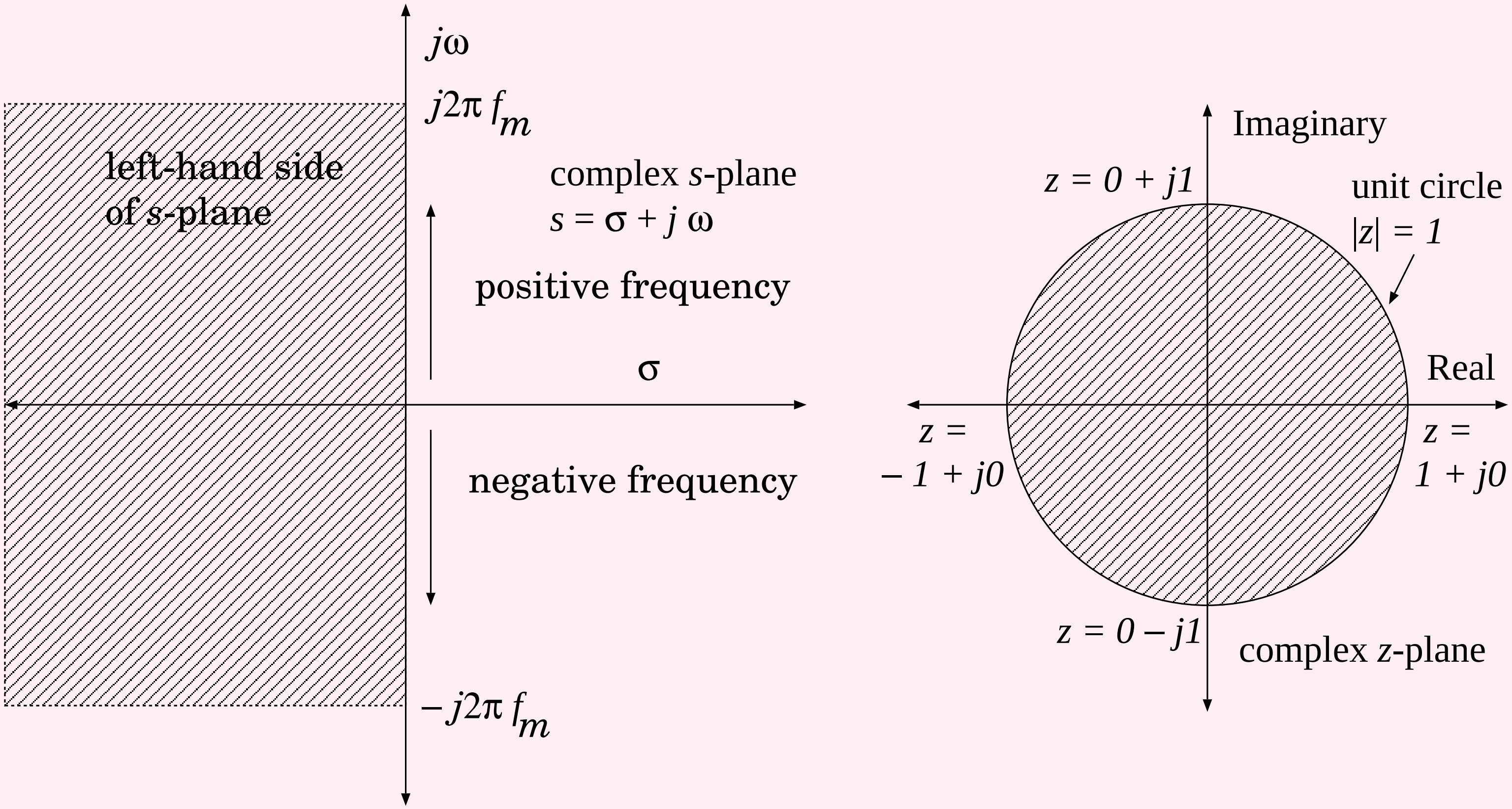
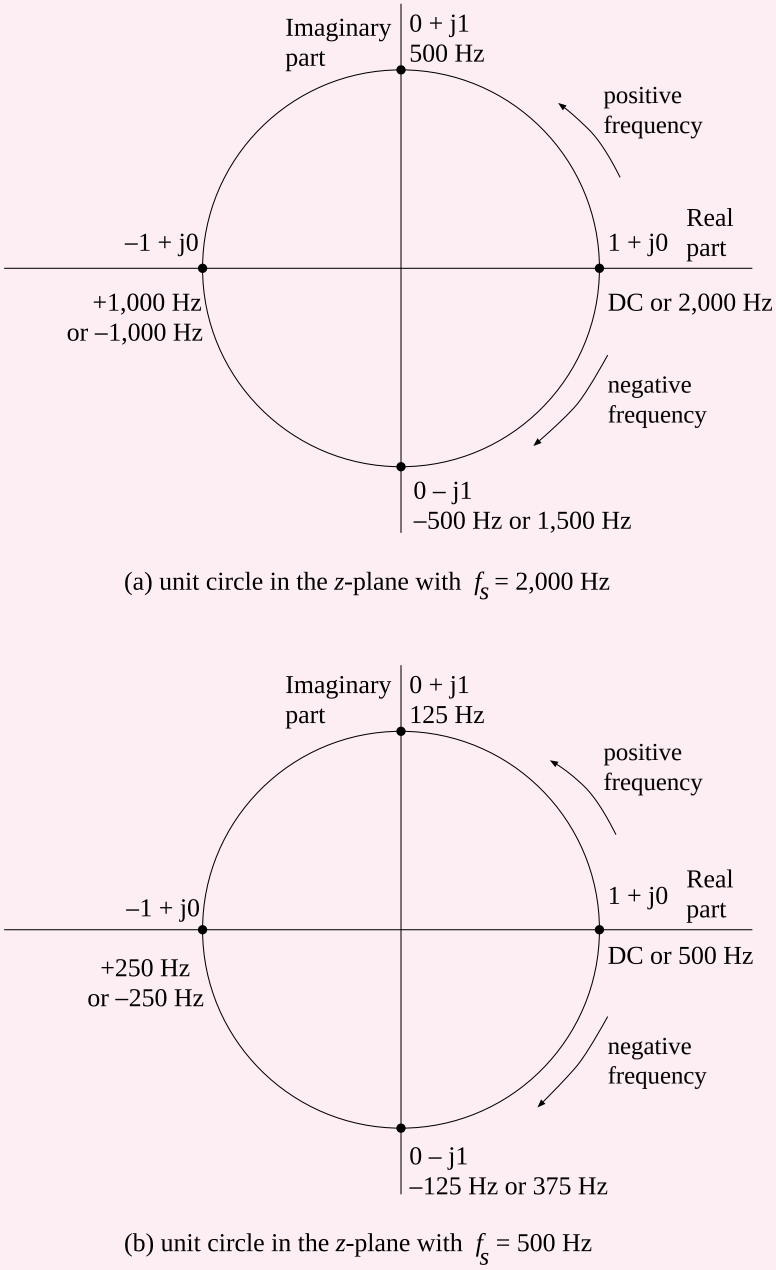


Figure 3.32: Transformation from the Laplace domain to the z-domain with $z = \exp(sT)$. $f_m = f_s/2$.

Figure 3.33 provides interpretation of the frequency variable around the unit circle in the z-domain for two different sampling frequencies, $f_s = 2,000\ Hz$ and $f_s = 500\ Hz$.

The unit circle is given by $z = \exp(j\omega T)$, and represents the frequency axis in a circular and periodic manner instead of the linear vertical axis in the s-plane.

The notions of a limited bandwidth of $\pm f_s/2$ or $[0, f_s]$ and the introduction of aliasing errors if the Nyquist sampling rate is not satisfied are demonstrated by the circular and periodic nature of the frequency axis.



(a) unit circle in the *z*-plane with f_s = 2,000 Hz

(b) unit circle in the *z*-plane with f_s = 500 Hz

Figure 3.33: Interpretation of the frequency variable with uniform sampling along the unit circle in the z-domain for two different sampling frequencies.

The Fourier transform may be seen as the Laplace transform evaluated on the imaginary axis with $s = j\omega$.

Evaluation of the z-transform on the unit circle in the z-plane gives us the Fourier transform:

$$\begin{aligned} X(\omega) &= \sum_{n=0}^{N-1} x(n)\ z^{-n}|_{z=\exp(j\omega T)} \\ &= \sum_{n=0}^{N-1} x(n)\ \exp(-j\omega nT). \end{aligned} \tag{3.66}$$

Using $\omega = 2\pi f$ and $T = 1/f_s$, we have the argument of the $\exp$ function above as $-j\omega nT = -j2\pi nf/f_s$.

We may now consider the ratio f/f_s to represent a normalized frequency variable in the range $[0, 1]$,

with zero corresponding to DC and

unity corresponding to f_s.

The result of multiplication of this ratio with 2π may be seen as division of the range $[0, 2\pi]$ into the frequency axis spanning the range $[0, f_s]$.

Thus, the variable T may be dropped.

3.4.3 The pole–zero plot

Consider a signal processing system or filter with the transfer function specified as a ratio of two polynomials in z or a rational function of z:

$$H(z) = \frac{Y(z)}{X(z)} = \frac{\Sigma_{k=0}^{N} \; b_k \, z^{-k}}{1 + \Sigma_{k=1}^{M} \; a_k \, z^{-k}}. \tag{3.67}$$

Equivalent time-domain difference equation of the filter:

$$y(n) = \sum_{k=0}^{N} \; b_k \; x(n-k) - \sum_{k=1}^{M} \; a_k \; y(n-k). \tag{3.68}$$

Let the roots of the numerator of Equation 3.67 be
$z_k,\ k = 1, 2, \ldots, N$:
zeros of the transfer function $H(z)$.

Let the roots of the denominator be $p_k,\ k = 1, 2, \ldots, M$:
poles of the transfer function $H(z)$.

A plot of the roots of $H(z)$ in the z-plane is the pole–zero plot of the system; see Figure 3.34.

The transfer function in terms of the poles and zeros:

$$H(z) = \frac{\Pi_{k=1}^{N} \left(1 - z_k z^{-1}\right)}{\Pi_{k=1}^{M} \left(1 - p_k z^{-1}\right)}. \tag{3.69}$$

$$H(z) = z^{(M-N)} \; \frac{\Pi_{k=1}^{N} \left(z - z_k\right)}{\Pi_{k=1}^{M} \left(z - p_k\right)}. \tag{3.70}$$

$(z - z_k)$: vector from an arbitrary point z to the zero z_k.

$(z - p_k)$: vector from an arbitrary point z to the pole p_k.

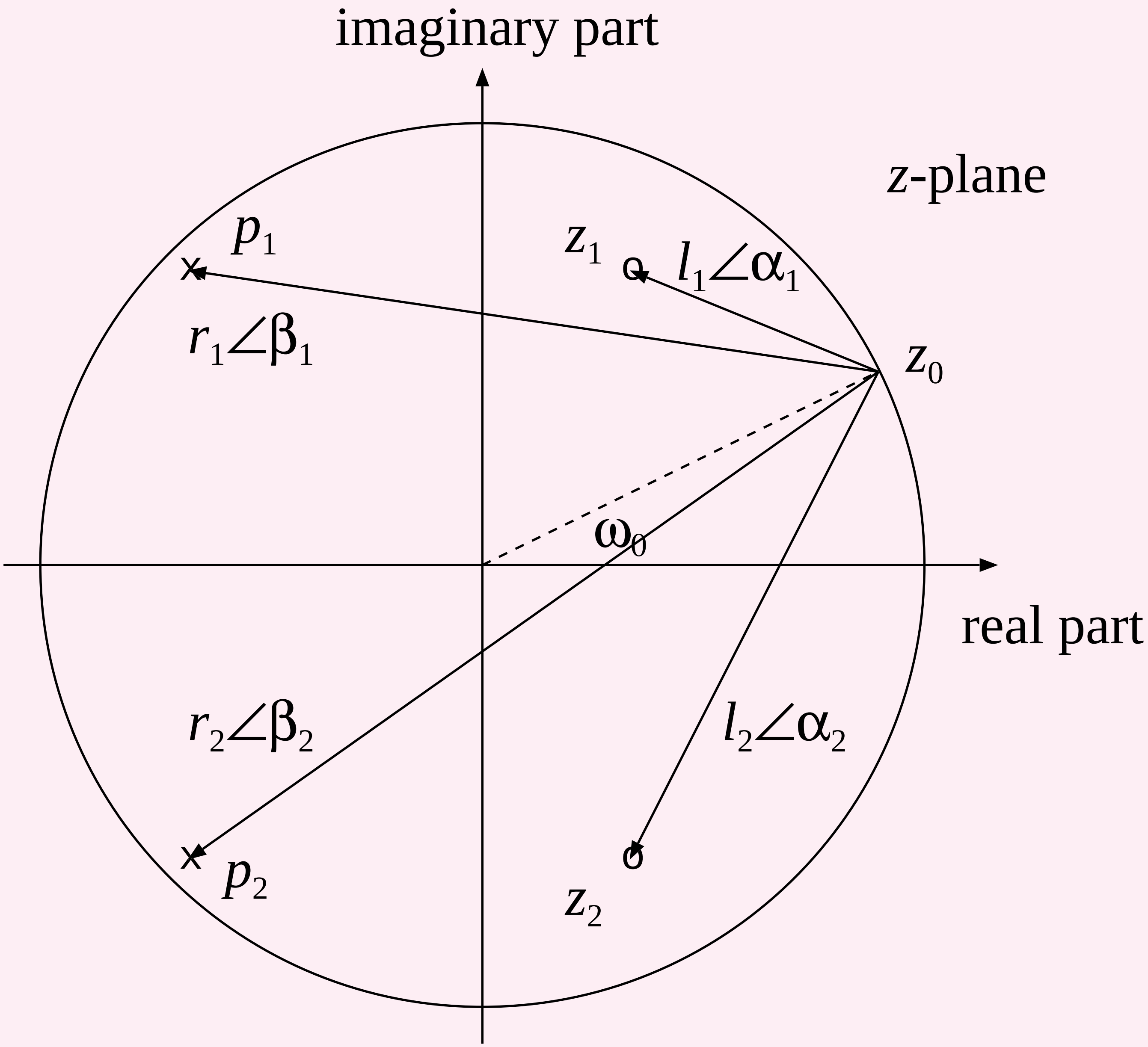


Figure 3.34: Derivation of the frequency response of a system from its pole–zero plot.

Now, consider the evaluation of $H(z)$ for a point z_0 on the unit circle in the z-plane, as shown in Figure 3.34.

This gives the frequency response of the system for the corresponding radian frequency ω_0:

$$H(\omega_0)|_{z=z_0} = z_0^{(M-N)} \; \frac{\Pi_{k=1}^{N} \; (z_0 - z_k)}{\Pi_{k=1}^{M} \; (z_0 - p_k)}. \qquad (3.71)$$

Let us represent the vector from z_0 to the zero z_1 as $(z_0 - z_1) = l_1 \angle \alpha_1$,

the vector from z_0 to the pole p_1 as $(z_0 - p_1) = r_1 \angle \beta_1$, etc.

$$|H(\omega_0)| = \frac{\Pi_{k=1}^{N} \; l_k}{\Pi_{k=1}^{M} \; r_k} \tag{3.72}$$

and

$$\angle H(\omega_0) = \angle z_0^{(M-N)} + \sum_{k=1}^{N} \; \alpha_k - \sum_{k=1}^{M} \; \beta_k . \tag{3.73}$$

Magnitude response (except for a gain factor):

ratio of the product of the distances from the corresponding point in the z-plane to all of the zeros of the system

to the product of the distances from the same point to all of the poles of the system.

Phase response (except for any additional linear phase):

difference between the sum of the angles of the vectors from the frequency point to all of the zeros of the system and

the sum of the angles of the vectors from the same frequency point to all of the poles of the system.

As the point z_0 approaches a zero of the system, one of the distances l_k will become small, leading to a low response.

If a zero is present on the unit circle, $l_k = 0$, resulting in zero response at the related frequency.

Zero on the unit circle: spectral null of the system.

As the frequency variable crosses from one side of a zero on the unit circle to the other side, there will be an associated 180° change in the phase response.

As z_0 approaches a pole of the system close to the unit circle, $r_k \approx 0$, leading to a high response at the related frequency.

Pole close to the unit circle: spectral peak or resonance.

The pole–zero plot of a system may be inspected to elicit qualitative information related to its frequency response.

3.4.4 The discrete Fourier transform

If we consider evaluation of the Fourier transform for only certain sampled values of the frequency, we could span the range $[0, 2\pi]$ with K samples, with even steps of $2\pi/K$.

This is equivalent to spanning the range $[0, 1]$ of the normalized frequency with K samples.

The discrete Fourier transform (DFT):

$$X(k) = \sum_{n=0}^{N-1} x(n)\ \exp\left(-j\frac{2\pi}{K}nk\right), \qquad (3.74)$$

$k = 0, 1, 2, \ldots, K - 1$.

If $x(n)$ has only N nonzero samples, we need only N samples of the DFT evenly spaced over the unit circle in the z-plane for exact recovery of $x(n)$ from $X(k)$. Then,

$$X(k) = \sum_{n=0}^{N-1} x(n)\ \exp\left(-j\frac{2\pi}{N}nk\right), \tag{3.75}$$

for $k = 0, 1, 2, \ldots, N-1$, and

$$x(n) = \frac{1}{N} \sum_{k=0}^{N-1} X(k)\ \exp\left(+j\frac{2\pi}{N}nk\right), \tag{3.76}$$

$n = 0, 1, 2, \ldots, N-1$, as the forward and inverse DFT.

Define a complex variable (vector or a phasor)

$$W_N = \exp\left(-j\frac{2\pi}{N}\right). \tag{3.77}$$

Then

$$X(k) = \sum_{n=0}^{N-1} x(n)\, W_N^{nk}, \tag{3.78}$$

for $k = 0, 1, 2, \ldots, N-1$.

With $N = 8$, Figure 3.35 shows the vectors or phasors representing the $N = 8$ roots of unity, with W_8^k, $k = 0, 1, 2, \ldots, 7$, where

$$W_8 = \exp\left(-j\tfrac{2\pi}{8}\right).$$

Given the relationship

$$W_N^{nk} = \exp\left(-j\frac{2\pi}{N}nk\right) = \cos\left(\frac{2\pi}{N}nk\right) - j\sin\left(\frac{2\pi}{N}nk\right), \tag{3.79}$$

the sinusoidal nature of the basis functions used in the DFT becomes clear.

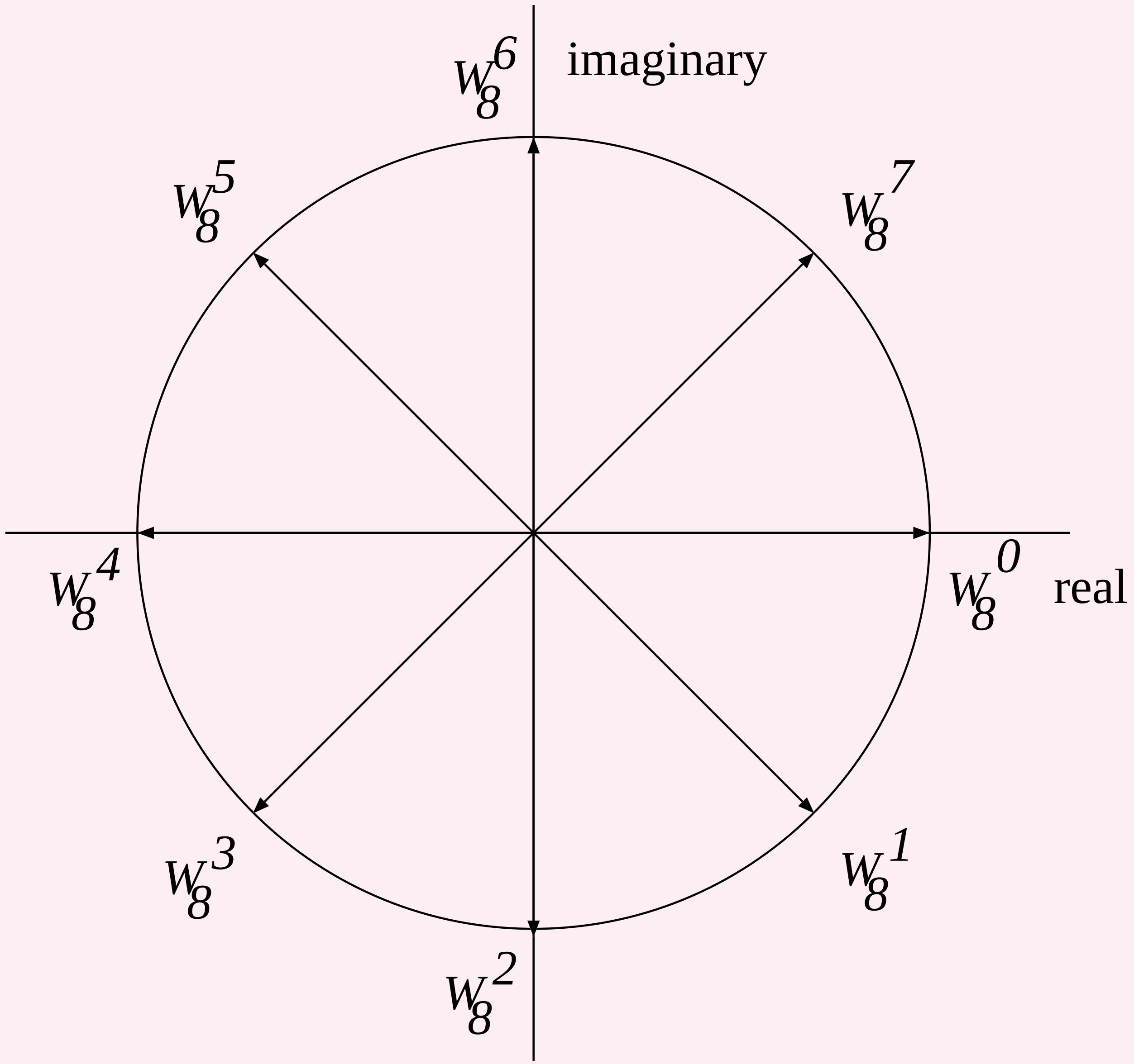


Figure 3.35: Vectors (or phasors) representing the $N = 8$ roots of unity, with W_8^k, $k = 0, 1, 2, \ldots, 7$, where $W_8 = \exp\left(-j\frac{2\pi}{8}\right)$. Based upon a similar figure by Hall.

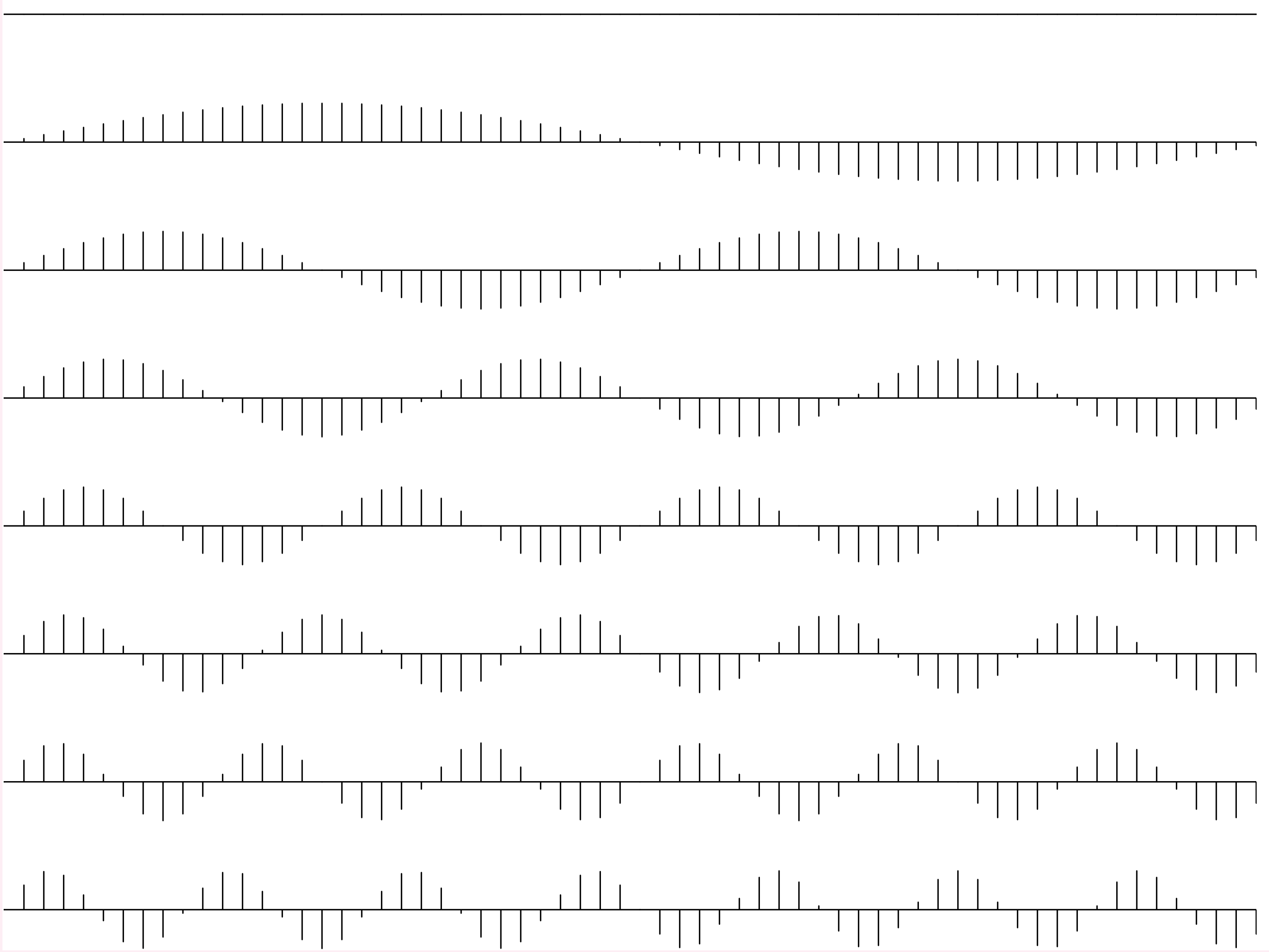


Figure 3.36: Sinusoidal basis functions used in the DFT. The eight plots (top to bottom) show the function $\sin(\frac{2\pi}{N}nk)$, for $k = 0, 1, 2, \ldots, 7$, with $N = 64$. Each sinusoidal function (except for $k = 0$) varies over the range $[-1, 1]$. The axis labels have been suppressed for improved visualization.

Consider the DFT relationship again, with the basis functions written in terms of the sin and cos functions:

$$
\begin{aligned}
X(k) &= \sum_{n=0}^{N-1} x(n)\ \exp\left(-j\frac{2\pi}{N}nk\right) \\
&= \sum_{n=0}^{N-1} x(n)\ \left\{\cos\left(\frac{2\pi}{N}nk\right) - j\sin\left(\frac{2\pi}{N}nk\right)\right\} \\
&= \sum_{n=0}^{N-1} x(n)\ \cos\left(\frac{2\pi}{N}nk\right) - j\sum_{n=0}^{N-1} x(n)\ \sin\left(\frac{2\pi}{N}nk\right).
\end{aligned}
\tag{3.80}
$$

The real part of $X(k)$ is given by the dot product of the given signal, $x(n)$, with the k^{th} $\cos$ basis function, $\cos\left(\frac{2\pi}{N}nk\right)$.

The imaginary part of $X(k)$ is given by the dot product with the corresponding $\sin$ function.

The dot product represents the projection of one vector or series of values on to another and indicates the extent of commonality between them.

The DFT coefficients indicate the amount or strength of each sinusoid present in the given signal.

This is the essence of the analysis of a signal in the frequency domain.

Inverse DFT as regeneration or synthesis of the original signal as a weighted combination of sinusoids of various frequencies, with weights = DFT coefficients:

$$\begin{aligned} x(n) &= \frac{1}{N} \sum_{k=0}^{N-1} X(k) \, \cos\left(\frac{2\pi}{N} nk\right) \\ &+ j \, \frac{1}{N} \sum_{k=0}^{N-1} X(k) \, \sin\left(\frac{2\pi}{N} nk\right), \end{aligned} \tag{3.81}$$

for $n = 0, 1, 2, \ldots, N-1$.

The periodic nature of the frequency variable and the range of the basis functions used in the DFT relationships leads to an important property:

The results of the DFT and inverse DFT are periodic.

Even when we start with a discrete-time signal of finite duration that is not periodic, its spectrum obtained via the DFT is periodic;

the result of application of the inverse DFT to the spectrum so obtained (and filtered) is periodic.

$$\begin{aligned} if \;\; y(n) &= x(n) * h(n), \\ then \;\; Y(k) &= X(k)\, H(k). \end{aligned} \tag{3.82}$$

The symmetry and periodicity of the DFT basis functions can be exploited to reduce the computational requirements: *the fast Fourier transform or FFT.*

$$W_N^{-nk} = \left(W_N^{nk}\right)^* . \tag{3.83}$$

$$W_N^{nk} = W_N^{n(k+N)} = W_N^{(n+N)k} . \tag{3.84}$$

It is usually computationally less expensive to use the second line of Equation 3.82 to realize convolution than the original definition of convolution itself:

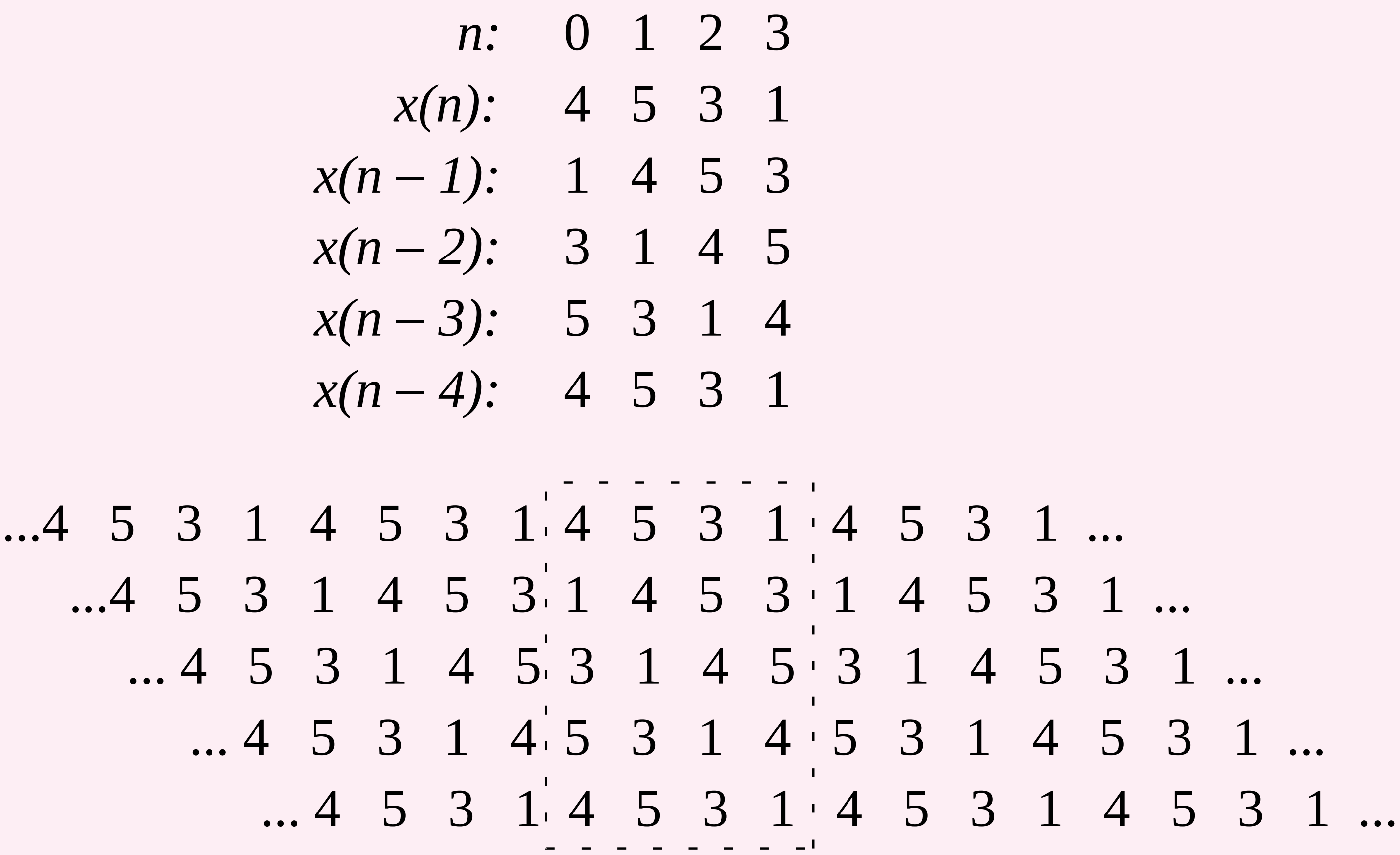
1. Compute the DFTs, $X(k)$ and $H(k)$, of the two signals given, $x(n)$ and $h(n)$.
2. Multiply the two DFTs, sample by sample, to get the DFT of the output, $Y(k) = X(k)\ H(k)$.
3. Compute the inverse DFT of $Y(k)$. The output signal, $y(n)$, is given by the real part of the result.

Convolution achieved in this manner is circular or periodic;

the results may not be the same as with linear convolution.

This arises due to the fact that the results of DFT and inverse DFT operations are periodic entities.

When we are interested in computing the output of an LSI system, we need to perform linear convolution.



n:	0	1	2	3
$x(n)$:	4	5	3	1
$x(n-1)$:	1	4	5	3
$x(n-2)$:	3	1	4	5
$x(n-3)$:	5	3	1	4
$x(n-4)$:	4	5	3	1

```
...4  5  3  1  4  5  3  1 ¦ 4  5  3  1 ¦ 4  5  3  1  ...
   ...4  5  3  1  4  5  3 ¦ 1  4  5  3 ¦ 1  4  5  3  1  ...
      ... 4  5  3  1  4  5 ¦ 3  1  4  5 ¦ 3  1  4  5  3  1  ...
         ... 4  5  3  1  4 ¦ 5  3  1  4 ¦ 5  3  1  4  5  3  1  ...
            ... 4  5  3  1 ¦ 4  5  3  1 ¦ 4  5  3  1  4  5  3  1  ...
```

Figure 3.37: A numerical illustration of circularly shifted versions of a periodic signal. The window shown in dashed lines on the lower set of signals over four cycles represents one period. Shifted versions of the signal are shown for one period in the upper part of the figure. The circular nature of the shift is evident in the upper illustration, even if the shift in the lower illustration is considered to be linear. This type of shifting is known as circular or periodic shifting. A sample that gets shifted out of the observation window on the right reenters the window from the left. See Figure 3.18 for an example of linear shifting.

In view of circular shifting of periodic signals, we can express the convolution of two periodic signals, having the same period of N samples, as

$$y_p(n) = \sum_{k=0}^{N-1} x_p(k)\, h_p[(n-k)\, mod\, N], \tag{3.85}$$

where the subscript p indicates that the signals are periodic, and $[(n-k)\, mod\, N]$ returns values within the range $[0, N-1]$, by adding or subtracting N, as required.

The result is also periodic, with period $= N$ samples.

n:	0	1	2	3
$x(n)$:	4	1	3	1
$h(n)$:	4	3	2	1
k:	0	1	2	3
$x(k)$:	4	1	3	1
$h(0-k)$:	4	1	2	3
$h(1-k)$:	3	4	1	2
$h(2-k)$:	2	3	4	1
$h(3-k)$:	1	2	3	4
$y(n)$:	26	21	24	19
n:	0	1	2	3

Figure 3.38: A numerical illustration of the circular or periodic convolution of two periodic discrete-time signals.

Given two signals, $x(n)$, $n = 0, 1, 2, \ldots, N_1$, and $h(n)$, $n = 0, 1, 2, \ldots, N_2$, the result of linear convolution of the two signals should have $N_1 + N_2 - 1$ samples.

The application of periodic convolution in place of linear convolution leads to erroneous results.

n:	0	1	2	3
$x(n)$:	4	1	3	1
$h(n)$:	3	2	1	0
k:	0	1	2	3
$x(k)$:	4	1	3	1
$h(0-k)$:	3	0	1	2
$h(1-k)$:	2	3	0	1
$h(2-k)$:	1	2	3	0
$h(3-k)$:	0	1	2	3
$y(n)$:	17	12	15	10
n:	0	1	2	3

Figure 3.39: A numerical illustration of the undesired effect of periodic convolution of two discrete-time signals when linear convolution is desired. Compare this with the illustration of linear convolution of the same signals in Figure 3.20.

To gain the computational benefits provided by the FFT, the DFT approach to convolution may be modified

by padding two given nonperiodic signals of unequal duration with zeros so as to have the same extended duration as that of the desired result,

and then considering them to represent one period each of their periodic extensions.

Algorithm to achieve linear convolution via the DFT:

1. Extend the two signals given, $x(n)$ and $h(n)$, to have N samples each, with $N \geq N_1 + N_2 - 1$, by appending or padding with zeros.
2. Compute the DFTs, $X(k)$ and $H(k)$, of the two extended signals, for $k = 0, 1, 2, \ldots, N - 1$.
3. Multiply the two DFTs, sample by sample, to get the DFT of the output, $Y(k) = X(k)\ H(k)$, $k = 0, 1, 2, \ldots, N - 1$.
4. Compute the inverse DFT of $Y(k)$. The output signal, $y(n)$, for $n = 0, 1, 2, \ldots, N_1 + N_2 - 1$, is given by the real part of the result.

The sample-by-sample product $Y(k) = X(k)\ H(k)$ cannot be computed if $X(k)$ and $H(k)$ are of different lengths or periods.

The final result $y(n)$ will, in general be complex,

and may have extra values for $N_1 + N_2 - 1 < n < N$ that are not of interest.

n:	0	1	2	3	4	5
$x(n)$:	4	1	3	1	0	0
$h(n)$:	3	2	1	0	0	0
k:	0	1	2	3	4	5
$x(k)$:	4	1	3	1	0	0
$h(0-k)$:	3	0	0	0	1	2
$h(1-k)$:	2	3	0	0	0	1
$h(2-k)$:	1	2	3	0	0	0
$h(3-k)$:	0	1	2	3	0	0
$h(4-k)$:	0	0	1	2	3	0
$h(5-k)$:	0	0	0	1	2	3
$y(n)$:	12	11	15	10	5	1
n:	0	1	2	3	4	5

Figure 3.40: A numerical illustration of the use of periodic convolution of two discrete-time and nonperiodic signals to achieve linear convolution by adequate zero padding. Compare this with the illustration of linear convolution of the same signals in Figure 3.20. With the expected number of samples in the result being six, the two input signals, with four and three samples each, have been padded with zeros to the same length of six samples. The effective period used in periodic convolution is six samples.

3.4.5 Properties of the Fourier transform

The Fourier transform is the most commonly used transform to study the frequency-domain characteristics of signals.

The Fourier transform uses sinusoids as its basis functions.

Projections are computed of the given signal $x(t)$ on to the complex exponential basis function of frequency ω $radians/s$, given by $\exp(j\omega t) = \cos(\omega t) + j\ \sin(\omega t)$, as

$$X(\omega) = \int_{-\infty}^{\infty} x(t)\ \exp(-j\omega t)\, dt, \tag{3.86}$$

or in the frequency variable f in Hz as

$$X(f) = \int_{-\infty}^{\infty} x(t)\ \exp(-j2\pi f t)\, dt. \tag{3.87}$$

The complex exponential function is conjugated in computing the projection.

In some fields, the forward Fourier transform is defined with $\exp(+j\omega t)$ in the integral.

Equations 3.86 and 3.87 represent *analysis* of the signal $x(t)$ with reference to the complex exponential basis functions.

Lower limit of the integral $= 0$ if the signal is causal;

upper limit = the signal’s duration for a finite-duration signal.

The value of $X(\omega)$ or $X(f)$ at each frequency $\omega = 2\pi f$ represents the amount or strength of the corresponding cosine and sine functions present in the signal $x(t)$.

In general, $X(\omega)$ is complex for real signals:

includes the magnitude and phase of the corresponding complex exponential.

The inverse transformation, representing *synthesis* of the signal $x(t)$ as a weighted combination of the complex exponential basis functions, is

$$x(t) = \frac{1}{2\pi} \int_{-\infty}^{\infty} X(\omega) \; \exp(j\omega t) \, d\omega$$

$$= \int_{-\infty}^{\infty} X(f) \; \exp(j2\pi f t) \, df. \qquad (3.88)$$

The second version with the frequency variable f in Hz may be more convenient than the first with ω in $radians/s$, due to the absence of $\frac{1}{2\pi}$.

If the forward Fourier transform is defined with $\exp(+j\omega t)$, the inverse Fourier transform will have $\exp(-j\omega t)$.

For a discrete-time signal $x(n)$, we may compute the Fourier transform with a continuous frequency variable ω as

$$X(\omega) = \sum_{n=-\infty}^{\infty} x(n)\ \exp(-j\omega n), \tag{3.89}$$

with the normalized-frequency range $0 \leq \omega \leq 2\pi$,

which is equivalent to $0 \leq f \leq 1$.

Lower limit $= 0$ if the signal is causal.

Upper limit $= N - 1$ for a signal with N samples.

Some of the important properties of the DFT:

- A signal $x(n)$ and its DFT $X(k)$ are both periodic.
- If $x(n)$ has N samples, its DFT $X(k)$ must be computed with at least N samples equally spaced over the normalized frequency range $0 \leq \omega \leq 2\pi$
  (around the unit circle in the z-plane)
  for complete representation of $X(\omega)$ and exact reconstruction of $x(n)$ via the inverse DFT of $X(k)$.

  One may use more than N samples to compute $X(k)$ with an FFT algorithm with $L = 2^M \geq N$ samples, where M is an integer, or to obtain $X(\omega)$ with finer frequency sampling than $2\pi/N$.

- The DFT is linear:

  the DFT of $ax(n) + by(n)$ is $aX(k) + bY(k)$,
  where $X(k)$ and $Y(k)$ are the DFTs of $x(n)$ and $y(n)$.

- The DFT of $x(n - n_o)$ is $\exp(-j\frac{2\pi}{N}kn_o)X(k)$.

  A time shift leads to a linear component being added to the phase of the original signal.

  As all sequences in DFT relationships are periodic, the shift operation should be a circular or periodic shift.

  If at least n_o zeros are present or are padded at the end of the signal before the shift operation, a circular shift will be equivalent to a linear shift.

- The DFT of $x(n) * h(n)$ is $X(k)H(k)$.

  The inverse DFT of $X(k)H(k)$ is $x(n) * h(n)$.

  Similarly, $x(n)h(n)$ and $X(k) * H(k)$ form a DFT pair.

  Convolution in one domain is equivalent to multiplication in the other.

  It is necessary for the signals in the relationships given above to have the same number of samples N.

  As all sequences in DFT relationships are periodic, the convolution operations in the relationships mentioned here are *periodic convolution* and not linear convolution.

Circular or periodic convolution is defined for periodic signals having the same period; the result will also be periodic with the same period.

The result of linear convolution of two signals $x(n)$ and $h(n)$ with different durations N_x and N_h samples will have a duration of $N_x + N_h - 1$ samples.

If linear convolution is desired via the inverse DFT of $X(k)H(k)$, the DFTs must be computed with $L \geq N_x + N_h - 1$ samples.

The individual signals should be padded with zeros at the end to make their effective durations equal for the sake of DFT computation and multiplication.

All signals and their DFTs are then periodic with the augmented period of L samples.

- The DFT of a real signal $x(n)$ possesses conjugate symmetry: $X(-k) = X^*(k)$.

  Real part and magnitude of $X(k)$: even-symmetric.

  Imaginary part and phase: odd-symmetric.

- Parseval's theorem: the total energy of the signal must be the same before and after Fourier transformation.

$$\int_{-\infty}^{\infty} |x(t)|^2 \, dt = \frac{1}{2\pi} \int_{-\infty}^{\infty} |X(\omega)|^2 \, d\omega, \qquad (3.90)$$

$$\sum_{n=0}^{N-1} |x(n)|^2 = \frac{1}{N} \sum_{k=0}^{N-1} |X(k)|^2.$$

The integral of $|X(\omega)|^2$ over all ω or the sum of $|X(k)|^2$ over all k represents the total energy of the signal, or average power, if divided by the signal's duration.

$|X(\omega)|^2$ and $|X(k)|^2$ represent the spread or *density* of the power of the signal along the frequency axis.

Note: A signal or function $x(n)$ possesses even symmetry if $x(-n) = x(n)$ or odd symmetry if $x(-n) = -x(n)$.

An arbitrary signal $x(n)$ may be expressed as a combination of a part with even symmetry (even part)

$$x_e(n) = \frac{1}{2}[x(n) + x(-n)] \tag{3.91}$$

and a part with odd symmetry (odd part)

$$x_o(n) = \frac{1}{2}[x(n) - x(-n)]. \tag{3.92}$$

$$x(n) = x_e(n) + x_o(n). \tag{3.93}$$

3.5 Time-domain Filters

Certain types of noise may be filtered directly

in the time domain using digital filters.

Advantage: spectral characterization of the signal and noise

not required — at least in a direct manner.

Time-domain processing may also be faster

than frequency-domain filtering.

3.5.1 Synchronized averaging

Linear filters fail when the signal and noise spectra overlap.

Synchronized signal averaging can separate a

repetitive signal from noise without distorting the signal.

ERP or SEP epochs may be obtained a number of times by

repeated application of the stimulus, and then

averaged using the stimulus as trigger to align the epochs.

If the noise is random with zero mean and uncorrelated

with the signal, averaging will improve the SNR.

$y_k(n)$: one realization of a signal,

with $k = 1, 2, \ldots, M$ representing the ensemble index,

$n = 1, 2, \ldots, N$ representing the time-sample index.

M: number of copies, events, epochs, or realizations.

N: number of samples in each signal.

$$y_k(n) = x_k(n) + \eta_k(n), \tag{3.94}$$

$x_k(n)$: original uncorrupted signal,

$\eta_k(n)$: noise in k^{th} copy of signal.

For each instant of time n, add M copies of signal:

$$\sum_{k=1}^{M} y_k(n) = \sum_{k=1}^{M} x_k(n) + \sum_{k=1}^{M} \eta_k(n); \; n = 1, 2, \ldots, N. \tag{3.95}$$

If the repetitions of the signal are identical and aligned,

$\Sigma_{k=1}^{M} x_k(n) = M x(n)$.

If noise is random, has zero mean and variance σ_η^2,

$\Sigma_{k=1}^{M} \eta_k(n) \longrightarrow 0$ as M increases,

with a variance of $M\sigma_\eta^2$.

RMS value of noise in the averaged signal $= \sqrt{M}\sigma_\eta$.

Thus SNR of signal increases by $\frac{M}{\sqrt{M}}$ or $\sqrt{M}$.

Larger the number of epochs or realizations averaged,

better the SNR of the result.

Synchronized averaging is a type of ensemble averaging.

Algorithmic description of synchronized averaging:

1. Obtain number of realizations of signal or event.
2. Determine reference point for each realization.

   Trigger, stimulus, QRS in ECG, etc.
3. Extract parts of the signal corresponding to the events and add them to a buffer.

   Various parts may have different durations.

   Alignment of copies at trigger point is important; the tail ends of all parts may not be aligned.
4. Divide the result by the number of events added.

Figure 3.41, upper two traces: two single-flash ERPs.

Averaging 10 and 20 flashes: third and fourth plots.

Visual ERPs: latencies of first major peak or positivity P120

(normal expected latency for adults is $120\ ms$),

trough or negativity before P120, labeled as N80;

trough following P120, labeled as N145.

N80, P120, and N145 latencies from averaged signal in

Trace 4 of Figure 3.41: 85.7, 100.7, and $117\ ms$.

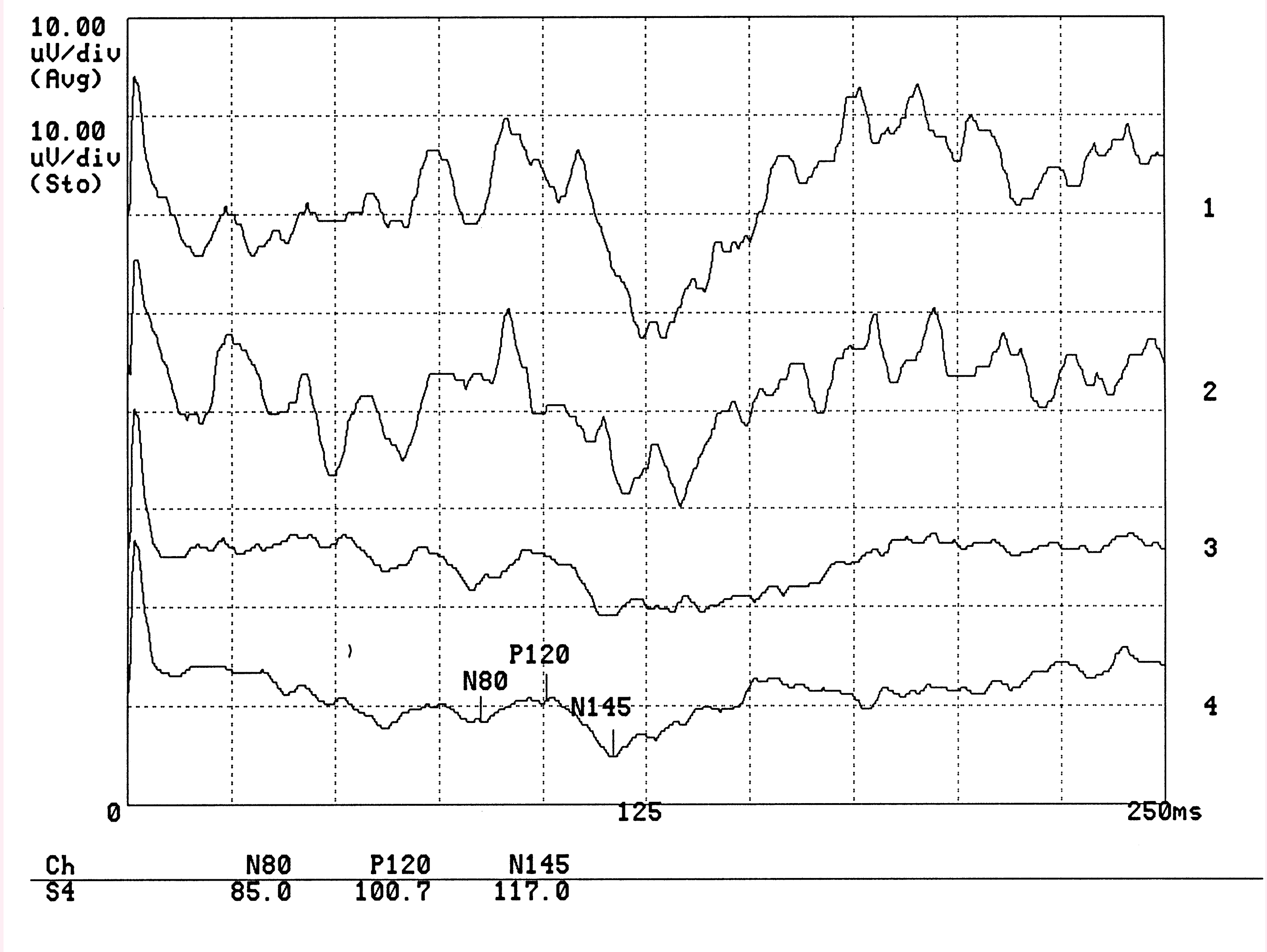


Figure 3.41: Traces 1 and 2: Two sample acquisitions of individual flash visual ERPs from the occipital midline (oz) position of a normal adult male. The ear lobes were used to form the reference lead (a1a2), and the left forehead was used as the reference (see Figure 1.37). Trace 3: Average of 10 ERPs. Trace 4: Average of 20 ERPs. The latencies of interest have been labeled on Trace 4 by an EEG technologist. See also Figure 3.2. Data courtesy of L. Alfaro and H. Darwish, Alberta Children's Hospital, Calgary.

Illustration of application:

Figure 3.42: noisy ECG signal over several beats.

To obtain trigger points:

sample QRS complex of $86\ ms$ duration

(86 samples at sampling rate $1,000\ Hz$)

extracted from the first beat and used as a template.

Template matching using normalized correlation coefficient:

$$\gamma_{xy}(k) = \frac{\Sigma_{n=0}^{N-1} [x(n) - \bar{x}][y(k - N + 1 + n) - \overline{y_k}]}{\sqrt{\Sigma_{n=0}^{N-1} [x(n) - \bar{x}]^2 \, \Sigma_{n=0}^{N-1} [y(k - N + 1 + n) - \overline{y_k}]^2}} \,, \tag{3.96}$$

$$\overline{y_k} = \frac{1}{N} \sum_{n=0}^{N-1} y(k - N + 1 + n),$$

$\overline{y_k}$: average of part of y used in template matching.

Operation is causal, valid from $k = N$ to last sample of y.

x: template; $\bar{x}$: average value of x.

k: time index of y at which template is placed.

Averaging inherent in cross-correlation (over N samples)

reduces effect of noise on template matching.

Appropriate threshold: obtain trigger point

to extract QRS locations in the ECG.

Threshold of 0.9: all 12 beats detected.

Figure 3.43: two ECG cycles extracted

and result of averaging the first 11 cycles in the signal.

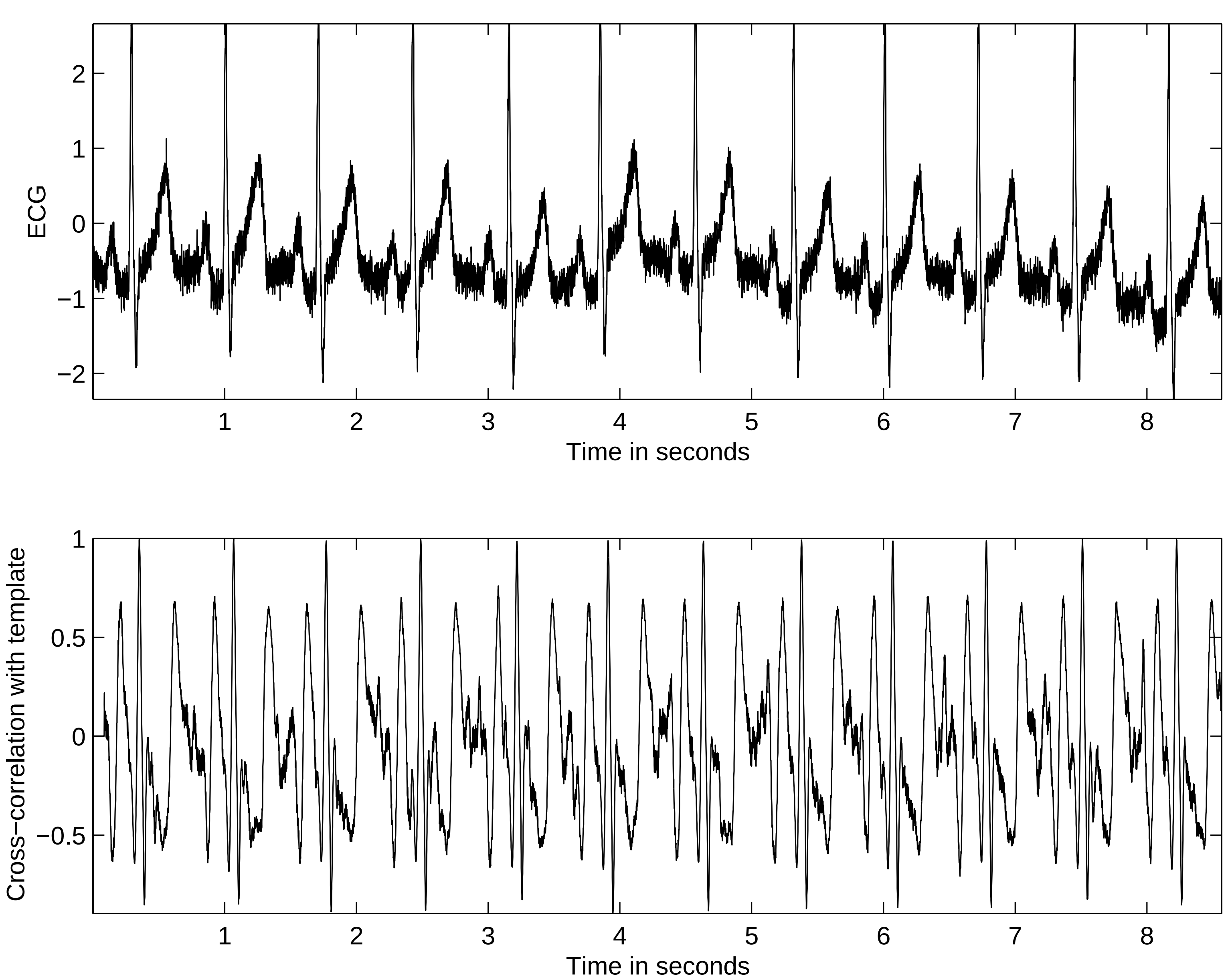


Figure 3.42: An ECG signal with noise (upper trace) and the result of cross-correlation (lower trace) with the QRS template selected from the first cycle. The cross-correlation coefficient is normalized to the range $(-1, 1)$.

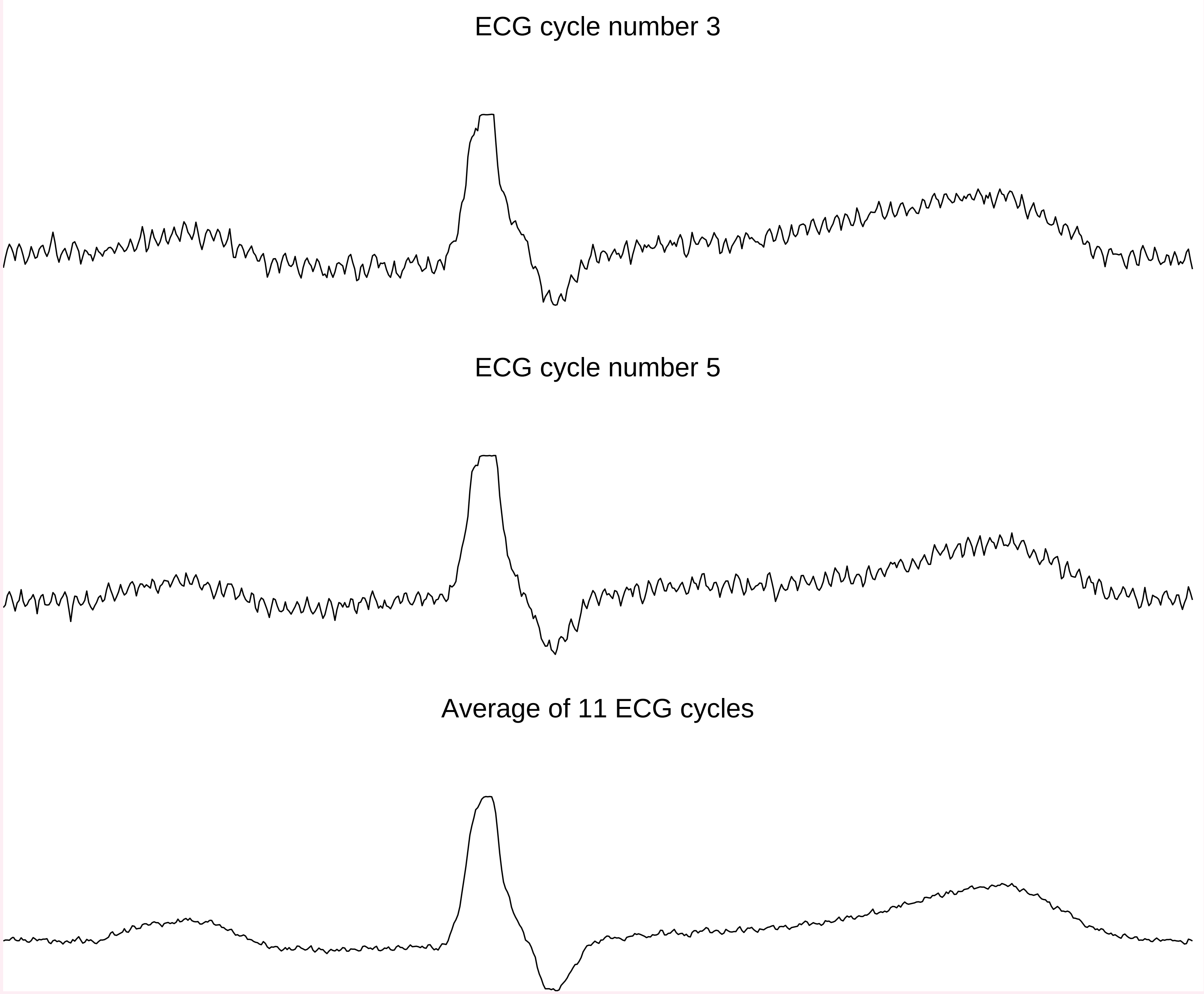


Figure 3.43: Upper two traces: two cycles of the ECG extracted from the signal in Figure 3.42. Bottom trace: the result of synchronized averaging of 11 cycles from the same ECG signal.

3.5.2 Moving-average filters

Problem:

Propose a time-domain technique to remove random noise given only one realization of the signal or event of interest.

Solution:

Ensemble of several realizations not available — synchronized averaging not possible.

Consider temporal averaging for noise removal.

Assumption: processes are ergodic.

Temporal statistics used instead of ensemble statistics.

Temporal window of samples moved to obtain

output at various points of time: moving-window averaging

or moving-average (MA) filter.

Average $\sim$ weighted combination of samples.

General form of MA filter:

$$y(n) = \sum_{k=0}^{N} b_k \, x(n-k), \tag{3.97}$$

x and y: input and output of filter.

b_k: filter coefficients or tap weights.

N: order of filter.

Effect of division by the number of samples used $(N+1)$

included in the values of the filter coefficients.

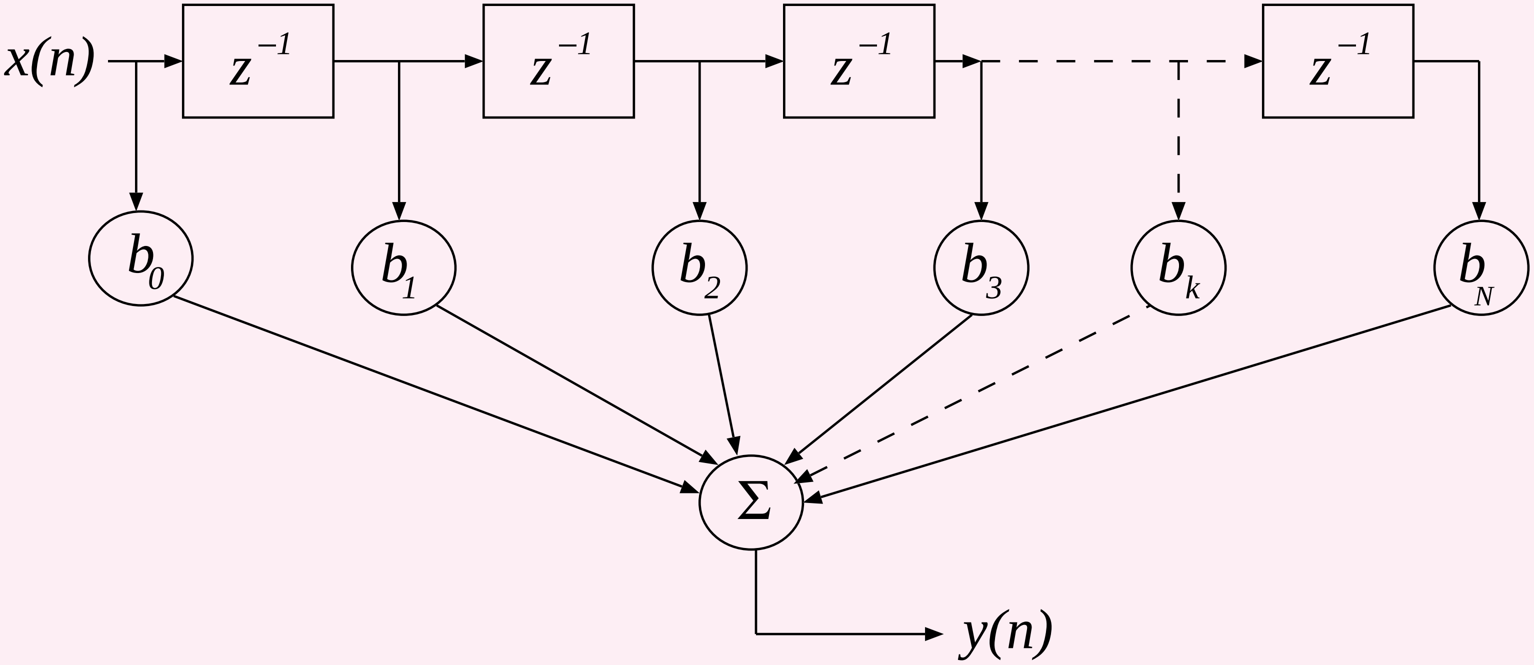


Figure 3.44: Signal-flow diagram of a moving-average filter of order N. Each block with the symbol z^{-1} represents a delay of one sample, and serves as a memory unit for the corresponding signal sample value.

Applying the z-transform, we get the transfer function:

$$H(z) = \frac{Y(z)}{X(z)} = \sum_{k=0}^{N} b_k\, z^{-k} = b_0 + b_1 z^{-1} + b_2 z^{-2} + \cdots + b_N z^{-N}, \tag{3.98}$$

$X(z)$ and $Y(z)$: z-transforms of $x(n)$ and $y(n)$.

MA filter to remove noise — von Hann or Hanning filter:

$$y(n) = \frac{1}{4}[x(n) + 2x(n-1) + x(n-2)]. \tag{3.99}$$

Impulse response: let $x(n) = \delta(n)$.

$$h(n) = \frac{1}{4}[\delta(n) + 2\delta(n-1) + \delta(n-2)]. \tag{3.100}$$

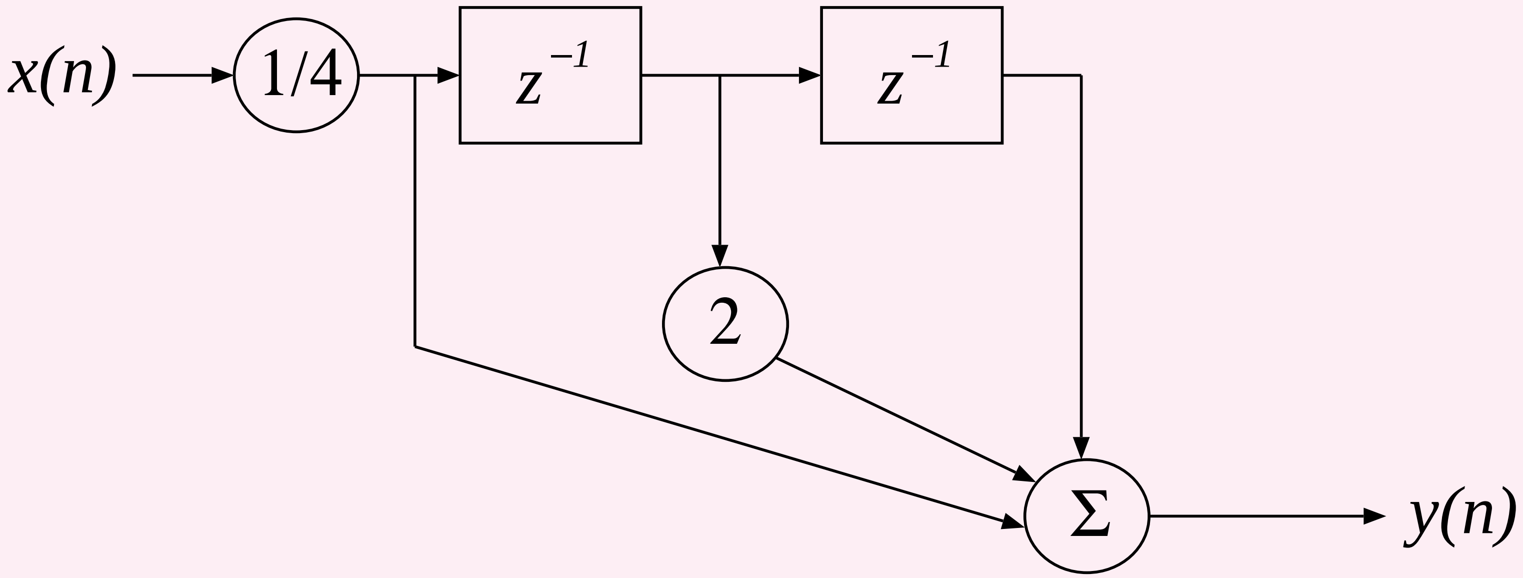


Figure 3.45: Signal-flow diagram of the Hann filter.

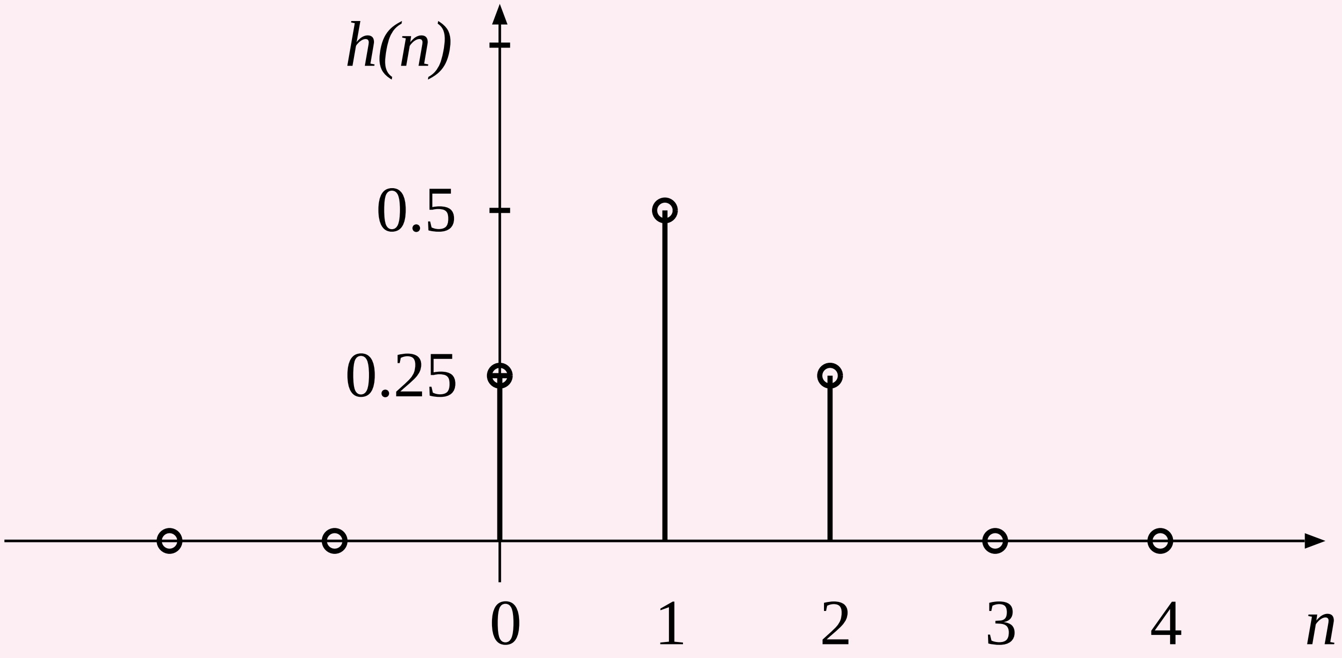


Figure 3.46: Impulse response of the Hann filter.

Applying the z-transform to Equation 3.99, we get

$$Y(z) = \frac{1}{4}[X(z) + 2\,z^{-1}X(z) + z^{-2}X(z)]. \qquad (3.101)$$

Transfer function of the Hann filter:

$$H(z) = \frac{1}{4}[1 + 2z^{-1} + z^{-2}] = \frac{1}{4}[1 + z^{-1}]^{2}. \qquad (3.102)$$

Double-zero at $z = -1$.

An MA filter is a finite impulse response (FIR) filter:

- Impulse response $h(k)$ has a finite number of terms: $h(k) = b_k,\ k = 0, 1, 2, \ldots, N$.
- An FIR filter may be realized nonrecursively with no feedback.
- Output depends only on the present input sample and a few past input samples.

- Filter is a set of tap weights of the delay stages.
- Transfer function has no poles except at $z = 0$: the filter is inherently stable.
- Filter has linear phase if the series of tap weights is symmetric or antisymmetric.

Note: A recursive filter uses previous values of the output to compute the current output value.

Frequency response: substitute $z = e^{j\omega T}$ in $H(z)$,

T: sampling interval in seconds, $T = 1/f_s$,

f: frequency in Hz,

f_s: sampling frequency,

ω: radian frequency, $\omega = 2\pi f$.

Set $T = 1$ and deal with normalized frequency in the range

$0 \leq \omega \leq 2\pi$ or $0 \leq f \leq 1$;

then $f = 1$ or $\omega = 2\pi$ represents the sampling frequency,

lower frequency values represented as

normalized fraction of f_s.

Frequency response of the Hann filter:

Frequency response of the Hann filter:

$$H(\omega) = H(z)|_{z=\exp(j\omega)} = \frac{1}{4}[1 + 2e^{-j\omega} + e^{-j2\omega}]. \quad (3.103)$$

$$\begin{aligned} H(\omega) &= \frac{1}{4}[1 + 2e^{-j\omega} + e^{-j2\omega}] \\ &= \frac{1}{4}\, e^{-j\omega}\, [e^{+j\omega} + 2 + e^{-j\omega}] \\ &= \frac{1}{4}\, e^{-j\omega}\, [2\cos(\omega) + 2] \\ &= \frac{1}{2}\, [1 + \cos(\omega)]\, e^{-j\omega}. \end{aligned} \quad (3.104)$$

Note: $e^{\pm j\omega} = \cos(\omega) \pm j\sin(\omega)$,
$e^{+j\omega} + e^{-j\omega} = 2\cos(\omega)$,
$e^{+j\omega} - e^{-j\omega} = 2j\sin(\omega)$.

Magnitude and phase responses:

$$|H(\omega)| = \left|\frac{1}{2}\{1 + \cos(\omega)\}\right| \tag{3.105}$$

$$\angle H(\omega) = -\omega. \tag{3.106}$$

Lowpass filter with linear phase.

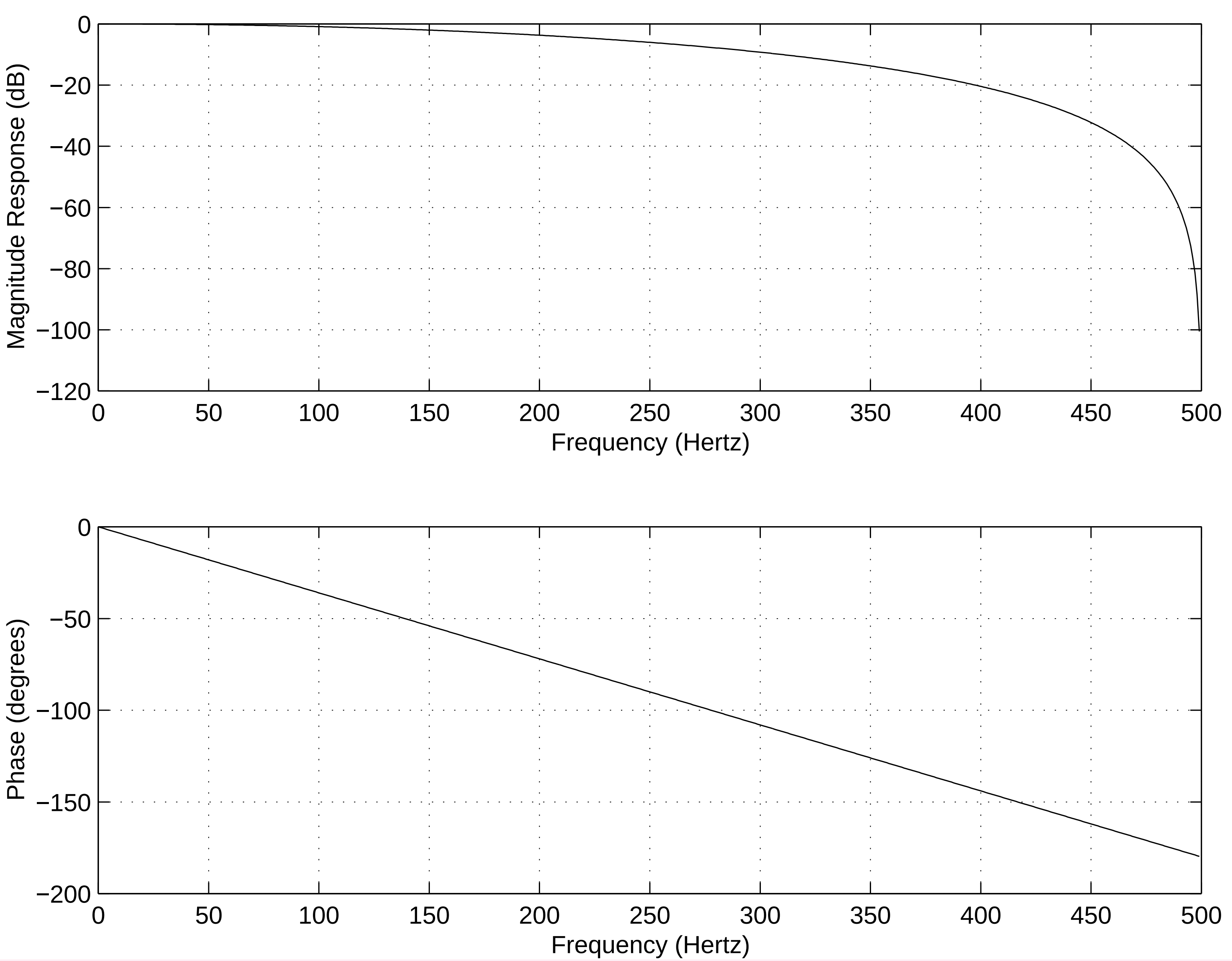


Figure 3.47: Magnitude and phase responses of the Hann (smoothing) filter.

For more smoothing, average signal samples over longer time windows; causes increased filter delay.

$$\begin{aligned} y(n) &= \frac{1}{8}\sum_{k=0}^{7} x(n-k) \\ &= \frac{1}{8}[x(n) + x(n-1) + x(n-2) + x(n-3) + x(n-4) \\ &+ x(n-5) + x(n-6) + x(n-7)]. \end{aligned} \tag{3.107}$$

$$\begin{aligned} h(n) &= \frac{1}{8}[\delta(n) + \delta(n-1) + \delta(n-2) + \delta(n-3) + \delta(n-4) \\ &+ \delta(n-5) + \delta(n-6) + \delta(n-7)]. \end{aligned} \tag{3.108}$$

Transfer function:

$$H(z) = \frac{1}{8} \sum_{k=0}^{7} z^{-k}. \qquad (3.109)$$

Frequency response $H(\omega)$

$$= \frac{1}{8} \sum_{k=0}^{7} \exp(-j\omega k)$$

$$= \frac{1}{4} \exp(-j7\omega/2) \qquad (3.110)$$

$$\times \; \{\cos(7\omega/2) + \cos(5\omega/2) + \cos(3\omega/2) + \cos(\omega/2)\}.$$

Zeros at $\frac{f_s}{8} = 125\ Hz$, $\frac{f_s}{4} = 250\ Hz$, $\frac{3f_s}{8} = 375\ Hz$,

and $\frac{f_s}{2} = 500\ Hz$.

More attenuation over $90 - 400\ Hz$ than the Hann filter.

Attenuation after $100\ Hz$ is nonuniform.

Phase response not linear, but piecewise linear.

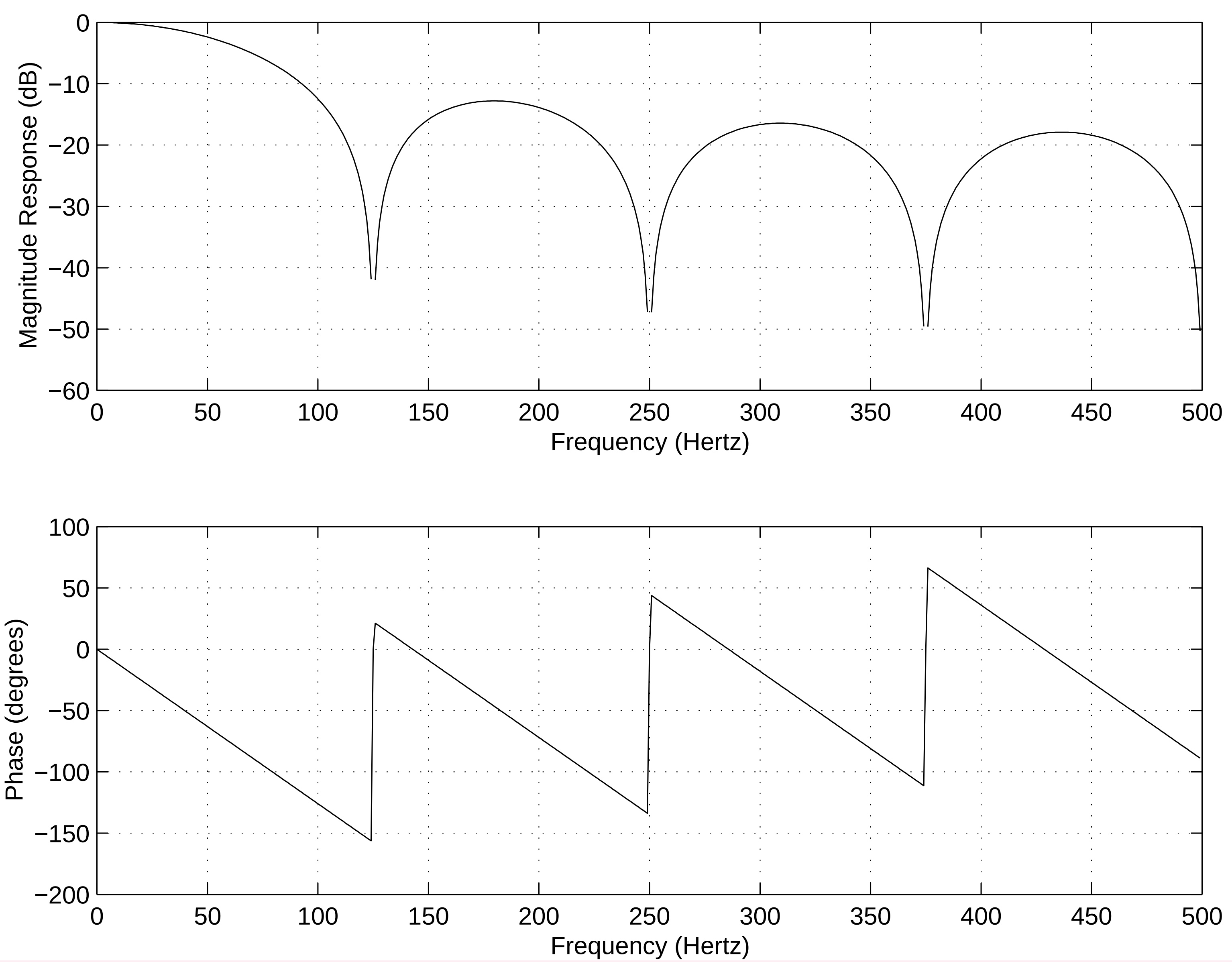


Figure 3.48: Magnitude and phase responses of the 8-point MA (smoothing) filter.

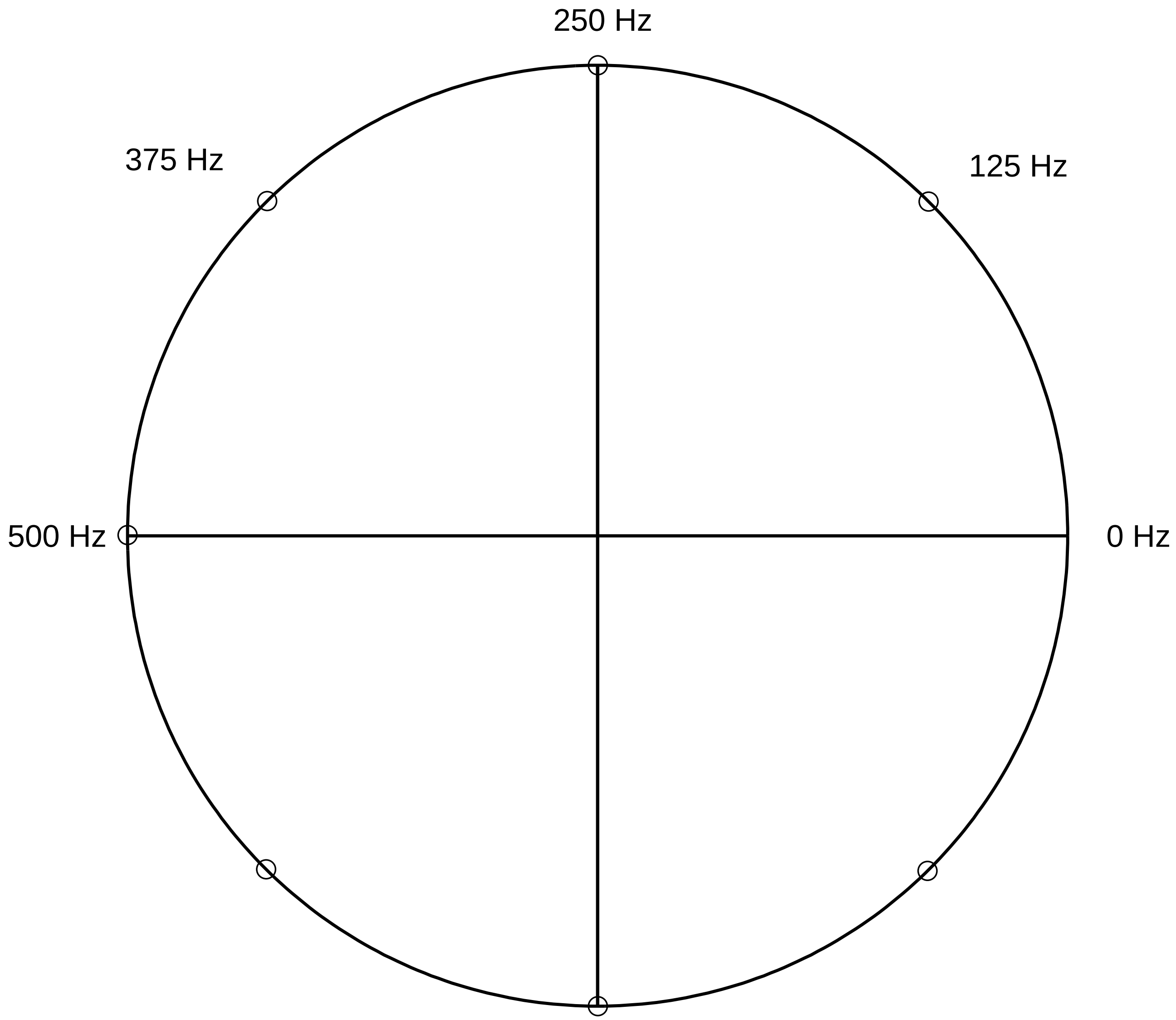


Figure 3.49: Pole-zero plot of the 8-point MA (smoothing) filter.

Relationship of MA filtering to integration:

Disregarding the $\frac{1}{8}$ scale factor,

the operation in Equation 3.107 may be interpreted as the

summation or integration of the signal from $n - 7$ to n.

Comparable integration of a continuous-time signal

$x(t)$ over the interval t_1 to t_2:

$$y(t) = \int_{t_1}^{t_2} x(t)\, dt. \tag{3.111}$$

General definition of the integral of a signal:

$$y(t) = \int_{-\infty}^{t} x(t)\, dt, \tag{3.112}$$

or, if the signal is causal,

$$y(t) = \int_{0}^{t} x(t)\, dt. \tag{3.113}$$

Fourier transforms of the signals:

$$Y(\omega) = \frac{1}{j\omega} X(\omega) + \pi X(0) \delta(\omega). \tag{3.114}$$

Frequency response of the integration operator:

$$H(\omega) = \frac{1}{j\omega}. \tag{3.115}$$

Magnitude and phase responses:

$$|H(\omega)| = \left|\frac{1}{\omega}\right|, \tag{3.116}$$

$$\angle H(\omega) = -\frac{\pi}{2}. \tag{3.117}$$

Gain of the filter reduces nonlinearly as frequency increases: filter has lowpass characteristics.

Integration or accumulation of a discrete-time signal

for all samples up to the present sample

results in the transfer function $H(z) = \frac{1}{1 - z^{-1}}$.

Such an operation is seldom used in practice.

Instead, a moving-window sum is computed

as in Equation 3.107.

The 8-point MA filter may be rewritten as

$$\begin{aligned} y(n-1) &= \frac{1}{8}[x(n-1)+x(n-2)+x(n-3)+x(n-4) \\ &\quad + x(n-5)+x(n-6)+x(n-7)+x(n-8)]. \end{aligned} \tag{3.118}$$

$$y(n) = y(n-1) + \frac{1}{8}x(n) - \frac{1}{8}x(n-8). \tag{3.119}$$

The recursive form clearly depicts integration.

Transfer function:

$$H(z) = \frac{1}{8}\left[\frac{1-z^{-8}}{1-z^{-1}}\right]. \tag{3.120}$$

Frequency response:

$$H(\omega) = \frac{1}{8}\left[\frac{1 - e^{-j8\omega}}{1 - e^{-j\omega}}\right] = \frac{1}{8}\, e^{-j\frac{7}{2}\omega} \left[\frac{\sin(4\omega)}{\sin(\frac{\omega}{2})}\right], \qquad (3.121)$$

which is equivalent to that in Equation 3.110.

Summation over a limited discrete-time window results in a frequency response having sinc-type characteristics;

Figure 3.48.

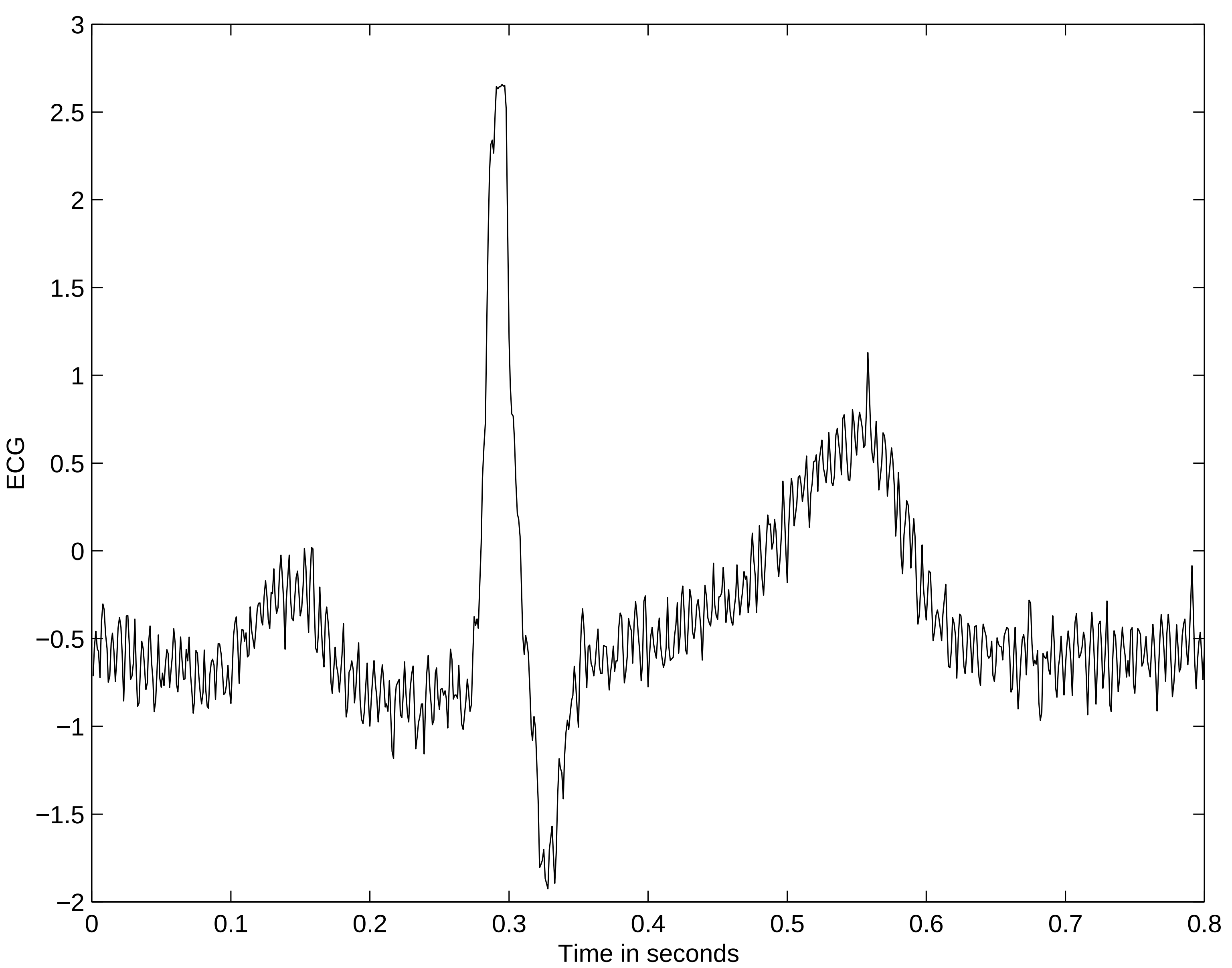


Figure 3.50: ECG signal with high-frequency noise; $f_s = 1,000\ Hz$.

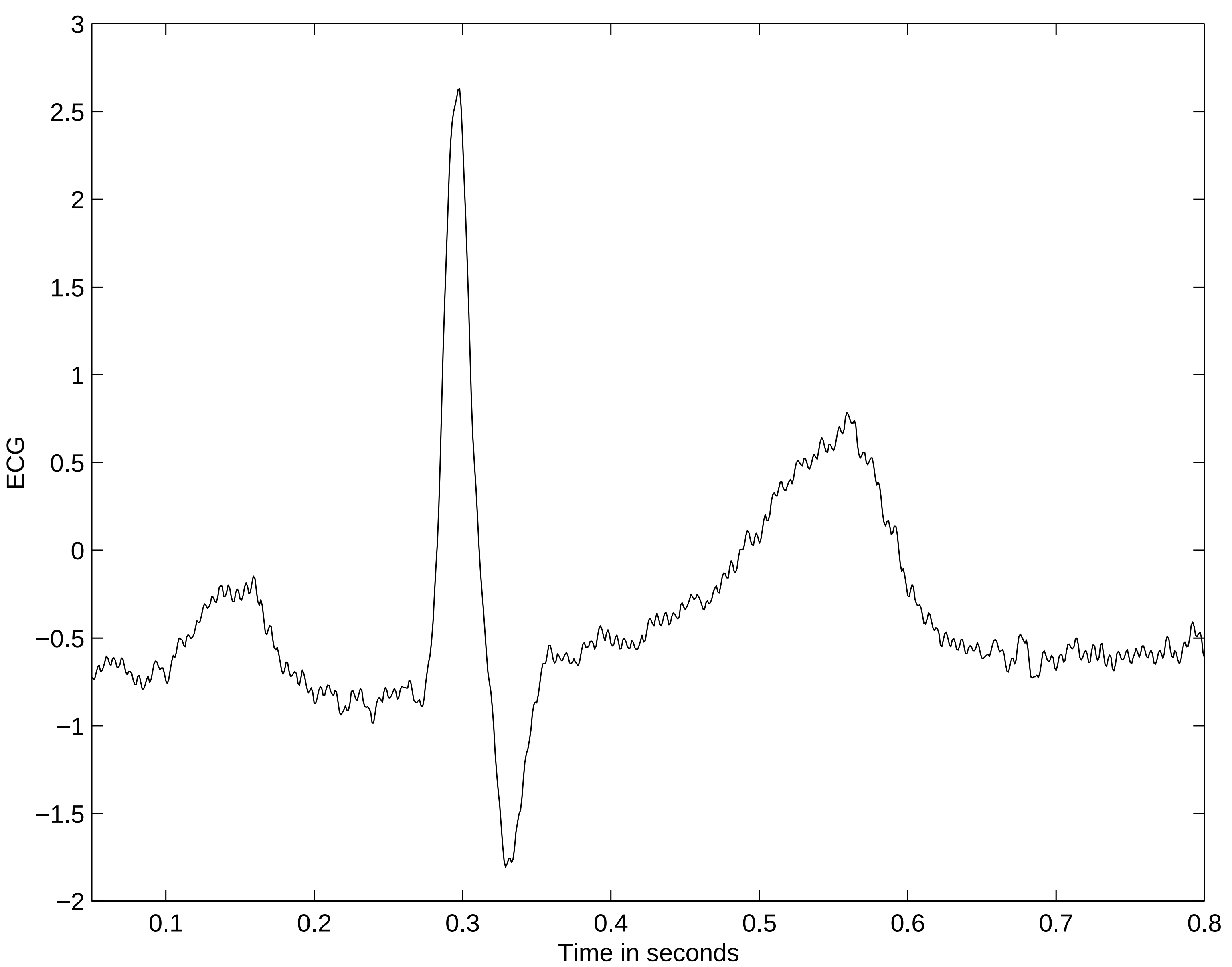


Figure 3.51: The ECG signal with high-frequency noise in Figure 3.50 after filtering by the 8-point MA filter shown in Figure 3.48.

3.5.3 Derivative-based operators to remove low-frequency artifacts

Problem: *Develop a time-domain technique to remove baseline drift in the ECG signal.*

Solution: The derivative operator in the time domain removes parts of the input that are constant.

Large changes in input → high values in output.

Ideal $\frac{d}{dt}$ operator in the time domain

results in multiplication of FT of original signal by

$j\,\omega = j\,2\pi\,f$ in the frequency domain.

If $X(f)$ = FT $[x(t)]$,

FT $\left[\frac{dx}{dt}\right]$ = $j\,2\pi\,f\,X(f)$ = $j\,\omega X(\omega)$.

Frequency response of the operation: $H(\omega) = j\,\omega$.

Gain increases linearly with frequency.

$H(0) = 0$:

DC component removed by the derivative operator.

Higher frequencies receive linearly increasing gain:

highpass filter.

The derivative operator may be used to remove DC,

suppress low-frequency components,

and boost high-frequency components.

Second-order derivative operator $\frac{d^2}{dt^2}$:

frequency response $H(\omega) = -\omega^2$,

quadratic increase in gain for higher frequencies.

In DSP, basic derivative = first-order difference:

$$y(n) = \frac{1}{T}\left[x(n) - x(n-1)\right]. \tag{3.122}$$

Scale factor including the sampling interval T required to obtain rate of change of signal with respect to true time.

Transfer function:

$$H(z) = \frac{1}{T}\left(1 - z^{-1}\right). \tag{3.123}$$

Zero at $z = 1$, the DC point.

Frequency response:

$$H(\omega) = \frac{1}{T}\left[1 - \exp(-j\omega)\right] = \frac{1}{T}\, j \, \exp\left(-j\frac{\omega}{2}\right)\left[2\sin\left(\frac{\omega}{2}\right)\right], \tag{3.124}$$

$$|H(\omega)| = \frac{2}{T}\left|\sin\left(\frac{\omega}{2}\right)\right|, \tag{3.125}$$

$$\angle H(\omega) = \frac{\pi}{2} - \frac{\omega}{2}. \tag{3.126}$$

High-frequency noise amplified significantly: noisy result.

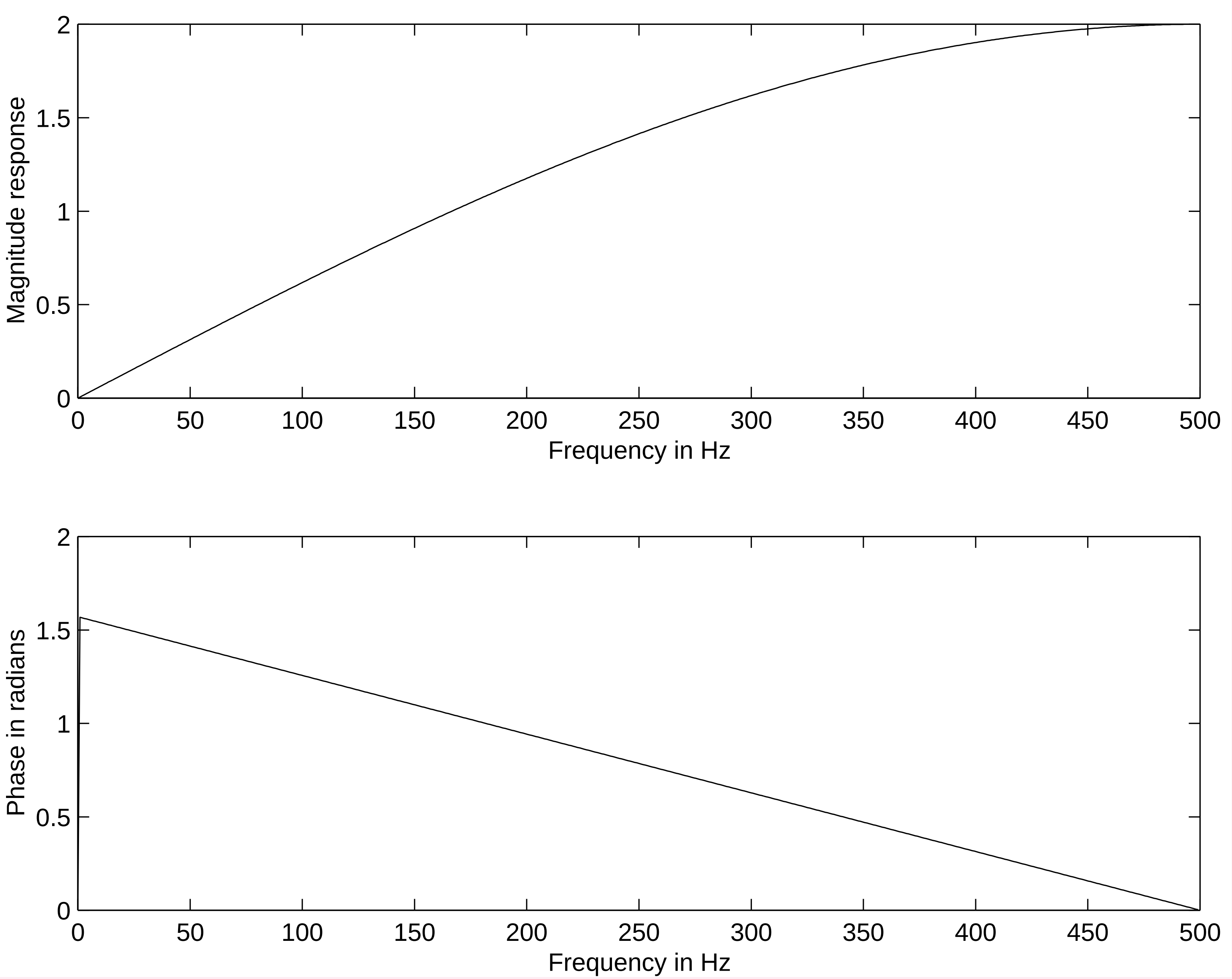


Figure 3.52: Magnitude and phase responses of the first-order difference operator. The magnitude response is shown on a linear scale in order to illustrate better its proportionality to frequency.

Three-point central difference:

Noise-amplification problem with the first-order difference operator in Equation 3.122 may be controlled by taking the average of two successive output values:

$$\begin{aligned} y_3(n) &= \frac{1}{2}\,[y(n) + y(n-1)] \\ &= \frac{1}{2T}\,[\{x(n) - x(n-1)\} + \{x(n-1) - x(n-2)\}] \\ &= \frac{1}{2T}\,[x(n) - x(n-2)]. \end{aligned} \tag{3.127}$$

$$H(z) = \frac{1}{2T}\left(1 - z^{-2}\right) = \left[\frac{1}{T}\left(1 - z^{-1}\right)\right] \left[\frac{1}{2}\left(1 + z^{-1}\right)\right]. \tag{3.128}$$

$$|H(\omega)| = \frac{1}{T}\left|\sin\left(\omega\right)\right|. \tag{3.129}$$

$$\angle H(\omega) = \frac{\pi}{2} - \omega. \tag{3.130}$$

Transfer function of three-point central-difference

= two filters in series (cascade)

= product of the transfer functions of the

first-order difference operator and a two-point MA filter.

Zeros at $z = 1$ and $z = -1$: bandpass filter.

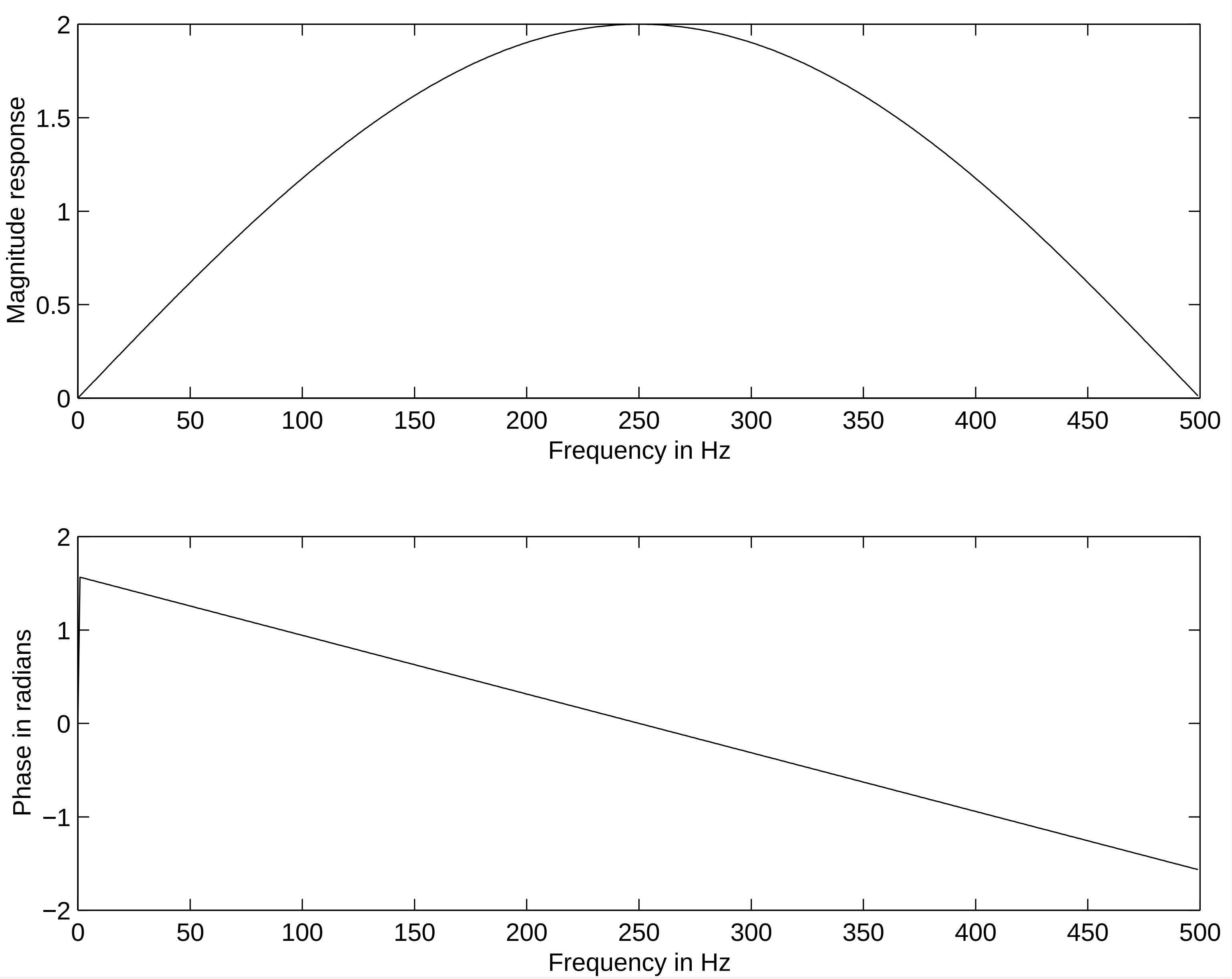


Figure 3.53: Magnitude and phase responses of the three-point central-difference operator. The magnitude response is shown on a linear scale.

Illustration of application: Figures 3.54 and 3.55.

Filtering the ECG signal with low-frequency noise in Figure 3.6, using first-order difference and three-point central difference.

Baseline drift has been removed, but the filters have also removed the slow P and T waves, and altered the QRS complexes: results unlike ECG signals.

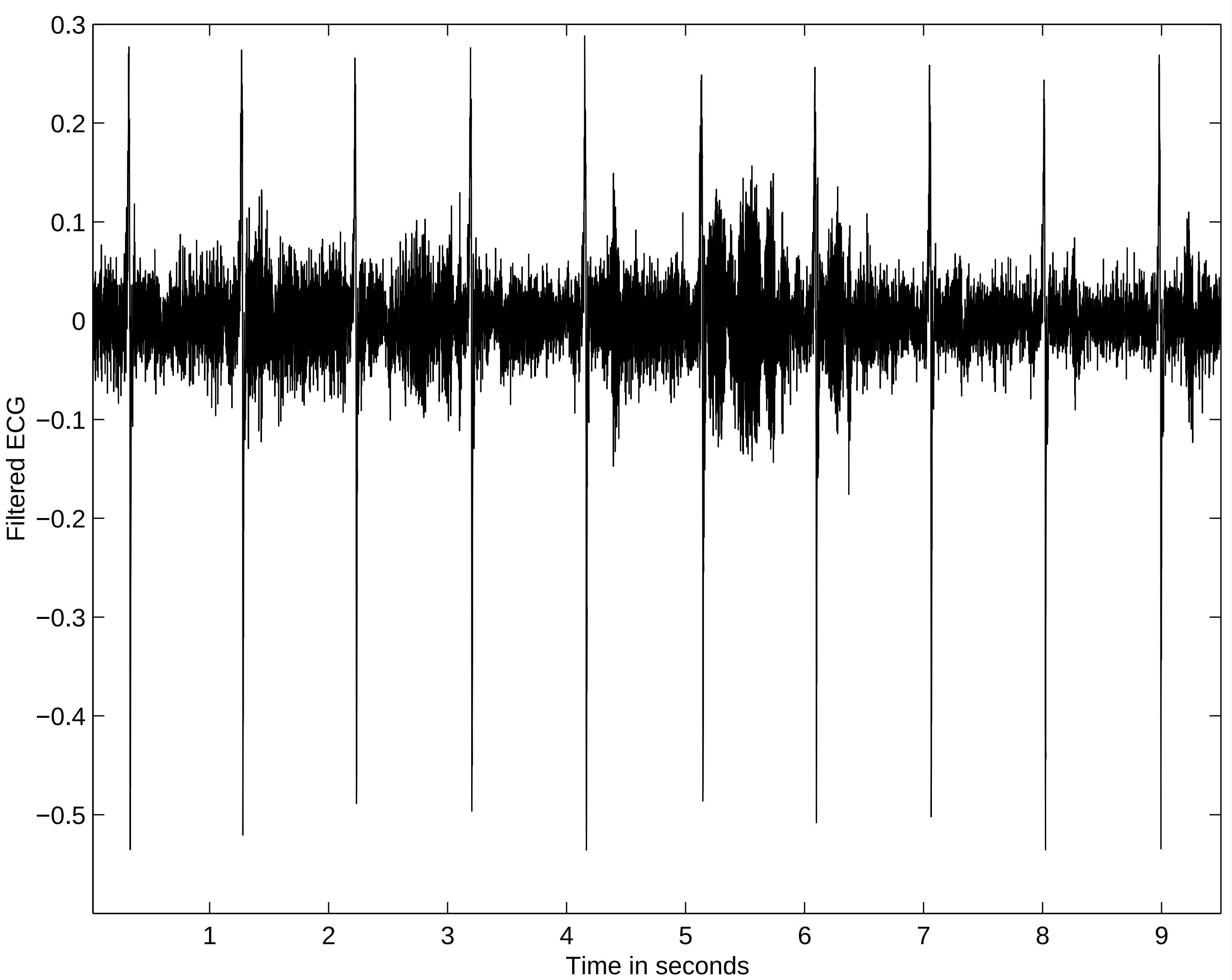


Figure 3.54: Result of filtering the ECG signal with low-frequency noise shown in Figure 3.6, using the first-order difference operator.

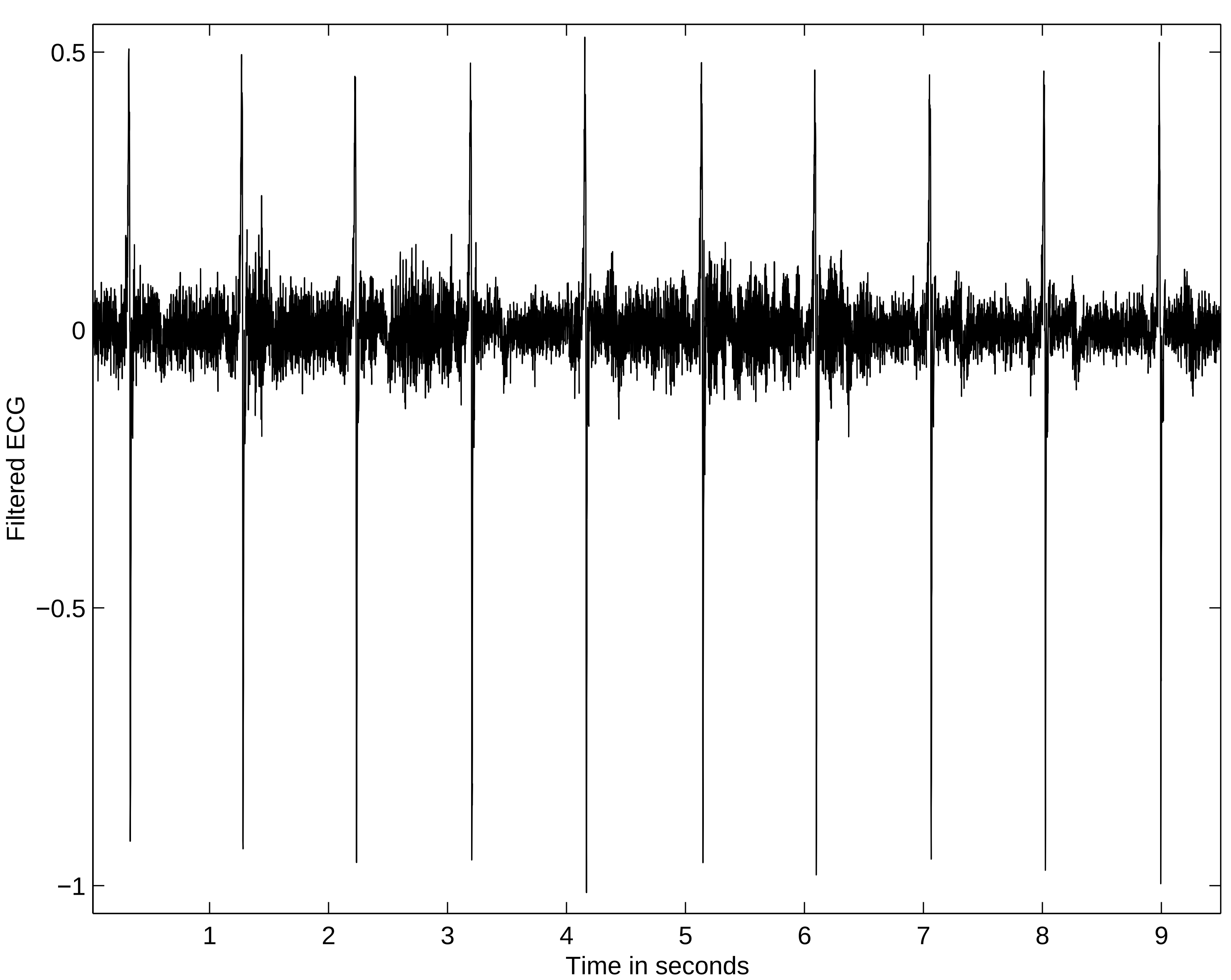


Figure 3.55: Result of filtering the ECG signal with low-frequency noise shown in Figure 3.6, using the three-point central-difference operator.

Problem: *How could we improve the performance of the basic first-order difference operator as a filter to remove low-frequency noise or baseline wander without distorting the QRS complex?*

Solution: Drawback of the first-order difference and the three-point central-difference operators — magnitude response remains low for a wide range of frequencies.

Need gain of filter to be close to unity after about $0.5\ Hz$.

The gain of a filter at specific frequencies may be boosted

by placing poles at related locations around the unit circle.

Stability: poles should be within the unit circle.

To maintain high gain at low frequencies,

place a pole on the real axis (zero frequency),

close to $z = 1$, e.g., $z = 0.995$.

$$H(z) = \frac{1}{T}\left[\frac{1 - z^{-1}}{1 - 0.995\ z^{-1}}\right], \tag{3.131}$$

$$H(z) = \frac{1}{T}\left[\frac{z - 1}{z - 0.995}\right]. \tag{3.132}$$

$$y(n) = \frac{1}{T}\,[x(n) - x(n-1)] + 0.995\ y(n-1). \tag{3.133}$$

Infinite impulse response (IIR) filter!

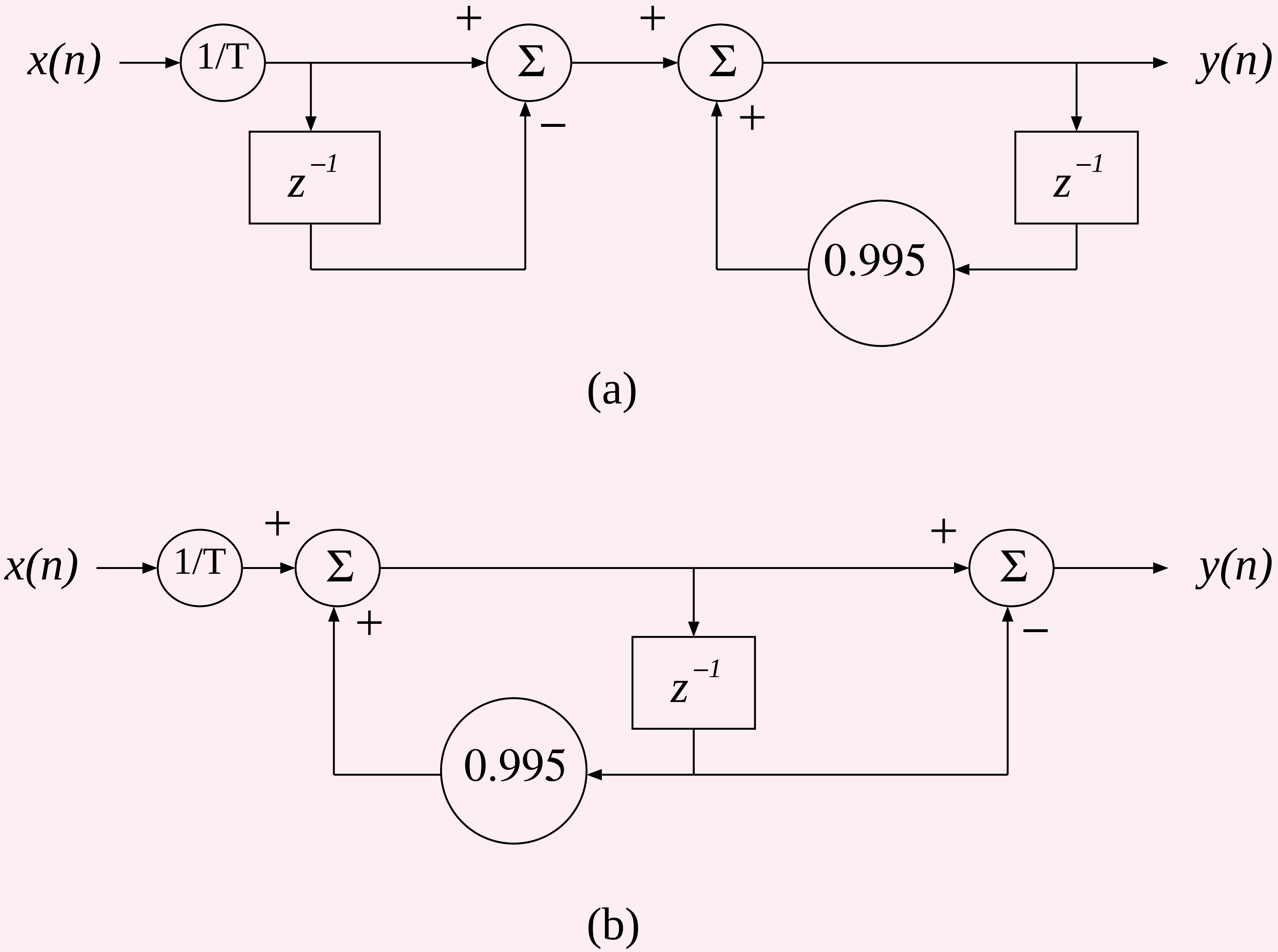


Figure 3.56: Two equivalent signal-flow diagrams of the filter to remove low-frequency noise or baseline wander. The form in (a) uses two delays, whereas that in (b) uses only one delay.

Graphical method for evaluation of response:

Evaluate transfer function at various points on the unit circle

in the z-plane, by letting $z = \exp(j\omega)$,

and evaluating $H(z)$ for various values of the frequency.

Magnitude transfer function $|H(z)| =$

product of the distances from z to all zeros of the system

divided by the product of the distances to the poles.

Phase response =

sum of the angles of the vectors from z to all the zeros

minus the sum of the angles to the poles.

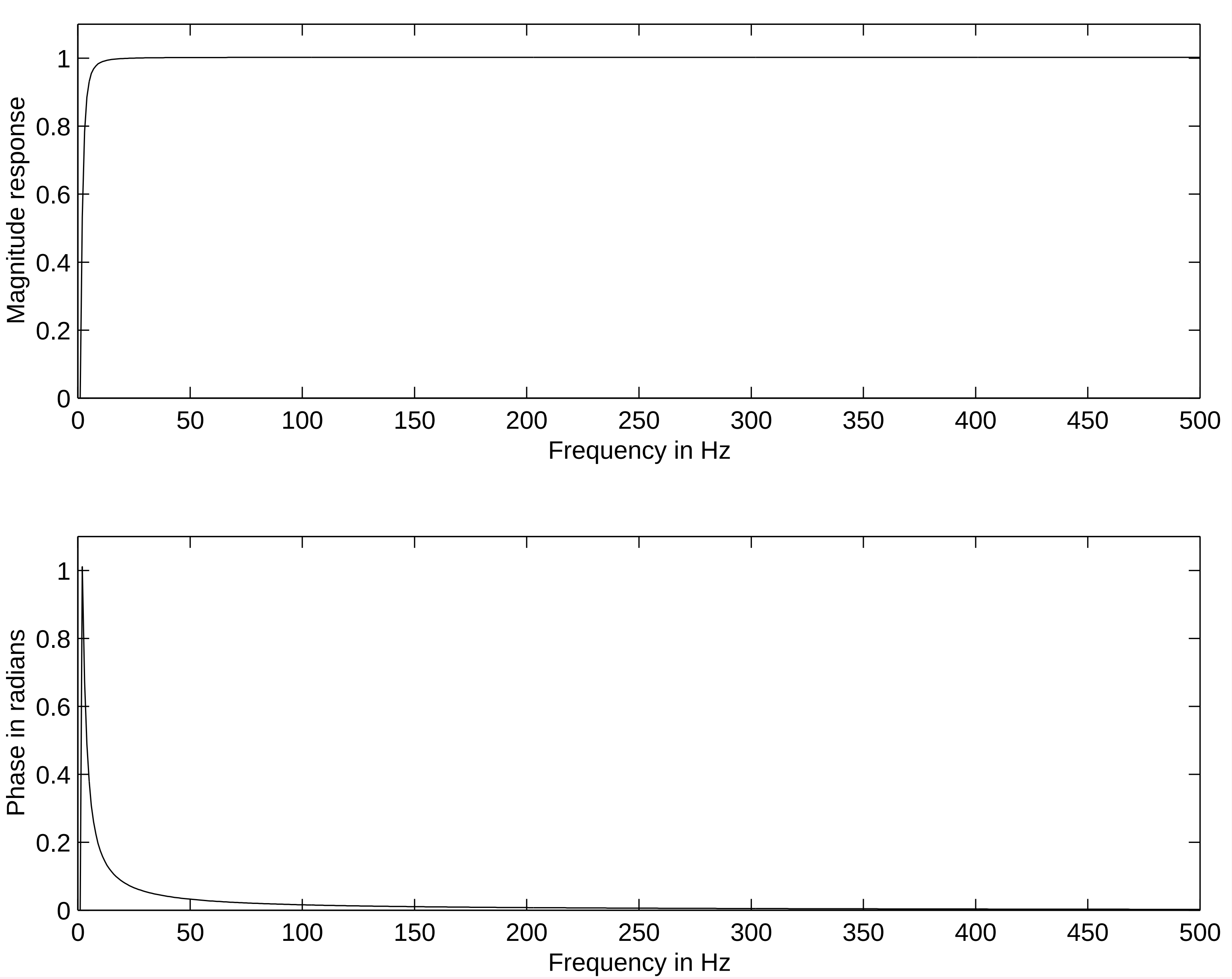


Figure 3.57: Normalized magnitude and phase responses of the filter to remove baseline wander as in Equation 3.131. The magnitude response is shown on a linear scale.

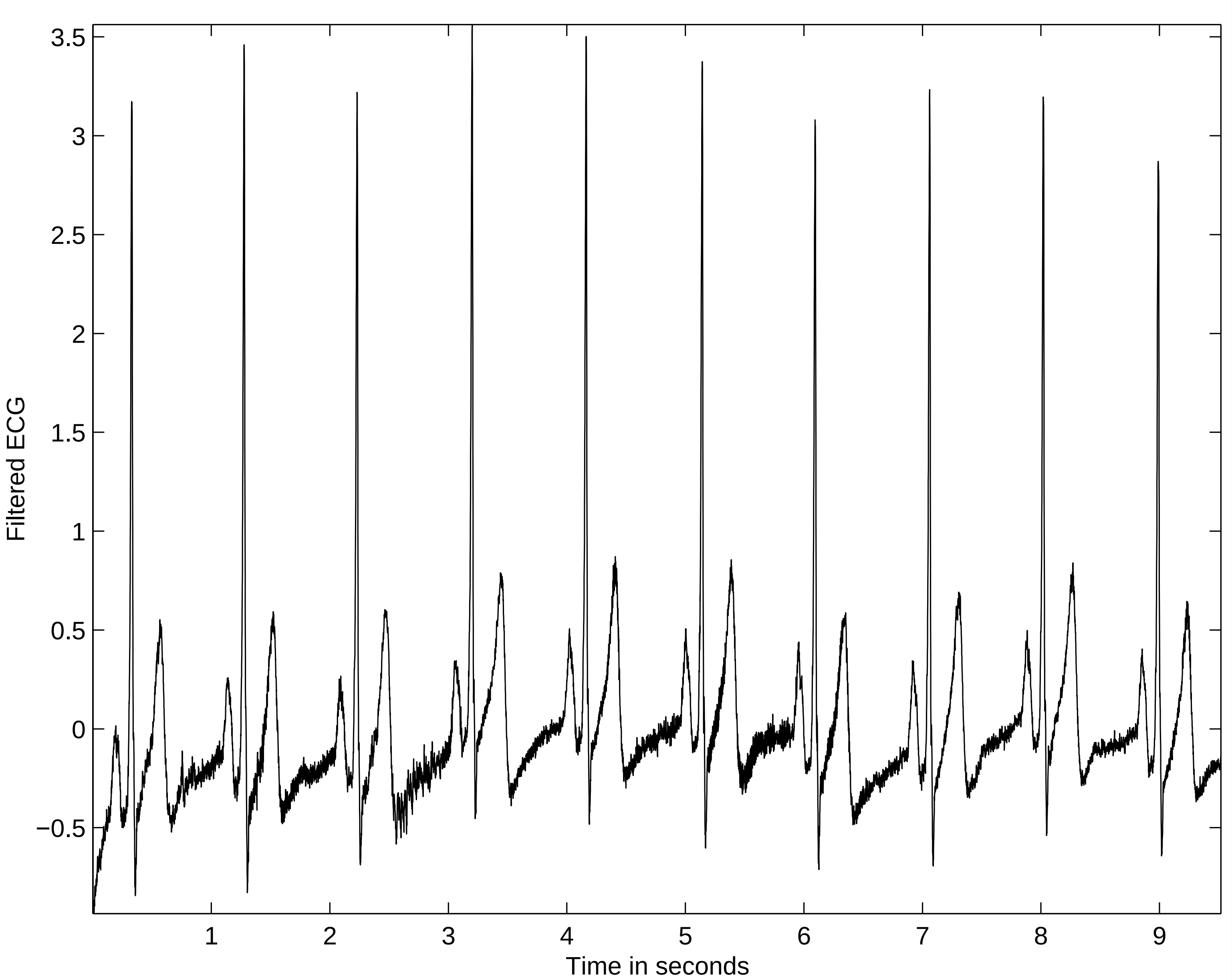


Figure 3.58: Result of processing the ECG signal with low-frequency noise shown in Figure 3.6, using the filter to remove baseline wander as in Equation 3.131. (Compare with the results in Figures 3.54 and 3.55.)

3.5.4 Various specifications of a filter

- Difference equation.
- Signal-flow diagram.
- Tap-weight or filter coefficients.
- Impulse response, $h(n)$.
- Transfer function, $H(z)$.
- Frequency response, $H(\omega)$, magnitude and phase.
- Pole–zero diagram and a gain factor.

The items listed above are interrelated, and any one of them may be used to derive any of the others.

3.6 Frequency-domain Filters

Filters may be designed in the frequency domain to provide specific lowpass, highpass, bandpass, or band-reject (notch) characteristics.

Implemented in software after obtaining FT of input signal, or converted into equivalent time-domain filters.

Most commonly used designs:

Butterworth, Chebyshev (Tschebyscheff), elliptic, Bessel.

Well-established in the analog-filter domain:

commence with an analog design $H(s)$ and apply the

bilinear transformation to obtain a digital filter $H(z)$.

Design a lowpass filter with the desired passband, transition, and stopband characteristics on a normalized-frequency axis, and then transform to the desired lowpass, highpass, bandpass, or band-reject filter.

Frequency-domain filters may also be specified directly

in terms of the values of the desired frequency response

at certain frequency samples only,

then transformed into the equivalent time-domain

filter coefficients via the inverse Fourier transform.

3.6.1 Removal of high-frequency noise: Butterworth lowpass filters

Problem: *Design a frequency-domain filter to remove high-frequency noise with minimal loss of signal components in the specified passband.*

Solution: Butterworth filter — maximally flat magnitude response in the passband.

Butterworth lowpass filter of order N:

first $2N - 1$ derivatives of squared magnitude response

are zero at $\Omega = 0$,

where Ω = analog radian frequency.

Butterworth filter response is monotonic in the passband

as well as in the stopband.

Basic Butterworth lowpass filter function:

$$|H_a(j\Omega)|^2 = \frac{1}{1 + \left(\frac{j\Omega}{j\Omega_c}\right)^{2N}}, \tag{3.134}$$

where H_a is the frequency response of the analog filter and

Ω_c is the cutoff frequency in $radians/s$.

Butterworth filter completely specified by

cutoff frequency Ω_c and order N.

As the order N increases, the filter response becomes

more flat in the passband, and

transition to the stopband becomes faster or sharper.

$|H_a(j\Omega_c)|^2 = \frac{1}{2}$ for all N.

Changing to the Laplace variable s,

$$H_a(s) H_a(-s) = \frac{1}{1 + \left(\frac{s}{j\Omega_c} \right)^{2N}} \, . \tag{3.135}$$

Poles of squared transfer function located with

equal spacing around a circle of radius Ω_c in the s-plane,

distributed symmetrically on either side of the

imaginary axis $s = j\Omega$.

No pole on the imaginary axis;

poles on the real axis for odd N.

Angular spacing between poles is $\frac{\pi}{N}$.

If $H_a(s)H_a(-s)$ has a pole at $s = s_p$,

it will have a pole at $s = -s_p$ as well.

For the filter coefficients to be real,

complex poles must appear in conjugate pairs.

To obtain a stable and causal filter, form $H_a(s)$ with

only the N poles on the left-hand side of the s-plane.

Pole positions in the s-plane given by

$$s_k = \Omega_c \, \exp\left[j\pi \left(\frac{1}{2} + \frac{(2k-1)}{2N} \right) \right], \qquad (3.136)$$

$k = 1, 2, \ldots, 2N$.

After the pole positions are obtained in the s-plane,

derive the transfer function in the analog Laplace domain:

$$H_a(s) = \frac{G}{(s-p_1)(s-p_2)(s-p_3)\cdots(s-p_N)}, \quad (3.137)$$

where p_k, $k = 1, 2, \ldots, N$, are the N poles

in the left-half of the s-plane,

and G is a gain factor specified as needed

or to normalize the gain at DC ($s = 0$) to be unity.

Bilinear transformation (BLT):

$$s = \frac{2}{T}\left[\frac{1 - z^{-1}}{1 + z^{-1}}\right]. \tag{3.138}$$

Butterworth circle in the s-plane maps to a

circle in the z-plane with real-axis intercepts at

$z = \frac{2-\Omega_c T}{2+\Omega_c T}$ and $z = \frac{2+\Omega_c T}{2-\Omega_c T}$.

Poles at $s = s_p$ and $s = -s_p$ in the s-plane map to

$z = z_p$ and $z = 1/z_p$.

Poles in the z-plane not uniformly spaced around

the transformed Butterworth circle.

For stability, all poles of $H(z)$ must lie

within the unit circle in the z-plane.

Unit circle in the z-plane: $z = e^{j\omega}$.

For points on the unit circle, we have

$$s = \sigma + j\Omega = \frac{2}{T}\left(\frac{1 - e^{-j\omega}}{1 + e^{-j\omega}}\right) = \frac{2j}{T}\tan\left(\frac{\omega}{2}\right). \qquad (3.139)$$

Continuous-time (s-domain) frequency variable Ω

related to discrete-time (z-domain) frequency variable ω as

$$\Omega = \frac{2}{T} \tan\left(\frac{\omega}{2}\right) \tag{3.140}$$

and

$$\omega = 2\, \tan^{-1}\left(\frac{\Omega T}{2}\right). \tag{3.141}$$

Nonlinear relationship warps frequency values:

mapped from the imaginary (vertical) axis in the s-plane

to the unit circle in the z-plane —

should be taken into account in specifying

cutoff frequencies.

$H_a(s)$ mapped to the z-domain by the BLT: $s = \frac{2}{T}\frac{1-z^{-1}}{1+z^{-1}}$.

$H(z)$ simplified to

$$H(z) = \frac{G' \, (1 + z^{-1})^N}{\Sigma_{k=0}^{N} \; a_k \; z^{-k}} \,, \tag{3.142}$$

a_k, $k = 0, 1, 2, \ldots, N$: filter coefficients or tap weights;

$a_0 = 1$;

G': gain, normalized so that $|H(z)| = 1$ at DC ($z = 1$).

IIR filter; N zeros at $z = -1$ due to the BLT.

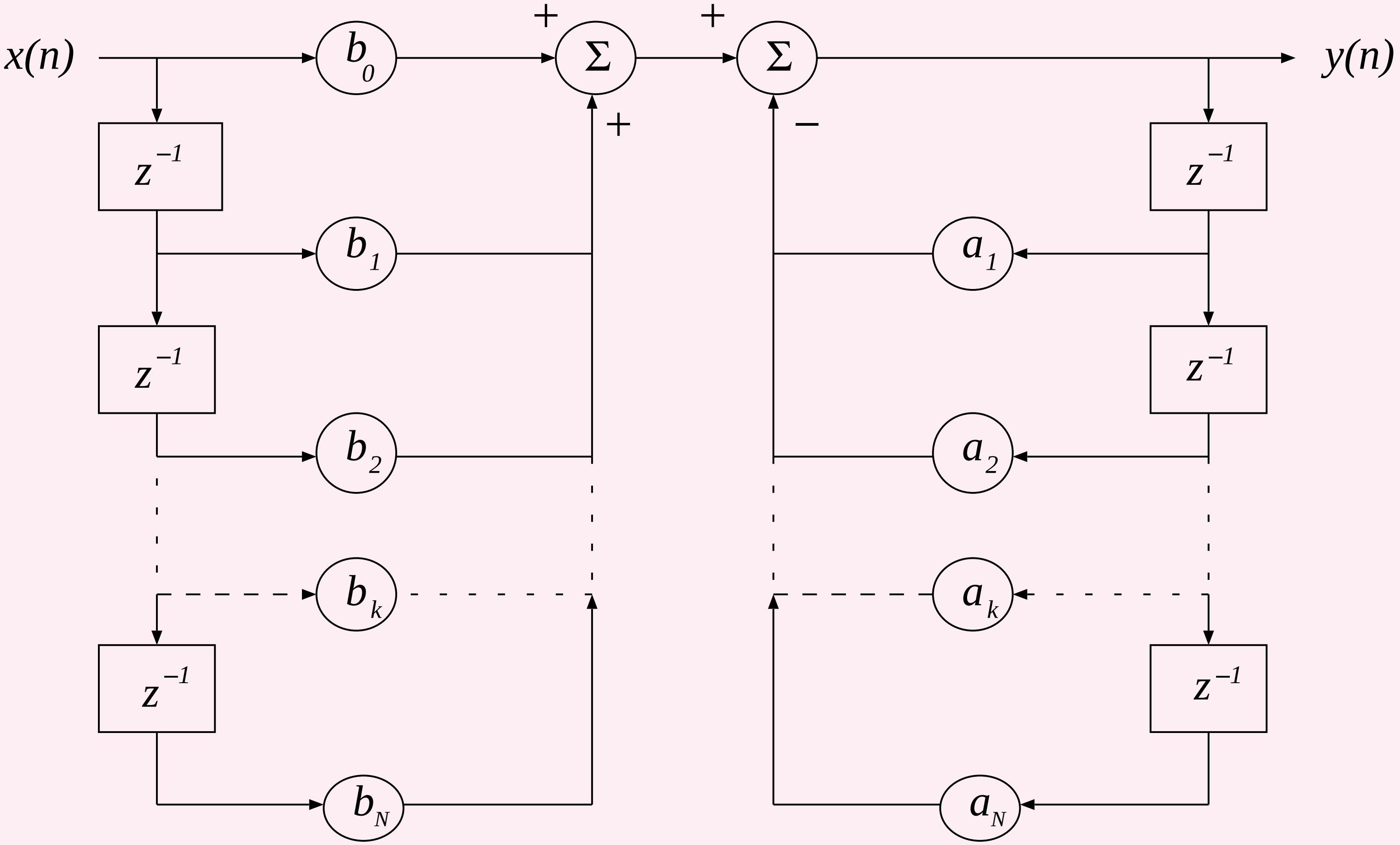


Figure 3.59: Signal-flow diagram of a direct realization of a generic infinite impulse response (IIR) filter. This form uses $2N$ delays and $2N + 1$ multipliers for a filter of order N.

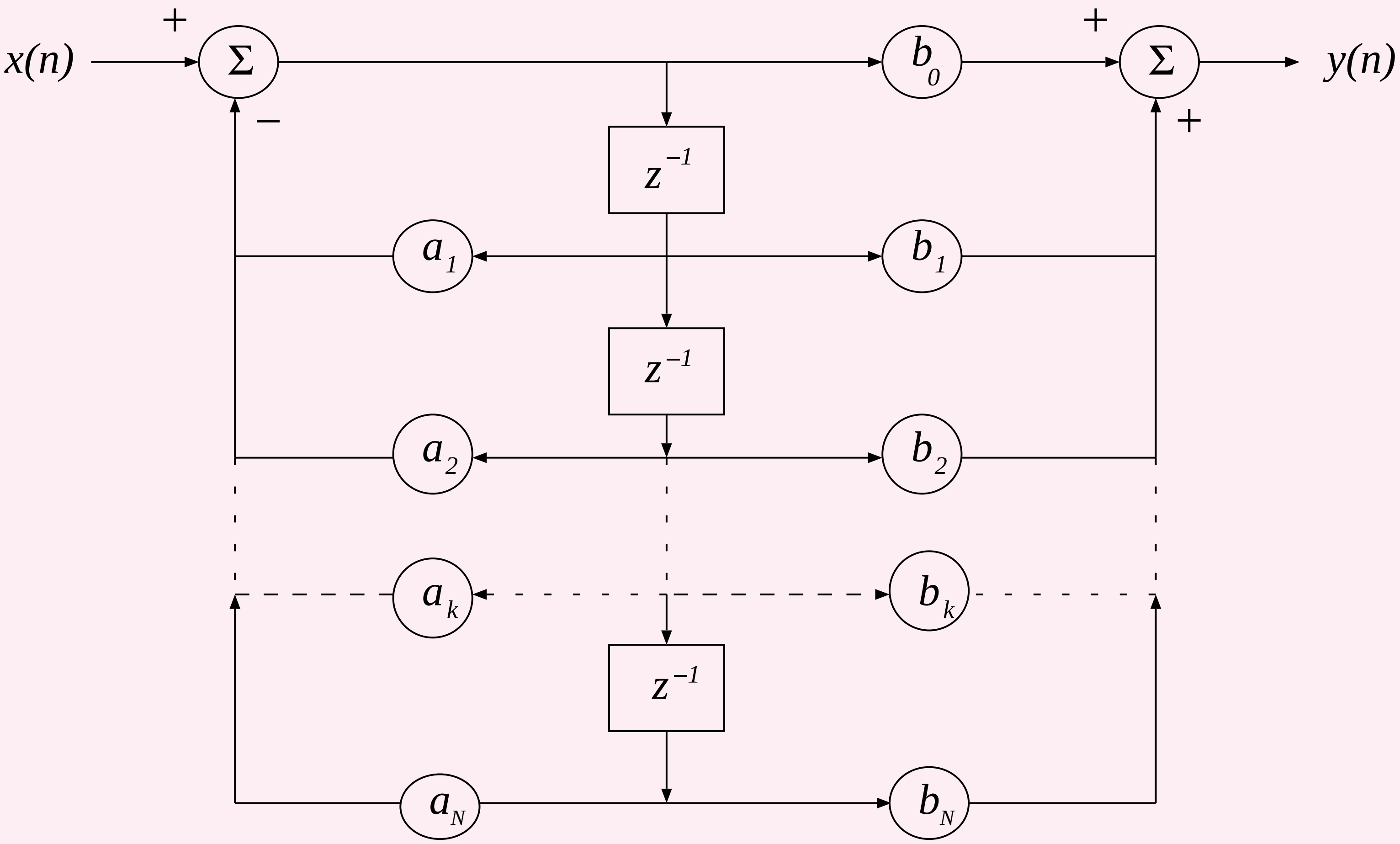


Figure 3.60: Signal-flow diagram of a realization of an IIR filter that uses only N delays and $(2N+1)$ multipliers for a filter of order N.

Time-domain representation required if

filter applied to data samples directly in time domain.

From transfer function $H(z)$ in Equation 3.142:

$$y(n) = \sum_{k=0}^{N} b_k \, x(n-k) - \sum_{k=1}^{N} a_k \, y(n-k). \qquad (3.143)$$

Coefficients b_k given by coefficients of expansion

of $G'(1+z^{-1})^N$.

MATLAB command *butter*.

Also possible to specify the Butterworth filter directly as

$$|H(\omega)|^2 = \frac{1}{1 + \left(\frac{\omega}{\omega_c}\right)^{2N}} \,, \tag{3.144}$$

ω normalized to $(0, 2\pi)$ for sampled signals;

equation valid only for the range $(0, \pi)$;

function over $(\pi, 2\pi)$ = reflection of function over $(0, \pi)$.

Cutoff frequency ω_c should be specified in the range $(0, \pi)$.

If the DFT is used:

$$|H(k)|^2 = \frac{1}{1 + \left(\frac{k}{k_c}\right)^{2N}}, \qquad (3.145)$$

k: index of DFT array for discretized frequency.

K: number of points in DFT array,

k_c index corresponding to cutoff frequency ω_c, $k_c = K \frac{\omega_c}{\omega_s}$;

Equation valid for $k = 0, 1, 2, \ldots, \frac{K}{2}$;

$H(k) = H(K - k)$, $k = \frac{K}{2} + 1, \ldots, K - 1$.

Note: DFT includes two unique values — DC in $H(0)$ and

folding-frequency component in $H(\frac{K}{2})$.

k: also used to represent normalized frequency in $(0, 1)$;

unity: sampling frequency;

0.5: maximum frequency in sampled signal

(folding frequency);

k_c specified in the range $(0, 0.5)$.

Note: MATLAB normalizes

half the sampling frequency to unity;

maximum normalized frequency in sampled signal $= 1$.

MATLAB and some other languages do not allow

array index = 0:

the indices mentioned above must be incremented by one.

Filtering in the frequency domain using the DFT:

1. Compute the DFT of the given signal
2. Multiply the result by $|H(k)|$
3. Compute the inverse DFT
4. Take the real part to obtain the filtered signal.

Note: Pay attention to shifting or folding of spectrum.

Advantage: no phase change is involved.

Note: Time-domain implementation required for

on-line, real-time applications.

Butterworth lowpass filter design example:

Specify two parameters: ω_c and N,

based on knowledge of the filter, signal, and noise.

Also possible to specify required minimum gain

at a certain frequency in the passband and

minimum attenuation at a frequency in the stopband.

With Equation 3.134, obtain two equations;

solve for the filter parameters ω_c and N.

Given the $3\ dB$ cutoff frequency f_c and order N:

1. Convert the specified $3\ dB$ cutoff frequency f_c to radians in the normalized range $(0, 2\,\pi)$ as $\omega_c = \frac{f_c}{f_s} 2\pi$.
   Then, $T = 1$. Prewarp the cutoff frequency ω_c by using Equation 3.140 and obtain Ω_c.
2. Derive the positions of the poles of the filter in the s-plane as given by Equation 3.136.
3. Form the transfer function $H_a(s)$ of the Butterworth lowpass filter in the Laplace domain by using the poles in the left-half plane only as given by Equation 3.137.

4. Apply the bilinear transformation as in Equation 3.138 and obtain the transfer function $H(z)$ as in Equation 3.142.
5. Convert to the coefficients b_k and a_k as in Equation 3.143.

Design Butterworth lowpass filter with

$f_c = 40\ Hz$, $f_s = 200\ Hz$, and $N = 4$.

$\omega_c = \frac{40}{200} 2\pi = 0.4\,\pi\ radians/s$.

Prewarped s-domain cutoff frequency is

$\Omega_c = \frac{2}{T} \tan\left(\frac{\omega_c}{2}\right) = 1.453085\ radians/s$.

Poles of $H_a(s)H_a(-s)$ placed around circle of

radius 1.453085

with angular separation of $\frac{\pi}{N} = \frac{\pi}{4}\ radians$.

Poles of interest located at angles $\frac{5}{8}\pi$ and $\frac{7}{8}\pi$

and the corresponding conjugate positions.

Coordinates of the poles of interest:

$(-0.556072 \pm j\,1.342475)$ and $(-1.342475 \pm j\,0.556072)$.

$$H_a(s) = \tag{3.146}$$

$$\frac{4.458247}{(s^2 + 1.112143s + 2.111456)(s^2 + 2.684951s + 2.111456)}.$$

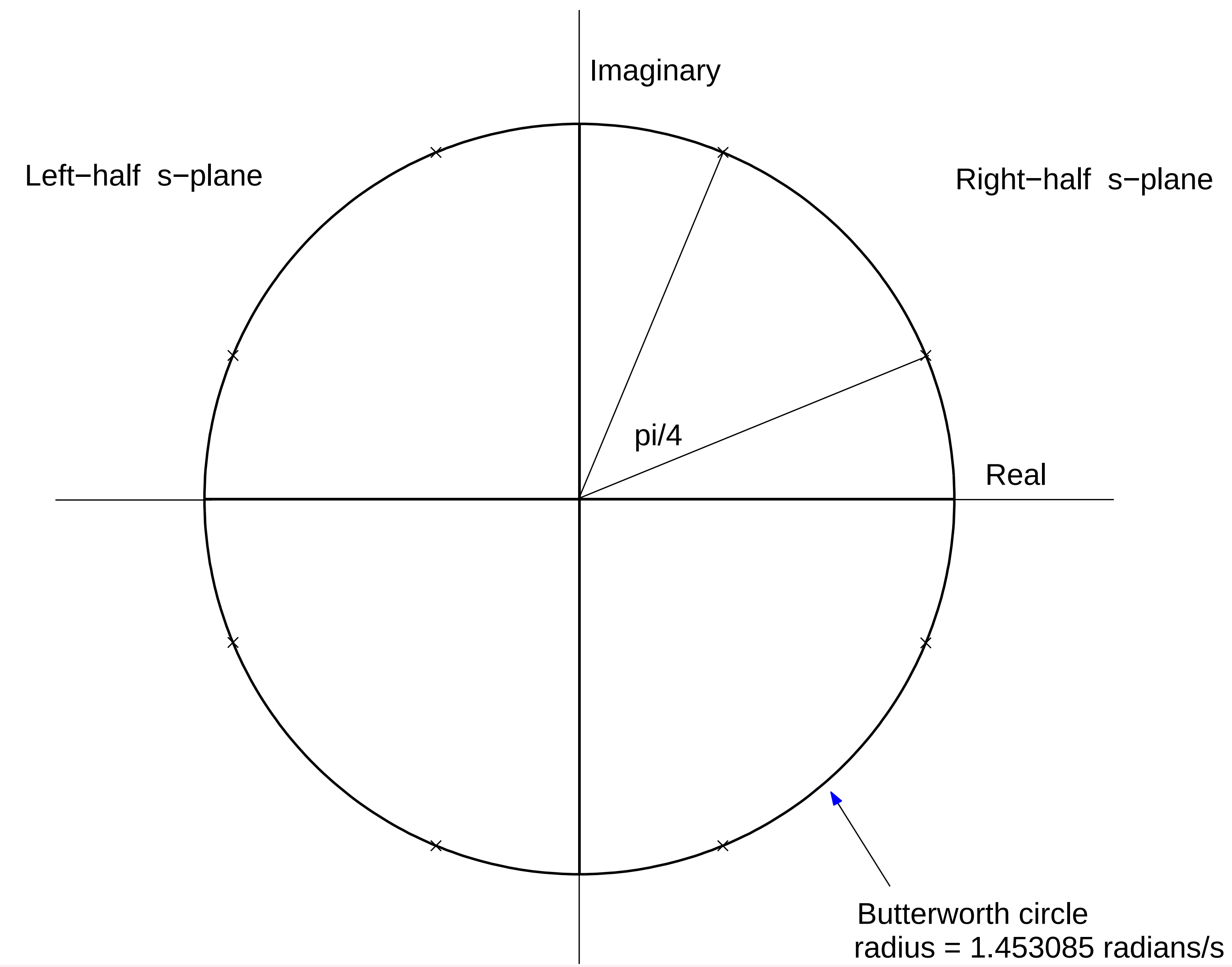


Figure 3.61: Pole positions in the s-plane of the squared magnitude response of the Butterworth lowpass filter with $f_c = 40\ Hz$, $f_s = 200\ Hz$, and $N = 4$.

Applying the bilinear transformation:

$$H(z) = \tag{3.147}$$

$$\frac{0.046583(1+z^{-1})^4}{(1-0.447765z^{-1}+0.460815z^{-2})(1-0.328976z^{-1}+0.064588z^{-2})}.$$

The filter has four poles at

$(0.223882 \pm j\,0.640852)$ and $(0.164488 \pm j\,0.193730)$,

and four zeros at $-1 + j\,0$.

b_k coefficients as in Equation 3.143:

$$\{0.0465829, 0.186332, 0.279497, 0.186332, 0.046583\},$$

a_k coefficients:

$$\{1, -0.776740, 0.672706, -0.180517, 0.029763\}.$$

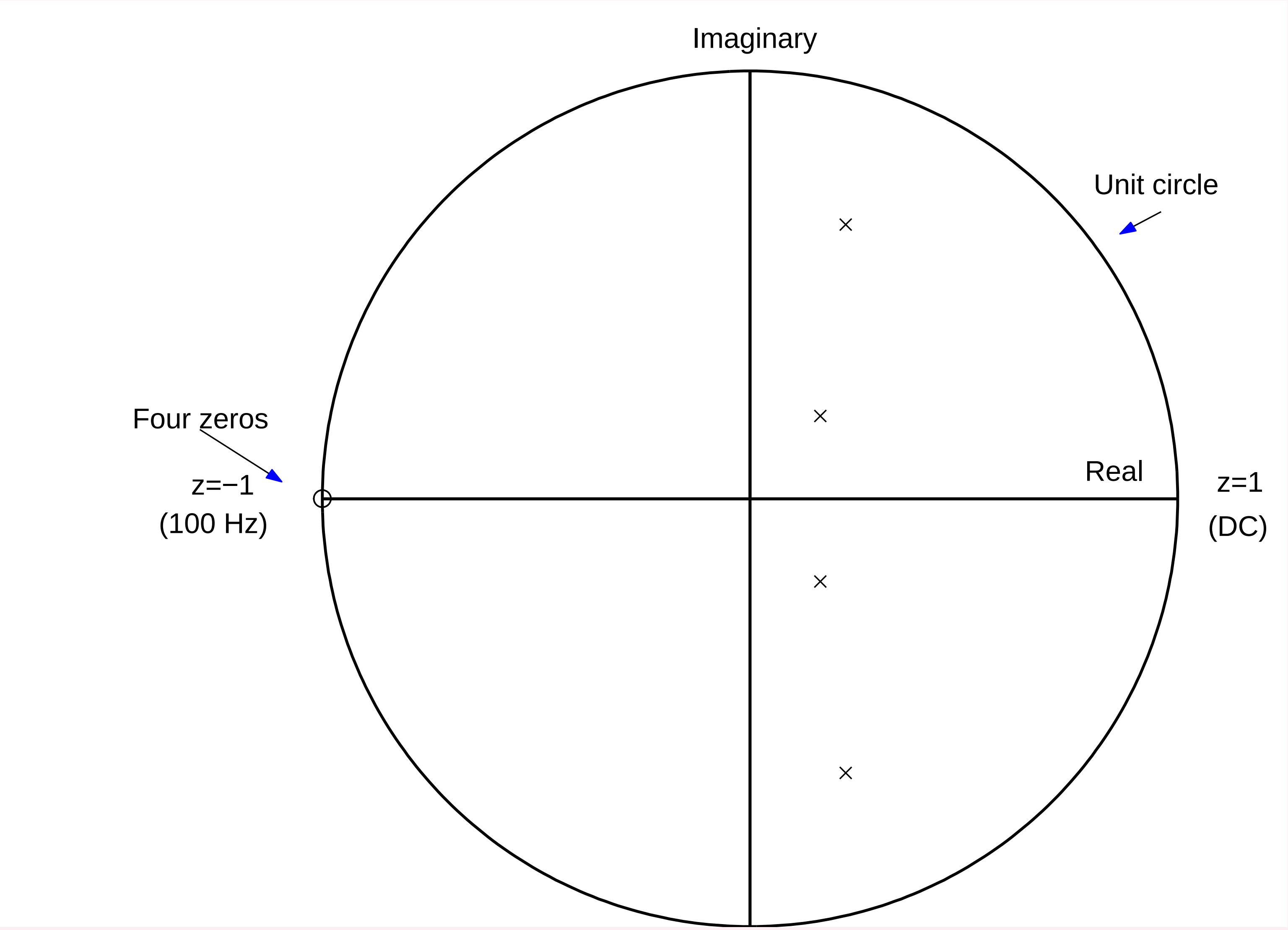


Figure 3.62: Positions of the poles and zeros in the z-plane of the Butterworth lowpass filter with $f_c = 40\ Hz$, $f_s = 200\ Hz$, and $N = 4$.

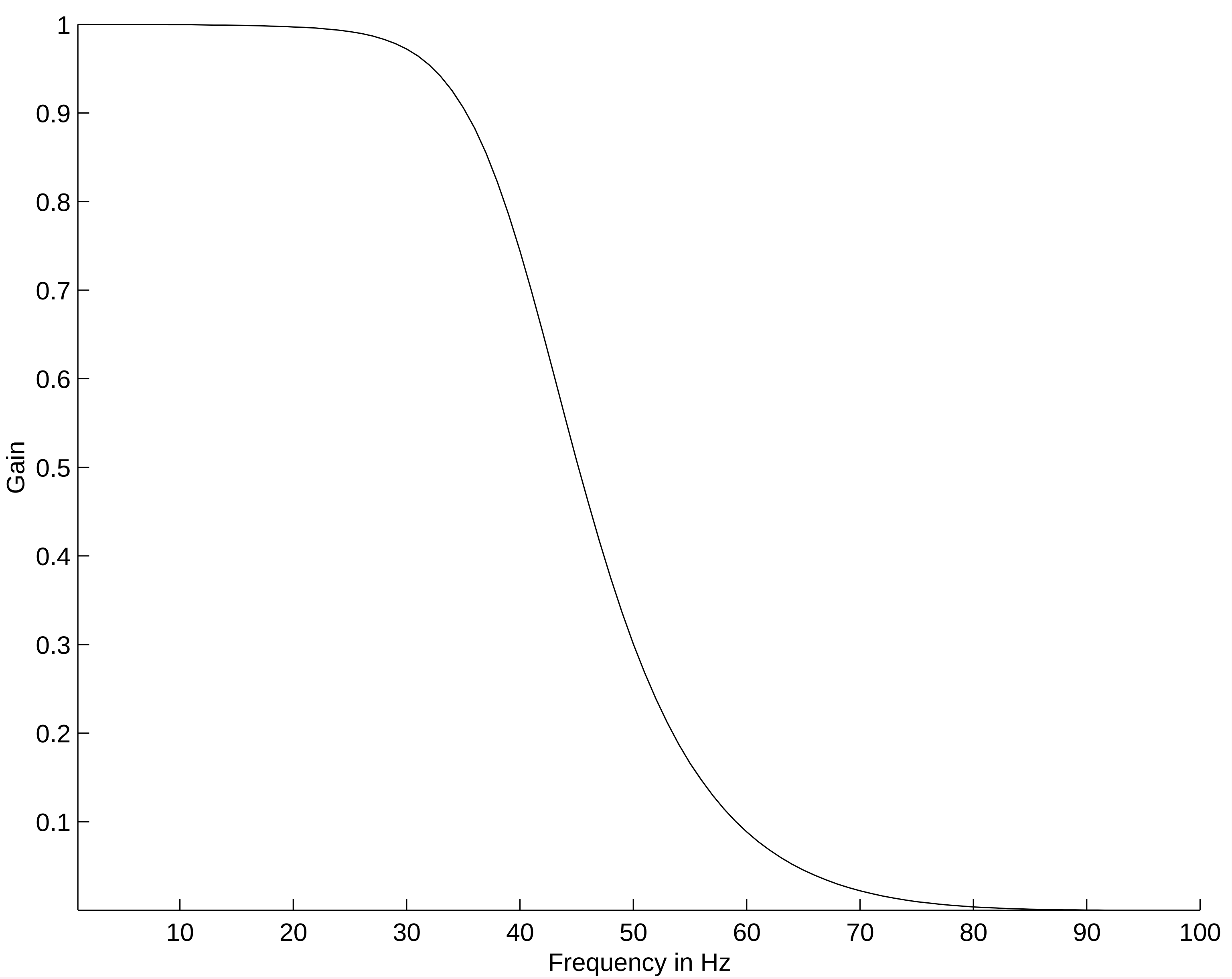


Figure 3.63: Magnitude response of the Butterworth lowpass filter with $f_c = 40\ Hz$, $f_s = 200\ Hz$, and $N = 4$.

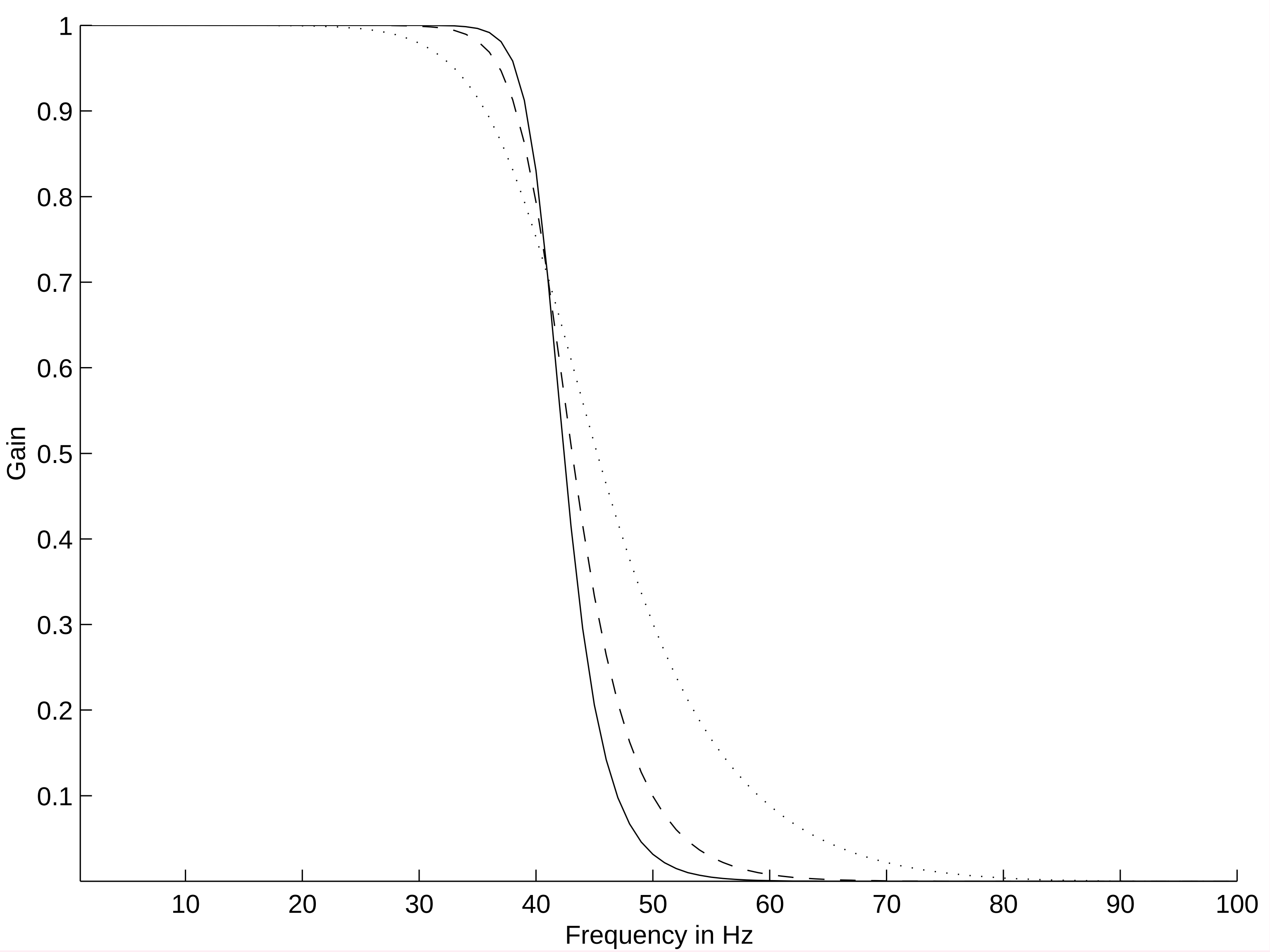


Figure 3.64: Magnitude responses of three Butterworth lowpass filters with $f_c = 40\ Hz$, $f_s = 200\ Hz$, and variable order: $N = 4$ (dotted), $N = 8$ (dashed), and $N = 12$ (solid). All three filters have gain $= 0.707$ at $40\ Hz$; the transition band becomes sharper as the order N is increased.

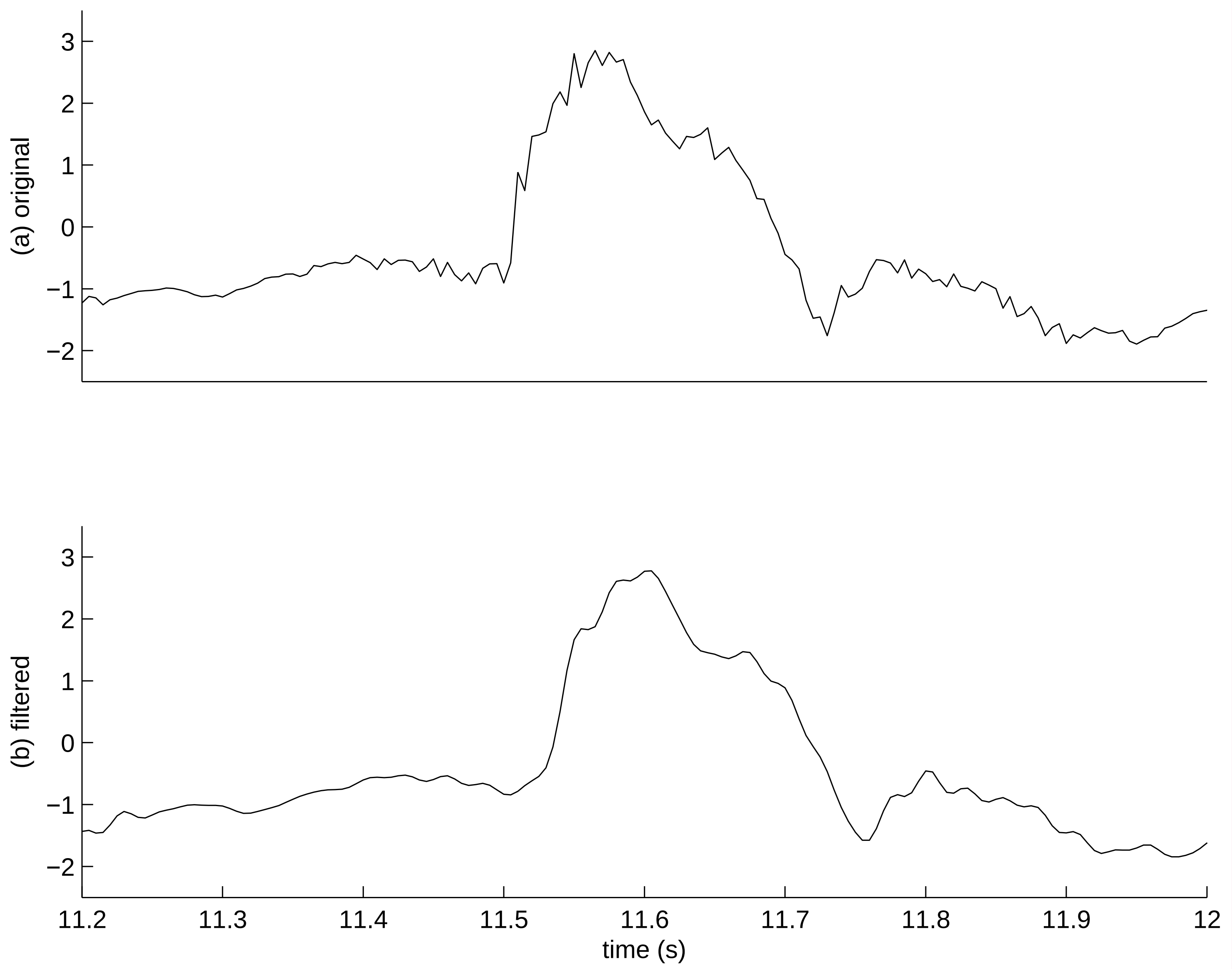


Figure 3.65: Upper trace: a carotid pulse signal with high-frequency noise. Lower trace: result of filtering using a Butterworth lowpass filter with $f_c = 40\ Hz$, $f_s = 200\ Hz$, and $N = 12$. The filtering operation was performed in the time domain using the MATLAB *filter* command.

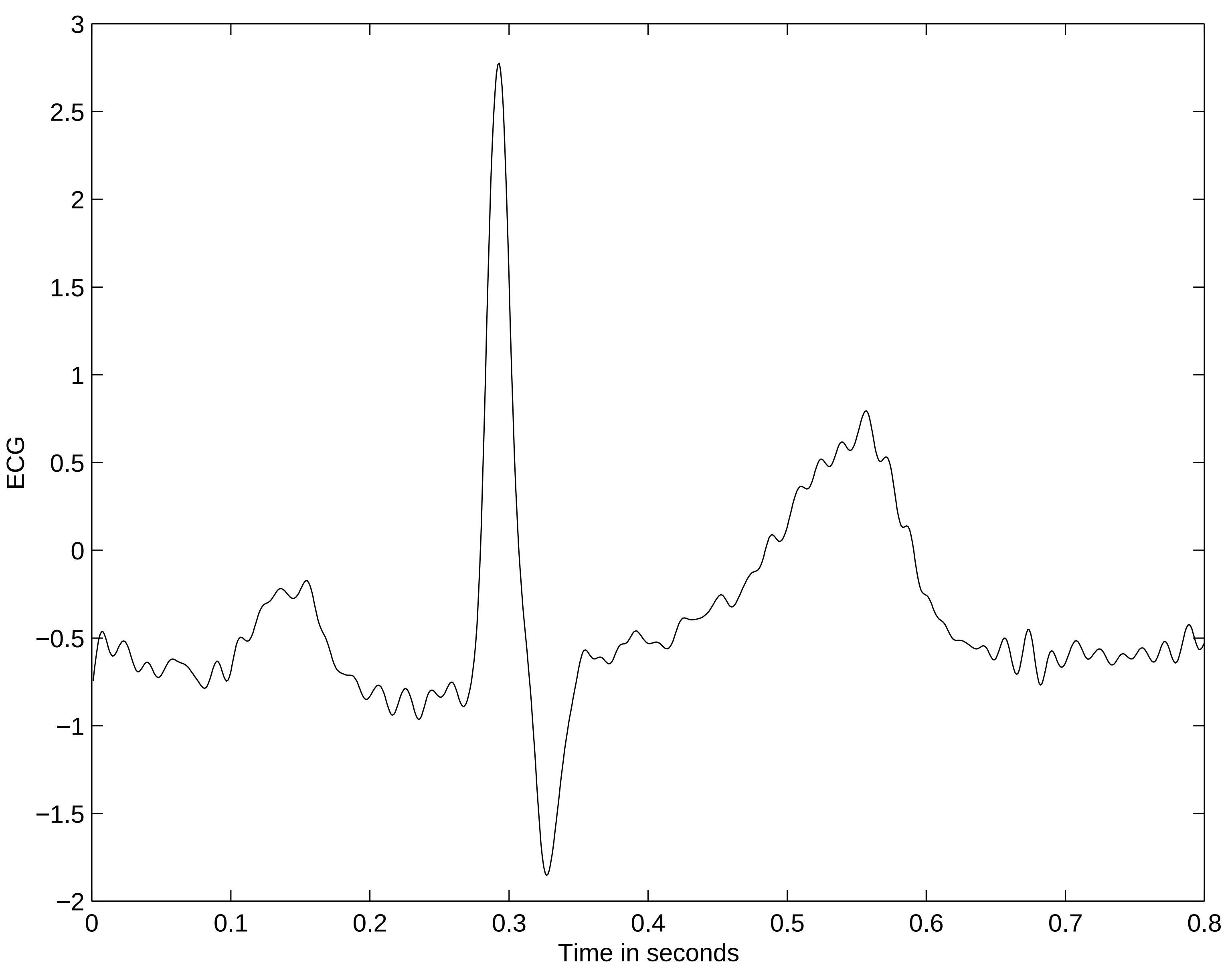


Figure 3.66: Result of frequency-domain filtering of the noisy ECG signal in Figure 3.50 with an eighth-order Butterworth lowpass filter with cutoff frequency = 70 Hz.

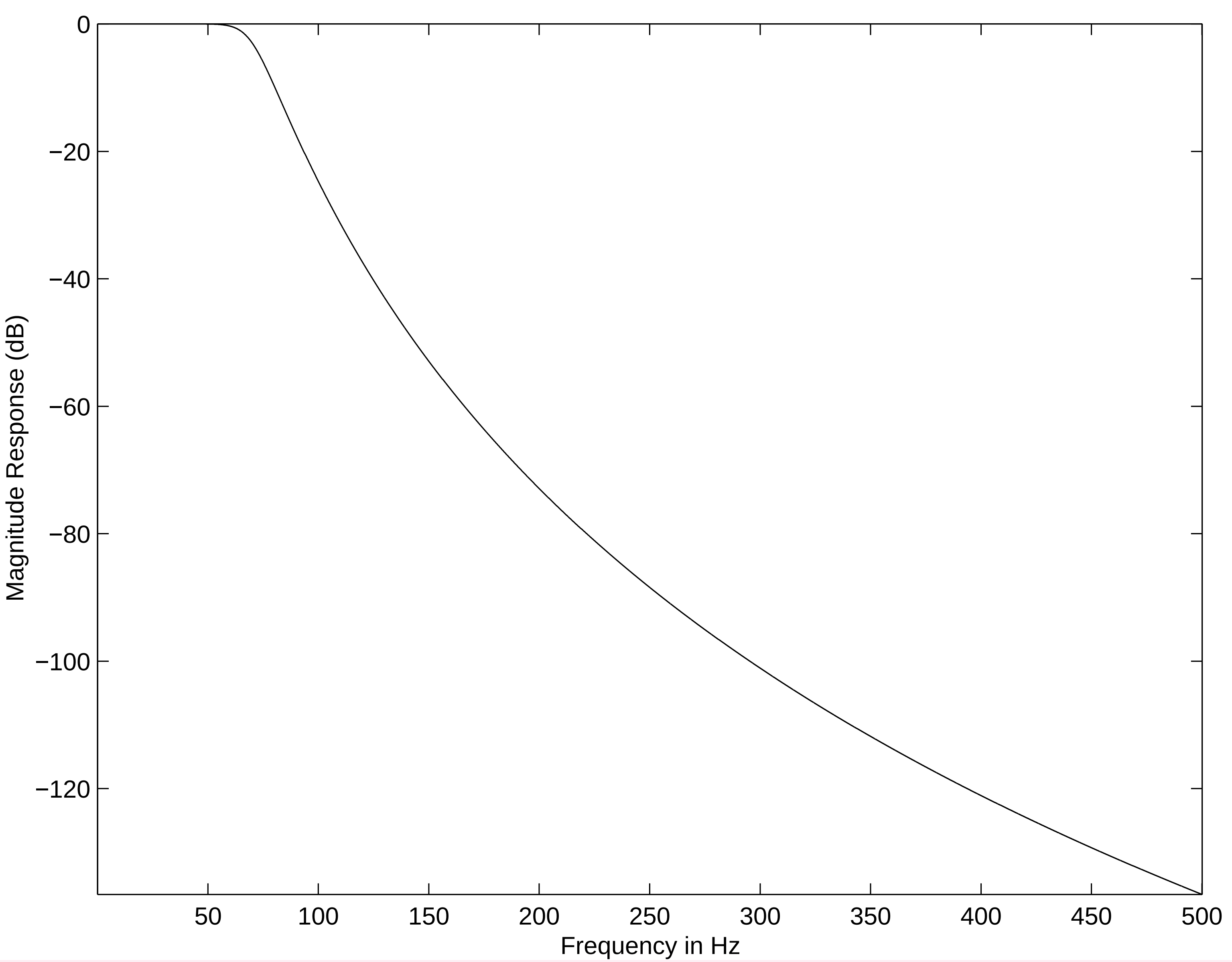


Figure 3.67: Frequency response of the eighth-order Butterworth lowpass filter with cutoff frequency = $f_c = 70\ Hz$ and $f_s = 1,000\ Hz$.

3.6.2 Removal of low-frequency noise: Butterworth highpass filters

Problem: *Design a frequency-domain filter*

to remove low-frequency noise with

minimal loss of signal components in the passband.

Solution:

Butterworth highpass filter specified directly

in the discrete-frequency domain as

$$|H(k)|^2 = \frac{1}{1 + \left(\frac{k_c}{k}\right)^{2N}}. \tag{3.148}$$

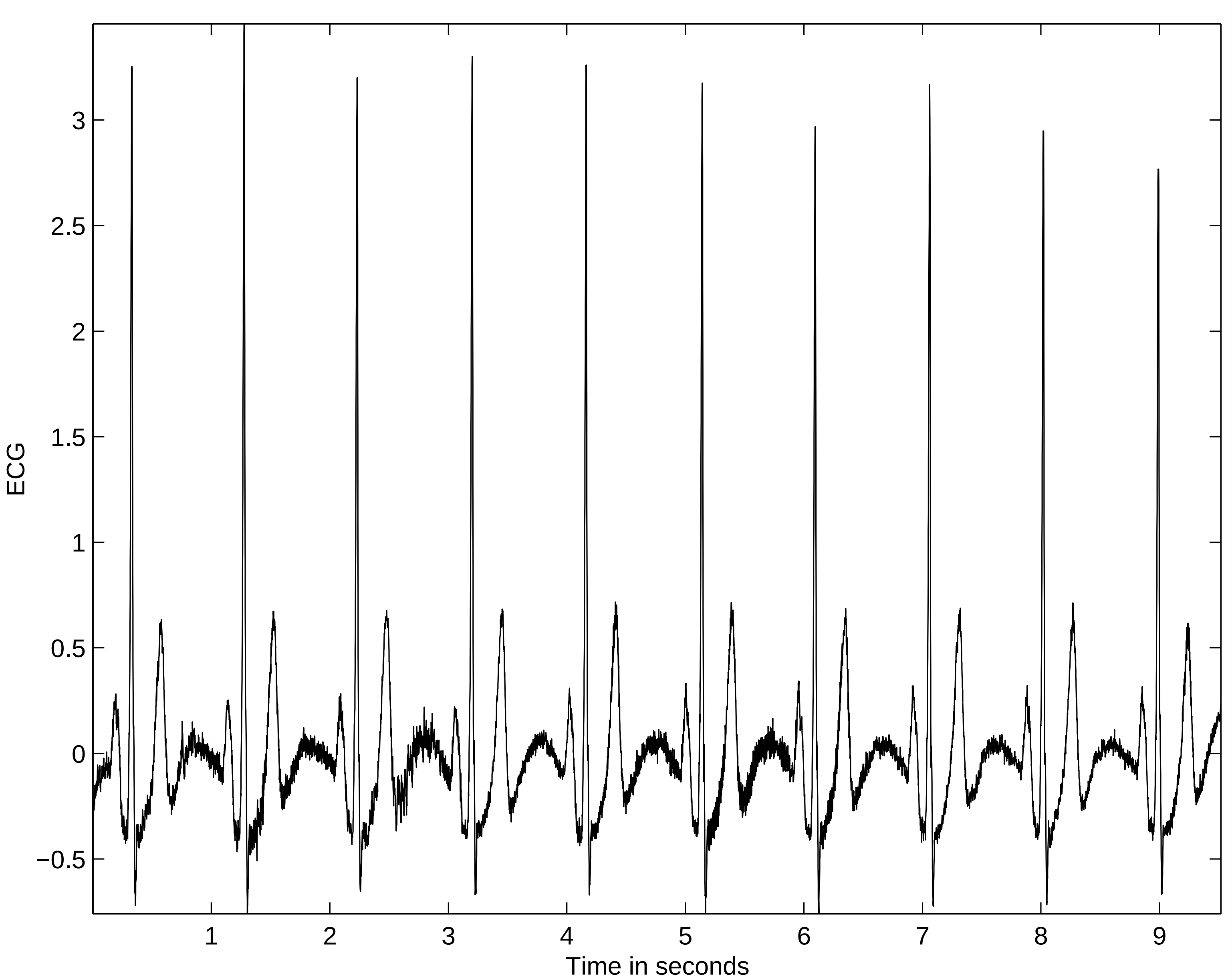


Figure 3.68: Result of frequency-domain filtering of the ECG signal with low-frequency noise in Figure 3.6 with an eighth-order Butterworth highpass filter with cutoff frequency $= 2\ Hz$. (Compare with the results in Figures 3.54, 3.55, and 3.58.)

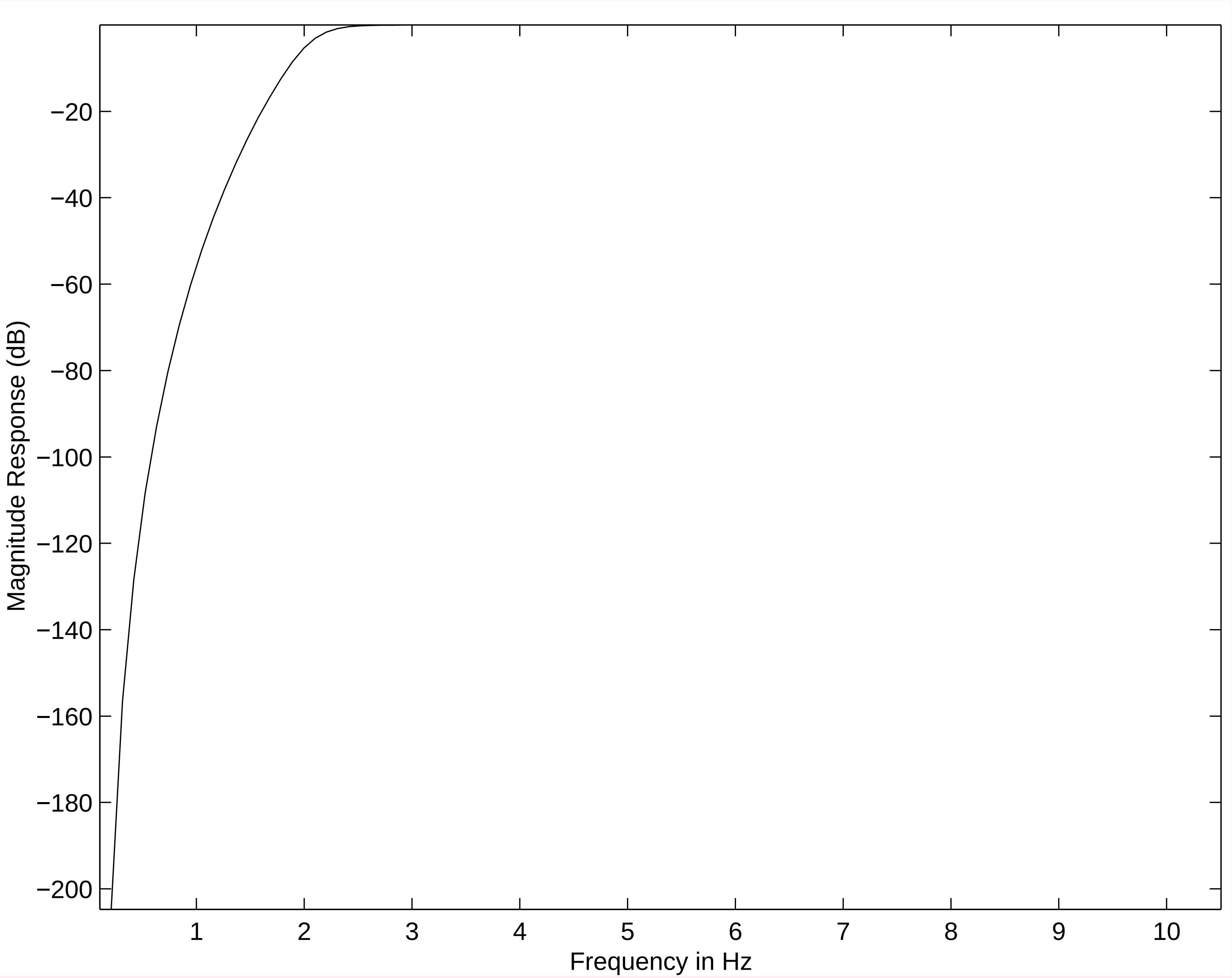


Figure 3.69: Frequency response of an eighth-order Butterworth highpass filter with cutoff frequency = $2\ Hz$. $f_s = 1,000\ Hz$. The frequency response is shown on an expanded scale for the range $0 - 10\ Hz$ only.

3.6.3 Removal of periodic artifacts: Notch and comb filters

Problem: *Design a frequency-domain filter to remove periodic artifacts such as power-line interference.*

Solution: Periodic interference may be removed by notch or comb filters with zeros on the unit circle in the z-domain at the specific frequencies to be rejected.

f_o: interference frequency.

Angles of complex conjugate zeros required: $\pm\frac{f_o}{f_s}(2\pi)$.

Radius of the zeros: unity.

If harmonics present, multiple zeros required at $\pm\frac{nf_o}{f_s}(2\pi)$,

n: orders of all harmonics present.

Notch filter design example:

Power-line interference at $f_o = 60\ Hz$; $f_s = 1,000\ Hz$.

Notch filter zeros at

$\omega_o = \pm \frac{f_o}{f_s}(2\pi)$ = $\pm 0.377\ radians = \pm 21.6^\circ$.

Zero locations: $\cos(\omega_o) \pm j \sin(\omega_o)$ or

$z_1 = 0.92977 + j0.36812$ and $z_2 = 0.92977 - j0.36812$.

$$\begin{aligned} H(z) &= (1 - z^{-1} z_1)(1 - z^{-1} z_2) \\ &= 1 - 1.85955 z^{-1} + z^{-2}. \end{aligned} \tag{3.149}$$

For $|H(1)| = 1$, divide $H(z)$ above by 0.14045.

Sharpness of notch may be improved by placing a few poles symmetrically around the zeros and inside the unit circle.

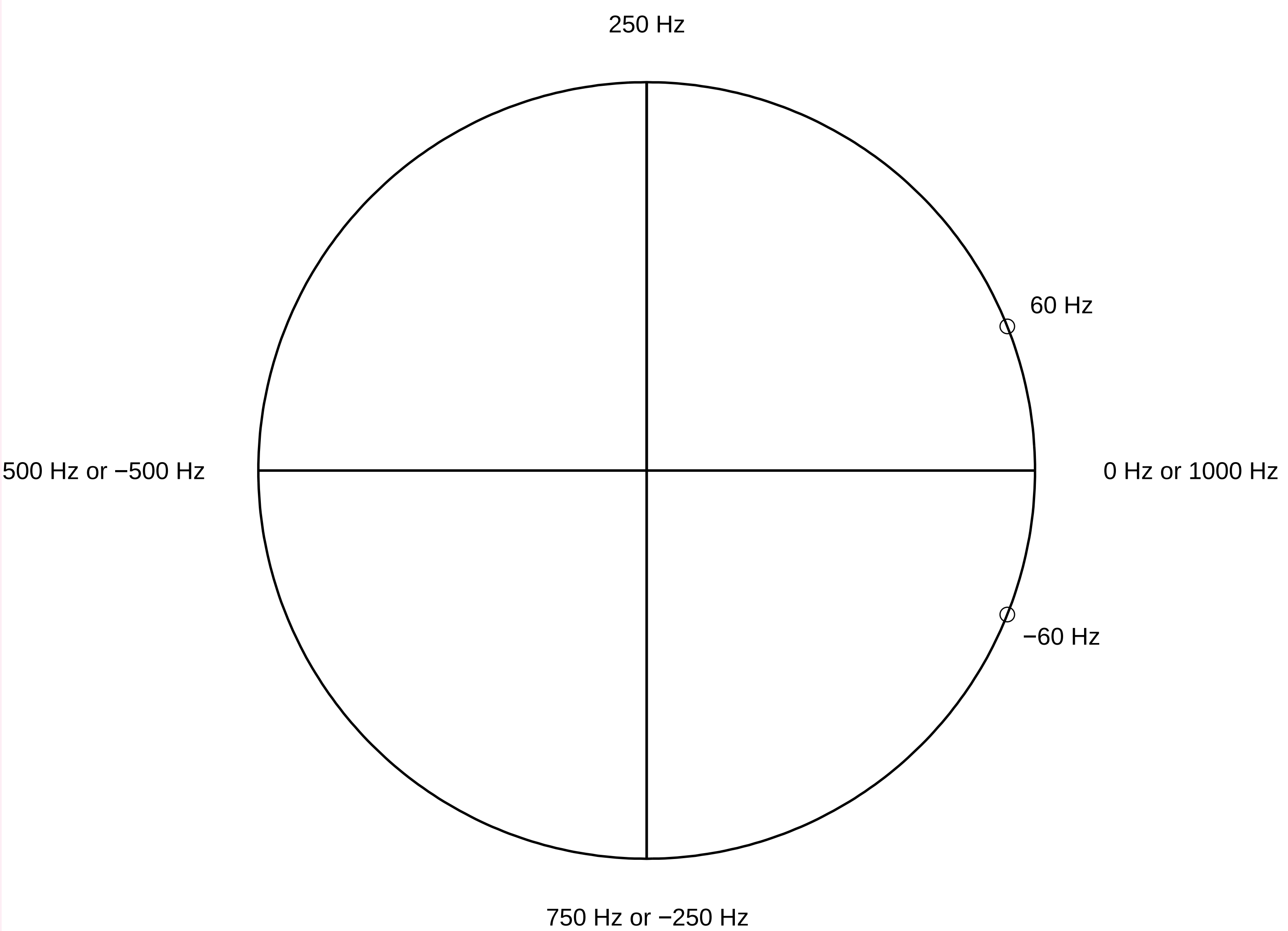


Figure 3.70: Zeros of the notch filter to remove $60\ Hz$ interference, the sampling frequency being $1,000\ Hz$.

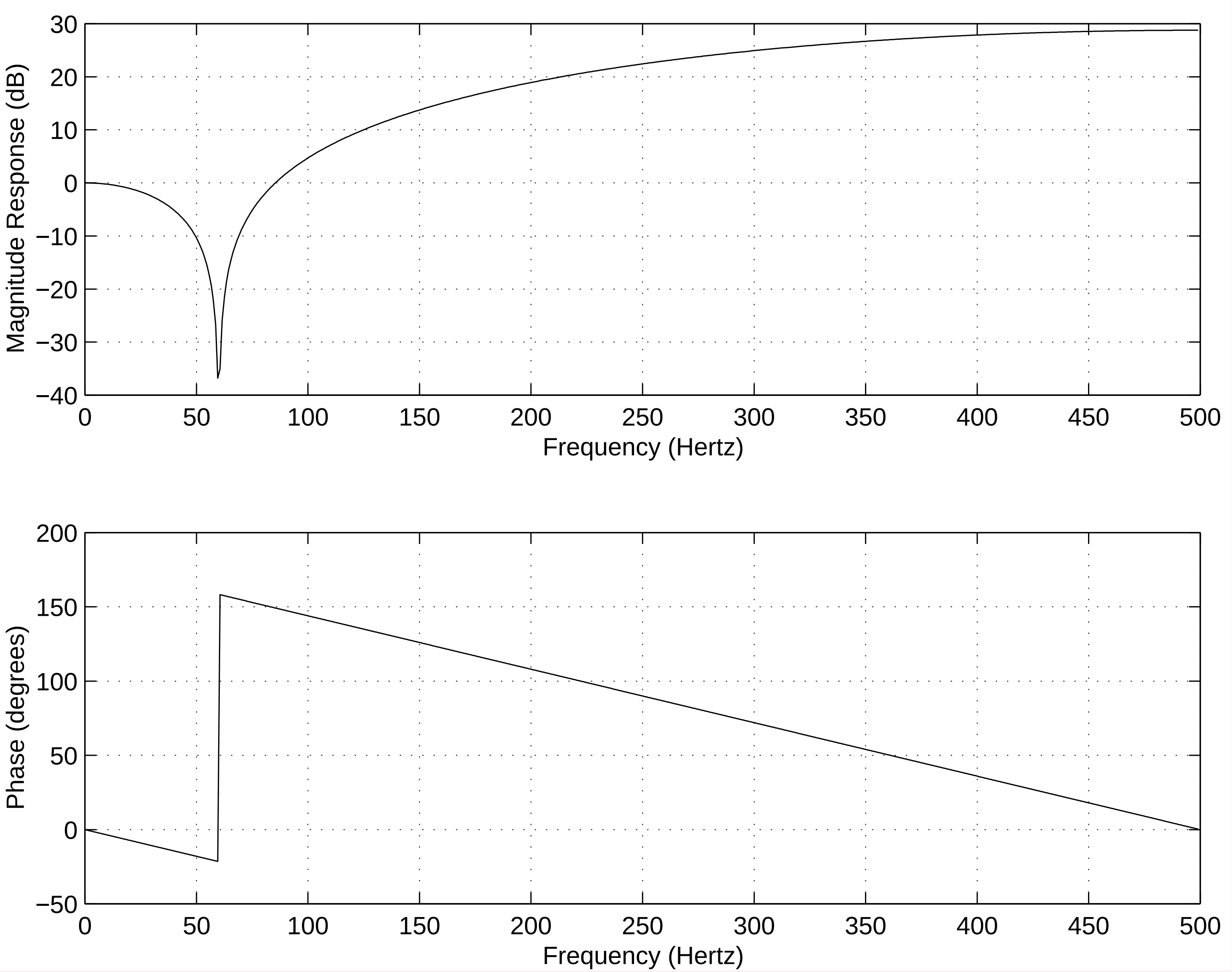


Figure 3.71: Magnitude and phase responses of the 60 Hz notch filter with zeros as shown in Figure 3.70. $f_s = 1,000\ Hz$. Note: Large gain at $f_s/2$ — could apply a lowpass filter.

Comb filter design example:

Periodic artifact with fundamental frequency $60\ Hz$ and

odd harmonics at $180\ Hz$, $300\ Hz$, and $420\ Hz$.

Zeros desired at $60\ Hz$, $180\ Hz$, $300\ Hz$, and $420\ Hz$,

or $\pm 21.6^\circ$, $\pm 64.8^\circ$, $\pm 108^\circ$, and $\pm 151.2^\circ$,

with 360° corresponding to $f_s = 1,000\ Hz$.

Coordinates of the zeros are $0.92977 \pm j0.36812$,

$0.42578 \pm j0.90483$, $-0.30902 \pm j0.95106$,

and $-0.87631 \pm j0.48175$.

$$\begin{aligned} H(z) &= G\,(1 - 1.85955z^{-1} + z^{-2})(1 - 0.85156z^{-1} + z^{-2}) \\ &\times\ (1 + 0.61803z^{-1} + z^{-2})(1 + 1.75261z^{-1} + z^{-2}) \end{aligned} \tag{3.150}$$

G: desired gain or scaling factor.

With G computed so that $|H(1)| = 1$:

$$\begin{aligned} H(z) &= 0.6310 - 0.2149z^{-1} + 0.1512z^{-2} \\ &- 0.1288z^{-3} + 0.1227z^{-4} - 0.1288z^{-5} \\ &+ 0.1512z^{-6} - 0.2149z^{-7} + 0.6310z^{-8}. \end{aligned} \tag{3.151}$$

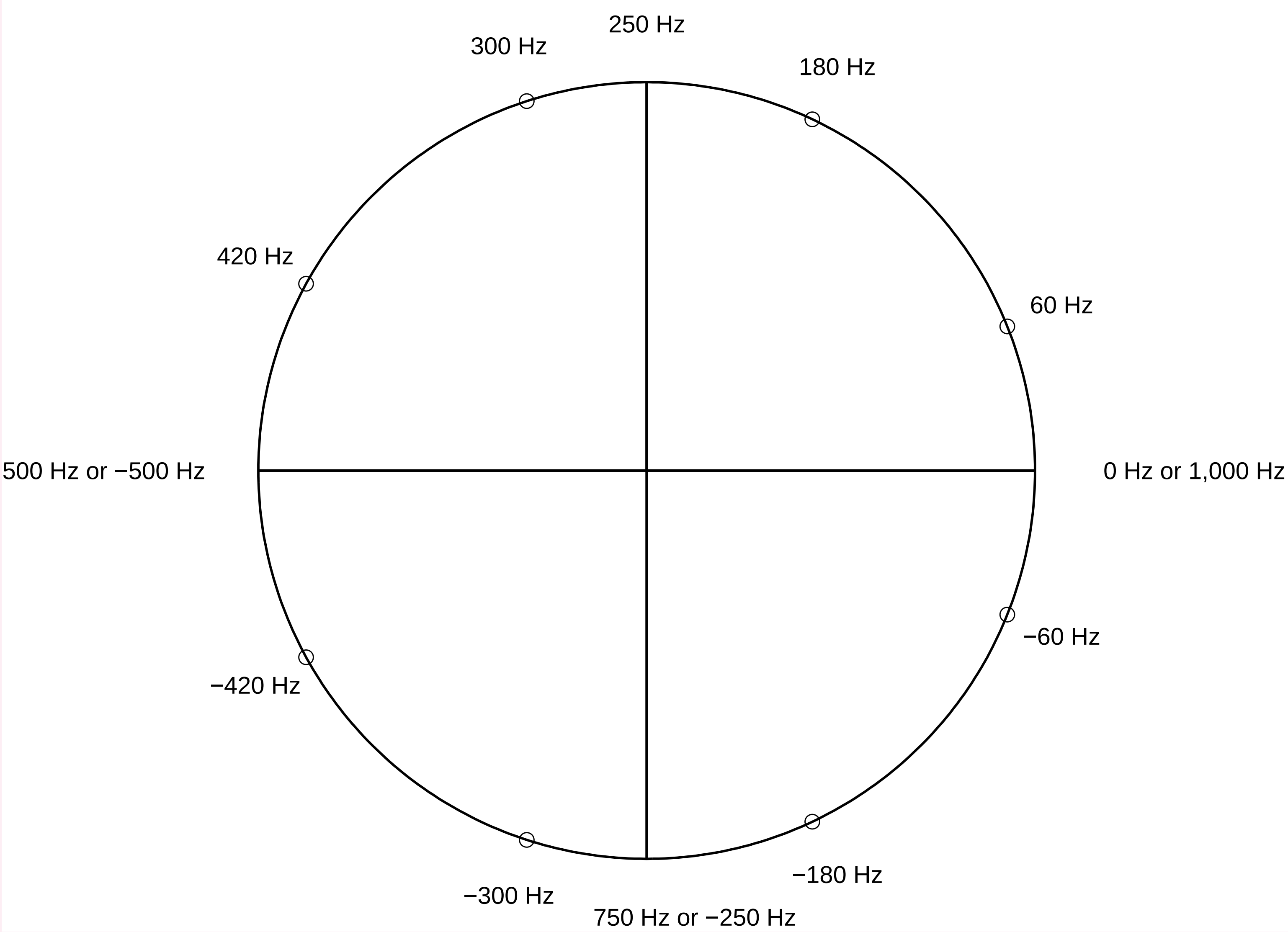


Figure 3.72: Zeros of the comb filter to remove $60\ Hz$ interference with odd harmonics; the sampling frequency is $1,000\ Hz$.

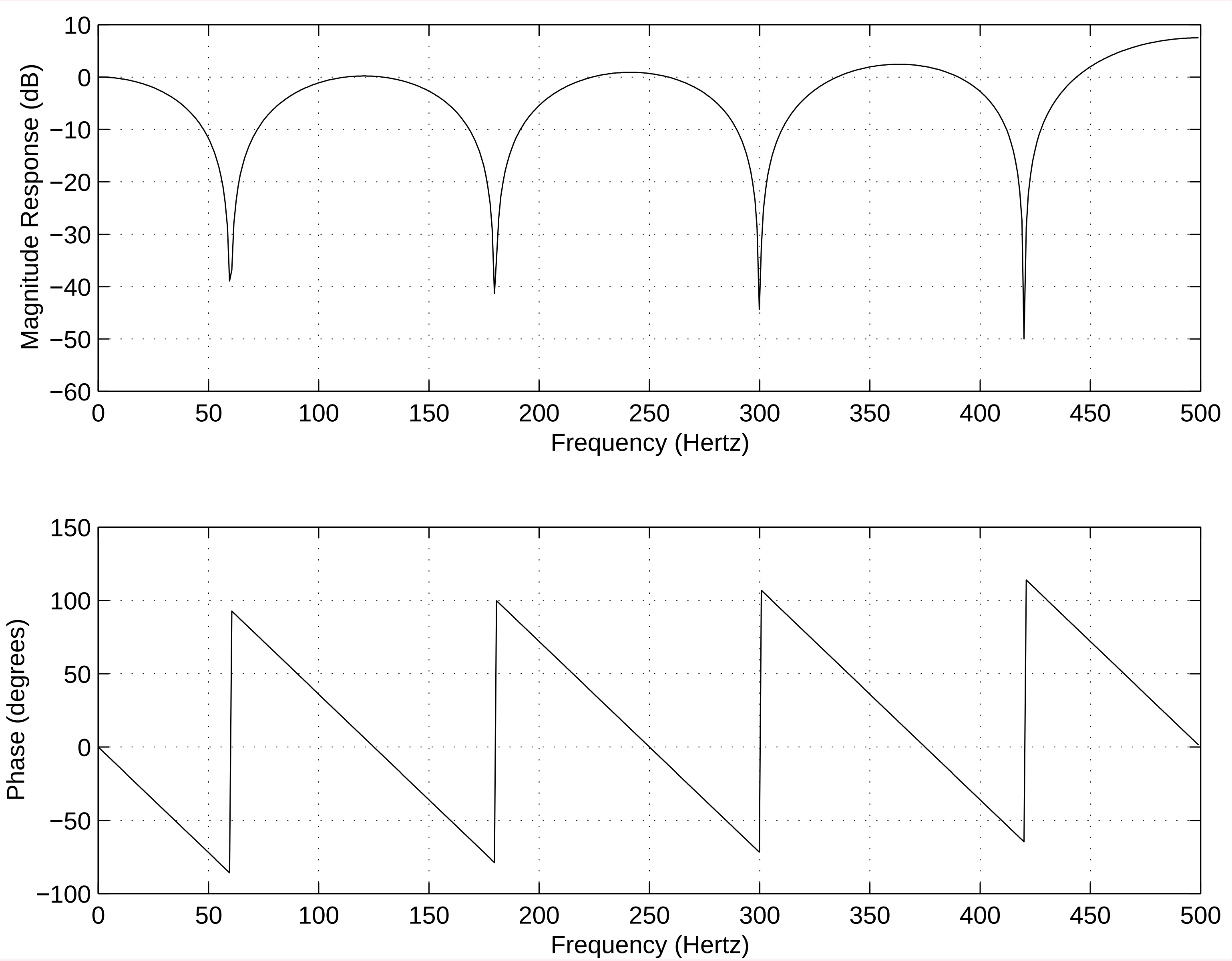


Figure 3.73: Magnitude and phase responses of the comb filter with zeros as shown in Figure 3.72. Note: Large gain at $f_s/2$ — could apply a lowpass filter.

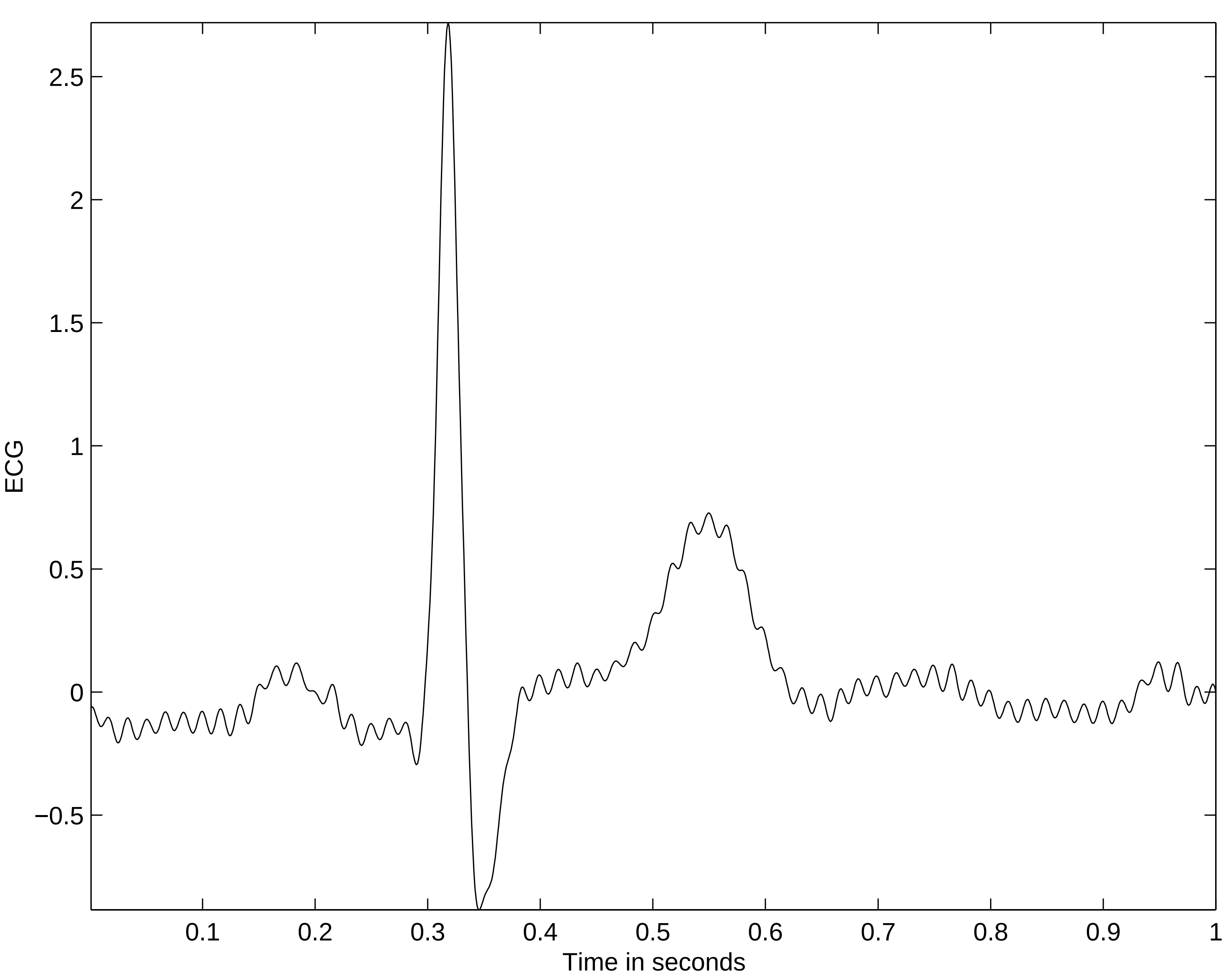


Figure 3.74: ECG signal with 60 Hz interference.

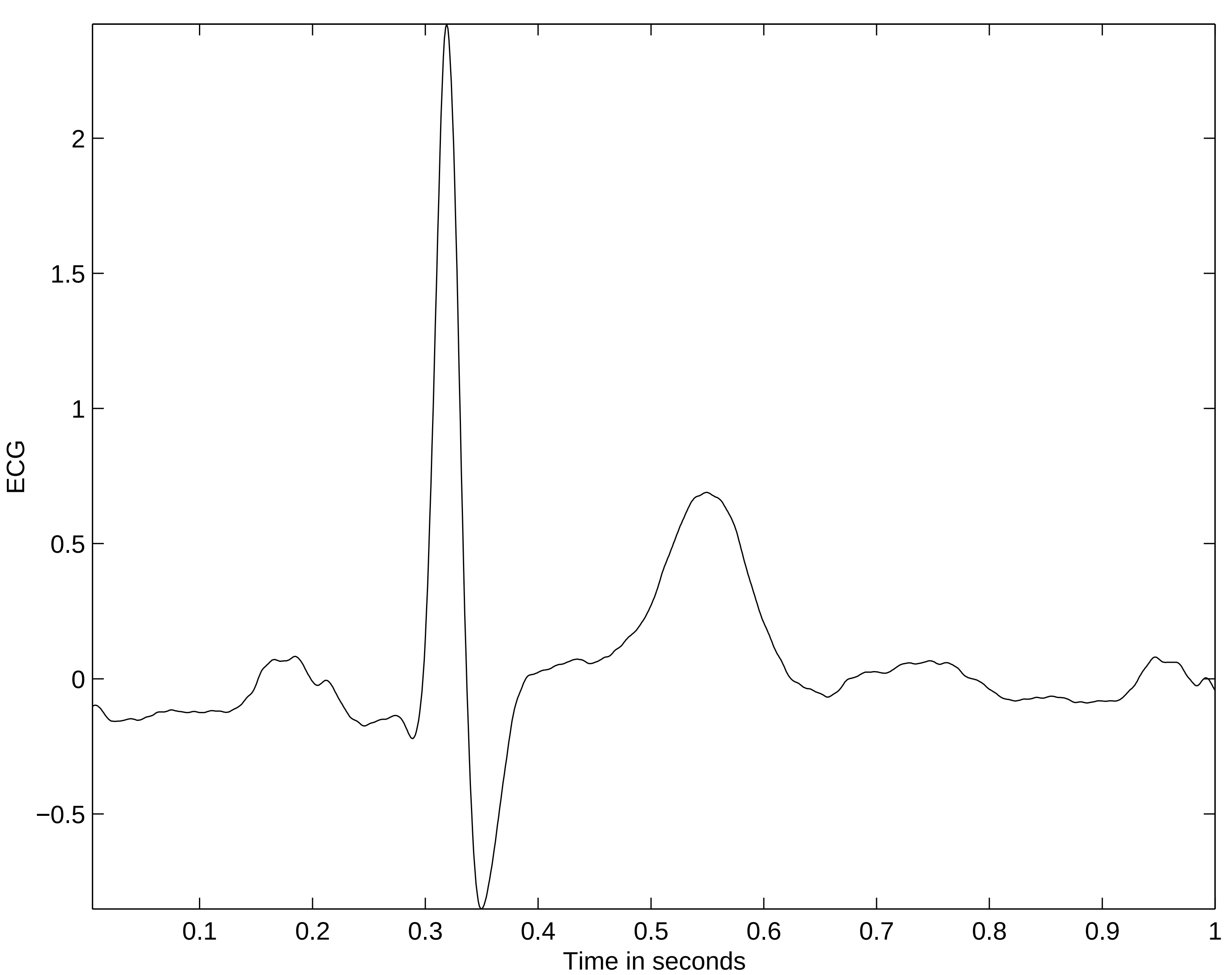


Figure 3.75: The ECG signal in Figure 3.74 after filtering with the 60 Hz notch filter shown in Figures 3.70 and 3.71.

To make the notch in the frequency response narrower or sharper, four poles were incorporated in the filter.

Frequencies of the poles: $59.95\ Hz$ and $60.05\ Hz$;

radii $= 0.4$.

Coordinates of the poles: $-0.1230 \pm j\ 0.3806$ and $-0.1242 \pm j\ 0.3802$.

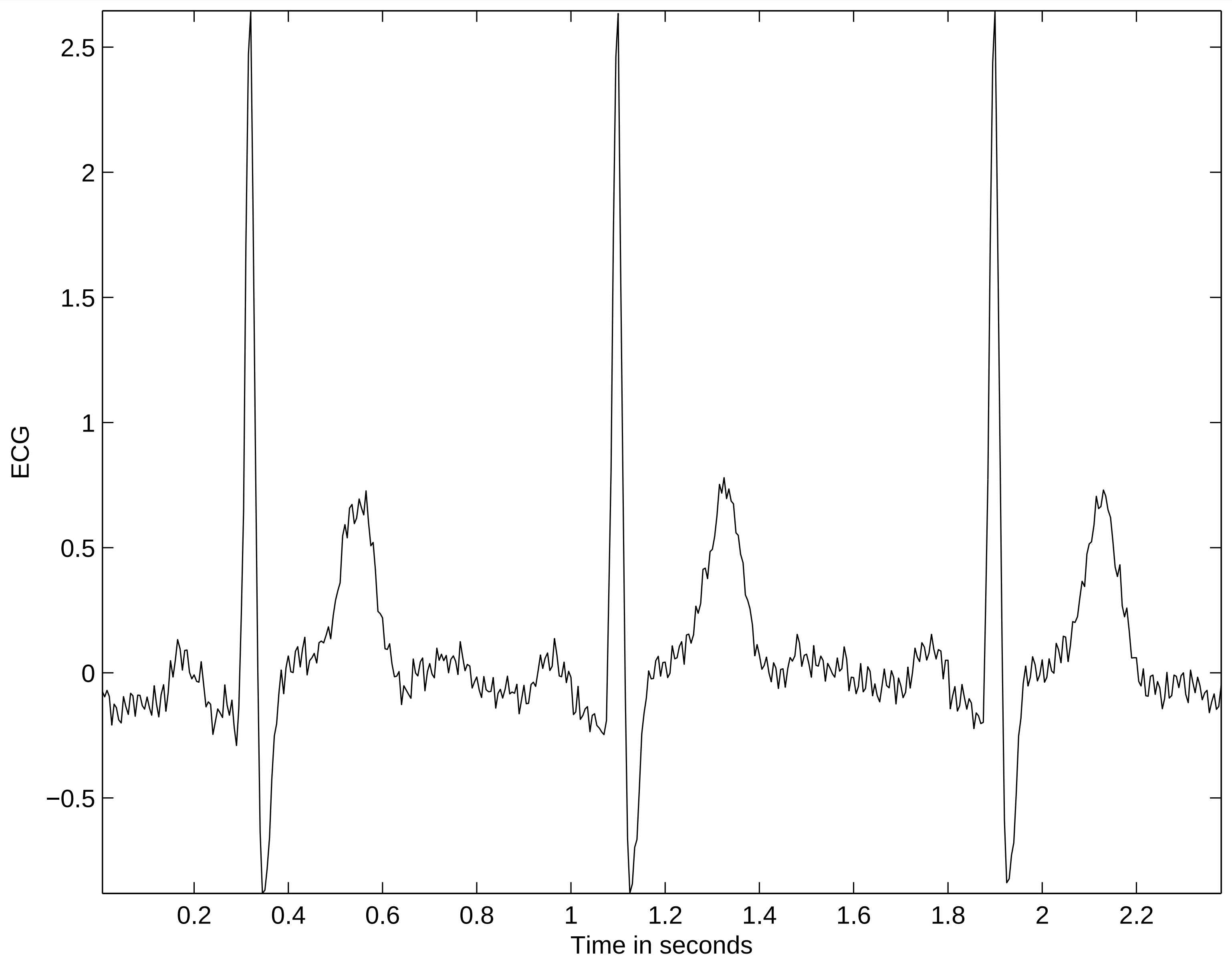


Figure 3.76: An ECG signal with power-line interference at 60 Hz; $f_s = 200\ Hz$.

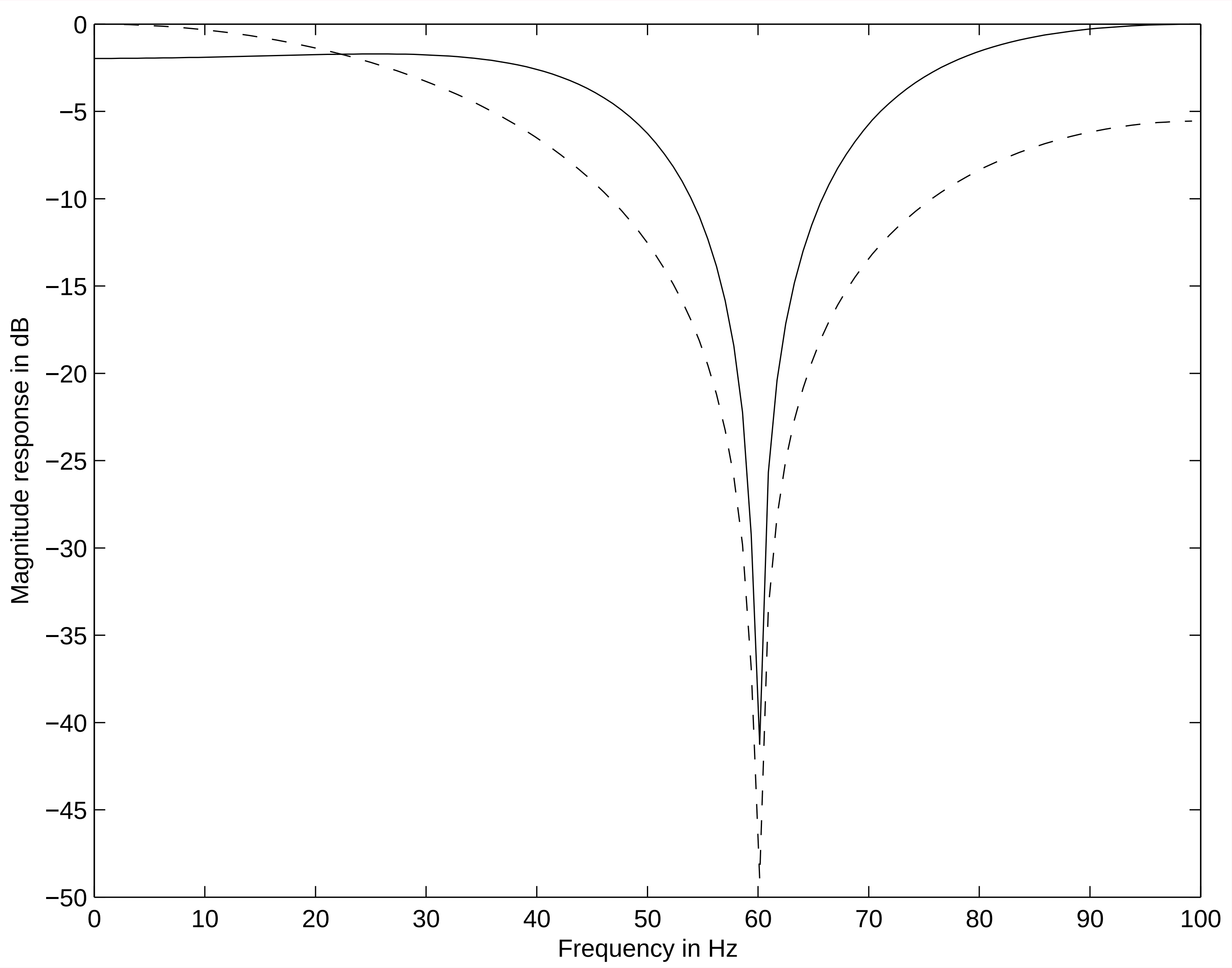


Figure 3.77: Dashed line: frequency response (magnitude, in dB) of a notch filter with two zeros at $60\ Hz$; $f_s = 200\ Hz$. The coordinates of the zeros in the z-plane are $-0.3090 \pm j\ 0.9511$. Solid line: response of the filter with the inclusion of four poles; see the text for details.

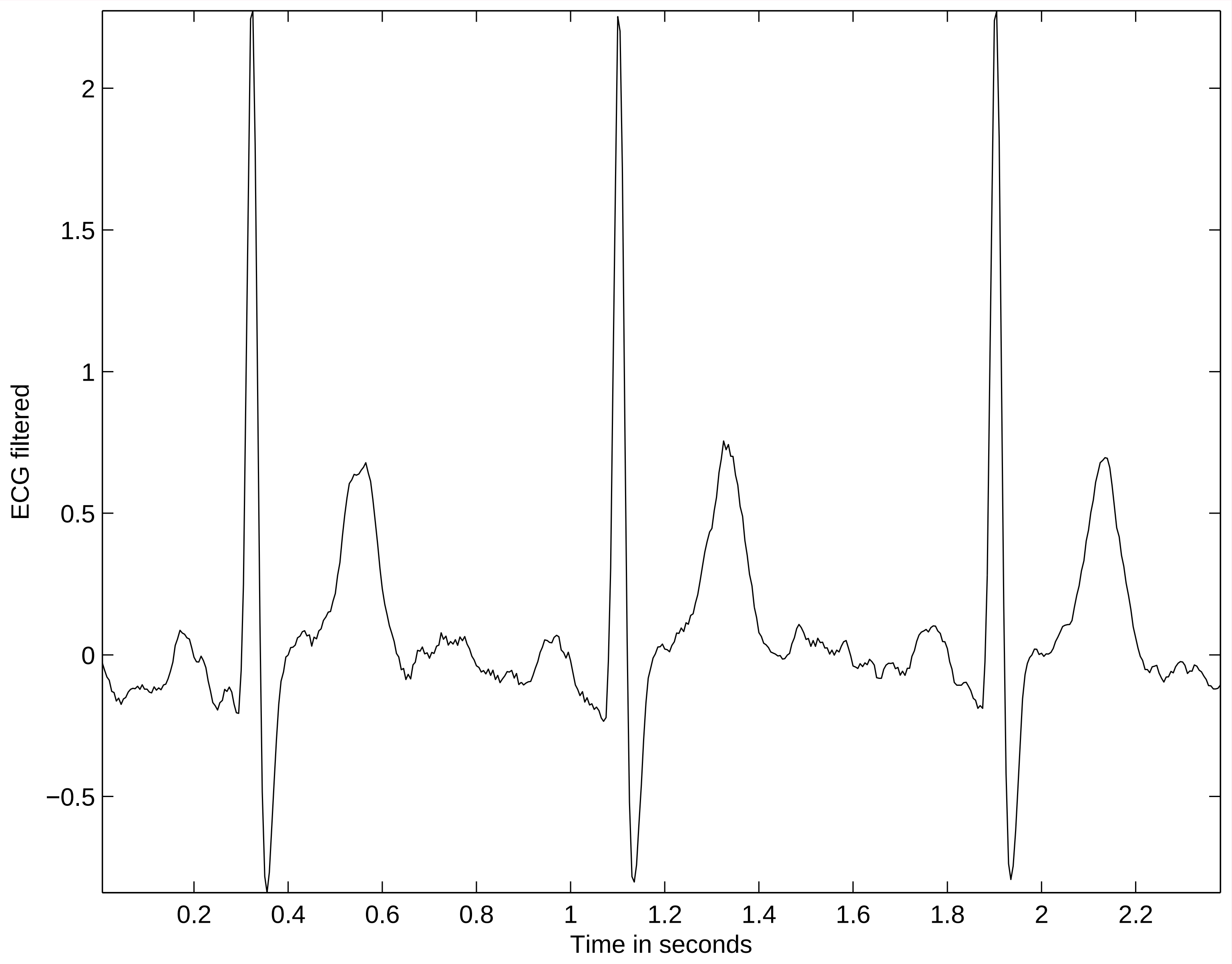


Figure 3.78: The result of filtering the noisy signal in Figure 3.76 using the notch filter with two zeros at 60 Hz.

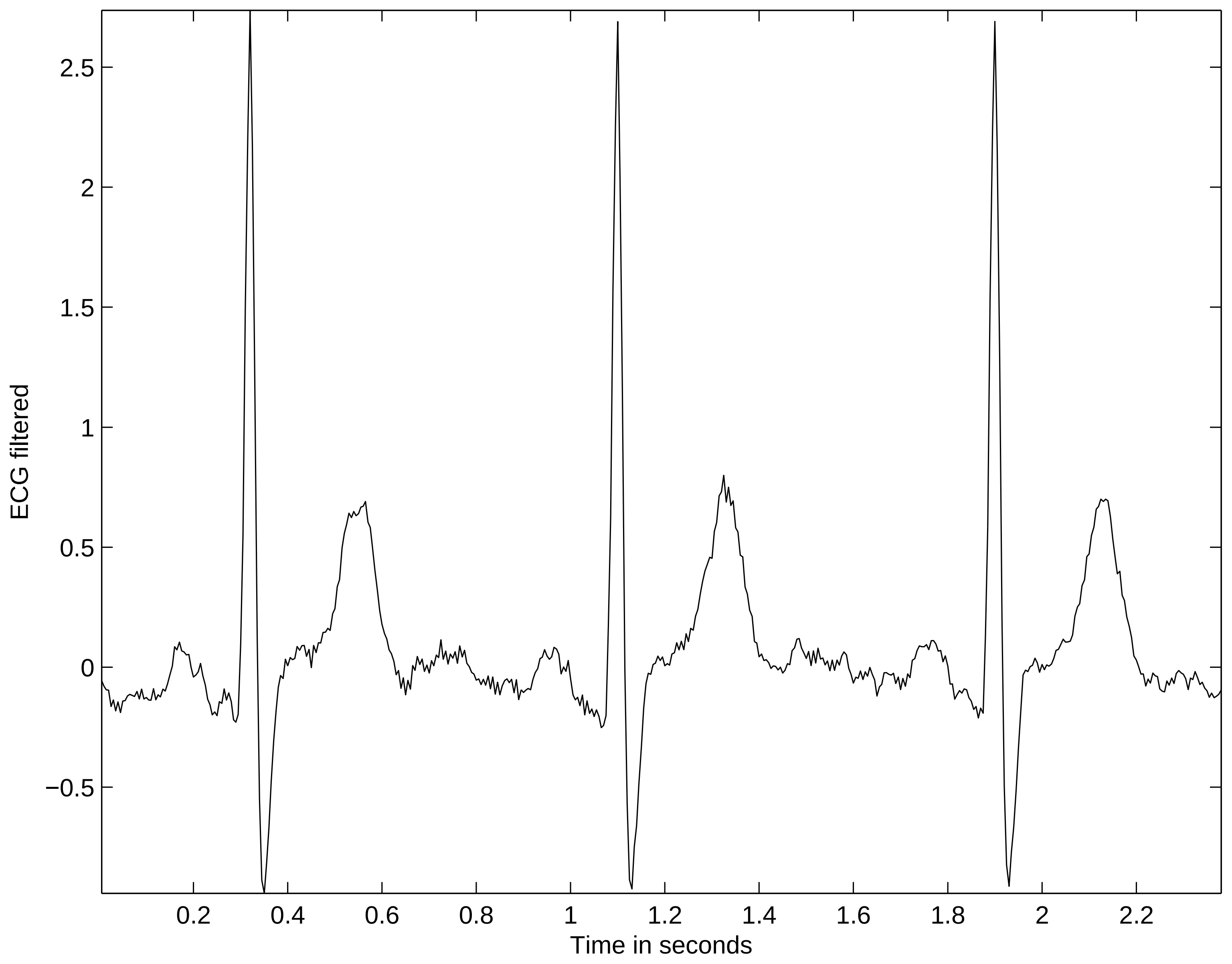


Figure 3.79: The result of filtering the noisy signal in Figure 3.76 using the notch filter with two zeros and four poles. Compare with the result in Figure 3.78.

3.7 Order-statistic Filters

Filters based on order statistics: nonlinear filters

First step: arrange in rank order, usually from the minimum to the maximum, the values of the signal in a moving window positioned at the current sample being processed.

i^{th} entry in the list = output of the i^{th} order-statistic filter.

- *Min filter*: the first entry in the rank-ordered list, useful in removing high-valued impulsive noise.
- *Max filter*: the last entry in the rank-ordered list, useful in removing low-valued impulsive noise.
- *Min/Max filter*: sequential application of the Min and Max filters, useful in removing impulsive noise of both of the types mentioned above.
- *Median filter*: the entry in the middle of the list. The median filter is the most popular and commonly used filter among the order-statistic filters.

- *α-trimmed mean filter*: the mean of a reduced or trimmed list; the first $\alpha \times 100\%$ and the last $\alpha \times 100\%$ of the entries in the original list are removed; $0 \leq \alpha < 0.5$.

  Outliers, which are samples with values substantially different from the rest of the samples in the list, are rejected by the trimming process.

  $\alpha \approx 0.5$ but < 0.5 leads to the rejection of the entire list except the median or a few values close to the median.

  The mean of the trimmed list provides a compromise between the generic mean and median filters.

- *L-filters*: weighted combination of all of the elements in the rank-ordered list.

  Appropriate weights facilitate the design of several nonlinear filters based on order statistics including those listed above.

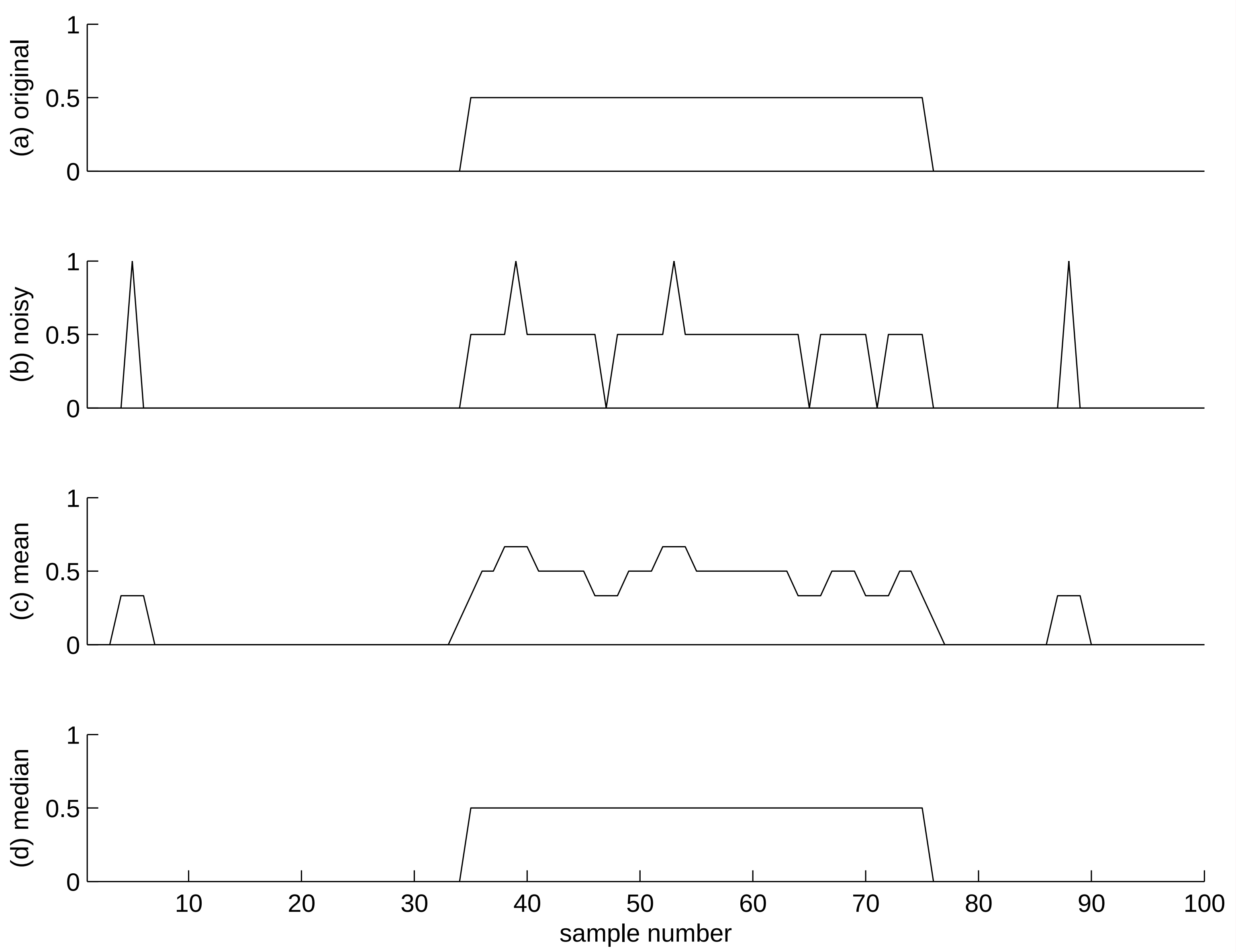


Figure 3.80: (a) A synthesized test signal with a rectangular pulse. (b) Test signal contaminated with simulated impulsive or shot noise. Result of filtering the noisy signal using (c) the mean and (d) the median with a sliding window having $M = 3$ samples.

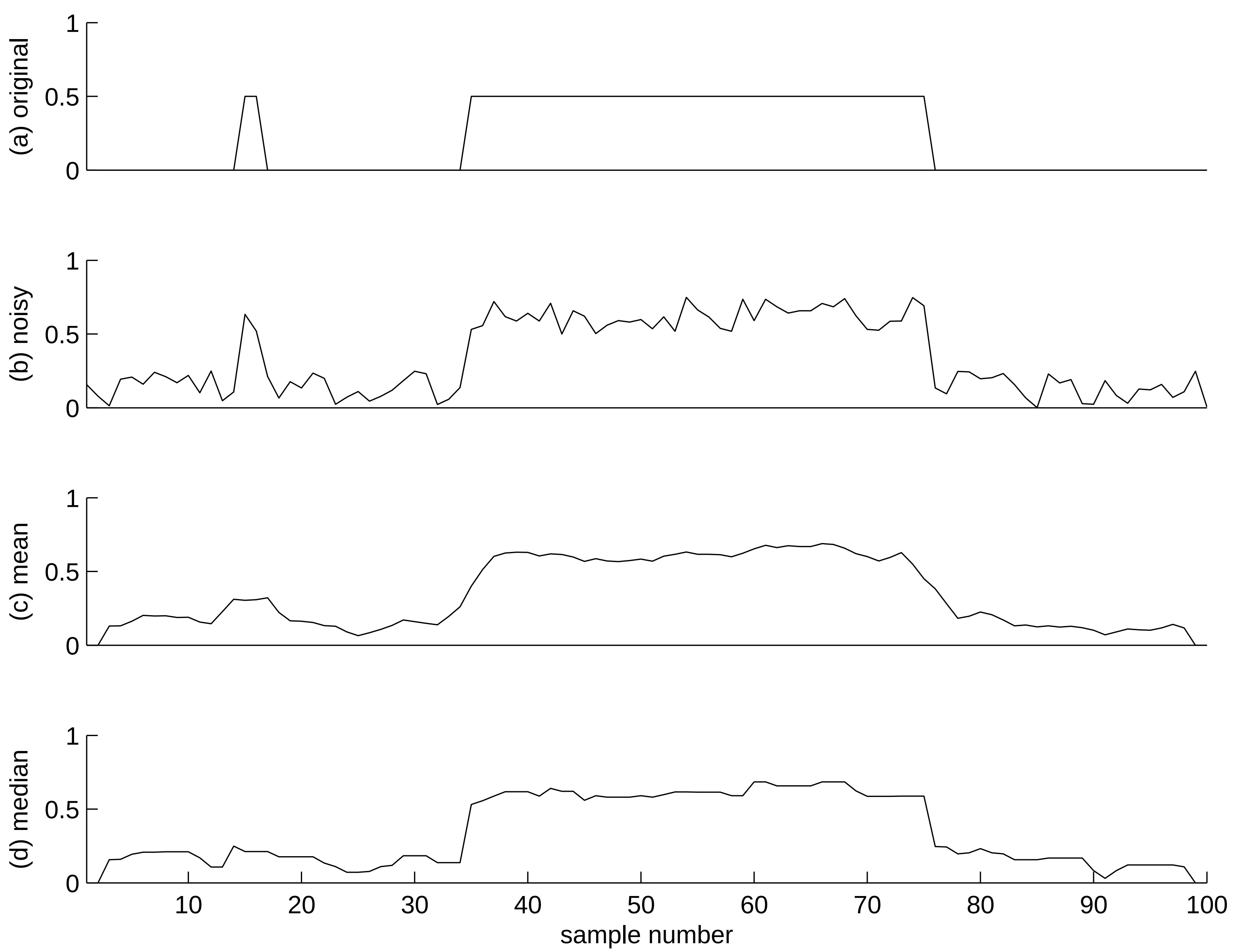


Figure 3.81: (a) A synthesized test signal with two rectangular pulses. (b) Degraded signal with uniformly distributed noise. Result of filtering the degraded signal using (c) the mean and (d) the median operation with a sliding window having $M = 5$ samples.

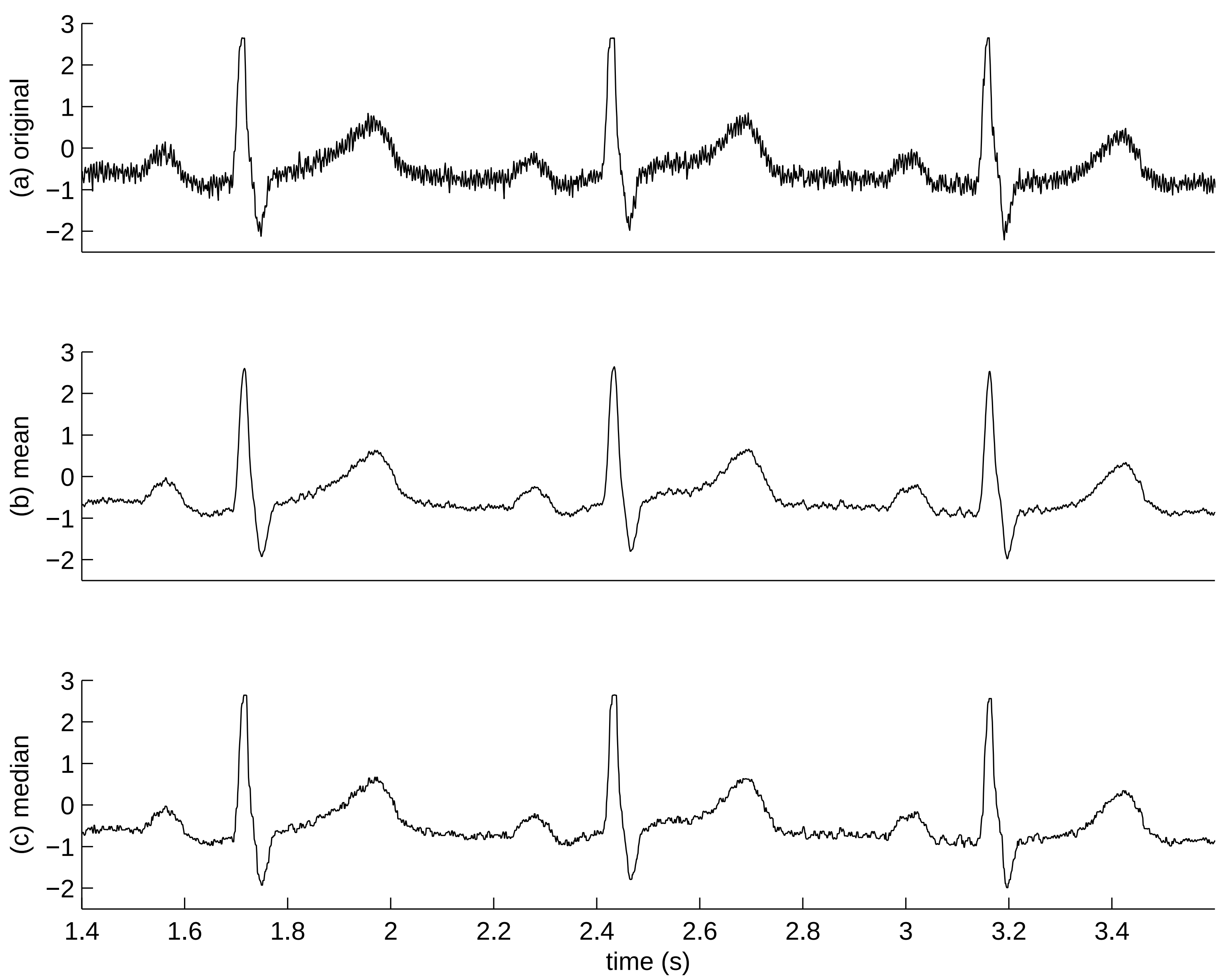


Figure 3.82: (a) An ECG signal with noise. Result of filtering the ECG signal using (b) the mean and (c) the median operation with a sliding window having $M = 9$ samples.

3.8 Optimal Filtering: The Wiener Filter

Problem: *Design an optimal filter to remove noise from a signal, given that the signal and noise processes are independent, stationary, random processes.*

You may assume the “desired” or ideal characteristics of the uncorrupted signal to be known.

The noise characteristics may also be assumed to be known.

Solution: Wiener filter theory provides for

optimal filtering by taking into account the

statistical characteristics of the signal and noise processes.

The filter parameters are *optimized* with reference to a

performance criterion.

The output is guaranteed to be the best achievable result

under the conditions imposed and the information provided.

Single-input, single-output, FIR filter with

real input signal values and real coefficients.

Figure 3.83: signal-flow diagram of a transversal filter

with coefficients or tap weights $w_i, i = 0, 1, 2, \ldots, M-1$,

input $x(n)$, and output $\tilde{d}(n)$.

Output $\tilde{d}(n)$ = an estimate of some "desired" signal

$d(n)$ that represents the ideal, uncorrupted signal.

If we assume that the desired signal is available,

estimation error between the output and the desired signal:

$$e(n) = d(n) - \tilde{d}(n). \tag{3.152}$$

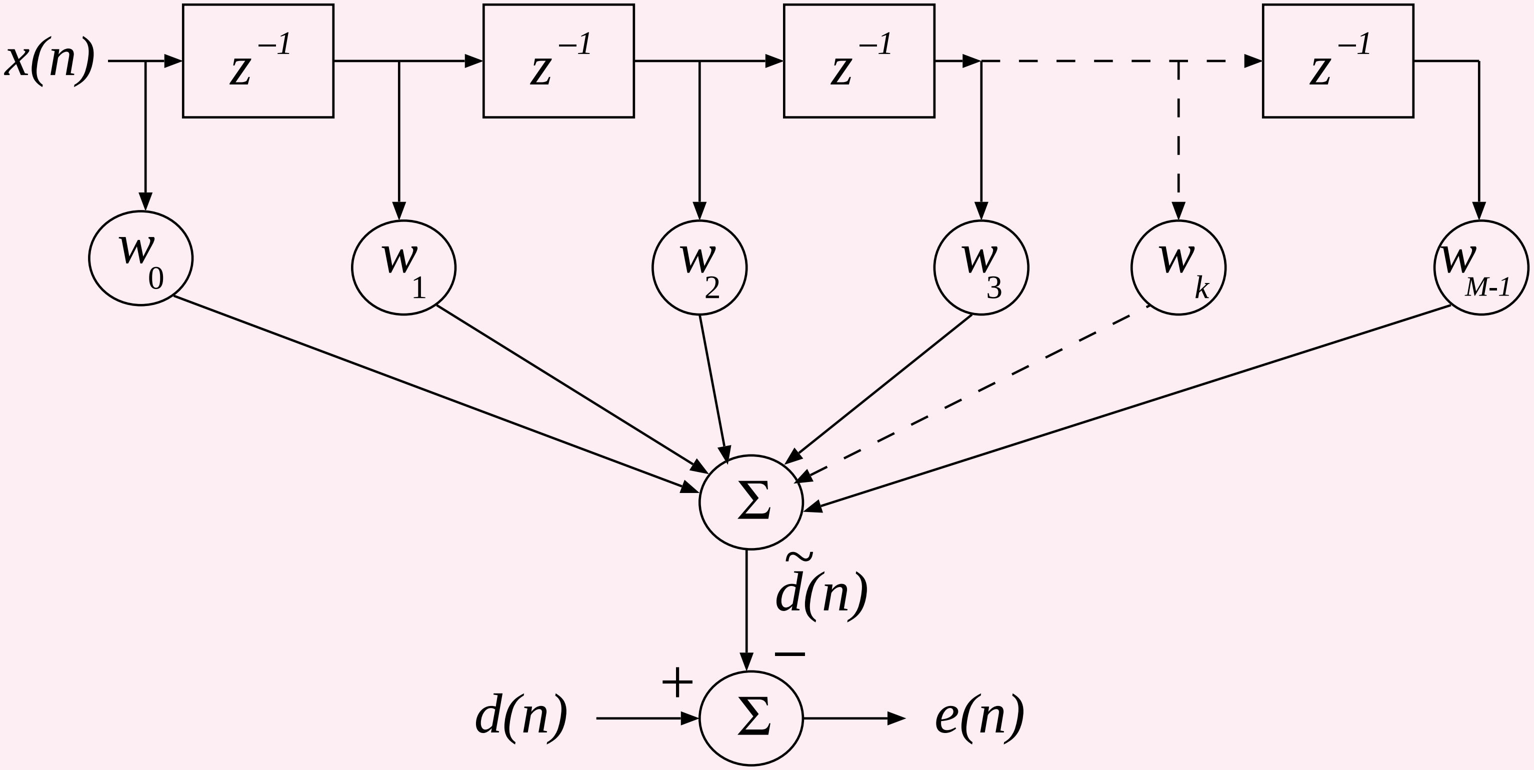


Figure 3.83: Block diagram of the Wiener filter.

$\tilde{d}(n)$ = output of a linear FIR filter

= convolution of the input $x(n)$

with the tap-weight sequence w_i:

$$\tilde{d}(n) = \sum_{k=0}^{M-1} w_k \, x(n-k). \tag{3.153}$$

w_i is also the impulse response of the filter.

For easier handling of the optimization procedures,

the tap-weight sequence may be written as an $M \times 1$

tap-weight vector:

$$\mathbf{w} = [w_0, w_1, w_2, \ldots, w_{M-1}]^T, \tag{3.154}$$

where the bold-faced character $\mathbf{w}$ represents a vector;

superscript T: vector transposition.

Tap weights convolved with M values of input.

Write the M input values as an $M \times 1$ vector:

$$\mathbf{x}(n) = [x(n), x(n-1), \ldots, x(n-M+1)]^T. \quad (3.155)$$

Vector $\mathbf{x}(n)$ varies with time: at a given instant n

the vector contains the current input sample $x(n)$ and the

preceding $(M-1)$ input samples $x(n-1)$ to $x(n-M+1)$.

Convolution in Equation 3.153 in a simpler form:

inner product or dot product of the vectors $\mathbf{w}$ and $\mathbf{x}(n)$:

$$\tilde{d}(n) = \mathbf{w}^T \mathbf{x}(n) = \mathbf{x}^T(n)\mathbf{w} = \langle \mathbf{x}, \mathbf{w} \rangle. \tag{3.156}$$

Estimation error:

$$e(n) = d(n) - \mathbf{w}^T \mathbf{x}(n). \tag{3.157}$$

Wiener filter theory estimates the tap-weight sequence that

minimizes the MS value of the estimation error;

output = *minimum mean-squared error* (MMSE) estimate

of the desired response: *optimal filter*.

Mean-squared error (MSE) defined as

$$\begin{aligned} J(\mathbf{w}) &= E[e^2(n)] \\ &= E[\{d(n) - \mathbf{w}^T \mathbf{x}(n)\}\{d(n) - \mathbf{x}^T(n)\mathbf{w}\}] \\ &= E[d^2(n)] - \mathbf{w}^T E[\mathbf{x}(n)d(n)] - E[d(n)\mathbf{x}^T(n)]\mathbf{w} \\ &+ \mathbf{w}^T E[\mathbf{x}(n)\mathbf{x}^T(n)]\mathbf{w}. \qquad (3.158) \end{aligned}$$

Expectation operator not applicable to $\mathbf{w}$.

Assumption: input vector $\mathbf{x}(n)$ and desired response $d(n)$

are jointly stationary. Then:

- $E[d^2(n)]$ = variance of $d(n)$ = σ_d^2, assuming that the mean of $d(n)$ is zero.

- $E[\mathbf{x}(n)d(n)] = M \times 1$ vector = cross-correlation between input vector $\mathbf{x}(n)$ and desired response $d(n)$:

$$\mathbf{\Theta} = E[\mathbf{x}(n)d(n)]. \tag{3.159}$$

$\mathbf{\Theta} = [\theta(0), \theta(-1), \ldots, \theta(1-M)]^T$, where

$$\theta(-k) = E[x(n-k)d(n)], \; k = 0, 1, 2, \ldots, M-1. \tag{3.160}$$

- $E[d(n)\mathbf{x}^T(n)]$ is the transpose of $E[\mathbf{x}(n)d(n)]$; therefore

$$\mathbf{\Theta}^T = E[d(n)\mathbf{x}^T(n)]. \tag{3.161}$$

- $E[\mathbf{x}(n)\mathbf{x}^T(n)]$ = autocorrelation of input vector $\mathbf{x}(n)$ computed as the outer product of the vector with itself:

$$\mathbf{\Phi} = E[\mathbf{x}(n)\mathbf{x}^T(n)] \tag{3.162}$$

or in its full $M \times M$ matrix form as $\mathbf{\Phi} =$

$$\begin{bmatrix} \phi(0) & \phi(1) & \cdots & \phi(M-1) \\ \phi(-1) & \phi(0) & \cdots & \phi(M-2) \\ \vdots & \vdots & \ddots & \vdots \\ \phi(-M+1) & \phi(-M+2) & \cdots & \phi(0) \end{bmatrix} . \tag{3.163}$$

The element in row k and column i of $\mathbf{\Phi}$ is

$$\phi(i-k) = E[x(n-k)x(n-i)], \qquad (3.164)$$

with the property that
$\phi(i-k) = \phi(k-i)$.
With the assumption of wide-sense stationarity,
the $M \times M$ matrix $\mathbf{\Phi}$ is completely specified by
the M values of the autocorrelation
$\phi(0), \phi(1), \ldots, \phi(M-1)$ for lags $0, 1, \ldots, M-1$.

MSE expression in Equation 3.158 is simplified to

$$J(\mathbf{w}) = \sigma_d^2 - \mathbf{w}^T \boldsymbol{\Theta} - \boldsymbol{\Theta}^T \mathbf{w} + \mathbf{w}^T \boldsymbol{\Phi} \mathbf{w}. \qquad (3.165)$$

MSE is a second-order function of the tap-weight vector $\mathbf{w}$.

To determine the optimal tap-weight vector, denoted by $\mathbf{w}_o$,

differentiate $J(\mathbf{w})$ with respect to $\mathbf{w}$,

set it to zero, and solve the resulting equation.

Note the following derivatives:

$$\frac{d}{d\mathbf{w}}(\mathbf{\Theta}^T\mathbf{w}) = \mathbf{\Theta},$$

$$\frac{d}{d\mathbf{w}}(\mathbf{w}^T\mathbf{\Theta}) = \mathbf{\Theta},$$

$$\frac{d}{d\mathbf{w}}(\mathbf{w}^T\mathbf{\Phi}\mathbf{w}) = 2\mathbf{\Phi}\mathbf{w}.$$

$$\frac{dJ(\mathbf{w})}{d\mathbf{w}} = -2\boldsymbol{\Theta} + 2\boldsymbol{\Phi}\mathbf{w} \rightarrow 0. \tag{3.166}$$

Condition for the optimal filter:

$$\boldsymbol{\Phi}\mathbf{w}_o = \boldsymbol{\Theta}. \tag{3.167}$$

Wiener–Hopf equation or the *normal equation*.

Optimal Wiener filter:

$$\mathbf{w}_o = \boldsymbol{\Phi}^{-1}\,\boldsymbol{\Theta}. \tag{3.168}$$

Wiener–Hopf equation in expanded form:

$$\begin{bmatrix} \phi(0) & \phi(1) & \cdots & \phi(M-1) \\ \phi(-1) & \phi(0) & \cdots & \phi(M-2) \\ \vdots & \vdots & \ddots & \vdots \\ \phi(-M+1) & \phi(-M+2) & \cdots & \phi(0) \end{bmatrix} \begin{bmatrix} w_{o0} \\ w_{o1} \\ \vdots \\ w_{o(M-1)} \end{bmatrix} =$$

$$(3.169)$$

$$\begin{bmatrix} \theta(0) \\ \theta(-1) \\ \vdots \\ \theta(1-M) \end{bmatrix} .$$

The matrix product is equivalent to the following:

$$\sum_{i=0}^{M-1} w_{oi}\, \phi(i-k) = \theta(-k),\;\; k = 0, 1, 2, \ldots, M-1. \tag{3.170}$$

Minimum MSE:

$$J_{\min} = \sigma_d^2 - \boldsymbol{\Theta}^T \boldsymbol{\Phi}^{-1} \boldsymbol{\Theta}. \tag{3.171}$$

Given the condition that the signals are stationary, we have

$\phi(i-k) = \phi(k-i)$ and $\theta(-k) = \theta(k)$.

Then, we may write Equation 3.170 as

$$\sum_{i=0}^{M-1} w_{oi}\, \phi(k-i) = \theta(k),\;\; k = 0, 1, 2, \ldots, M-1. \tag{3.172}$$

This is equivalent to the convolution relationship

$$w_{ok} * \phi(k) = \theta(k). \tag{3.173}$$

Applying the Fourier transform:

$$W(\omega) S_{xx}(\omega) = S_{xd}(\omega). \tag{3.174}$$

Note: FT(ACF) = PSD and FT(CCF) = CSD.

Wiener filter frequency response:

$$W(\omega) = \frac{S_{xd}(\omega)}{S_{xx}(\omega)} . \tag{3.175}$$

$S_{xx}(\omega)$: PSD of the input signal and

$S_{xd}(\omega)$: cross-spectral density (CSD)

between the input signal and the desired signal.

Derivation of the optimal filter requires specific knowledge

about the input $x(n)$ and the desired response $d(n)$:

autocorrelation $\mathbf{\Phi}$ of the input $x(n)$ and

cross-correlation $\mathbf{\Theta}$ between $x(n)$ and $d(n)$.

In practice, desired response $d(n)$ not known,

but it is possible to obtain an estimate of its

temporal or spectral statistics $\longrightarrow$ estimate $\mathbf{\Theta}$.

Wiener filter to remove noise:

Input $x(n)$ = desired original signal $d(n)$ + noise $\eta(n)$:

$$x(n) = d(n) + \eta(n). \tag{3.176}$$

Vector notation:

$$\mathbf{x}(n) = \mathbf{d}(n) + \boldsymbol{\eta}(n). \tag{3.177}$$

$\boldsymbol{\eta}(n)$: vector representation of noise $\eta(n)$.

Autocorrelation matrix of input:

$$\mathbf{\Phi} = E[\mathbf{x}(n)\mathbf{x}^T(n)] = E[\{\mathbf{d}(n) + \boldsymbol{\eta}(n)\}\{\mathbf{d}(n) + \boldsymbol{\eta}(n)\}^T]. \tag{3.178}$$

Assume that the noise process is statistically independent of the signal process, and that the mean of at least one of the processes is zero. Then,

$$E[\mathbf{d}(n)\boldsymbol{\eta}^T(n)] = E[\boldsymbol{\eta}^T(n)\mathbf{d}(n)] = E[\boldsymbol{\eta}(n)]E[\mathbf{d}(n)] = 0. \tag{3.179}$$

$$\boldsymbol{\Phi} = E[\mathbf{d}(n)\mathbf{d}^T(n)] + E[\boldsymbol{\eta}(n)\boldsymbol{\eta}^T(n)] = \boldsymbol{\Phi}_d + \boldsymbol{\Phi}_\eta. \quad (3.180)$$

$\boldsymbol{\Phi}_d$, $\boldsymbol{\Phi}_\eta$: $M \times M$ autocorrelation matrices of signal and noise.

$$\begin{aligned}\boldsymbol{\Theta} &= E[\mathbf{x}(n)d(n)] = E[\{\mathbf{d}(n) + \boldsymbol{\eta}(n)\}d(n)] \\ &= E[\mathbf{d}(n)d(n)] = \boldsymbol{\Phi}_{1d}. \end{aligned} \tag{3.181}$$

$\boldsymbol{\Phi}_{1d}$: $M \times 1$ autocorrelation vector of the desired signal.

Optimal Wiener filter:

$$\mathbf{w}_o = (\mathbf{\Phi}_d + \mathbf{\Phi}_\eta)^{-1} \ \mathbf{\Phi}_{1d}. \tag{3.182}$$

Frequency response of the Wiener filter:

$$S_{xx}(\omega) = S_d(\omega) + S_\eta(\omega), \tag{3.183}$$

$$S_{xd}(\omega) = S_d(\omega), \tag{3.184}$$

$$W(\omega) = \frac{S_d(\omega)}{S_d(\omega) + S_\eta(\omega)} = \frac{1}{1 + \frac{S_\eta(\omega)}{S_d(\omega)}}. \tag{3.185}$$

$S_d(\omega)$ and $S_\eta(\omega)$: PSDs of desired signal and noise.

Designing the optimal filter requires knowledge

of the PSDs of the desired signal and

the noise process (or models thereof).

Illustration of application:

Figure 3.84: ECG signal with noise.

Piecewise linear model of the desired version of the signal

obtained by concatenating linear segments similar to

P, QRS, and T waves

in amplitude, duration, and interval in given noisy signal.

Baseline of the model signal set to zero.

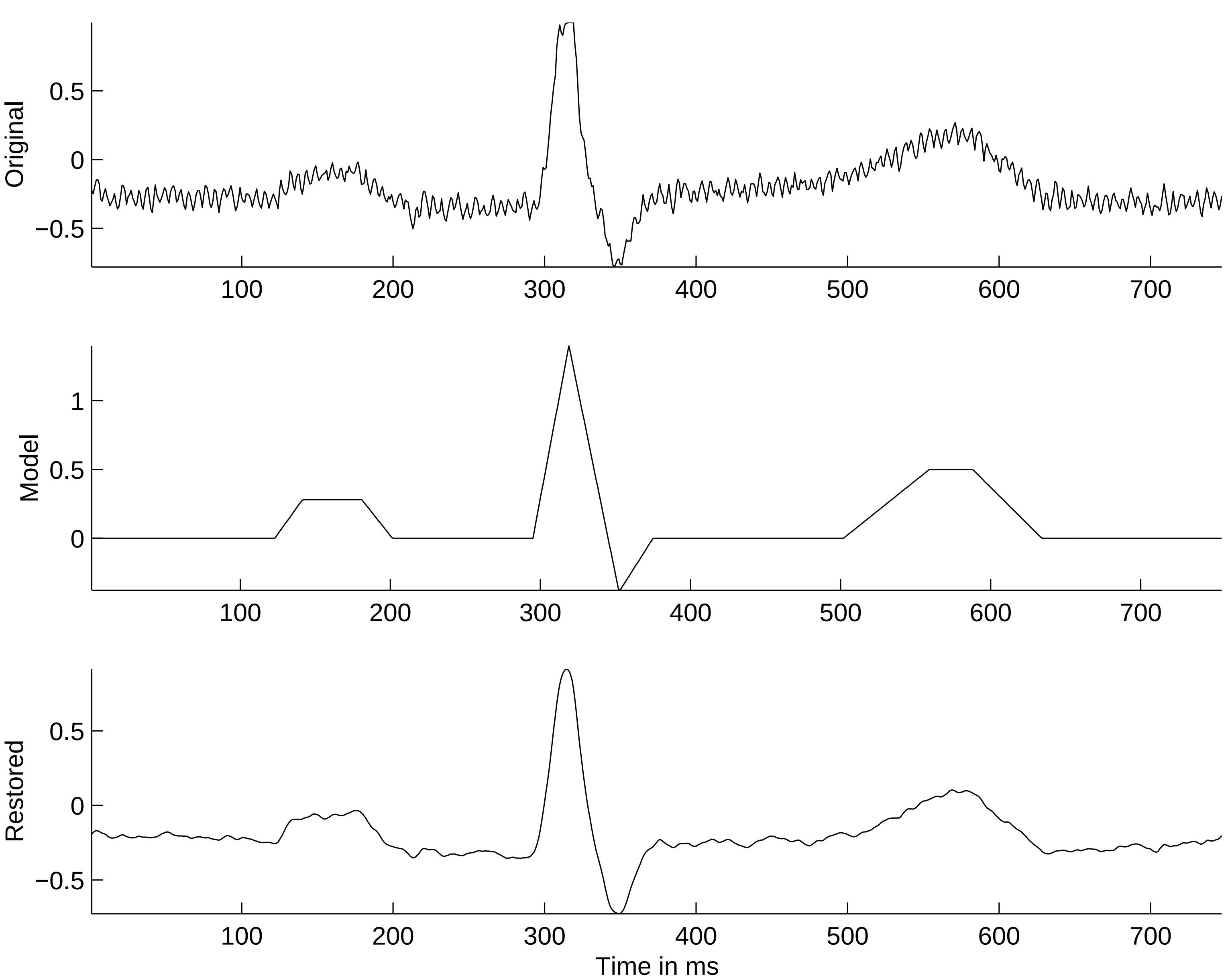


Figure 3.84: From top to bottom: one cycle of the noisy ECG signal in Figure 3.5 (labeled as Original); a piecewise linear model of the desired noise-free signal (Model); and the output of the Wiener filter (Restored).

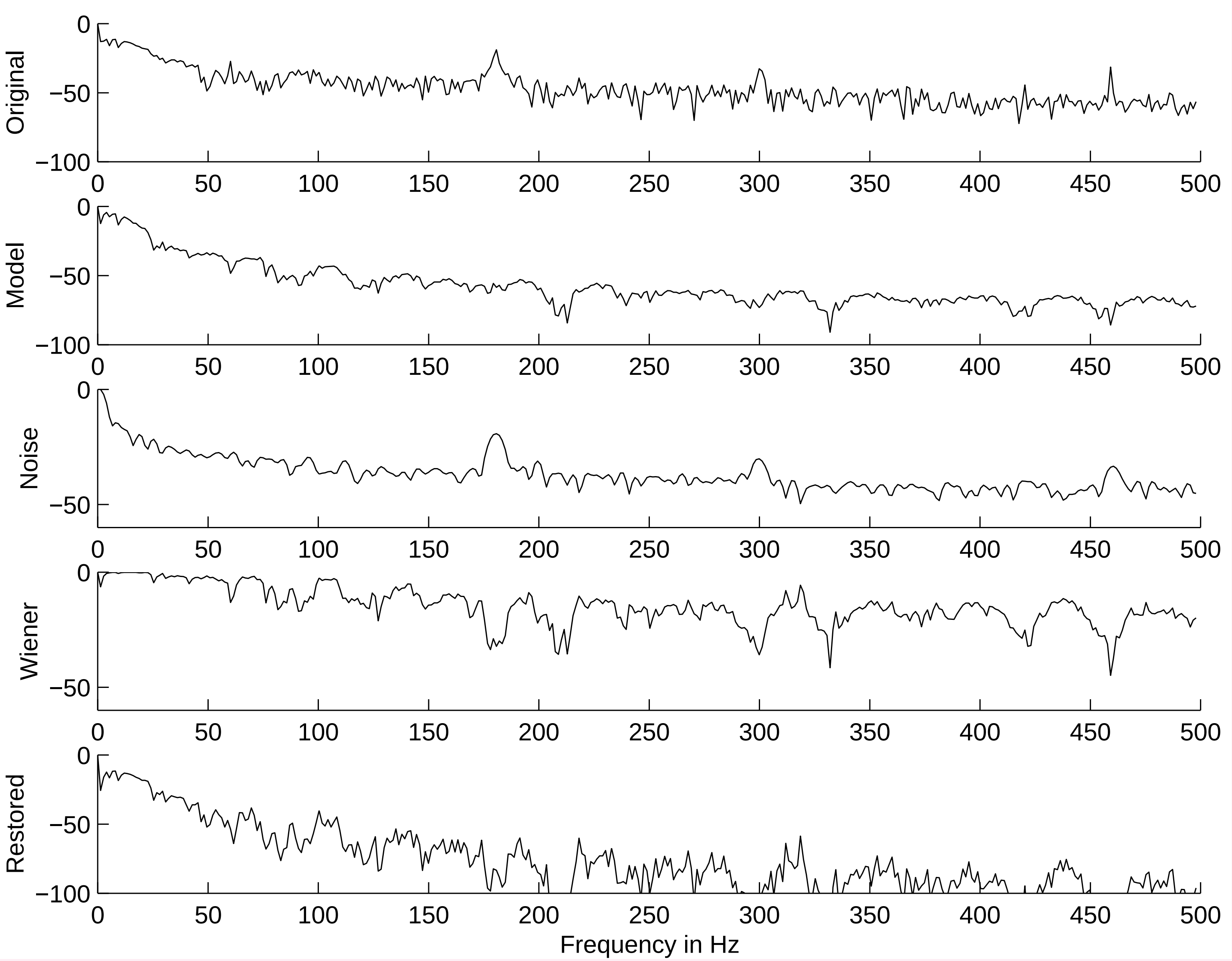


Figure 3.85: From top to bottom: log PSD (in dB) of the given noisy signal (labeled as Original); log PSD of the noise-free model (Model); estimated log PSD of the noise process (Noise); log frequency response of the Wiener filter (Wiener); and log PSD of the filter output (Restored).

T – P intervals between successive cardiac cycles:

interbeat intervals typically = isoelectric baseline.

Then, any activity in these intervals constitutes noise.

Four T – P intervals were selected from the noisy signal

in Figure 3.5, and their Fourier power spectra

were averaged to derive the noise PSD

$S_\eta(\omega)$ required in the Wiener filter (Equation 3.185).

Estimated log PSD of noise shown in Figure 3.85.

Wiener filter derived with models of noise and signal PSDs:

obtained from the given signal itself!

No cutoff frequency specified.

Another experiment: noise-free ECG signal obtained from a subject under controlled conditions; $f_s = 200\ Hz$.

One-second-long segment selected to derive the desired signal PSD (model) and design the Wiener filter.

Noisy ECG recorded from same subject including artifacts related to motion and contraction of the limbs.

Ten segments of noise selected from noisy ECG in the T–P intervals and average PSD computed.

Impulse response of Wiener filter = inverse Fourier transform of frequency response as in Equation 3.185.
Even symmetry due to real-valued frequency response.

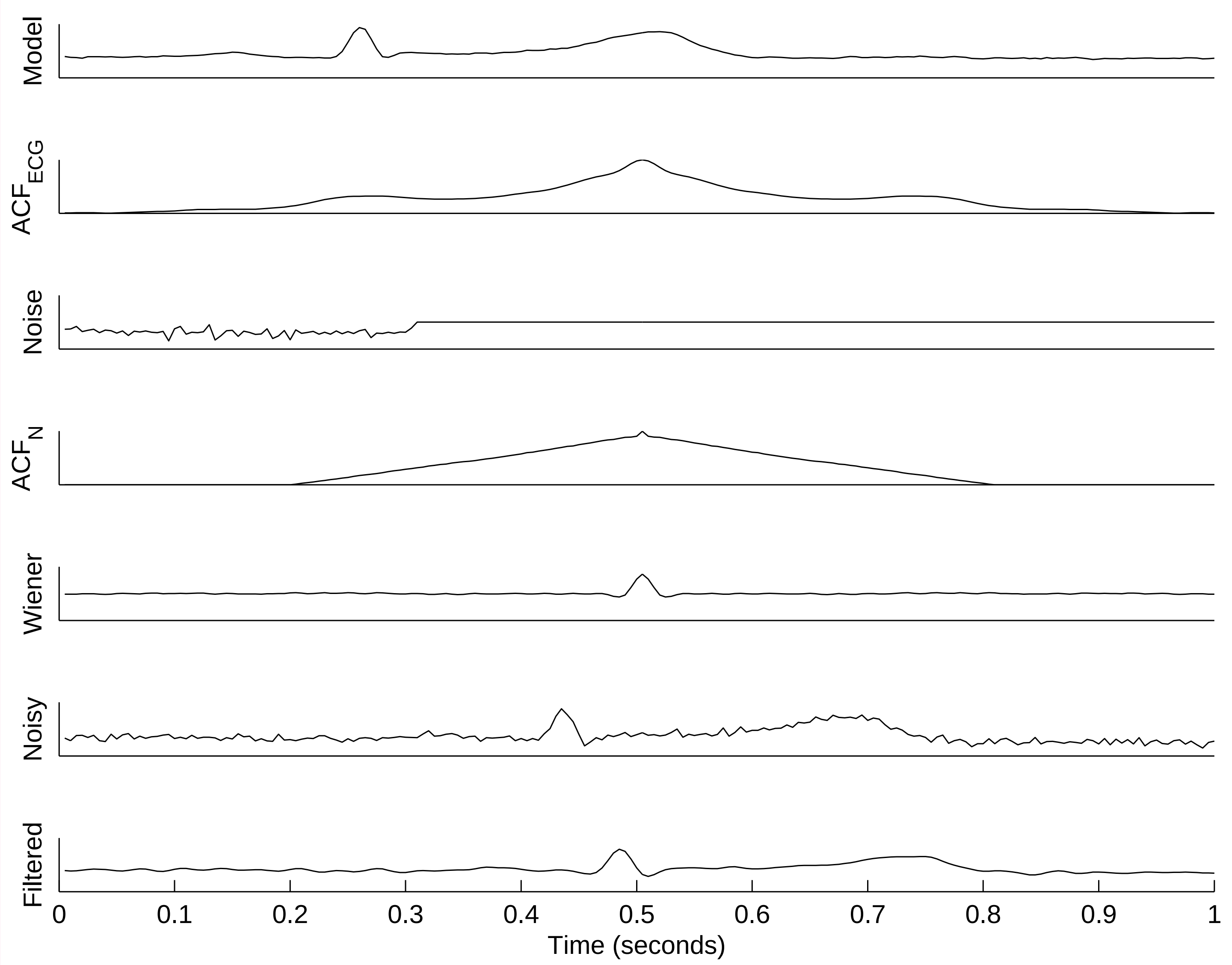


Figure 3.86: From top to bottom: one cycle of the noise-free ECG of a subject (labeled as Model); the ACF of the noise-free ECG; a sample segment of noise from a noisy ECG of the same subject (the actual noise segment has a duration of $0.3\ s$ but has been padded with zeros to the same length as the ECG, which is $1\ s$); the ACF of the noise (ACF_N) obtained using 10 segments; the impulse response of the Wiener filter (shifted for causality); a segment of the noisy ECG to be filtered; and the corresponding filtered result. The amplitude labels have been suppressed to prevent clutter. ECG signal data courtesy of Emily Marasco and Matthew LaRocque, University of Calgary. See also Figure 3.87.

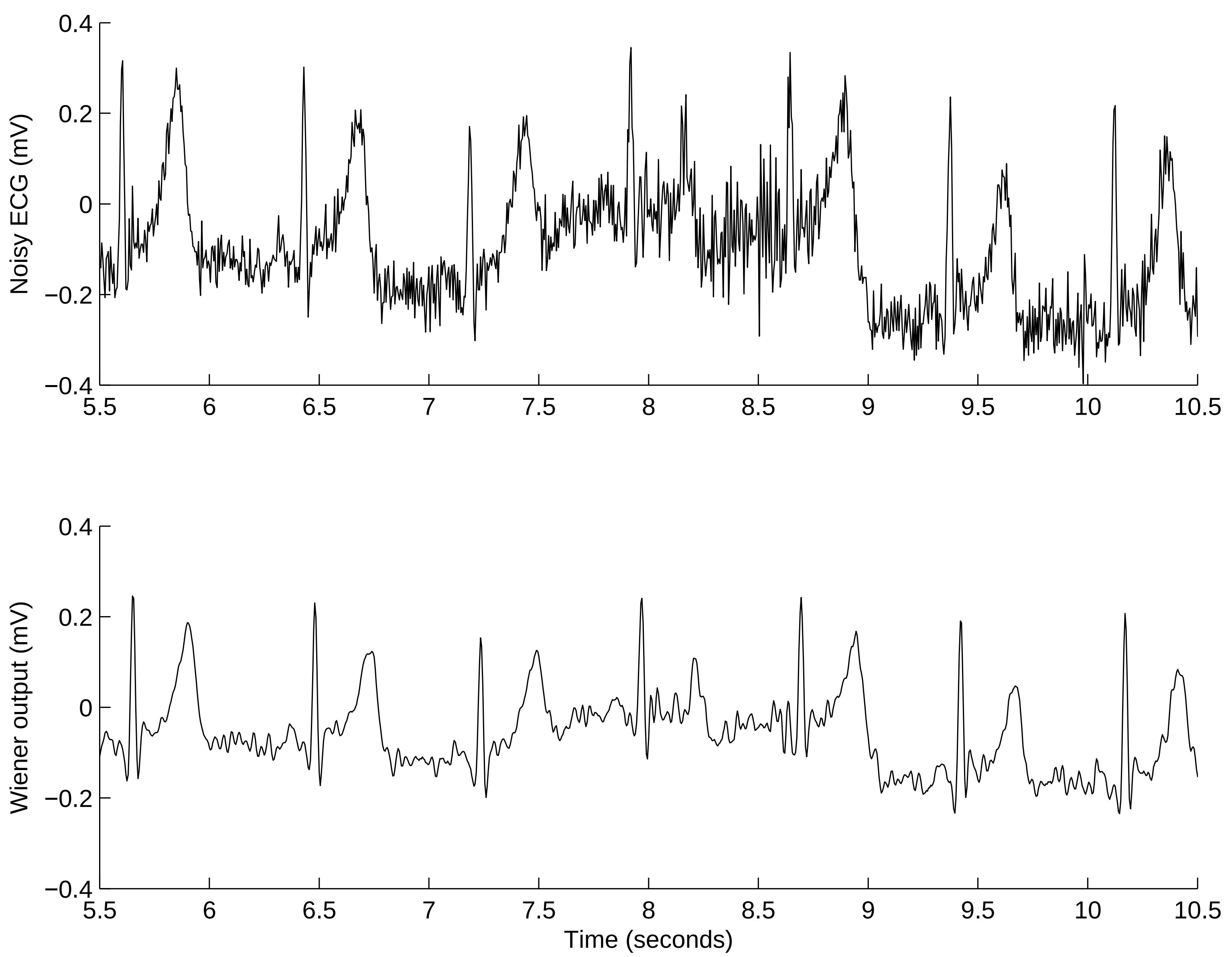


Figure 3.87: From top to bottom: a segment of the noisy ECG to be filtered and the corresponding filtered result. ECG signal data courtesy of Emily Marasco and Matthew LaRocque, University of Calgary. See also Figure 3.86.

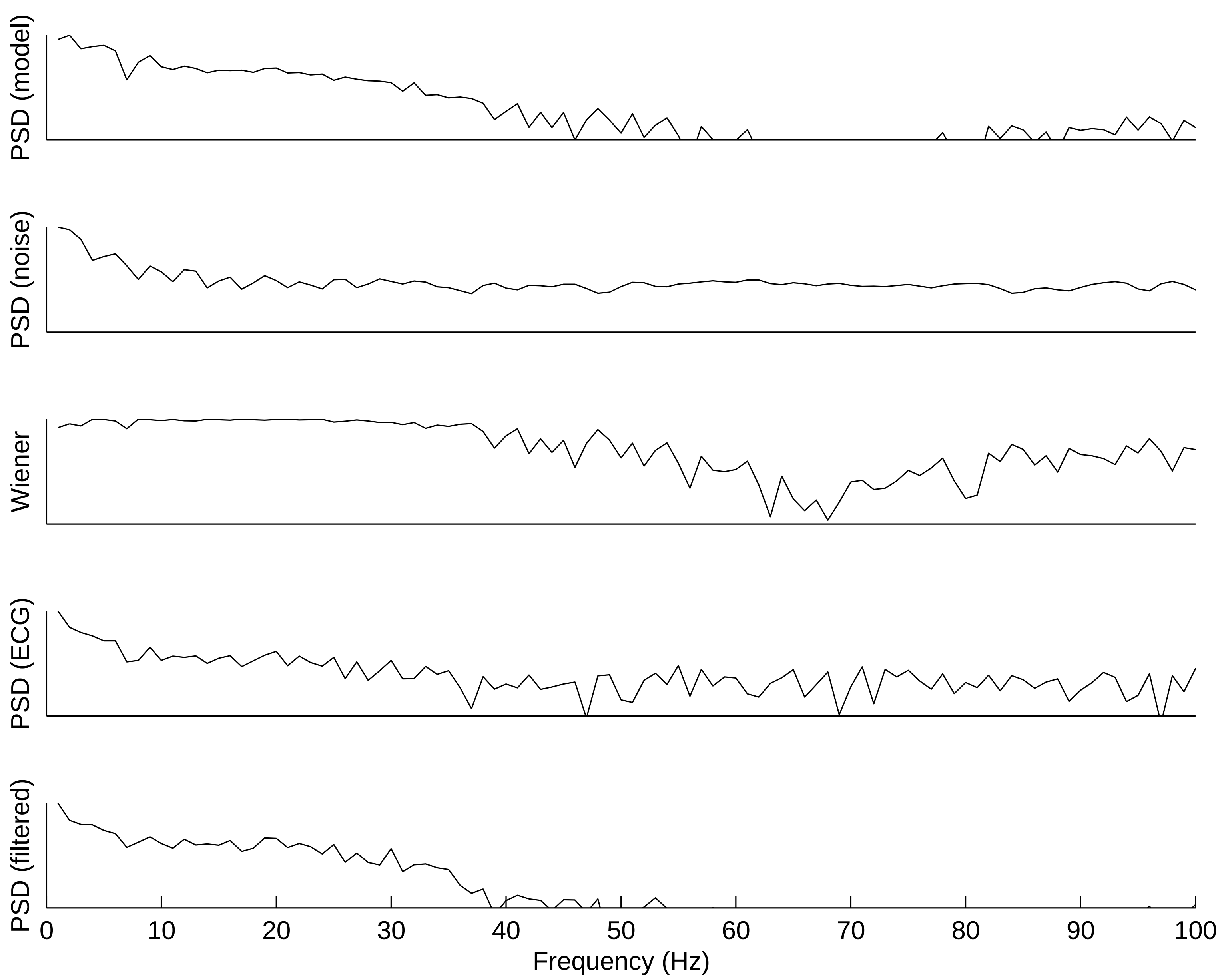


Figure 3.88: From top to bottom: log PSD (in dB) of the noise-free ECG (model); log PSD of the noise obtained using 10 segments; log frequency response of the Wiener filter; log PSD of the noisy ECG; and log PSD of the filtered result. The display of all PSDs is limited to the range $[-50, 0]$ dB. $f_s = 200$ Hz. The PSDs of ECG signals were obtained using only one cardiac cycle of each as shown in Figure 3.86.

3.9 Adaptive Filters for Removal of Interference

Filters with fixed characteristics, tap weights, or coefficients are suitable when the characteristics of the signal and noise are stationary and known.

Such filters are not applicable when the characteristics of the signal and/or noise vary with time:

when they are nonstationary.

They are also not suitable when the spectral contents of the signal and the interference overlap significantly.

Problem: *Design an optimal filter to remove a nonstationary interference from a nonstationary signal.*

An additional channel of information related to the interference is available for use.

The filter should continuously adapt to the changing characteristics of the signal and interference.

Solution:

Need to address two different concerns:

1. The filter should be *adaptive;*

   the tap-weight vector of the filter varies with time:

   adaptive filter or adaptive noise canceler (ANC).

2. The filter should be *optimal:* LMS and RLS algorithms.

3.9.1 The adaptive noise canceler

“Primary input” or observed signal $x(n)$ is a mixture of the signal of interest $v(n)$ and the “primary noise” $m(n)$:

$$x(n) = v(n) + m(n). \tag{3.186}$$

It is desired that the interference or noise $m(n)$ be estimated and removed from $x(n)$ in order to obtain the signal of interest $v(n)$.

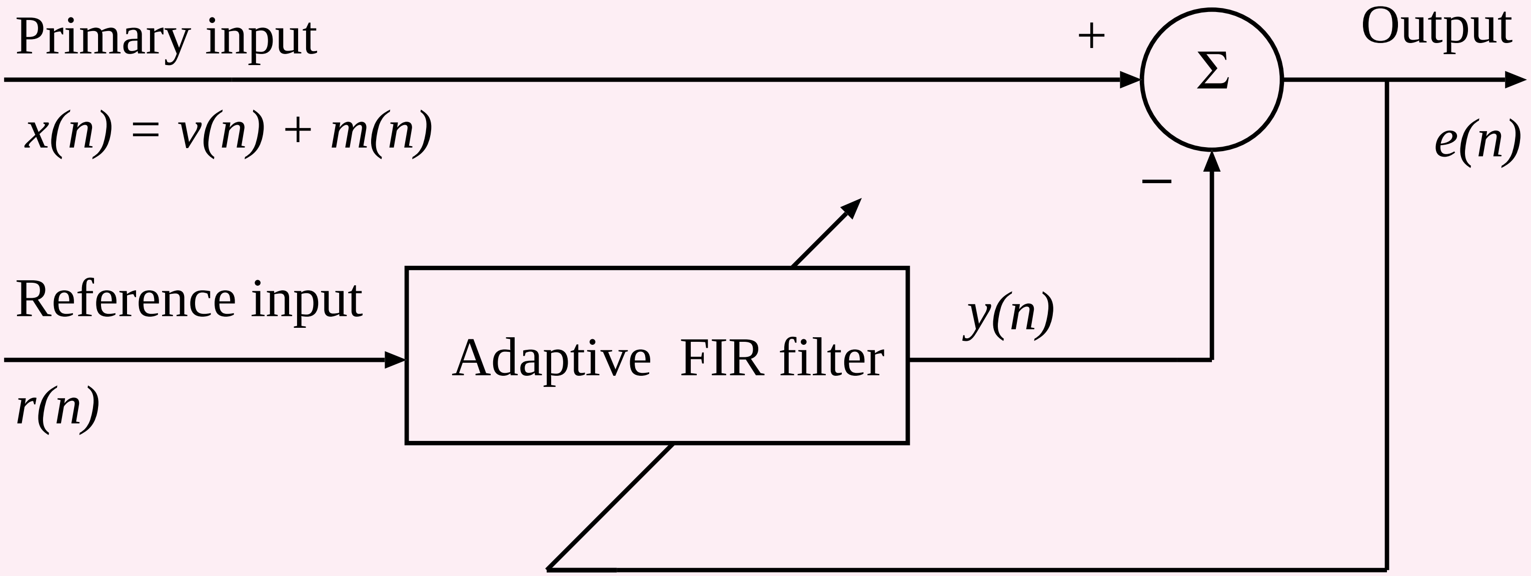


Figure 3.89: Block diagram of a generic adaptive noise canceler (ANC) or adaptive filter.

It is assumed that $v(n)$ and $m(n)$ are uncorrelated.

ANC requires a second input: “reference input” $r(n)$,

uncorrelated with the signal of interest $v(n)$

but closely related to or correlated with the interference or

noise $m(n)$ in some manner that need not be known.

The ANC filters or modifies the reference input $r(n)$

to obtain a signal $y(n)$ as close to noise $m(n)$ as possible.

$y(n)$ is subtracted from primary input

to estimate desired signal:

$$\tilde{v}(n) = e(n) = x(n) - y(n). \tag{3.187}$$

Assume that the signal of interest $v(n)$,

the primary noise $m(n)$,

the reference input $r(n)$, and the

primary noise estimate $y(n)$ are

statistically stationary and have zero means.

Note: The requirement of stationarity will be removed later

when the expectations are computed in moving windows.

$v(n)$ is uncorrelated with $m(n)$ and $r(n)$, and

$r(n)$ is correlated with $m(n)$.

$$\begin{aligned} e(n) &= x(n) - y(n) \\ &= v(n) + m(n) - y(n). \end{aligned} \tag{3.188}$$

$y(n) = \tilde{m}(n)$: estimate of the primary noise

obtained at the output of the adaptive filter.

Taking the square and expectation of both sides

of Equation 3.188:

$$E[e^2(n)] = E[v^2(n)] + E[\{m(n) - y(n)\}^2] \qquad (3.189)$$

$$+ \; 2E[v(n)\{m(n) - y(n)\}].$$

Because $v(n)$ is uncorrelated with $m(n)$ and $y(n)$ and all of them have zero means, we have

$$\begin{aligned} E[v(n)\{m(n) - y(n)\}] &= E[v(n)]\; E[m(n) - y(n)] \\ &= 0. \end{aligned} \tag{3.190}$$

$$E[e^2(n)] = E[v^2(n)] + E[\{m(n) - y(n)\}^2]. \tag{3.191}$$

Output $e(n)$ used to control the adaptive filter.

In ANC, the objective is to obtain an output $e(n)$ that is a least-squares fit to the desired signal $v(n)$.

Achieved by feeding the output back to the adaptive filter and adjusting the filter to minimize the total output power.

System output: error signal for the adaptive process.

Signal power $E[v^2(n)]$ unaffected as the filter is adjusted

to minimize $E[e^2(n)]$; the minimum output power is

$$\min\ E[e^2(n)] = E[v^2(n)] + \min\ E[\{m(n) - y(n)\}^2]. \quad (3.192)$$

As the filter is adjusted so that $E[e^2(n)]$ is minimized,

$E[\{m(n) - y(n)\}^2]$ is minimized.

Thus the filter output $y(n)$ is the

MMSE estimate of the primary noise $m(n)$.

Moreover, when $E[\{m(n) - y(n)\}^2]$ is minimized,

$E[\{e(n) - v(n)\}^2]$ is minimized,

because from Equation 3.188 we have

$$e(n) - v(n) = m(n) - y(n). \tag{3.193}$$

Adapting the filter to minimize the total output power is equivalent to causing the *output* $e(n)$ *to be the MMSE estimate of the signal of interest* $v(n)$ for the given structure and adjustability of the adaptive filter and for the given reference input.

Output $e(n)$ contains signal of interest $v(n)$ and noise.

From Equation 3.193, the output noise is given by

$$e(n) - v(n) = \tilde{v}(n) - v(n) = m(n) - y(n).$$

Minimizing $E[e^2(n)]$ minimizes $E[\{m(n) - y(n)\}^2]$;

therefore

minimizing the total output power

minimizes the output noise power.

Because the signal component $v(n)$

in the output remains unaffected,

minimizing the total output power

maximizes the output SNR.

Note from Equation 3.191 that the

output power is minimum when $E[e^2(n)] = E[v^2(n)]$.

When this condition is achieved, $E[\{m(n) - y(n)\}^2] = 0$.

We then have $y(n) = m(n)$ and $e(n) = v(n)$:

then, the output is a perfect and noise-free

estimate of the desired signal.

Optimization of the filter may be performed by

expressing the error in terms of the tap-weight vector

and applying the procedure of choice.

$$y(n) = \sum_{k=0}^{M-1} w_k \, r(n-k). \tag{3.194}$$

w_k, $k = 0, 1, 2, \ldots, M-1$, are the tap weights,

and M is the order of the filter.

Estimation error $e(n)$ or output of ANC:

$$e(n) = x(n) - y(n). \tag{3.195}$$

Define the tap-weight vector at time n as

$$\mathbf{w}(n) = [w_0(n), w_1(n), \ldots, w_{M-1}(n)]^T. \tag{3.196}$$

Tap-input vector at each time instant n:

$$\mathbf{r}(n) = [r(n), r(n-1), \ldots, r(n-M+1)]^T. \qquad (3.197)$$

Then, estimation error $e(n)$:

$$e(n) = x(n) - \mathbf{w}^T(n)\,\mathbf{r}(n). \qquad (3.198)$$

3.9.2 The least-mean-squares adaptive filter

Adjust the tap-weight vector to minimize the MSE.

Squaring the estimation error $e(n)$ in Equation 3.198:

$$\begin{aligned} e^2(n) &= x^2(n) - 2\,x(n)\,\mathbf{r}^T(n)\,\mathbf{w}(n) \\ &+ \mathbf{w}^T(n)\,\mathbf{r}(n)\,\mathbf{r}^T(n)\,\mathbf{w}(n). \end{aligned} \tag{3.199}$$

Squared error is a second-order or quadratic function of the tap-weight vector and the inputs, and may be depicted as a concave hyperparaboloidal, bowl-like surface.

Aim of optimization: reach the bottom of the bowl-like function.

Gradient-based methods may be used for this purpose.

Taking the expected values of the entities in Equation 3.199

and the derivative with respect to the tap-weight vector,

we may derive the Wiener–Hopf equation for the ANC.

The LMS algorithm takes a simpler approach:

assume the square of the instantaneous error

in Equation 3.199 to stand for an estimate of the MSE.

LMS algorithm based on the method of steepest descent:

new tap-weight vector $\mathbf{w}(n+1)$ given by the

present tap-weight vector $\mathbf{w}(n)$ plus a

correction proportional to the negative of the

gradient $\nabla e^2(n)$ of the squared error:

$$\mathbf{w}(n+1) = \mathbf{w}(n) - \mu\, \nabla e^2(n). \tag{3.200}$$

Parameter μ controls stability and rate of convergence:

larger the value of μ, larger is the gradient of the error

that is introduced, and the faster is the convergence.

LMS algorithm approximates $\nabla e^2(n)$ by the

derivative of the squared error in Equation 3.199

with respect to the tap-weight vector as

$$\begin{aligned}\tilde{\nabla} e^2(n) &= -2\, x(n)\, \mathbf{r}(n) + 2\, \{\mathbf{w}^T(n)\, \mathbf{r}(n)\}\, \mathbf{r}(n) \\ &= -2\, e(n)\, \mathbf{r}(n). \qquad (3.201)\end{aligned}$$

Using this estimate of the gradient in Equation 3.200, we get

$$\mathbf{w}(n+1) = \mathbf{w}(n) + 2\,\mu\;e(n)\;\mathbf{r}(n). \tag{3.202}$$

This is known as the Widrow–Hoff LMS algorithm.

Advantages of LMS algorithm:

simplicity and ease of implementation.

Although the method is based on the MSE and gradient-based optimization, the filter expression itself is free of differentiation, squaring, or averaging.

Illustration of application:

Zhang et al. used a two-stage adaptive LMS filter to cancel muscle-contraction interference from VAG signals.

First stage: to remove measurement noise;

second stage: to cancel the muscle signal.

Optimization of step size μ by using an RMS-error-based misadjustment factor and a time-varying estimate of input signal power.

$$\mathbf{w}(n+1) = \mathbf{w}(n) + 2\,\mu(n)\;e(n)\;\mathbf{r}(n). \qquad (3.203)$$

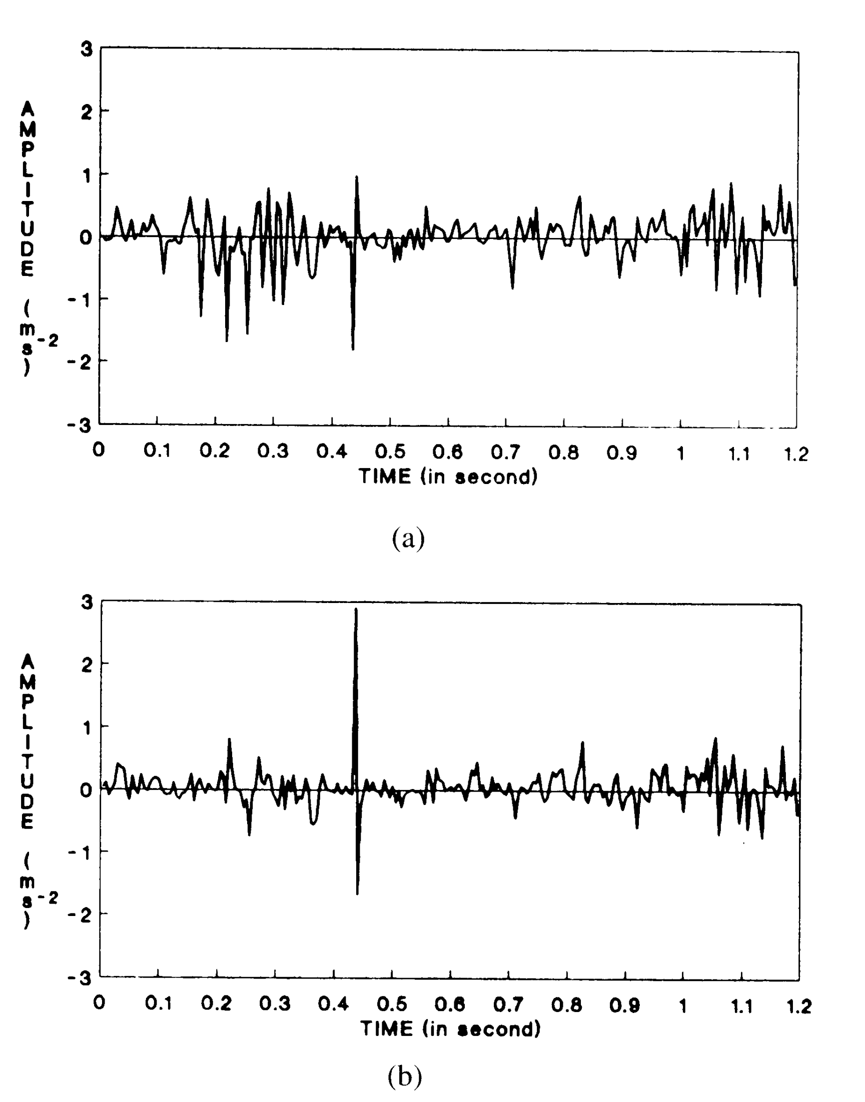
$$\mu(n) = \frac{\mu}{(M+1)\;\bar{x}^2(n)\,(\alpha, r(n), \bar{x}^2(n-1))}, \qquad (3.204)$$

$0 < \mu < 1$.

$\bar{x}^2(n)$ time-varying function of $\alpha, r(n)$, and $\bar{x}^2(n-1)$:

$\bar{x}^2(n) = \alpha\; r^2(n) + (1-\alpha)\;\bar{x}^2(n-1)$.

Forgetting factor α in the adaptation process, $0 \leq \alpha \ll 1$.



(a)

(b)

Figure 3.90: LMS-filtered versions of the VAG signals recorded from the mid-patella and the tibial tuberosity positions, as shown in Figure 3.11 (traces (b) and (c), right-hand column). The muscle-contraction signal recorded at the distal rectus femoris position was used as the reference input (Figure 3.11, right-hand column, trace (a)). The recording setup is shown in Figure 3.10. Reproduced with permission from Y.T Zhang, R.M. Rangayyan, C.B. Frank, and G.D. Bell, Adaptive cancellation of muscle-contraction interference from knee joint vibration signals, *IEEE Transactions on Biomedical Engineering,* 41(2):181–191, 1994. ©IEEE.

3.9.3 The recursive least-squares adaptive filter

Not included in this course.

3.10 Selecting an Appropriate Filter

Synchronized or ensemble averaging is possible when:

- The signal is statistically stationary, (quasi)periodic, or cyclostationary.
- Multiple realizations of signal of interest available.
- A trigger point or time marker is available, or can be derived to extract and align the copies of the signal.
- The noise is a stationary random process that is uncorrelated with the signal and has a zero mean (or a known mean).

Temporal MA filtering is suitable when:

- The signal is statistically stationary at least over the duration of the moving window.
- The noise is a zero-mean random process that is stationary at least over the duration of the moving window and is independent of the signal.
- The signal is a slow (low-frequency) phenomenon.
- Fast, on-line, real-time filtering is desired.

Frequency-domain fixed filtering is applicable when:

- The signal is statistically stationary.
- The noise is a stationary random process that is statistically independent of the signal.
- The signal spectrum is limited in bandwidth compared to that of the noise (or vice-versa).
- Loss of information in the spectral band removed by the filter does not seriously affect the signal.
- On-line, real-time filtering is not required (if implemented in the spectral domain via the Fourier transform).

The optimal Wiener filter can be designed if:

- The signal is statistically stationary.
- The noise is a stationary random process that is statistically independent of the signal.
- Specific details (or models) are available regarding the ACFs or the PSDs of the signal and noise.

Adaptive filtering is called for and possible when:

- The noise or interference is not stationary and not necessarily a random process.
- The noise is uncorrelated with the signal.
- No information is available about the spectral characteristics of the signal and noise, which may also overlap significantly.
- A second source or recording site is available to obtain a reference signal strongly correlated with the noise but uncorrelated with the signal.

An adaptive filter acts as a fixed filter

when the signal and noise are stationary.

An adaptive filter can also act as a notch or

a comb filter when the interference is periodic.

All of the filters mentioned above are applicable

only when the noise is additive.

Homomorphic filtering may be used as a preprocessing step

if signals are combined with operations other than addition.

3.11 Application: Removal of Artifacts in ERP Signals

Kamath et al. applied synchronized averaging to improve the SNR of cortical evoked potentials or ERPs related to electrical and mechanical stimulation of the esophagus.

They observed that habituation took place as the number of stimuli was increased beyond a certain limit;

use of the ERPs obtained after habituation in averaging led to a reduction in the SNR.

Let $y_k(n)$ represent one realization or observation of an ERP, with $k = 1, 2, \ldots, M$ representing the ensemble index, and $n = 0, 1, 2, \ldots, N-1$ representing the time-sample index.

$M =$ number of realizations of the ERP available;
$N =$number of the time samples in each signal.

$$y_k(n) = x_k(n) + \eta_k(n). \tag{3.228}$$

Result of ensemble averaging for $n = 0, 1, 2, \ldots, N-1$:

$$\begin{aligned} \bar{y}(n) &= \frac{1}{M} \sum_{k=1}^{M} y_k(n) \\ &= \frac{1}{M} \sum_{k=1}^{M} x_k(n) + \frac{1}{M} \sum_{k=1}^{M} \eta_k(n). \end{aligned} \tag{3.229}$$

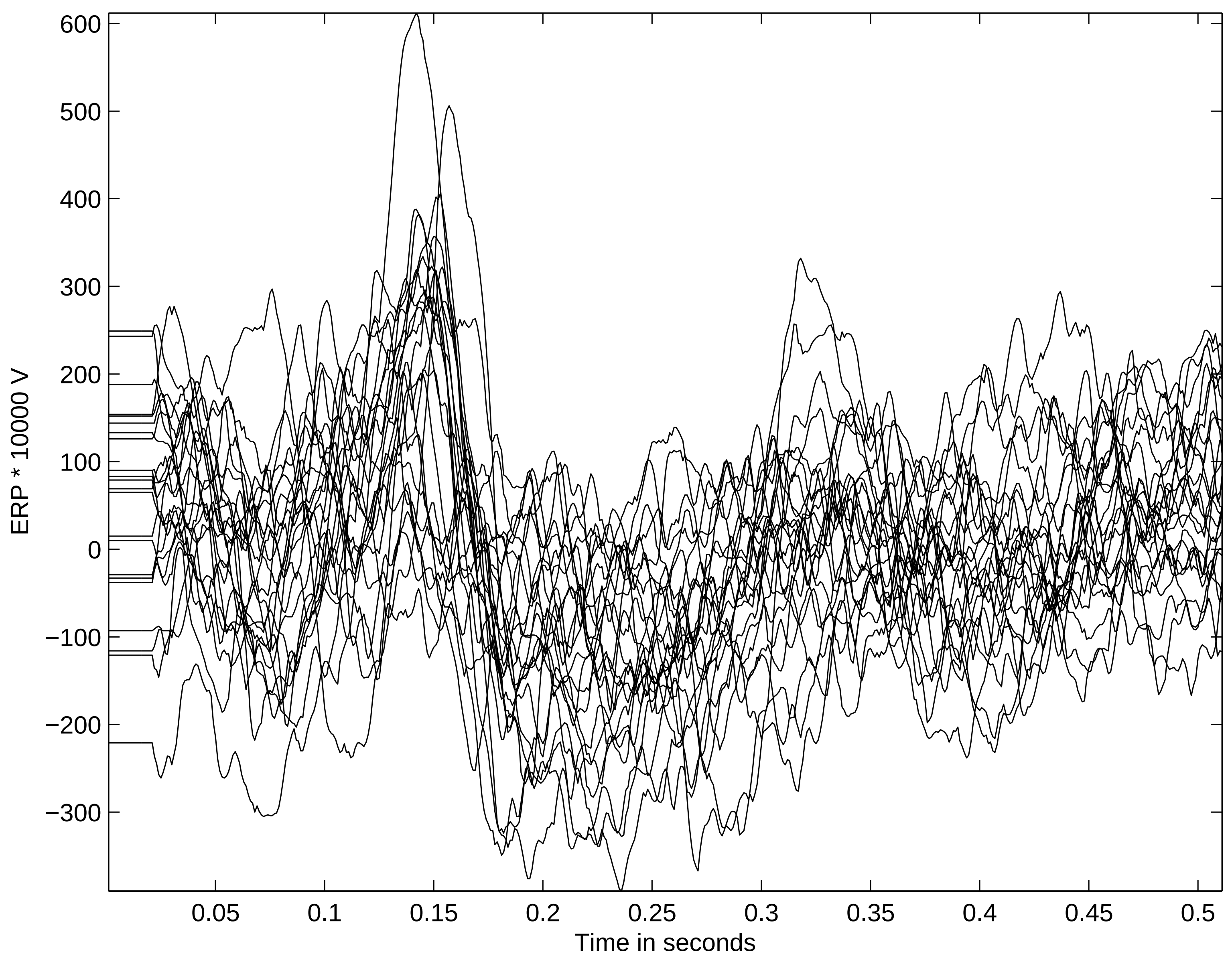


Figure 3.96: Superimposed plots of 24 cortical ERPs related to electrical stimulation of the esophagus. Data courtesy of M.V. Kamath, McMaster University, Hamilton, ON, Canada.

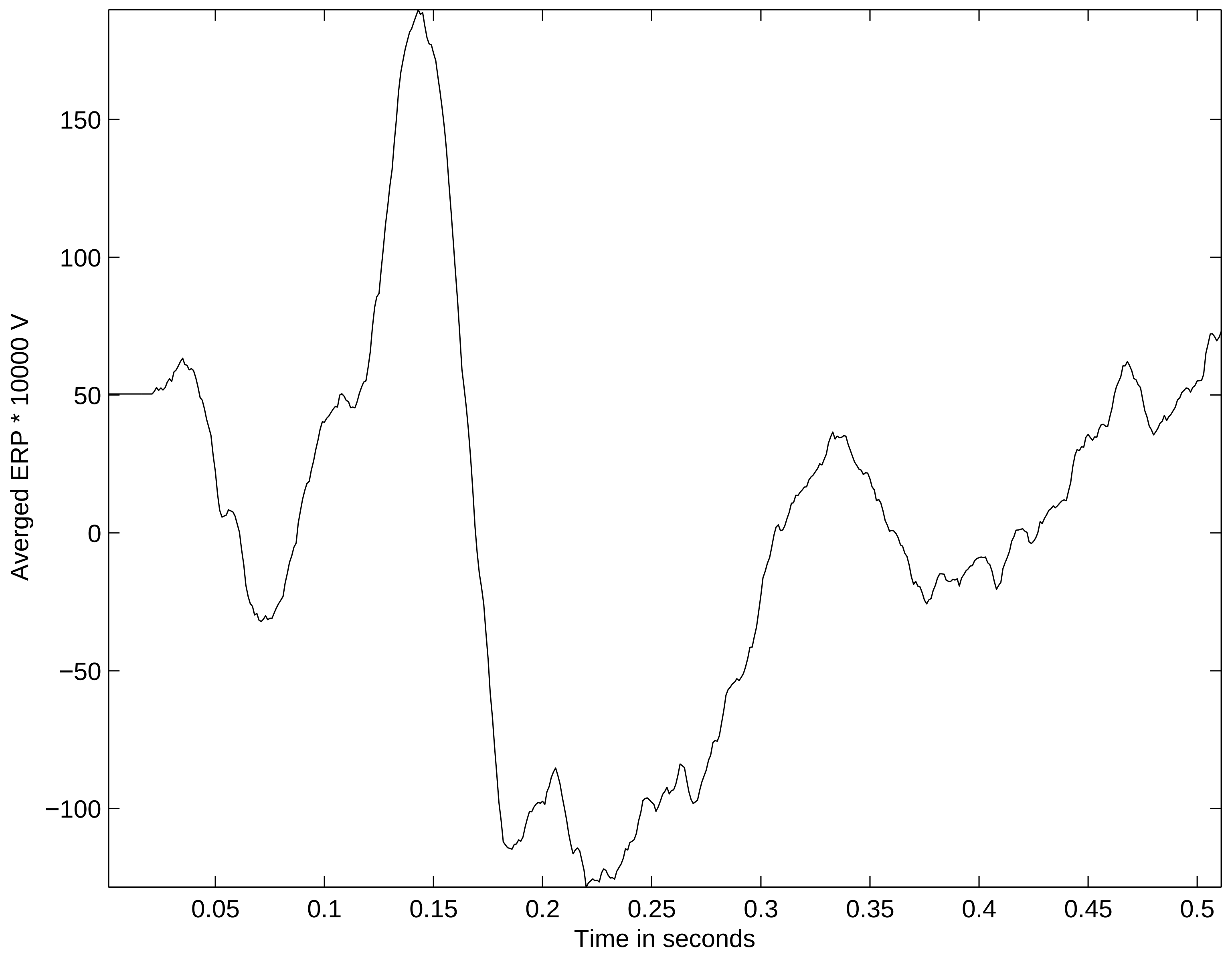


Figure 3.97: Result of synchronized averaging of all of the 24 ERPs shown in Figure 3.96.

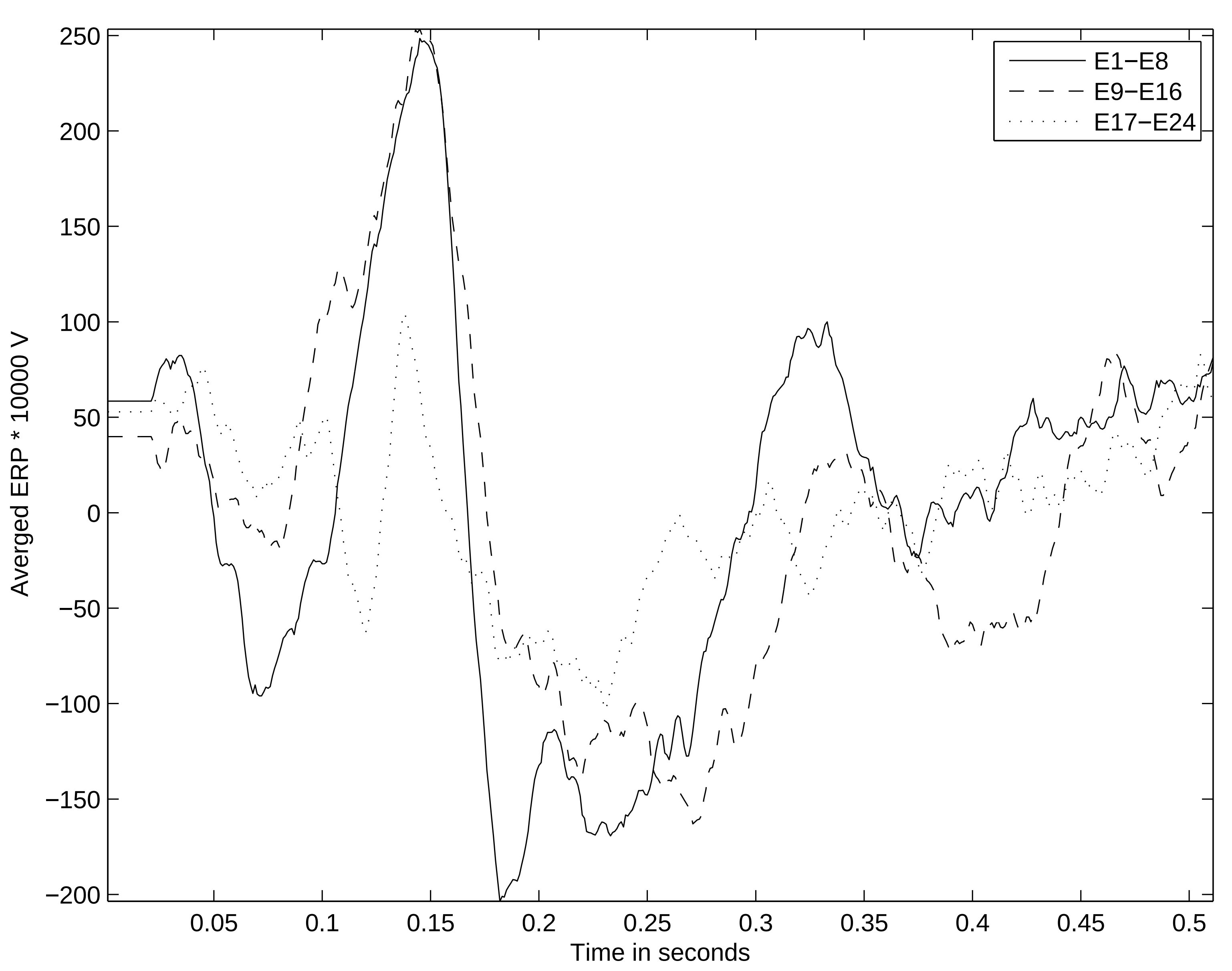


Figure 3.98: Results of synchronized averaging of selected sets of the ERPs shown in Figure 3.96. The three plots shown were obtained by averaging ERPs in groups of eight, for $k = 1, 2, \ldots, 8$ (solid line); $k = 9, 10, \ldots, 16$ (dashed line); and $k = 17, 18, \ldots, 24$ (dotted line).

Section 3.5.1, we saw that synchronized averaging can, theoretically, improve the SNR by the factor of $\sqrt{M}$.

Noise power in the output:

$$\sigma_\eta^2 = \frac{1}{NT(M-1)} \sum_{k=1}^{M} \sum_{n=0}^{N-1} [y_k(n) - \bar{y}(n)]^2 . \qquad (3.230)$$

$T = 0.001\ s$ is the sampling interval.

Signal power in the output:

$$\sigma_{\bar{y}}^2 = \frac{1}{NT} \left\{ \sum_{n=0}^{N-1} [\bar{y}(n)]^2 \right\} - \frac{\sigma_\eta^2}{M} . \qquad (3.231)$$

$$SNR = \frac{\sigma_{\bar{y}}^2}{\sigma_\eta^2} . \qquad (3.232)$$

Euclidean distance between the original ERP signals and the averaged signal:

$$D = \frac{1}{M} \sum_{k=1}^{M} \sqrt{\sum_{n=0}^{N-1} [y_k(n) - \bar{y}(n)]^2}\,. \qquad (3.233)$$

SNR for the average of all of the 24 ERPs: 0.37.

SNRs for the results of averaging the first, second, and third sets of eight ERPs: 0.73, 0.65, and 0.06.

Repeated stimulation of a biological system could lead to habituation and fatigue, which could cause the response to be substantially different.

3.12 Application: Removal of Artifacts in the ECG

Problem: *Figure 3.99 (top trace) shows an ECG signal with a combination of baseline drift, high-frequency noise, and power-line interference.*

Design filters to remove the artifacts.

Solution: HPF, LPF, and comb filter in series (cascade).

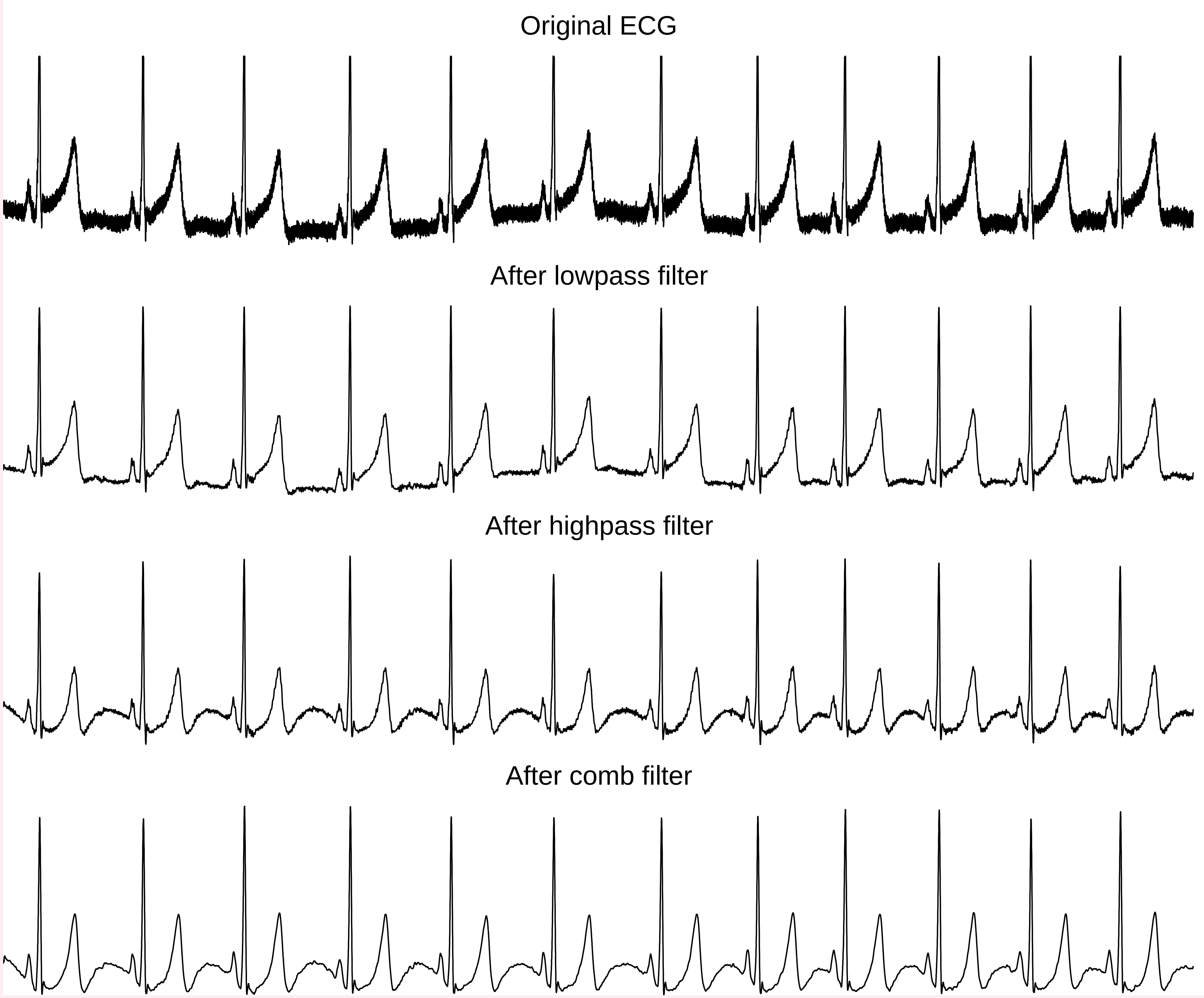


Figure 3.99: ECG signal with a combination of artifacts and its filtered versions. The duration of the signal is 10.7 s.

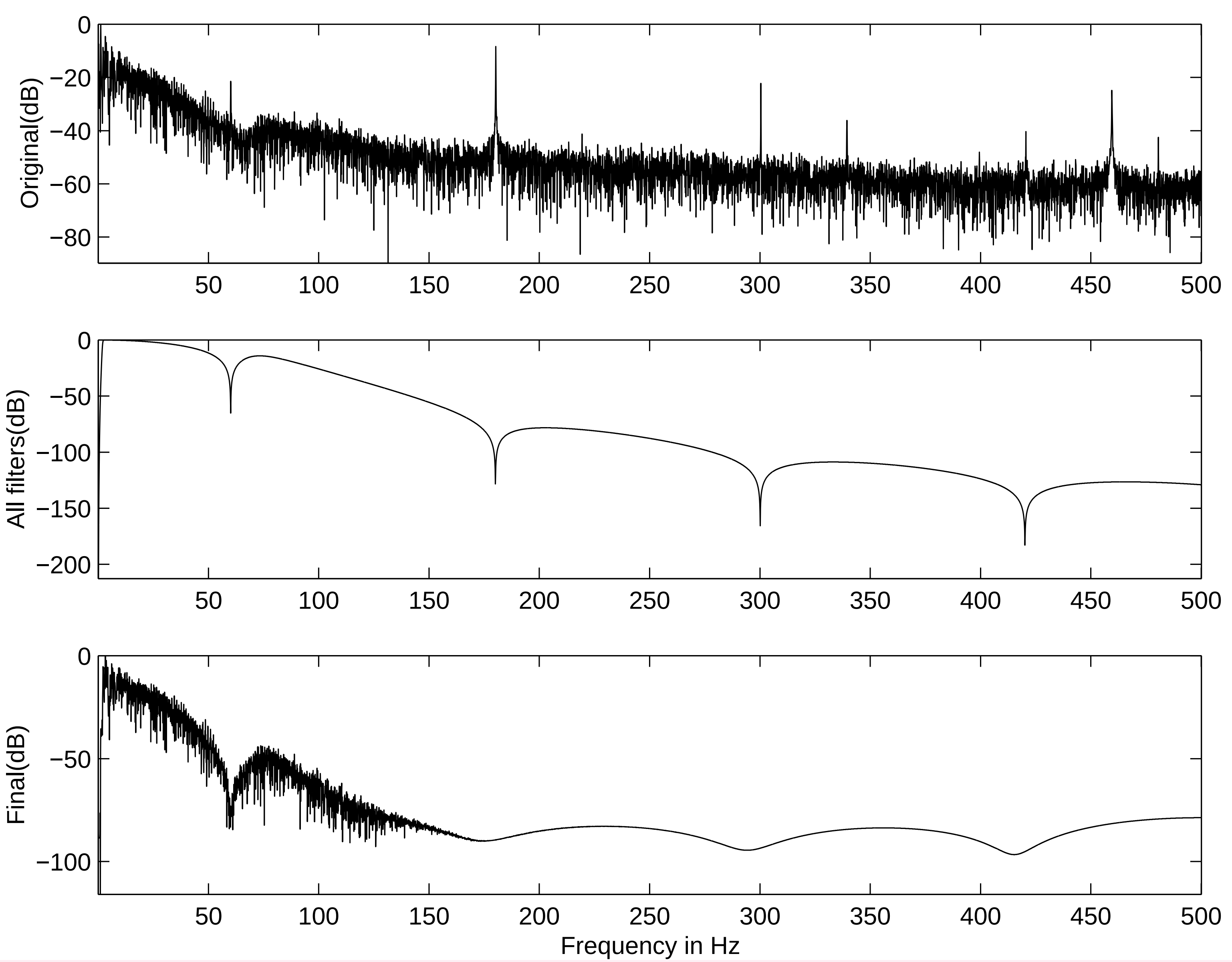


Figure 3.100: Top and bottom plots: Power spectra of the ECG signals in the top and bottom traces of Figure 3.99. Middle plot: Frequency response of the combination of lowpass, highpass, and comb filters. The cutoff frequency of the highpass filter is $2\ Hz$; the highpass portion of the frequency response is not clearly seen in the plot.

3.13 Application: Adaptive Cancellation of the Maternal ECG to Obtain the Fetal ECG

Problem: *Propose an adaptive noise cancellation filter to remove the maternal ECG signal from the abdominal-lead ECG shown in Figure 3.9 to obtain the fetal ECG.*

Chest-lead ECG signals of the mother may be used for reference.

Solution: Widrow et al. proposed a multiple-reference

ANC for removal of the maternal ECG in order

to obtain the fetal ECG.

Combined ECG obtained from a single abdominal lead.

Maternal ECG was obtained via four chest leads.

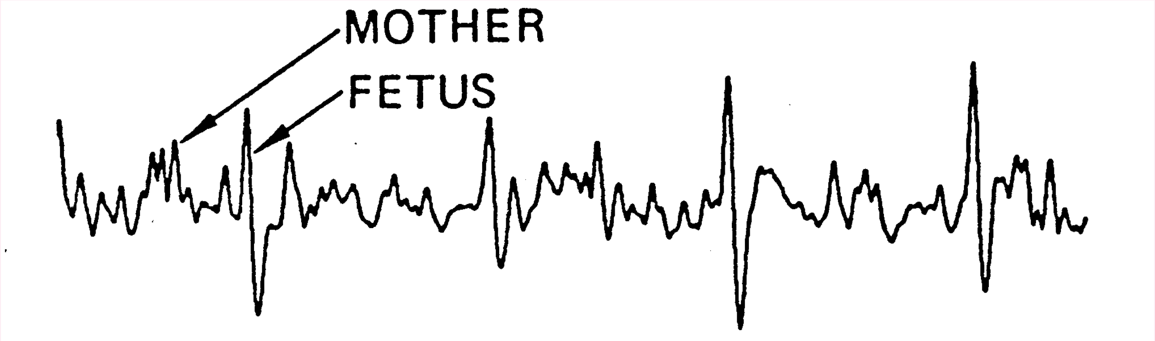


Figure 3.101: Result of adaptive cancellation of the maternal chest ECG from the abdominal ECG in Figure 3.9. The QRS complexes extracted correspond to the fetal ECG. Reproduced with permission from B. Widrow, J.R. Glover, Jr., J.M. McCool, J. Kaunitz, C.S. Williams, R.H. Hearn, J.R. Zeidler, E. Dong, Jr., R.C. Goodlin, Adaptive noise cancelling: Principles and applications, Proceedings of the IEEE, 63(12):1692–1716, 1975. ©IEEE.