Complement on Digital Spectral Analysis and Optimal Filtering: Theory and Exercises

Random Signals Analysis (MVE136)

Mats Viberg
Department of Signals and Systems
Chalmers University of Technology
412 96 Göteborg, Sweden

Sept. 2011

1 Introduction

The purpose of this document is to provide some complementing material on digital statistical signal processing, and in particular spectral analysis and optimal filtering. The theory complements that in Chapters 10 and 11 of [2, 3]. While the presentation in [2, 3] is largely based on continuous-time, most practical implementations use digital signal processing. The available sampling rates (and thereby signal bandwidths) is continuously increasing, and today it is feasible with real-time digital signal processing even at GHz bandwidths.

2 Continuous-Time Signals

Most signals we deal with are sampled versions of continuous-time stochastic processes. It is therefore natural to define many signal properties for the continuous-time "physical" signal rather than the digital representation. The spectrum of a wide-sense stationary (WSS) signal $x(t)$ is defined in [2, 3] defined as

$$S_x(F) = \lim_{T\to\infty} \frac{E\{|X_T(F)|^2\}}{2T}\,,$$

where $X_T(F)$ is the Fourier Transform of the signal truncated to $-T < t < T$:

$$X_T(F) = \int_{-T}^{T} x(t)e^{-j2\pi F t}dt$$

The frequency variable F has units Hz (revolutions per second), and $S_x(F)$ is a Power Spectral Density in [power per Hz] (see below).
The *Wiener-Khintchine(-Einstein)* theorem states that $S_x(F)$ is the Fourier transform of the auto-correlation function:

$$S_x(F) = \int_{-\infty}^{\infty} r_x(\tau)e^{-j2\pi F\tau}d\tau\,,$$

where

$$r_x(\tau) = E\{x(t)x(t-\tau)\}$$

is the ACF (auto-correlation function) of $x(t)$. For a WSS signal, $E\{x(t)x(t-\tau)\}$ does not depend on absolute time t, but only the time lag τ. The inverse transform reads

$$r_x(\tau) = \int_{-\infty}^{\infty} S_x(F) e^{j2\pi F\tau} dF\,.$$

Evaluating this at lag $\tau = 0$, we see that the power is the integral over the spectrum:

$$\sigma_x^2 = E\{x^2(t)\} = r_x(0) = \int_{-\infty}^{\infty} S_x(F) dF\,.$$

Let us now perform a thought experiment: apply an ideal narrowband bandpass filter with passband $(F, F+\Delta F)$ to $x(t)$ and measure the power of the resulting signal $x_F(t)$. Since this has no energy outside the passband, we get

$$E\{x_F^2(t)\} = \int_{F}^{F+\Delta F} S_x(F) dF \approx S_x(F)\Delta F\,.$$

Thus, as $\Delta F \to 0$ we see that

$$S_x(F) = \frac{E\{x_F^2(t)\}}{\Delta F} \quad \text{[power per Hz]},$$

which justifies that $S_x(F)$ is a power spectral density. Indeed, a bank of narrowband bandpass filters followed by power measurement is a possible way to estimate the spectrum of a signal. However, this has largely been replaced by the digital spectral analysis techniques to be presented in Section 4.

3 Random Processes in Discrete Time

This section gives a brief review of relevant properties of discrete-time stochastic processes, assuming that the reader is already familiar with the continuous-time case. The reader is also assumed to be familiar with basic concepts in discrete-time signals and systems, including the discrete-time Fourier transform (DTFT) and digital filters. See, e.g., [4] for a thorough background in signals and systems.

Autocorrelation function Just like for deterministic signals, it is of course perfectly possible to sample random processes. The discrete-time version of a continuous-time signal $x(t)$ is here denoted $x[n]$, which corresponds to $x(t)$ sampled at the time instant $t = nT$. The sampling rate is $F_s = 1/T$ in Hz, and T is the sampling interval in seconds. The index n can be interpreted as normalized (to the sampling rate) time. To simplify matters, it will throughout be assumed that all processes are (wide-sense) stationary, real-valued and zero-mean. Similar to the deterministic case, it is in principle possible to reconstruct a bandlimited continuous-time stochastic process $x(t)$ from its samples $\{x[n]\}_{n=-\infty}^{\infty}$. In the random case the reconstruction must be given a more precise statistical meaning, like convergence in the mean-square sense. To avoid the technical difficulties with sampling the random signal $x(t)$ we will instead consider sampling the ACF $r_x(\tau)$. The sampled ACF is then given by

$$r_x[k] = r_x(\tau)|_{\tau=kT}\,.$$

From the usual sampling theorem (for deterministic signals), we infer that $r_x(\tau)$ can uniquely be reconstructed from the samples $\{r_x[k]\}_{k=-\infty}^{\infty}$, provided its Fourier transform (which is the discrete-time spectrum) is band-limited to $|F| < F_s/2$. Whether or not the actual signal realization $x(t)$ can be reconstructed is a different question, which will not be further dealt with here.
Note that the sampled continuous-time ACF is also the ACF of the sampled discrete-time signal $x[n]$:

$$r_x[k] = E\{x[n]x[n-k]\}$$

Clearly, $r_x[k]$ does not depend on absolute time n due to the stationarity assumption. Also, one can show that the autocorrelation is symmetric, $r_x[k] = r_x[-k]$ (conjugate symmetric in the complex case), and that $|r_x[k]| \leq r_x[0] = \sigma_x^2$ for all k (by Cauchy-Schwartz' inequality).
For two discrete-time stationary processes $x[n]$ and $y[n]$, we define the cross-correlation function as

$$r_{xy}[k] = E\{x[n]y[n-k]\}\,.$$

The cross-correlation measures how related the two processes are. This is useful, for example for determining who well one can predict a desired but unmeasurable signal $y[n]$ from the observed $x[n]$. If $r_{xy}[k] = 0$ for all k, we say that $x[n]$ and $y[n]$ are uncorrelated. It should be noted that the cross-correlation function is not necessarily symmetric, and it does not necessarily have its maximum at $k = 0$. For example, if $y[n]$ is a time-delayed version of $x[n]$, $y[n] = x[n-l]$, then $r_{xy}[k]$ peaks at lag $k = l$. Thus, the cross-correlation can be used to estimate the time-delay between two measurements of the same signal, perhaps taken at different spatial locations. This can be used, for example for synchronization in a communication system.

Power Spectrum While several different definitions can be made, the power spectrum is here simply introduced as the Discrete-Time Fourier Transform (DTFT) of the autocorrelation function:

$$P_x(e^{j\omega}) = \sum_{n=-\infty}^{\infty} r_x[n]e^{-j\omega n} \tag{1}$$

The variable ω is "digital" (or normalized) angular frequency, in radians per sample. This corresponds to "physical" frequency $\Omega = \omega/T$ rad/s. The spectrum is a real-valued function by construction, and it is also non-negative if $r_x[n]$ corresponds to a "proper" stochastic process.
Clearly, the DTFT is periodic in ω, with period 2π. The "visible" range of frequencies is $\omega \in (-\pi, \pi)$, corresponding to $\Omega \in (-\pi F_s, \pi F_s)$. The frequency $\Omega_N = \pi F_s$, or $F_N = F_s/2$ is recognized as the Nyqvist frequency. Thus, in order not to loose information when sampling a random process, its spectrum must be confined to $|\Omega| \leq \Omega_N$. If this is indeed the case, then we know that the spectrum of $x[n]$ coincides with that of $x(t)$ (up to a scaling by T). This implies that we can estimate the spectrum of the original continuous-time signal from the discrete samples.
The inverse transform is

$$r_x[n] = \frac{1}{2\pi}\int_{-\pi}^{\pi} P_x(e^{j\omega})e^{j\omega n}d\omega\,,$$

so the autocorrelation function and the spectrum are a Fourier transform pair. In particular, at lag 0 we have the signal power,

$$r_x[0] = E\{x^2[n]\} = \frac{1}{2\pi}\int_{-\pi}^{\pi} P_x(e^{j\omega})d\omega\,,$$

which is the integral of the spectrum. If we again perform the thought experiment of measuring the power of the signal, filtered by an ideal filter with the passband $(f, f+\Delta f)$, $f = \omega/(2\pi)$ being the normalized frequency in revolutions per sample, we find that

$$P_x(e^{j\omega}) = \frac{E\{x_f^2[n]\}}{\Delta f}\,.$$

This leads again to the interpretation of $P_x(e^{j\omega})$ as a power spectral density, with unit power (e.g. voltage squared) per *normalized* frequency Δf. This is expected, since the continuous- and discrete-time spectra are related by $S_x(F) = TP_x(e^{j2\pi f})$, where $f = F/F_s$.

White Noise The basic building block when modeling smooth stationary spectra is discrete-time white noise. This is simply a sequence of uncorrelated random variables, usually of zero mean and time-invariant power. Letting $e[n]$ denote the white noise process, the autocorrelation function is therefore

$$r_e[k] = E\{e[n]e[n-k]\} = \begin{cases} \sigma_e^2 & k = 0 \\ 0 & k \neq 0 \end{cases}\,,$$

where $\sigma_e^2 = E\{e^2[n]\}$ is the noise power. Such noise naturally arises in electrical circuits, for example due to thermal noise. The most commonly used noise model is additive white Gaussian noise (AWGN), sometimes called "university noise". The noise samples are then drawn from a $N(0, \sigma_e)$ distribution. However, it is a mistake to believe that white noise is always Gaussian distributed. For example, the impulsive noise defined by

$$e[n] = \begin{cases} \mathcal{N}(0, \sigma_e/\sqrt{\epsilon}) & \text{w.p. } \epsilon \\ 0 & \text{w.p. } 1-\epsilon \end{cases},$$

where $0 < \epsilon \ll 1$, is a stationary white noise, whose realizations have very little resemblance with AWGN. Two examples are shown in Figure 1. Yet, both are white noises with identical second-order properties. This illustrates that the second-order properties do not contain all information about a random signal,

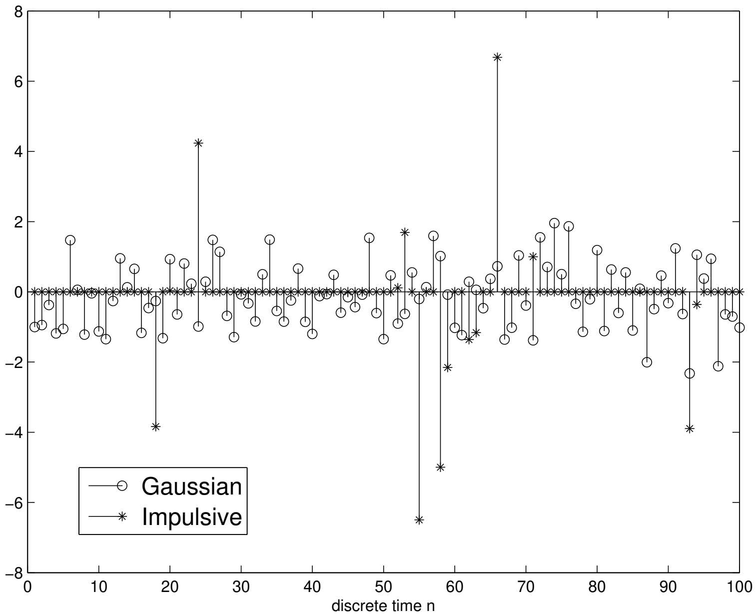


Figure 1: *Realizations of an AWGN and an impulsive noise with $\epsilon = 0.1$. Both are stationary white noises with identical second-order properties.*

this is true only in the case of a Gaussian process.
The spectrum of a white noise is given by

$$P_e(e^{j\omega}) = \sum_{n=-\infty}^{\infty} r_e[n] e^{-j\omega n} = \sigma_e^2 \, .$$

In other words, the white noise spectrum is flat over the region $(-\pi, \pi)$. One can infer that in order for a discrete-time noise to be white, the spectrum of the corresponding continuous-time signal should obey

$$P_e(\Omega) = \begin{cases} \sigma_e^2 F_s & |\Omega| \le \pi F_s \\ 0 & |\Omega| > \pi F_s \end{cases}$$

This is usually called *critical sampling.* If the sampling rate is higher than necessary, the discrete-time noise samples are in general correlated, although the spectrum of the continuous-time noise is flat over its bandwidth.

AR, MA and ARMA Processes Similar to the continuous-time case, it is very easy to carry the spectral properties over to a filtered signal. Let the discrete-time filter $H(z)$ (LTI-system) have impulse response $\{h[n]\}_{n=-\infty}^{\infty}$. Assume that the stationary process $e[n]$ is applied as input to the filter, see Figure 2. The output $x[n]$ is then given by

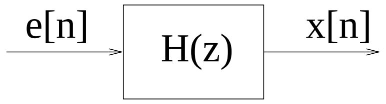


Figure 2: *Filtering a discrete-time random process by a digital filter.*

$$x[n] = h[n] * e[n] = \sum_{k=-\infty}^{\infty} h[k]e[n-k]$$

Provided the filter is stable, it is easy to show that the output signal is also a stationary stochastic process, and that the input and output spectra are related by

$$P_x(e^{j\omega}) = |H(e^{j\omega})|^2 P_e(e^{j\omega}) .$$

Also, the cross-spectrum, which is the DTFT of the cross-correlation, is given by

$$P_{xe}(e^{j\omega}) = \sum_{n=-\infty}^{\infty} r_{xe}[n]e^{-j\omega n} = H(e^{j\omega})P_e(e^{j\omega}) .$$

In particular, when the input signal $e[n]$ is a white noise, the output spectrum is given by

$$P_x(e^{j\omega}) = \sigma_e^2 \, |H(e^{j\omega})|^2 .$$

In other words, if we are good at shaping the amplitude characteristics $|H(e^{j\omega})|$ of a digital filter, we can use this to model the spectrum of a stationary stochastic process. Indeed, the spectral factorization theorem tells us that every smooth spectra can be well approximated by a filtered white noise, provided the filter has a sufficiently high order. We will here only consider finite orders. Thus, let the transfer function of the filter be

$$H(z) = \frac{B(z)}{A(z)} ,$$

where the numerator and denominator polynomials (in z^{-1}) are given by

$$\begin{aligned} B(z) &= 1 + b_1 z^{-1} + \cdots + b_q z^{-q} \\ A(z) &= 1 + a_1 z^{-1} + \cdots + a_p z^{-p} . \end{aligned}$$

Let the input to the filter be a white noise process $e[n]$. The filter output is given by the difference equation

$$x[n] + a_1 x[n-1] + \cdots + a_p x[n-p] = e[n] + b_1 e[n-1] + \cdots + b_q e[n-q]$$

Thus, the signal $x[n]$ is an auto-regression onto its past values along with a moving "average" (perhaps weighted average would have been a better name) of the noise. Therefore, $x[n]$ is called an auto-regressive moving average (ARMA) process. Sometimes the orders are emphasized by writing ARMA(p, q). The spectrum of an ARMA process is obtained as

$$P_x(e^{j\omega}) = \sigma_e^2 \frac{|B(e^{j\omega})|^2}{|A(e^{j\omega})|^2} .$$

Thus, shaping the spectrum of a process is reduced to appropriately selecting the filter coefficients and the noise power.
For the important special case when $q = 0$, $H(z) = 1/A(z)$ is an all-pole filter. The process $x[n]$ is called an AR process, defined by

$$x[n] + a_1 x[n-1] + \cdots + a_p x[n-p] = e[n] .$$

The spectrum of an AR process is given by

$$P_x(e^{j\omega}) = \frac{\sigma_e^2}{|A(e^{j\omega})|^2} .$$

Similarly, when $p = 0$, $H(z) = B(z)$ has only zeros and we get the MA process

$$x[n] = e[n] + b_1 e[n-1] + \cdots + b_q e[n-q]$$

with

$$P_x(e^{j\omega}) = \sigma_e^2 \, |B(e^{j\omega})|^2 .$$

In general, it is difficult to say when an AR model is preferred over an MA or ARMA model. However, it is more easy to get peaky spectra by using poles close to the unit circle. Similarly, spectra with narrow dips (notches) are better modeled with zeros. See Figure 3 for some examples. The plots are produced

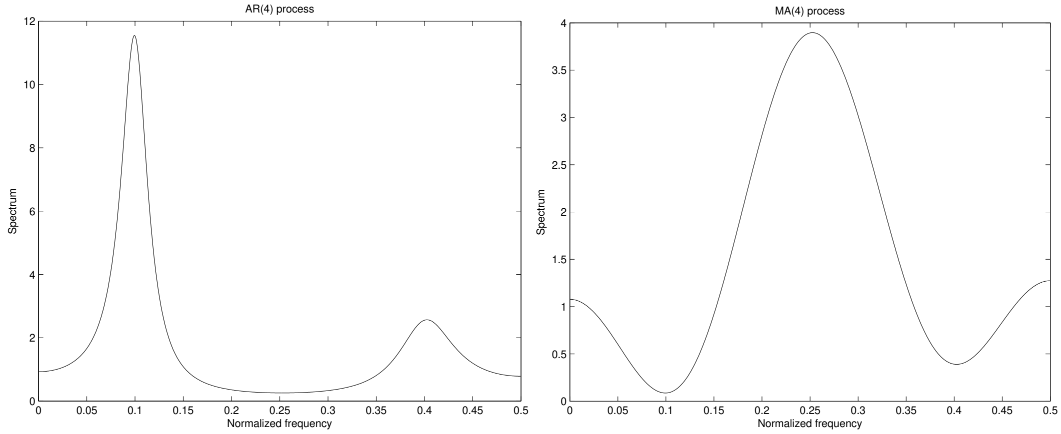


Figure 3: *Examples of spectra for the case of AR(4) and MA(4) models. The AR part better models narrow peaks, whereas the MA part is better with notches and broad peaks.*

with $\sigma_e^2 = 1$ and

$$B(z) = A(z) = 1 - 0.16z^{-1} - 0.44z^{-2} + 0.12z^{-3} + 0.52z^{-4} .$$

The roots of this polynomial, which are the zeros and poles, respectively, are located at $0.9e^{\pm 2\pi 0.1 j}$ and $0.8e^{\pm 2\pi 0.4 j}$. This explains why we see a notch (MA process) and a peak (AR process) at the normalized frequencies 0.1 and 0.4 revolutions per sample, with the notch (peak) at 0.1 being more pronounced because the zero (pole) is closer to the unit circle. In practice, AR models are often preferred because they lead to simpler estimation algorithms than do MA or ARMA models, and also because we are more often interested in peaks and peak locations than other properties of the spectrum.

ACF for an AR Process: The Yule-Walker Equations We saw previously that it is very easy to compute the spectrum of an ARMA process once its polynomial coefficients are known. In principle, the autocorrelation function can then be computed by the inverse DTFT. However, for the important special case of an AR process there is a simpler and more practical procedure. Recall the AR difference equation:

$$x[n] + a_1 x[n-1] + \cdots + a_p x[n-p] = e[n]$$

Multiply both sides of the equation by $x[n-k]$ for $k \geq 0$ and take expectation. This leads to the relation

$$r_x[k] + a_1 r_x[k-1] + \cdots + a_p r_x[k-p] = E\{e[n]x[n-k]\}$$

To evaluate the right hand side, first assume that $k > 0$. Then $x[n-k]$ only depends on past values of the noise, $\{e[n-k],\ e[n-k-1],\ , e[n-k-2], \dots\}$. We conclude that $x[n-k],\ k > 0$ is uncorrelated with $e[n]$, so $E\{e[n]x[n-k]\} = 0$ for $k > 0$. When $k = 0$, we use the AR equations to express $x[n]$ as a function of past values:

$$E\{e[n]x[n]\} = E\{e[n](-a_1 x[n-1] - \cdots - a_p x[n-p] + e[n])\} = \sigma_e^2 .$$

Thus we have proved the relation

$$r_x[k] + \sum_{l=1}^{p} a_l r_x[k-l] = \sigma_e^2\,\delta[k]\,,\ k \geq 0\,, \tag{2}$$

where $\delta[k] = 0$ for $k \neq 0$ and $\delta[0] = 1$. The relation (2), for $k = 0, 1, \ldots, p$ is known as the Yule-Walker (YW) equations. Since this is a linear relation between the AR coefficients and the autocorrelation function, it can be used to efficiently solve for one given the other. If the autocorrelation function is given, we can put the equation for $k = 1, 2, \ldots, N$ on matrix form as:

$$\begin{bmatrix} r_x[0] & r_x[1] & \cdots & r_x[p-1] \\ r_x[1] & r_x[0] & \cdots & r_x[p-2] \\ \vdots & \vdots & \ddots & \vdots \\ r_x[p-1] & r_x[p-2] & \cdots & r_x[0] \end{bmatrix} \begin{bmatrix} a_1 \\ a_2 \\ \vdots \\ a_p \end{bmatrix} = - \begin{bmatrix} r_x[1] \\ r_x[2] \\ \vdots \\ r_x[p] \end{bmatrix}. \tag{3}$$

This linear system of equations is usually referred to as the Normal Equations. We can express the matrix equation as

$$\mathbf{R}_x \mathbf{a} = -\mathbf{r}_x\,,$$

where $\mathbf{R}_x$ is called the autocorrelation matrix (of size $p \times p$). The solution w.r.t. $\mathbf{a} = [a_1, a_2, \ldots, a_p]^T$ now follows as $\mathbf{a} = -\mathbf{R}_x^{-1}\mathbf{r}_x$. In practice, more efficient methods for computing $\mathbf{a}$ are often used, in particular if p is large and/or if $\mathbf{R}_x$ is nearly singular.

Conversely, if $a_k,\ k = 1, 2, \cdots, p$ and σ_e^2 are known in advance, then we can use the same set of Yule-Walker equations to solve for $r_x[k]$, $k = 0, 1, \cdots, p$. For example, when $p = 2$ we get

$$\begin{bmatrix} 1 & a_1 & a_2 \\ a_1 & 1+a_2 & 0 \\ a_2 & a_1 & 1 \end{bmatrix} \begin{bmatrix} r_x[0] \\ r_x[1] \\ r_x[2] \end{bmatrix} = \begin{bmatrix} \sigma_e^2 \\ 0 \\ 0 \end{bmatrix}$$

which can easily be solved for $(r_x[0], r_x[1], r_x[2])^T$.

Example 3.1 ACF of an AR(2) process

We demonstrate the Yule-Walker procedure by computing the auto-correlation function for the signal model

$$x[n] - 1.5x[n-1] + 0.7x[n-2] = e[n]\,,$$

where $\sigma_e^2 = 1$. The relation between the auto-correlation and the AR coefficients given above is in this case:

$$\begin{bmatrix} 1 & -1.5 & 0.7 \\ -1.5 & 1.7 & 0 \\ 0.7 & -1.5 & 1 \end{bmatrix} \begin{bmatrix} r_x[0] \\ r_x[1] \\ r_x[2] \end{bmatrix} = \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix}$$

Solving the linear system gives

$$r_x[0] \approx 8.85\,, \quad r_x[\pm 1] \approx 7.81\,, \quad r_x[\pm 2] \approx 5.52$$

The auto-correlation function at higher lags are computed recursively by

$$r_x[k] = 1.5r_x[k-1] - 0.7r_x[k-2]\,,\ k > 0$$

The auto-correlation function for this process is plotted in Figure 4. To verify the above calculation, we can insert the auto-correlation function into the normal equations:

$$\begin{bmatrix} 8.85 & 7.81 \\ 7.81 & 8.85 \end{bmatrix} \begin{bmatrix} a_1 \\ a_2 \end{bmatrix} = - \begin{bmatrix} r_x[7.81] \\ r_x[5.52] \end{bmatrix}$$

Solving this linear system gives $a_1 = -1.50$ and $a_2 = 0.70$ as expected. Also,

$$\sigma_e^2 = r_x[0] + a_1 r_x[1] + a_2 r_x[2] = 8.85 - 1.5 \cdot 7.81 + 0.7 \cdot 5.52 = 1.00\,,$$

which again agrees with our expectation. □

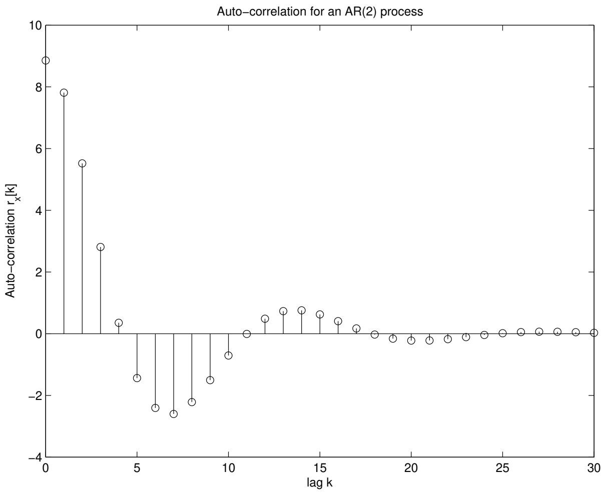


Figure 4: *Auto-correlation function for the AR(2) process* $x[n] - 1.5x[n-1] + 0.7x[n-2] = e[n]$.

ACF for MA and ARMA Processes Also in the case of a pure MA process, the auto-correlation function is easy to compute. Consider the MA model:

$$x[n] = e[n] + b_1 e[n-1] + \cdots + b_q e[n-q]$$

To obtain the variance $r_x[0]$, just square both sides of the equation and take expectation:

$$\begin{aligned} E\{x^2[n]\} = r_x[0] &= E\{(e[n] + b_1 e[n-1] + \cdots + b_q e[n-q])(e[n] + b_1 e[n-1] + \cdots + b_q e[n-q])\} \\ &= (1 + b_1^2 + \cdots b_q^2)\sigma_e^2 \end{aligned}$$

where we have used that $e[n]$ is white noise. For $r_x[1]$, multiply the MA equation by $x[n-1]$ and take expectation:

$$\begin{aligned} r_x[1] &= E\{(e[n] + b_1 e[n-1] + \cdots + b_q e[n-q])(e[n-1] + b_1 e[n-2] + \cdots + b_q e[n-q-1])\} \\ &= (b_1 + b_2 b_1 + \cdots b_q b_{q-1})\sigma_e^2 \,. \end{aligned}$$

The same principle is used to compute $r_x[k],\ k = 2, 3, \ldots$, noting that $r_x[k] = 0$ for $k > q$. Thus, an MA process has a finite correlation length, given by its order q. Note that although we can easily compute the auto-correlation function given the MA coefficients and the noise variance, the converse is in general not as simple. This is because the relation between $r_x[k]$ and the b_k:s is not linear as in the case of an AR process. From the above we see that it is a quadratic relation. Although this may sound simple enough, a computationally efficient and robust (to errors in $r_x[k]$) method to compute an MA model $\{b_k\}_{k=1}^q$, σ_e^2 from the auto-correlation coefficients $\{r_x[k]\}_{k=0}^q$ is not known to date. The reader is referred to [1, 5] for approximate methods with acceptable computational cost and statistical properties.

Example 3.2 ACF of an MA(2) process
Consider the second-order MA process:

$$y[n] = e[n] - 1.5e[n-1] + 0.5e[n-2] \,,$$

where $e[n]$ is WGN with $E\{e^2[n]\} = \sigma_e^2$. By direct calculation we get

$$\begin{aligned} E\{y^2[n]\} &= E\{(e[n] - 1.5e[n-1] + 0.5e[n-2])(e[n] - 1.5e[n-1] + 0.5e[n-2])\} \\ &= E\{e^2[n]\} + E\{(-1.5e[n-1])^2\} + E\{(0.5e[n-2])^2\} \\ &= \sigma_e^2 + 1.5^2\sigma_e^2 + 0.5^2\sigma_e^2 = 3.5\sigma_e^2. \end{aligned}$$

Similarly,

$$\begin{aligned}
E\{y[n]y[n-1]\} &= E\{(e[n]-1.5e[n-1]+0.5e[n-2])(e[n-1]-1.5e[n-2]+0.5e[n-3])\} \\
&= E\{-1.5e^2[n-1]\}+E\{(0.5\times -1.5)e^2[n-2]\} \\
&= -1.5\sigma_e^2 - 0.75\sigma_e^2 = -2.25\sigma_e^2
\end{aligned}$$

and

$$\begin{aligned}
E\{y[n]y[n-2]\} &= E\{(e[n]-1.5e[n-1]+0.5e[n-2])(e[n-2]-1.5e[n-3]+0.5e[n-4])\} \\
&= E\{0.5e^2[n-2]\} = 0.5\sigma_e^2.
\end{aligned}$$

Thus, the ACF is given by

$$r_y[k] = \begin{cases} 3.5\sigma_e^2 & , \ k=0 \\ -2.25\sigma_e^2 & , \ k=\pm 1 \\ 0.5\sigma_e^2 & , \ k=\pm 2 \\ 0 & , \ |k|>2 \end{cases} .$$

□

Computing the auto-correlation function for an ARMA process requires in general considerably more effort than in the pure AR or MA cases. A common way to solve this problem is to exploit that the auto-correlation is the inverse transform of the spectrum, which for an ARMA model is given by:

$$P_x(e^{j\omega}) = \sigma_e^2 \frac{|B(e^{j\omega})|^2}{|A(e^{j\omega})|^2} .$$

The inverse transform is

$$r_x[n] = \frac{1}{2\pi}\int_{-\pi}^{\pi} P_x(e^{j\omega})e^{j\omega n}d\omega ,$$

and the integral can be solved using calculus of residues. For low model orders, especially when the roots of $A(z)$ are real-valued, it is easier to use the inverse Z-transform:

$$r_x[n] = \mathcal{Z}^{-1}\{P_x(z)\} = \mathcal{Z}^{-1}\left\{\sigma_e^2 \frac{B(z)B(z^{-1})}{A(z)A(z^{-1})}\right\} .$$

For simple cases, the inverse transform can be carried out using partial fraction expansion (with long division if $q \geq p$) and table lookup. For the special case of an ARMA(1,1)-process, there is an even simpler procedure. Consider the model

$$x[n]+a_1x[n-1] = e[n]+b_1e[n-1]$$

First, square both sides of the equation and take expectation, yielding

$$r_x[0](1+a_1^2)+2a_1r_x[1] = (1+b_1^2)\sigma_e^2$$

Next, multiply the ARMA(1,1) equation by $x[n-1]$ and take expectation. This gives

$$r_x[1]+a_1r_x[0] = 0+b_1\sigma_e^2 ,$$

where we have used that $x[n-1]$ is uncorrelated with $e[n]$ and that $E\{e[n-1]x[n-1]\} = \sigma_e^2$. We now have two equations in the two unknowns $r_x[0]$ and $r_x[1]$, which can easily be solved given numerical values of a_1, b_1 and σ_e^2. To obtain $r_x[k]$ for higher lags, multiply the ARMA equation by $x[n-k]$, $k>1$ and take expectation. This gives

$$r_x[k]+a_1r_x[k-1] = 0 , \ k>1.$$

Thus, $r_x[k]$ can be recursively computed once $r_x[1]$ is known. The above relation is known as the *extended Yule-Walker equations*, and a similar relation is obtained in the general ARMA(p,q) case for lags $k>q$. This shows that a linear relation between the auto-correlation function and the AR parameters $\{a_k\}$ can be obtained also in the ARMA case. However, the auto-correlation coefficients are still non-linear functions of the MA parameters $\{b_k\}$.

Spectral Lines By spectral factorization, any smooth spectrum can be well approximated by an ARMA process of sufficiently high order. But what about spectra that are not smooth? The so-called Wold decomposition tells us that all stationary stochastic processes (i.e. even those with non-smooth spectra) can be decomposed into a "purely" stochastic part and a perfectly predictable part. The stochastic part is represented as an ARMA model as before. The remaining part is represented as a sum of sinusoids:

$$s[n] = \sum_{k=1}^{p} A_k \cos(\omega_k n + \phi_k)\,.$$

The frequencies ω_k and the amplitudes A_k are usually regarded as deterministic (fixed), whereas the initial phases ϕ_k are modeled as independent and uniformly distributed in $(0, 2\pi)$. It is then easy to show that $s[n]$ in fact becomes a stationary stochastic process with zero mean and autocorrelation function

$$r_s[k] = \sum_{k=1}^{p} \frac{A_k^2}{2} \cos(\omega_k n)\,.$$

Although $s[n]$ is formally a stochastic process, it is clear that we can perfectly predict the future values of $s[n]$ if we only know sufficiently many initial conditions. The interested reader can show that both $s[n]$ and $r_s[n]$ obey the following homogenous difference equation of order $2p$:

$$s[n] + a_1 s[n-1] + \cdots + a_{2p} s[n-2p] = 0\,,$$

where the corresponding polynomial $A(z)$ has its roots on the unit circle, at $\{e^{\pm j\omega_k}\}_{k=1}^{p}$.
Taking the DTFT of $r_s[k]$, the spectrum is obtained as

$$P_s(e^{j\omega}) = \frac{\pi}{2} \sum_{k=1}^{p} A_k^2 \left\{\delta(\omega - \omega_k) + \delta(\omega + \omega_k)\right\}.$$

In other words, the spectrum consists of infinitely narrow spectral peaks, located at the frequencies $\pm\omega_k$. We get peaks also at the negative frequencies, because $\cos \omega n = (e^{j\omega n} + e^{-j\omega n})/2$ consists of two "phasors" rotating in opposite directions. The angular speed of these phasors are the frequencies.

4 Digital Spectral Analysis

We have previously argued that the spectrum of a bandlimited continuous-time signal $x(t)$ can in principle be inferred from its discrete-time samples $x[n]$. However, computing the spectrum involves averaging over infinitely many realizations of the signal. In practice, one has usually access to only one realization. Fortunately, for a well-behaved stationary signal, we can replace ensemble averaging by time averaging (ergodicity). Thus, given an infinite observation time, the spectrum of a signal could be perfectly estimated. In practice, the data length is limited by the available time to deliver a result, the computer processor speed and memory size, and/or the time over which the process can be considered stationary. Thus, the problem of spectral estimation is to estimate the spectrum $P_x(e^{j\omega})$ based on a finite sample $\{x[0], x[1], \ldots, x[N-1]\}$. The number of data required will depend on the demanded accuracy as well as the nature of the true spectrum. It is also different for different methods as we shall see.

4.1 Non-Parametric Methods

The simplest and most straightforward way to estimate the power spectrum of a signal is to use non-parametric methods, also known as Fourier-based (or window-based) methods. The main advantages of these methods are that 1) they do not require specification of a signal model, and 2) they are easy to implement and computationally inexpensive due to the efficient fast Fourier transform (FFT). The disadvantage is that the frequency resolution is limited by the length of the observation time.

The Periodogram

The classical approach to spectral estimation is to use the so-called periodogram. For a given data sequence $x[n]$, $n = 0, 1, \cdots, N-1$, the periodogram is the normalized magnitude square of the DTFT:

$$\hat{P}_{per}(e^{j\omega}) = \frac{1}{N}|X_N(e^{j\omega})|^2 \tag{4}$$

$$X_N(e^{j\omega}) = \sum_{n=0}^{N-1} x[n]e^{-j\omega n} \tag{5}$$

We remember from the deterministic Signals and Systems theory that the finite-interval DTFT (and hence the periodogram) has a limited frequency resolution. Two spectral peaks that are spaced less than $\Delta\omega \approx 2\pi/N$ apart will show up as a single peak in $\hat{P}_{per}(e^{j\omega})$. Still, smoother parts of the spectrum of a deterministic signal will generally be well approximated by the periodogram. Unfortunately, this is not the case when $x[n]$ is stochastic. The periodogram will then reflect the spectral contents of the particular realization of $x[n]$ that has been observed. In general, we are more interested in the average behavior, given by the "true" spectrum (1). To connect the two, it is useful to rewrite (4) in the form

$$\hat{P}_{per}(e^{j\omega}) = \sum_{k=-N+1}^{N-1} \hat{r}_x[k]e^{-j\omega} \tag{6}$$

where $\hat{r}_x[k]$ is the sample autocorrelation function

$$\hat{r}_x[k] = \frac{1}{N}\sum_{n=k}^{N-1} x[n]x[n-k]\,.$$

The interested reader is encouraged to prove the equivalence of (4) and (6). It is now easy to show that

$$\lim_{N\to\infty} E\{\hat{P}_{per}(e^{j\omega})\} = P_x(e^{j\omega})$$

which shows that the periodogram is an asymptotically unbiased estimator. Under some regularity conditions, it is further possible to show [1] that

$$E\{(\hat{P}_{per}(e^{j\omega}) - P_x(e^{j\omega}))^2\} = P_x^2(e^{j\omega}) + R_N\,,$$

where the reminder term R_N tends to zero as $N \to \infty$. Thus, the standard deviation of the periodogram is approximately as large as the quantity (the spectrum) it is supposed to estimate. In its basic form, the periodogram is apparently an unacceptably noisy estimator. Fortunately, there are modifications with much better behavior.

The Modified Periodogram

The DTFT has a limited resolution, due to the inherent windowing of the data. We can express the DTFT as

$$X_N(e^{j\omega}) = \sum_{n=-\infty}^{\infty} w_R[n]x[n]e^{-j\omega n}\,,$$

where $w_R[n]$ is the length-N rectangular window

$$w_R[n] = \begin{cases} 1 & 0 \le n \le N-1 \\ 0 & \text{otherwise} \end{cases}$$

Window functions are usually characterized in the frequency domain in terms of the mainlobe width $\Delta\omega$ and the peak sidelobe level. Among all windows of a given length, the rectangular has the smallest $\Delta\omega$, given by $\Delta\omega = 2\pi/N$ (radians/sample). Figure 5 shows the periodogram for two sinusoids of different frequencies. In the left plot, the sinusoids are well separated, and two peaks are clearly seen at the right

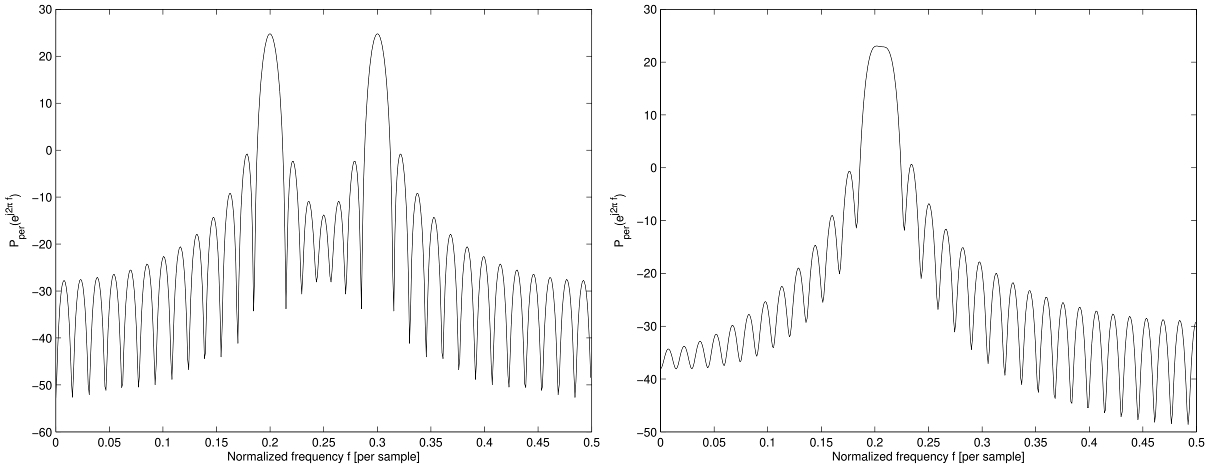


Figure 5: *Periodogram for two sinusoids of frequencies $f_1 = 0.2$, $f_2 = 0.3$ (left plot), and $f_1 = 0.2$, $f_2 = 0.21$ (right plot). The number of samples is $N = 64$, so the frequency resolution is $\Delta f = 1/N = 0.0156$. In the right plot, $f_2 - f_1 = 0.01 < \Delta f$, and the frequencies cannot be resolved.*

frequencies. In the right plot, the periodogram fails to resolve the two frequency components, and only one (broad) peak is visible. We also see in the plot that the peak sidelobe of the rectangular window is only about 13 dB below the mainlobe. This is independent of the number of samples. Therefore, weaker signal components can be masked by a neighboring strong frequency peak. This effect, called the "masking phenomenon", can be alleviated by using a different window function:

$$X_w(e^{j\omega}) = \sum_{n=-\infty}^{\infty} w[n]x[n]e^{-j\omega n} \, .$$

The frequency resolution is then traded for better sidelobe suppression. For example, the so-called Hamming window has a frequency resolution of about $1.3 \times (2\pi/N)$ and peak sidelobe -43 dB [1]. Another useful choice is the Chebyshev window, which has a constant sidelobe level that can be chosen by the user (lower sidelobes mean of course less resolution). Figure 6 illustrates the operation of four different windows. The weak frequency component is masked when no windowing is used, but clearly visible with the Hamming and Chebyshev windows. The periodogram using windowed data is usually called the *modified* periodogram. The windowing has a negligible effect on the variance, so for a random signal it needs to be combined with other modifications.

Bartlett's and Welch's Methods: Periodogram Averaging

To overcome the disadvantage of high variance in the periodogram or the modified periodogram, Bartlett proposed to divide the data into $K = N/M$ non-overlapping segments. For each segment of data a "local" periodogram is applied. This is followed by averaging all periodograms as follows:

$$P_B(e^{j\omega}) = \frac{1}{K}\sum_{i=1}^{K} \hat{P}_{per}^{(i)}(e^{j\omega}) \, .$$

If the local periodograms are considered approximately uncorrelated, the averaging leads to a reduction of the variance by a factor of $K = N/M$:

$$E\{(\hat{P}_B(e^{j\omega}) - P_x(e^{j\omega}))^2\} \approx \frac{M}{N} P_x^2(e^{j\omega}) \, .$$

The price for this is a reduced frequency resolution. Since each data length is only $M = N/K$ samples, the resolution is $\Delta\omega \approx 2\pi/M = 2\pi K/N$. Provided N is large enough, a "sufficient" resolution will anyway be obtained, and for a fixed M, the variance of $\hat{P}_B(e^{j\omega})$ will tend to zero as $N \to \infty$.

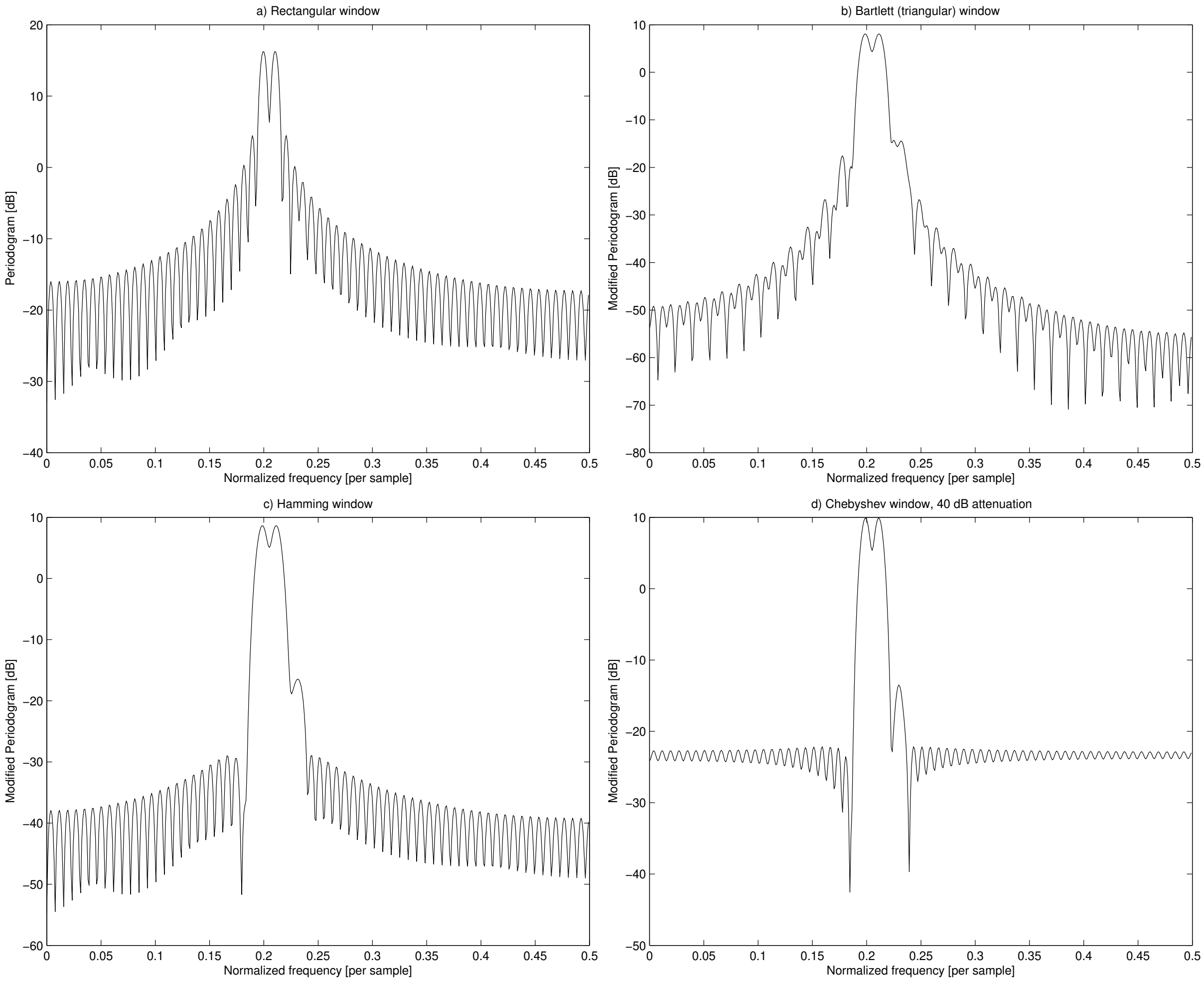


Figure 6: *Spectral estimates for three sinusoids of frequencies* $f \in \{0.20, 0.21, 0.2295\}$*. The location of the* $f = 0.2295$ *component is at a sidelobe of the other components, and since its amplitude is 26 dB below the others it is completely masked by the strong neighbors when no windowing is used. The data windows are: a) Rectangular (no windowing), b) Bartlett (triangular), c) Hamming, and d) Chebyshev with 40 dB attenuation. The number of samples is* $N = 128$.

A further refinement of the technique of periodogram averaging is Welch's method. The idea is to use overlapping segments to increase the frequency resolution (or the number of segments). To reduce the dependence between the segments, modified periodograms are used. For a well chosen window function, a better trade-off between resolution (bias) and variance is in general achieved, and at the same time the masking problem is addressed. For more details, see [1].

Blackman-Tukey's Method: Smoothed Periodogram

The source of the problem with the high variance in the periodogram is clearly visible in the form (6). Each sample correlation $\hat{r}_x[k]$ is based on averaging $N - k$ data samples. Thus, the variance of $\hat{r}_x[k]$ increases with increasing k, until at $k = N - 1$ there is no averaging at all. A very natural idea is then to introduce a weighting in (6), which gives successively smaller weights for increasing lag parameter k. Denoting the correlation window $w_{lag}[k]$, which is non-zero for $-M \leq k \leq M$, the so-called Blackman-

Tukey estimate is given by

$$\hat{P}_{BT}(e^{j\omega}) = \sum_{k=-M}^{M} w_{lag}[k]\hat{r}_x[k]e^{-jk\omega} \,. \tag{7}$$

The original periodogram is retrieved if $M = N$ and a rectangular window is used. An appealing interpretation of $\hat{P}_{BT}(e^{j\omega})$ is obtained by looking at (7) as the DTFT of $w_{lag}[k]\hat{r}_x[k]$:

$$\hat{P}_{BT}(e^{j\omega}) = \sum_{k=-\infty}^{\infty} w_{lag}[k]\hat{r}_x[k]e^{-jk\omega} = \text{DTFT}\{w_{lag}[k]\hat{r}_x[k]\}$$

(since $w_{lag}[k] = 0$ for $|k| > M$). Multiplication in the time domain corresponds to convolution in the frequency domain, so we can also write

$$\hat{P}_{BT}(e^{j\omega}) = \frac{1}{2\pi} W_{lag}(e^{j\omega}) * \hat{P}_{per}(e^{j\omega}) = \frac{1}{2\pi}\int_{-\pi}^{\pi} \hat{P}_{per}(e^{j(\omega-\xi)})W_{lag}(e^{j\xi})d\xi \,,$$

where $W_{lag}(e^{j\omega})$ is the DTFT of the window function and $\hat{P}_{per}(e^{j\omega})$ is the periodogram, which is the DTFT of $\hat{r}_x[k]$. Thus, $\hat{P}_{BT}(e^{j\omega})$ is in effect a *smoothed* periodogram. Think of $W_{lag}(e^{j\omega})$ as a rectangular box of size $\Delta\omega$. Then, the BT (Blackman-Tukey) spectral estimate at a certain frequency ω is obtained by a local averaging (=smoothing) of the periodogram, in the interval $(\omega-\Delta\omega/2, \omega+\Delta\omega/2)$. In practice, the window function has a non-rectangular mainlobe which implies some weighted averaging, and also sidelobes which may result in frequency masking.

It is clear that the smoothing reduces the variance, at the price of a reduced frequency resolution (bias). It can be shown that (e.g. [1])

$$\begin{aligned} E\{\hat{P}_{BT}(e^{j\omega})\} &\approx \frac{1}{2\pi} P_x(e^{j\omega}) * W_{lag}(e^{j\omega}) \\ E\{(\hat{P}_{BT}(e^{j\omega}) - P_x(e^{j\omega}))^2\} &\approx \frac{1}{N}\sum_{k=-M}^{M} w_{lag}^2[k]\, P_x^2(e^{j\omega}) \end{aligned}$$

Given these formulas, the length and the shape of the lag window is selected as a tradeoff between the variance and the bias of the estimate. For $M \to \infty$, the Fourier transform of the lag window becomes more and more like a Dirac pulse, and the BT estimate becomes unbiased. In practice, M should be chosen large enough so that all details in the spectrum can be recovered (which of course depends on the shape of the true spectrum), but yet $M \ll N$ to reduce the variance. In general, the BT method gives the best trade-off among the non-parametric spectral estimation methods presented here.

Example 4.1 Non-Parametric Spectral Estimation
A set of $N = 1000$ data samples were generated according to the AR model

$$x[n] - 1.5x[n-1] + 0.7x[n-2] = e[n]\,,$$

where $e[n]$ is $\mathcal{N}(0,1)$ white noise. Figure 7, left plot, shows the true spectrum along with the Periodogram estimate. It is clear that the Periodogram is a very noise estimate of the spectrum, although "on average" it looks OK. In Figure 7, right plot, the BT spectrum estimate is shown for various lengths M of the lag window. A Hamming window is used. It is clear that reducing M lowers the variance of the estimate at the price of a reduced resolution. In this case, $M \in (20, 50)$ appears to be a good choice. □

4.2 Parametric Spectral Estimation

As we previously saw, non-parametric spectral estimation methods have limitations in terms of frequency resolution and estimation variance. An attempt to overcome these limitations is to use parametric modeling of the measured signal. Such methods are expected to perform much better when the "true" spectrum can be well approximated with a model using few parameters in the chosen model class. This is because the variance of the estimated spectrum is in general proportional to the number of estimated parameters. We shall only consider signals with AR models in this review (for other models, see e.g. [1, 5]).

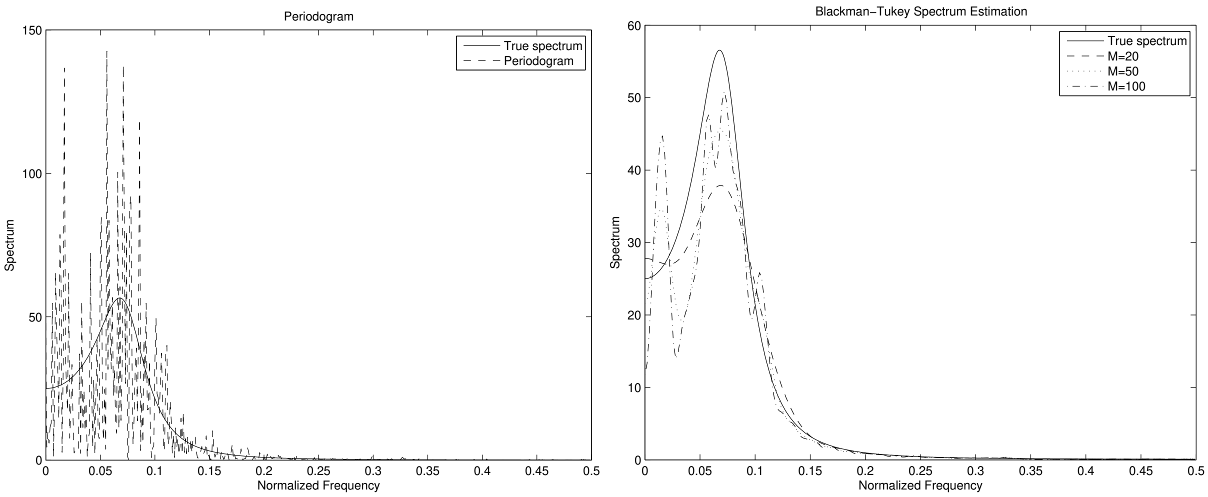


Figure 7: *Examples of non-parametric spectrum estimation using the Periodogram (left plot) and the Blackman-Tukey method (right plot). The true spectrum is an AR(2) process.*

AR Models

In an AR(p) model, the data sample at time n is assumed to be the weighted combination of the previous data samples $x[n-k]$, $k = 1, \dots, p$, and a "driving" white noise $e[n]$,

$$x[n] = -\sum_{k=1}^{p} a_k x[n-k] + e[n] \,.$$

This is equivalent to $x[n]$ being the output of an all-pole system $H(z) = 1/A(z)$, whose input is the white noise $e[n]$. Given the above model, the spectrum of $x[n]$ is represented as

$$P_x(e^{j\omega}) = \frac{\sigma_e^2}{|A(e^{j\omega})|^2} \,.$$

Therefore, the spectral estimation is now reduced to selecting the model order p, finding the model parameters $\{a_k\}_{k=1}p$, and estimating the innovation variance σ_e^2. Now, recall the normal equations (3). Given measured data $\{x[n]\}_{n=0}^{N-1}$, a natural idea is to replace the true (but unknown) autocorrelations $r_x[k]$ in (3) with the sample autocorrelations:

$$\hat{r}_x[k] = \frac{1}{N} \sum_{n=k}^{N-1} x[n]x[n-k] \,, \; k = 0, 1, \dots, p \,.$$

The sample version of the normal equations are now expressed as

$$\begin{bmatrix} \hat{r}_x[0] & \hat{r}_x[1] & \cdots & \hat{r}_x[p-1] \\ \hat{r}_x[1] & \hat{r}_x[0] & \cdots & \hat{r}_x[p-2] \\ \vdots & \vdots & \ddots & \vdots \\ \hat{r}_x[p-1] & \hat{r}_x[p-2] & \cdots & \hat{r}_x[0] \end{bmatrix} \begin{bmatrix} a_1 \\ a_2 \\ \vdots \\ a_p \end{bmatrix} = - \begin{bmatrix} \hat{r}_x[1] \\ \hat{r}_x[2] \\ \vdots \\ \hat{r}_x[p] \end{bmatrix}$$

The solution is given by $\hat{\mathbf{a}} = -\hat{\mathbf{R}}_x^{-1}\hat{\mathbf{r}}_x$ with obvious notation. Several numerically efficient algorithms exist for computing the estimate, including the so-called Levinson-type recursions, see [1]. However, for our purposes, it suffices to type `a=-R\r` in Matlab, or use a suitable command from the Signal Processing or System Identification toolboxes.
Once the AR parameters are estimated, the noise variance is computed using (2) for $k = 0$, leading to

$$\hat{\sigma}_e^2 = \hat{r}_x[0] + \sum_{k=1}^{p} \hat{a}_l \hat{r}_x[k] \,.$$

Several methods have been proposed to estimate the model order p. The basic idea of most methods is to compute the noise variance $\hat{\sigma}_e^2$ for increasing values of $p = 1, 2, \dots,$ and observe the decrease of the residual noise power. When the "true" order has been found (if there is one), the noise power is expected to level out, i.e. further increase of p does not yield any significant decrease of $\hat{\sigma}_e^2$. The difference among the various suggested methods lies in determining what to mean by a "significant" decrease. See, e.g., [5] for details.

Example 4.2 AR Spectral Estimation
Consider the same AR(2) model as in Example 4.1. For the same $N = 1000$ samples, the sample autocorrelation function was calculated to

$$\hat{r}_x[0] = 7.74\,, \quad \hat{r}_x[1] = 6.80\,, \quad \hat{r}_x[2] = 4.75\,.$$

Solving the YW equations for the AR parameters (assuming the correct model order $p = 2$ is known) then gives $\hat{a}_1 = -1.50$, $\hat{a}_2 = 0.70$, which in fact agrees with the true parameters up to the given precision. The estimated noise variance is found to be

$$\hat{\sigma}_e^2 = 7.74 - 1.50 \cdot 6.80 + 0.70 \cdot 4.75 = 0.87\,,$$

which is slightly too low. The resulting YW-based spectrum estimate is displayed in Figure 8. It is remarked that the `pyulear` command in Matlab gives a spectrum estimate with the same shape, but the level is too low. □

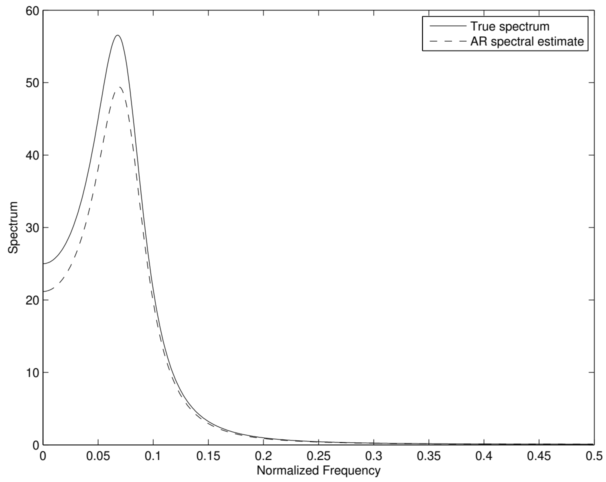


Figure 8: *Parametric AR spectral estimation using the YW method. Both the estimated and the true spectrum is an AR(2) model.*

In the above example, the YW method is applied to data that are generated according to an AR model with known order. As one could expect, the resulting spectrum estimate agrees very well with the true spectrum. With measured data, it is of course never the case that an AR model can give a perfect description of the true spectrum. The best one can hope for is that a "sufficiently good" approximation can be found using a not too high model order. It this is the case, the parametric AR-based spectrum estimate will in general give better performance than the non-parametric methods, particularly for short data records (N small).

5 Optimal Filtering

One of the main tasks in digital signal processing is to estimate signals (or parameters of signals) from noise corrupted measurements. For example, background noise can disturb a recorded speech signal so that it becomes unintelligible, or an image taken from an infrared camera shows thermal noise that significantly reduces the image quality. The noise is often modeled as additive, although in many cases there are also other distortions, including filtering ("blurring") and phase-noise, which is multiplicative. Assuming additive noise, the measured signal $x[n]$ can be described as a stochastic process,

$$x[n] = s[n] + w[n]$$

where $s[n]$ denotes the signal of interest and $w[n]$ is the noise (or disturbance). The optimal filtering problem is to construct a linear filter $H(z)$ such that the filtered signal $\hat{d}[n]$ is the "best" estimate of some reference signal $d[n]$. Here, the reference signal $d[n]$ can be the signal $s[n]$ itself, in which case we talk about the "filtering" problem. If $d[n] = s[n+k],\ k > 0$ or $d[n] = x[n+k],\ k > 0$, we have the "prediction" problem, and if $d[n] = s[n-k]$ the problem is known as "smoothing". Other interesting cases include the so-called deconvolution, or equalization, problem, where $s[n]$ is a filtered (distorded) version of $d[n]$. Figure 5 depicts the process of optimal linear filtering for signal estimation. An objective

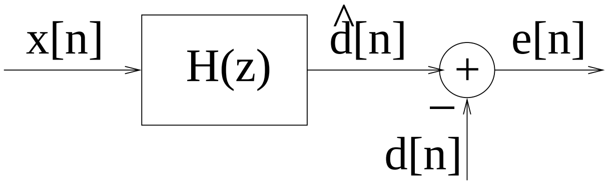


Figure 9: Optimal linear filter for signal estimation.

criterion is required to quantify what we mean by "optimal". A commonly used criterion is the mean square error (MSE). The filter coefficients are then selected to minimize the MSE

$$\text{MSE} = E\{(d[n] - \hat{d}[n])^2\}\,.$$

Since we are only looking for linear estimates, the optimal $\hat{d}[n]$ will be termed a linear minimum mean square error (LMMSE) estimate. In the case of Wiener filtering, the observed data $x[n]$ is assumed to be wide-sense stationary. This leads to a time-invariant filter $H(z)$, so that also $\hat{d}[n]$ is stationary. For time-varying signal models (including transients), the LMMSE principle leads to the Kalman filter, see e.g. [1].

Generally, the estimate $\hat{d}[n]$ is a linear combination of measured data samples

$$\hat{d}[n] = \sum_k h[k]x[n-k]\,.$$

The range of k, i.e., the extent of the impulse response $h[k]$, distinguishes between different cases of interest. Inserting the above into the MSE expression leads to

$$\text{MSE}(\mathbf{h}) = E\left\{(d[n] - \sum_k h[k]x[n-k])^2\right\}\,,$$

where the dependence of the MSE on $\mathbf{h}$, the vector of (possibly infinitely many) impulse response coefficients, has been stressed. It is clear that the MSE is a concave quadratic function of the impulse response coefficients. Hence, we can find the minimum simply by differentiating w.r.t. $h[l]$, l ranging over the same values as k, and setting the derivatives to zero. Assuming we can interchange the order of differentiation and expectation, we get

$$\frac{\partial}{\partial h[l]}\text{MSE}(\mathbf{h}) = E\left\{\frac{\partial}{\partial h[l]}(d[n] - \sum_k h[k]x[n-k])^2\right\} = E\left\{2(d[n] - \sum_k h[k]x[n-k])(-x[n-l])\right\}$$

Evaluating the expectation now yields

$$\frac{\partial}{\partial h[l]}\text{MSE}(\mathbf{h}) = -2r_{dx}[l] + 2\sum_k h[k]r_x[l-k]$$

Equating the derivative to zero shows that the optimal filter coefficients must satisfy

$$\sum_k h[k]r_x[l-k] = r_{dx}[l]\,. \tag{8}$$

This relation, for a suitable range of k and l, is known as the Wiener-Hopf (W-H) equations.

5.1 Causal FIR Wiener Filters

We shall first review case of a causal finite impulse response (FIR) filter. The estimate of $d[n]$ is then based on p past samples of $x[n]$:

$$\hat{d}[n] = \sum_{k=0}^{p-1} h[k]x[n-k]$$

Thus, $H(z)$ is the FIR filter

$$H(z) = \sum_{k=0}^{p-1} h[k]z^{-k}$$

For this case, the W-H equations become

$$\sum_{k=0}^{p-1} h[k]r_x[l-k] = r_{dx}[l]\,,\ l = 0, 1, \ldots, p-1.$$

This is a set of p linear equations in the p unknowns $h[0], \ldots, h[p-1]$. The solution is found by expressing the W-H equations in matrix form:

$$\begin{bmatrix} r_x[0] & r_x[1] & \cdots & r_x[p-1] \\ r_x[1] & r_x[0] & \cdots & r_x[p-2] \\ \vdots & \vdots & \ddots & \vdots \\ r_x[p-1] & r_x[p-2] & \cdots & r_x[0] \end{bmatrix} \begin{bmatrix} h[0] \\ h[1] \\ \vdots \\ h[p-1] \end{bmatrix} = \begin{bmatrix} r_{dx}[0] \\ r_{dx}[1] \\ \vdots \\ r_{dx}[p-1] \end{bmatrix}$$

We can express this compactly as $\mathbf{R}_x\mathbf{h} = \mathbf{r}_{dx}$, and the solution is given by $\mathbf{h}_{FIR} = \mathbf{R}_x^{-1}\mathbf{r}_{dx}$. This gives the coefficients of the optimal FIR Wiener filter. The minimum MSE can be shown to be

$$\text{MSE}(\mathbf{h}_{FIR}) = r_d[0] - \sum_{k=0}^{p-1} r_{dx}[k]h_{FIR}[k] = r_d[0] - \mathbf{r}_{dx}^T\mathbf{h}_{FIR}$$

Once the optimal filter has been found, the best estimate (in the LMMSE sense) is given by

$$\hat{d}[n] = \sum_{k=0}^{p-1} h_{FIR}[k]x[n-k] = \mathbf{h}_{FIR}^T\mathbf{x}[n]\,.$$

To find the optimal LMMSE filter requires apparently that the autocorrelation sequence $r_x[k]$ and cross correlation sequence $r_{dx}[k]$, for $k = 0, 1, \cdots, p-1$, are known in advance. In many practical situations this information is unavailable, and in addition the statistical properties may be (slowly) time-varying. In such a case, it may be possible to apply a so-called *adaptive* filter, which learns the optimal filter coefficients from data. In this context, FIR filters are highly useful, since they are always guaranteed to be stable. Adaptive filtering will be explored in the forthcoming course Applied Signal Processing.

Example 5.1 FIR Wiener Filter
Suppose a desired signal $d[n]$ is corrupted by additive noise $w[n]$, so the measured signal $x[n]$ is given by $x[n] = d[n] + w[n]$. The desired signal is modeled by the AR(1) equation

$$d[n] - 0.5d[n-1] = e_d[n]\,,$$

whereas the disturbance is modeled by

$$w[n] + 0.5w[n-1] = e_w[n]\,.$$

Here, $e_d[n]$ and $e_w[n]$ are zero-mean white noises with variance $\sigma^2_{ed} = \sigma^2_{ew} = 1$. The signal and noise spectra are displayed in Figure 10, left plot. The signal is of low-pass character whereas the noise is high-pass. The auto-correlation functions are computed by the Yule-Walker equations, which for $d[n]$ reads

$$\begin{aligned} r_d[0] - 0.5r_d[1] &= 1 \\ r_d[1] - 0.5r_d[0] &= 0 \end{aligned}$$

with solution $r_d[0] = 4/3$, $r_d[1] = 2/3$, and in general $r_d[k] = \frac{4}{3}0.5^{|k|}$. Similarly, $r_w[k] = \frac{4}{3}(-0.5)^{|k|}$. Assuming $d[n]$ and $w[n]$ to be independent, we get $r_x[k] = r_d[k] + r_w[k]$ and $r_{dx}[k] = r_d[k]$. Thus, the W-H equations for an FIR Wiener filter of length $p = 2$ is obtained as

$$\begin{bmatrix} 8/3 & 0 \\ 0 & 8/3 \end{bmatrix} \begin{bmatrix} h[0] \\ h[1] \end{bmatrix} = \begin{bmatrix} 4/3 \\ 2/3 \end{bmatrix}$$

with solution $h[0] = 0.5$ and $h[1] = 0.25$. The magnitude of the optimal filter is shown in Figure 10, right plot. As expected, the Wiener filter is passing frequencies where the signal dominates the noise and

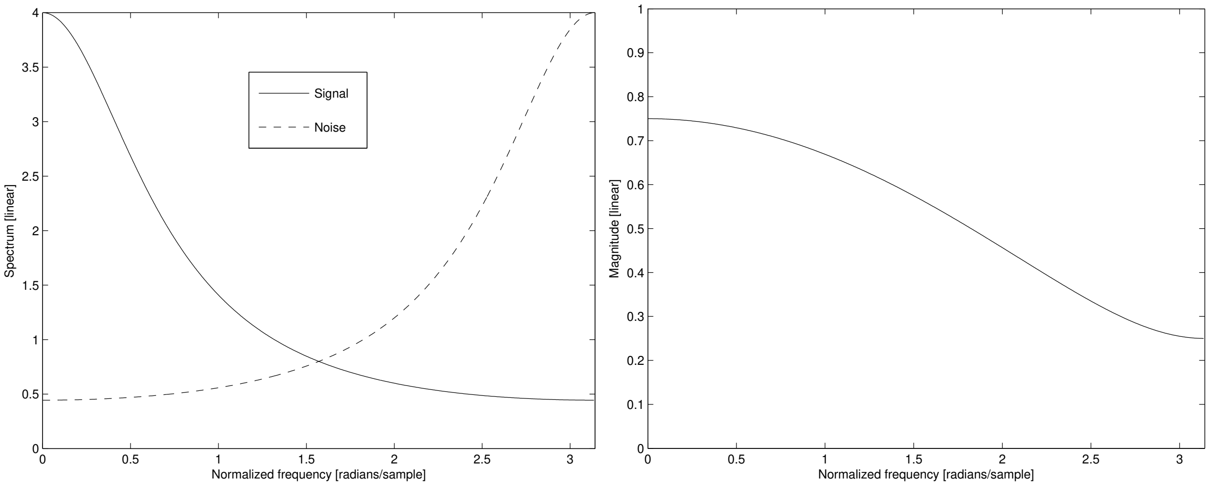


Figure 10: *Signal and noise spectra (left plot), and magnitude of the corresponding length $p = 2$ FIR Wiener filter (right plot).*

suppresses those where the noise is stronger. The minimum MSE is reduced from $\sigma^2_w = 4/3$ (no filtering) to

$$\text{MSE}_{min} = 4/3 - 0.5 \times (4/3) - 0.25 \times (2/3) = 0.5\,.$$

□

Example 5.2 Linear Prediction
Consider once again the AR(2) model

$$x[n] - 1.5x[n-1] + 0.7x[n-2] = e[n]$$

where $e[n]$ is zero-mean white noise with variance $\sigma_e^2 = 1$. Suppose we wish to predict the future values of $\{x[n]\}$ based on past observations. We choose an FIR predictor of length $p = 2$:

$$\hat{x}[n+k] = h[0]x[n] + h[1]x[n-1]\,,$$

where $k > 0$. Thus $d[n] = x[n+k]$ in this case. The coefficients $\{h[0], h[1]\}$ should be chosen to minimize the MSE

$$\text{MSE}(h[0], h[1]) = E\{(x[n+k] - \hat{x}[n+k])^2\}\,.$$

The WH equations are in this case given by:

$$\begin{bmatrix} r_x[0] & r_x[1] \\ r_x[1] & r_x[0] \end{bmatrix} \begin{bmatrix} h[0] \\ h[1] \end{bmatrix} = \begin{bmatrix} r_x[k] \\ r_x[k+1] \end{bmatrix}$$

Solving this 2×2 linear system gives:

$$\begin{bmatrix} h[0] \\ h[1] \end{bmatrix} = \frac{1}{r_x^2[0] - r_x^2[1]} \begin{bmatrix} r_x[0]r_x[k] - r_x[1]r_x[k+1] \\ r_x[0]r_x[k+1] - r_x[1]r_x[k] \end{bmatrix}$$

The auto-correlation function for the given AR model was calculated in Example 3.1:

$$r_x[0] \approx 8.85\,, r_x[1] \approx 7.81\,, r_x[2] \approx 5.52\,, r_x[3] \approx 2.81\,, r_x[4] \approx 0.35\,\ldots$$

Inserting these values into the expression for the optimal filter coefficients yields for $k = 1$:

$$\{h[0], h[1]\}_{k=1} = \{1.5, -0.7\}$$

Thus, the optimal one-step ahead predictor is given by

$$\hat{x}[n+1] = 1.5x[n] - 0.7x[n-1]\,.$$

This is an intuitively appealing result, as seen by rewriting the AR model as

$$x[n+1] = 1.5x[n] - 0.7x[n-1] + e[n+1]\,.$$

At time n we have no information about $e[n+1]$, so the best is to replace it with its mean, which is zero. This results in our optimal predictor. The prediction error is in this case simply the white noise $e[n+1]$, and the minimum MSE is:

$$\text{MSE}_{k=1} = 8.85 - 1.5 \cdot 7.81 + 0.7 \cdot 5.52 = 1.00$$

as expected.
For $k = 3$ we get the filter coefficients:

$$\{h[0], h[1]\}_{k=3} = \{1.28, -2.31\}$$

The optimal three-step ahead predictor is therefore given by:

$$\hat{x}[n+3] = 1.28x[n] - 2.31x[n-1]\,.$$

The minimum MSE is now obtained as:

$$\text{MSE}_{k=3} = 8.85 - 1.28 \cdot 2.81 + 2.31 \cdot 0.35 = 6.06\,,$$

which is considerably higher than for $k = 1$. This is not surprising; prediction is of course more difficult the longer the prediction horizon is. □

5.2 Non-causal IIR Wiener Filters

The FIR filter is limited in the sense that its impulse response is required to be finite. One can expect that an infinite impulse response (IIR) filter can give more flexibility, and thus smaller MSE. The most flexible case is when the estimate $\hat{d}[n]$ is allowed to depend on all possible data:

$$\hat{d}[n] = \sum_{k=-\infty}^{\infty} h[k]x[n-k]$$

Thus, $H(z)$ is the non-causal IIR filter

$$H(z) = \sum_{k=-\infty}^{\infty} h[k]z^{-k}$$

In this case, the W-H equations are infinitely many, and they depend on an infinite set of unknowns:

$$\sum_{k=-\infty}^{\infty} h[k]r_x[l-k] = r_{dx}[l]\,,\ -\infty < l < \infty\,.$$

The solution is easily obtained by transforming the W-H equations to the frequency domain. Note that the term on the left-hand side is the convolution of $h[k]$ and $r_x[k]$. By taking the Z-transform, this is converted to multiplication:

$$H(z)P_x(z) = P_{dx}(z)\,.$$

Recall that $P_x(e^{j\omega})$ is the DTFT of $r_x[k]$, so $P_x(z)$ is the Z-transform. Likewise, $P_{dx}(z)$ denotes the Z-transform of the cross-correlation $r_{dx}[n]$. The optimal non-causal IIR filter is now easily obtained as

$$H_{nc}(z) = \frac{P_{dx}(z)}{P_x(z)}$$

If desired, the impulse response coefficients can be computed by inverse Z-transform.
An interesting special case is obtained if $x[n] = s[n] + w[n]$, where $s[n]$ and $w[n]$ are uncorrelated, and $d[n] = s[n]$. Then, it follows that

$$P_x(z) = P_s(z) + P_w(z)\,, \quad P_{dx}(z) = P_s(z)\,,$$

and the non-causal LMMSE estimate becomes (in the DTFT-domain)

$$H_{nc}(e^{j\omega}) = \frac{P_s(e^{j\omega})}{P_s(e^{j\omega}) + P_w(e^{j\omega})} = \frac{SNR(\omega)}{SNR(\omega)+1}\,,$$

where $SNR(\omega) = P_s(e^{j\omega})/P_w(e^{j\omega})$ is the Signal-to-Noise Ratio at the frequency ω. The nice interpretation is that the optimal filter should be $H_{nc}(e^{j\omega}) \approx 1$ at those frequencies where the SNR is high, and $H_{nc}(e^{j\omega}) \approx 0$ where the SNR is poor. When $P_s(e^{j\omega}) = P_w(e^{j\omega})$, which corresponds to 0 dB SNR, the filter should have a gain of about -6 dB. Also note that $H_{nc}(e^{j\omega})$ is purely real in this case, i.e. there is no phase distortion. This is of course possible only for a non-causal filter. However, in contrast to the analog case, non-causal digital filters can in fact be implemented in practice, just not in real time. Given a pre-recorded batch of data, the transfer function $H_{nc}(e^{j\omega})$ can, for example, be applied in the frequency domain, by first taking the DTFT of $\{x[n]\}$. The filtering will suffer from end effects (at both sides!) due to the finiteness of data. To optimally deal with initial effects requires the Kalman filter.
Once the optimal filter is found, the minimum MSE is computed by

$$\begin{aligned}
\mathrm{MSE}(H_{nc}(z)) &= r_d[0] - \sum_{k=-\infty}^{\infty} r_{dx}[k]h_{nc}[k] \\
&= \frac{1}{2\pi}\int_{-\pi}^{\pi} \left[P_d(e^{j\omega}) - H_{nc}(e^{j\omega})P_{dx}^*(e^{j\omega})\right] d\omega\,.
\end{aligned}$$

5.3 Causal IIR Wiener Filters

While the non-causal IIR Wiener filter yields the smallest possible MSE, it is not implementable in real-time. In the causal IIR Wiener filter, the signal estimate depends only on past measurements:

$$\hat{d}[n] = \sum_{k=0}^{\infty} h[k]x[n-k]$$

Thus, $H(z)$ is the causal IIR filter

$$H(z) = \sum_{k=0}^{\infty} h[k]z^{-k}$$

The W-H equations are now given by

$$\sum_{k=0}^{\infty} h[k]r_x[l-k] = r_{dx}[l]\,,\ 0 \le l < \infty\,.$$

Since the value of the left-hand side is unspecified for $l < 0$, a direct Z-transform (or DTFT) can no longer be applied. Instead, a more elaborate procedure, similar to the derivation at pp. 432-434 [2] or at pp. 491-493 [3], for the continuous-time case is required. We use a slightly different technique, which is based on introducing a non-causal "dummy" sequence $nc[l]$, which is added to the right-hand side of the W-H equations:

$$\sum_{k=0}^{\infty} h[k]r_x[l-k] = r_{dx}[l] + nc[l]\,,\ -\infty < l < \infty\,,$$

where $nc[l] = 0$ for $l \ge 0$. Now, we can take the Z-transform, which results in

$$H_c(z) = \frac{P_{dx}(z) + NC(z)}{P_x(z)}$$

We must now select the non-causal transfer function $NC(z)$ so that $H_c(z)$ becomes causal. Towards this goal, the spectrum of $x[n]$ is factorized as

$$P_x(z) = Q(z)Q(z^{-1})\,,$$

where the spectral factor $Q(z)$ is causal (it has all of its zeros inside the unit circle), whereas $Q(z^{-1})$ is anti-causal. This process is known as spectral factorization, and it is always possible because $P_x(e^{j\omega})$ is real-valued and non-negative. The causal IIR Wiener filter is now expressed as

$$H_c(z) = \frac{1}{Q(z)}\,\frac{P_{dx}(z) + NC(z)}{Q(z^{-1})}\,.$$

The next step is to split $P_{dx}(z)/Q(z^{-1})$ into its causal and anti-causal parts:

$$\frac{P_{dx}(z)}{Q(z^{-1})} = \left[\frac{P_{dx}(z)}{Q(z^{-1})}\right]_+ + \left[\frac{P_{dx}(z)}{Q(z^{-1})}\right]_-\,.$$

It is now clear that the anti-causal dummy filter $NC(z)$ should be chosen as

$$NC(z) = -Q(z^{-1})\,\left[\frac{P_{dx}(z)}{Q(z^{-1})}\right]_-\,.$$

Inserting this into the Wiener filter formula shows that the optimal causal IIR Wiener filter is given by

$$H_c(z) = \frac{1}{Q(z)}\,\left[\frac{P_{dx}(z)}{Q(z^{-1})}\right]_+\,.$$

The minimum MSE is again obtained as

$$\begin{aligned}\mathrm{MSE}(H_c(z)) &= r_d[0] - \sum_{k=-\infty}^{\infty} r_{dx}[k]h_{nc}[k] \\ &= \frac{1}{2\pi}\int_{-\pi}^{\pi}\left[P_d(e^{j\omega}) - H_c(e^{j\omega})P_{dx}^*(e^{j\omega})\right]d\omega\,.\end{aligned}$$

Relations among the minimum MSEs In general, the minimum MSEs for the different types of Wiener filters satisfy the following relations

$$\text{MSE}(\mathbf{h}_{FIR}) \geq \text{MSE}(H_c(z)) \geq \text{MSE}(H_{nc}(z))\,.$$

However, as the length of the FIR filter tends to infinity, we have $\text{MSE}(\mathbf{h}_{FIR}) \to \text{MSE}(H_c(z))$. Thus, for "large enough" values of p, the difference between the two is negligible. The difference between $H_c(z)$ and $H_{nc}(z)$ is mainly due to the phase. Both can approximate any amplitude characteristic arbitrarily well, but $H_{nc}(z)$ is bound to be causal, which generally introduces a time-delay and possibly also other phase distortions.

6 Supplementary Exercises on Digital Spectral Analysis and Optimal Filtering

1. We wish to estimate the spectrum of a stationary stochastic process using the method of averaged periodograms. The analog signal is sampled at a sampling frequency of 1 kHz. Suppose a frequency resolution of 1 Hz and a relative variance of 1% is desired for the estimated spectrum. How long time (in seconds) does it take to collect the required data?

2. It is desired to estimate the spectrum of a stationary continuous-time stochastic process $\{x_a(t)\}$. An ideal lowpass filter with $F_p = 5$ kHz is first applied to $\{x_a(t)\}$. The signal is then sampled, and a total of 10 s of data is collected. The spectrum is estimated by averaging non-overlapping periodograms. The periodogram lengths are selected to give a frequency resolution of approximately 10 Hz. Emilia uses a sampling frequency of $F_s = 10$ kHz, whereas Emil suggests that $F_s = 100$ kHz should give a lower variance of the spectrum estimate, since more data is obtained. Determine the normalized variance for the two choices of F_s, and investigate if Emil is right!

3. Let $x[n]$ be a stationary stochastic process. Suppose the spectrum is estimated as

$$\hat{P}_x\left(e^{j\omega}\right) = \sum_{n=-100}^{100} \hat{r}_x[n]w[n]e^{-j\omega n}$$

   where

$$w[n] = \begin{cases} e^{-0.1|n|} & ; \quad |n| \leq 100 \\ 0 & ; \quad n > 100 \end{cases}$$

   and $\hat{r}_x[n]$ is based on $N = 10000$ samples.

   Determine the approximate normalized variance

$$\nu = \frac{\text{Var}\left(\hat{P}_x(e^{j\omega})\right)}{P_x^2(e^{j\omega})}$$

4. Consider the following three signal models:

$$x[n] + 0.6x[n-1] - 0.2x[n-2] = e[n]$$
$$x[n] = e[n] + 0.8e[n-1] + 0.2e[n-2]$$
$$x[n] + 0.8x[n-1] = e[n] - 0.2e[n-1]$$

   where $e[n]$ is a zero-mean white noise with variance $\sigma_e^2 = 1$. Determine the autocorrelation function $r_x[k]$ and the spectrum $P_x(e^{j\omega})$ for each of the cases!

5. In a certain communication system simulation, it is necessary to generate samples from a low-pass random process. Suppose the specification is that $P_x(e^{j\omega})$ should be of "low-pass character", with DC-level 1 and 3-dB cutoff frequency $\omega_c = \pi/3$.

It is known that any "smooth" spectrum can be represented as a filtered white noise, and based on this Emilia suggests to use a first-order AR-process:

$$x[n] + ax[n-1] = e[n] \,.$$

Emil prefers a first-order MA-process:

$$x[n] = e[n] + be[n-1] \,.$$

Select the coefficients $\{a, \sigma_e^2\}$ and $\{b, \sigma_e^2\}$, respectively, so that the specifications are met, and comment on the result!

6. The spectrum of a stationary stochastic process is to be estimated from the data:

$$x[n] = \{0.6, -0.7, 0.2, 0.3\} \,.$$

Due to the small sample support, a simple AR(1)-model is exploited:

$$x[n] + a_1 x[n-1] = e[n] \,.$$

Determine estimates of the AR-parameter a_1 and the white noise variance σ_e^2. Based on these, give a parametric estimate of the spectrum $P_x(e^{j\omega})$!

7. We wish to estimate an Auto-Regressive model for a measured signal $x[n]$. The covariance function is estimated based on $N = 1000$ data points as

$$\hat{r}_x(0) = 7.73, \quad \hat{r}_x(1) = 6.80, \quad \hat{r}_x(2) = 4.75, \quad \hat{r}_x(3) = 2.36, \quad \hat{r}_x(4) = 0.23$$

Use the Yule-Walker method to estimate a model

$$x[n] + a_1 x[n-1] + a_2 x[n-2] = e[n]$$

for the signal, where $e[n]$ is white noise. Also give an estimate of the noise variance σ_e^2!

8. We are given a noise-corrupted measurement $x[n]$ of a desired signal $d[n]$. The observed signal is thus modeled by

$$x[n] = d[n] + w[n] \,.$$

Suppose $w[n]$ is white noise with variance $\sigma_w^2 = 1$. The desired signal is represented by the low-pass AR-model

$$d[n] = 0.9d[n-1] + e[n] \,,$$

where $e[n]$ is white noise with variance σ_e^2. An estimate of the desired signal is generated by filtering the measurement as

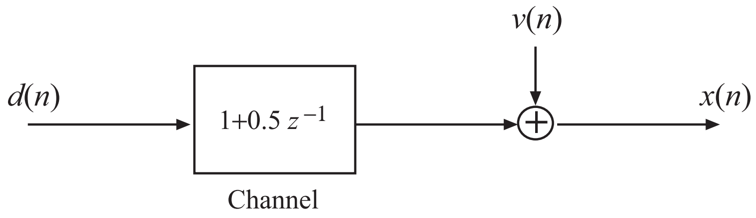
$$\hat{d}[n] = h_0 x[n] + h_1 x[n-1] \,.$$

Determine h_0 and h_1 such that $E[(d[n] - \hat{d}[n])^2]$ is minimized. Then sketch the amplitude characteristics of the resulting filter $h_0 + h_1 z^{-1}$ for σ_e^2 "small", "medium" and "large", respectively. Explain the result!

9. The goal of this problem is to equalize a communication channel using an FIR Wiener filter. The desired signal $d[n]$ is described as

$$d[n] - 0.8d[n-1] = e[n],$$

where $e[n]$ is zero-mean white noise with variance $\sigma_e^2 = 1$. The measured signal $x[n]$ is given by



where $v[n]$ is a zero-mean white noise with variance $\sigma_v^2 = 0.1$. Determine a filter

$$\hat{d}[n] = w_0\, x[n] + w_1\, x[n-1]$$

so that $E\left[\left(d[n] - \hat{d}[n]\right)^2\right]$ is minimized!

10. A 2:nd order digital FIR notch filter can be designed by placing the zeros at $e^{\pm j\omega}$. The resulting transfer function is

$$H(z) = 1 - 2\cos\omega z^{-1} + z^{-2}.$$

Thus, a natural approach for estimating the frequency of a sine-wave is to apply a constrained FIR filter

$$W(z) = 1 - wz^{-1} + z^{-2}$$

to the measured signal, and select w such that the output power is minimized. The frequency could then be estimated as $\omega = \arccos \frac{w}{2}$, provided $|w| < 2$. Now, suppose the measured signal is

$$x[n] = A\cos(\omega_0 n + \varphi) + v[n],$$

where φ is a $U(0, 2\pi)$ random phase and $v[n]$ is a zero-mean stationary white noise. The filter output is given by

$$y[n] = x[n] - wx[n-1] + x[n-2]\,.$$

Find the w that minimizes $E[y^2[n]]$, and compute $\hat{\omega} = \arccos \frac{w}{2}$. Then, prove that $\hat{\omega} = \omega_0$ for $\sigma_v^2 = 0$, but that the estimate is otherwise biased.

7 Solutions to Selected Problems

1. In the method of periodogram averaging (Bartlett's method), the available N samples are split into $K = N/M$ non-overlapping blocks, each of length M samples. The periodogram for each sub-block is computed, and the final estimate is the average of these. The variance of each sub-periodogram is

$$Var\{\hat{P}_{per}^{(i)}(e^{j\omega})\} \approx P_x^2(e^{j\omega})$$

Thus, the normalized variance is

$$\frac{Var\{\hat{P}_{per}^{(i)}(e^{j\omega})\}}{P_x^2(e^{j\omega})} \approx 1$$

or 100%. Assuming the sub-periodograms to be uncorrelated, averaging reduces the normalized variance to

$$\frac{Var\{\hat{P}_B(e^{j\omega})\}}{P_x^2(e^{j\omega})} \approx 1/K$$

To achieve a normalized variance of 0.01 (1%) thus requires averaging $K \geq 100$ sub-periodograms. Each sub-periodogram has the approximate frequency resolution $\Delta\omega \approx 2\pi/M$, or $\Delta f \approx 1/M$. In un-normalized frequency this becomes $\Delta F \approx F_s/M$ Hz. With $F_s = 1000$ and a desired frequency resolution of $\Delta F \leq 1$ Hz, the length of each sub-block must be $M \geq 1000$. Together with $K \geq 100$, this shows that the number of available samples must be $N = KM \geq 10^5$. At 1 kHz, the data collection takes at least 100 seconds.

3. For Blackman-Tukey's method, the normalized variance is given by

$$\nu = \frac{Var\{\hat{P}_{BT}(e^{j\omega})\}}{P_x^2(e^{j\omega})} \approx \frac{1}{N}\sum_{k=-M}^{M} w_{lag}^2[k]\,,$$

where $w_{lag}[k]$ is the lag window. In this case we get

$$\sum_{k=-M}^{M} w_{lag}^2[k] = \sum_{k=-100}^{100} e^{-0.2|k|} = 2\sum_{k=0}^{100} e^{-0.2k} - 1 = 2\frac{1-e^{-200}}{1-e^{-0.2}} - 1 \approx 10\,.$$

Thus, the normalized variance is $\nu = 10/N = 10^{-3}$.

4. The first model is an AR(2)-process:

$$x[n] + 0.6x[n-1] - 0.2x[n-2] = e[n]$$

Multiplying by $x[n-k]$, $k \geq 0$, and taking expectation leads to the relation

$$r_x[k] + 0.6r_x[k-1] - 0.2r_x[k-2] = \sigma_e^2\,\delta[k]$$

(Yule-Walker). With $k = \{0, 1, 2\}$, we get three equations in three unknowns (see also the theory complement):

$$\begin{bmatrix} 1 & 0.6 & -0.2 \\ 0.6 & 1-0.2 & 0 \\ -0.2 & 0.6 & 1 \end{bmatrix} \begin{bmatrix} r_x[0] \\ r_x[1] \\ r_x[2] \end{bmatrix} = \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix}$$

Solving this, for example using Matlab, leads to $r_x[0] = 2.4$, $r_x[1] = -1.8$ and $r_x[2] = 1.6$. We can then continue for $k = 3, 4, \ldots$ to get the autocorrelation at any lag, for example,

$$r_x[3] = -0.6r_x[2] + 0.2r_x[1] = -1.3\,, \quad r_x[4] = -0.6r_x[3] + 0.2r_x[2] = 1.1\,.$$

It is of course also possible to solve the homogenous difference equation

$$r_x[k] + 0.6r_x[k-1] - 0.2r_x[k-2] = 0\,,$$

with the given "initial conditions", to give the general solution. The spectrum of the AR model is given by

$$P_x(e^{j\omega}) = \frac{\sigma_e^2}{|A(e^{j\omega})|^2} = \frac{1}{(1+0.6e^{-j\omega}-0.2e^{-j2\omega})(1+0.6e^{j\omega}-0.2e^{j2\omega})} = \frac{1}{1.4+0.96\cos\omega-0.4\cos 2\omega}$$

The second model is an MA(2) process:

$$x[n] = e[n] + 0.8e[n-1] + 0.2e[n-2]$$

In this case, the autocorrelation function is finite. First, multiply by $x[n]$ and take expectation, which leads to

$$r_x[0] = E\{(e[n] + 0.8e[n-1] + 0.2e[n-2])^2\} = (1 + 0.8^2 + 0.2^2)\sigma_e^2 = 1.68\,.$$

Next, multiply by $x[n-1]$ and take expectation:

$$r_x[1] = E\{(e[n]+0.8e[n-1]+0.2e[n-2])(e[n-1]+0.8e[n-2]+0.2e[n-3])\} = (0.8+0.2\times 0.8)\sigma_e^2 = 0.96.$$

Finally, multiplying by $x[n-2]$ and taking expectation leads to

$$r_x[2] = E\{(e[n] + 0.8e[n-1] + 0.2e[n-2])(e[n-2] + 0.8e[n-3] + 0.2e[n-4])\} = 0.2\,.$$

Multiplying by $x[n-k]$, $k > 2$ and taking expectation wee see that $r_x[k] = 0$, $|k| > 2$. The spectrum of the MA process is given by

$$P_x(e^{j\omega}) = \sigma_e^2|B(e^{j\omega})|^2 = (1+0.8e^{-j\omega}+0.2e^{-j2\omega})(1+0.8e^{j\omega}+0.2e^{j2\omega}) = 1.68+1.92\cos\omega+0.4\cos 2\omega$$

The third model is the ARMA(1,1) process:

$$x[n] + 0.8x[n-1] = e[n] - 0.2e[n-1]$$

In general, computing the autocorrelation function for an ARMA model is quite difficult. However, in the ARMA(1,1) case the following procedure is easiest. First, multiply by $x[n-1]$ and take expectation, which gives:

$$r_x[1] + 0.8r_x[0] = -0.2\sigma_e^2\,.$$

Second, square the ARMA model equation and again take expectation:

$$E\{(x[n] + 0.8x[n-1])^2\} = E\{(e[n] - 0.2e[n-1])^2\}\,.$$

This leads to

$$1.64r_x[0] + 1.6r_x[1] = 1.04\sigma_e^2\,.$$

Now we have two equations in the two unknowns $r_x[0]$ and $r_x[1]$:

$$\begin{bmatrix} 0.8 & 1 \\ 1.64 & 1.6 \end{bmatrix} \begin{bmatrix} r_x[0] \\ r_x[1] \end{bmatrix} = \begin{bmatrix} -0.2 \\ 1.04 \end{bmatrix}$$

Applying the usual formula for inverting a 2×2 matrix, the solution is easily obtained as

$$\begin{bmatrix} r_x[0] \\ r_x[1] \end{bmatrix} = \frac{1}{0.8 \times 1.6 - 1 \times 1.64} \begin{bmatrix} 1.6 & -1 \\ -1.64 & 0.8 \end{bmatrix} \begin{bmatrix} -0.2 \\ 1.04 \end{bmatrix} \approx \begin{bmatrix} 3.8 \\ -3.2 \end{bmatrix}.$$

The remaining auto-correlation parameters can be obtained by multiplying the ARMA equation by $x[n-k],\ k \geq 2$ and taking expectation:

$$r_x[k] + 0.8r_x[k-1] = 0\,,\ k \geq 2.$$

Thus, $r_x[2] = -0.8r_x[1] \approx 2.6$ etc. In general, $r_x[k] = (-0.8)^{k-1}r_x[1] = (-0.8)^{k-1}(-3.2)$ for $k \geq 2$. The spectrum of the ARMA model is given by

$$P_x(e^{j\omega}) = \sigma_e^2 \frac{|B(e^{j\omega})|^2}{|A(e^{j\omega})|^2} = \frac{(1-0.2e^{j\omega})(1-0.2e^{-j\omega})}{(1+0.8e^{j\omega})(1+0.8e^{-j\omega})} = \frac{1-0.4\cos\omega}{1+1.6\cos\omega}$$

5. For Emilia's AR(1) model, the spectrum is given by

$$P_x(e^{j\omega}) = \frac{\sigma_e^2}{(1-ae^{j\omega})(1-ae^{-j\omega})} = \frac{\sigma_e^2}{1+a^2-2a\cos\omega}$$

DC-level 1 means $P_x(e^{j0}) = 1$, which gives $\sigma_e^2 = 1 + a^2 - 2a = (1-a)^2$. Cutoff frequency $\omega_c = \pi/3$ means $P_x(e^{j\pi/3}) = 1/2$, and since $\cos\pi/3 = 1/2$ we get

$$\frac{(1-a)^2}{1+a^2-a} = 0.5$$

The solutions are $a = (3 \pm \sqrt{5})/2$, from which we choose the stable solution (pole inside the unit circle): $a = (3-\sqrt{5})/2 \approx 0.382$, and $\sigma_e^2 \approx 0.382$.

Emil's MA(1) model gives

$$P_x(e^{j\omega}) = \sigma_e^2(1 + b^2 + 2b\cos\omega)$$

Thus we get $\sigma_e^2 = (1+b)^{-2}$ and

$$\frac{1+b^2+b}{(1+b)^2} = 0.5$$

This leads to $b^2 + 1 = 0$, which does not have any real-valued solutions. The conclusion is that an MA(1) model cannot meet the specifications. In general, AR-models can be made "sharper" than MA-models for a given order. An MA-model is of course also possible in this case, but it requires a higher order (how high?).

6. The Yule-Walker method gives the estimate

$$\hat{a}_1 = -\hat{r}_x[0]^{-1}\hat{r}_x[1]\,.$$

With the given data, the sample autocorrelation function is calculated as

$$\hat{r}_x[0] = \frac{1}{4}(0.6^2 + 0.7^2 + 0.2^2 + 0.3^2) = 0.245$$

and

$$\hat{r}_x[1] = \frac{1}{4}(0.6 \times (-0.7) + (-0.7) \times 0.2 + 0.2 \times 0.3) = -0.125$$

Thus, we get

$$\hat{a}_1 = \frac{0.125}{0.245} \approx 0.51\,.$$

The noise variance estimate follows as

$$\hat{\sigma}_e^2 = \hat{r}_x[0] + \hat{a}_1\hat{r}_x[1] \approx 0.18\,.$$

8. The optimal filter (in the LMMSE sense) is given by the Wiener-Hopf (W-H) equations:

$$\sum_k h[k]r_x[l-k] = r_{dx}[l]\,.$$

In this case, the filter is a causal FIR filter with two taps (first-order filter):

$$\hat{d}[n] = h_0 x[n] + h_1 x[n-1]\,.$$

Thus, the W-H equations are

$$\sum_{k=0}^{1} h_k r_x[l-k] = r_{dx}[l]\,,\ l = 0, 1.$$

In matrix form, this becomes

$$\begin{bmatrix} r_x[0] & r_x[1] \\ r_x[1] & r_x[0] \end{bmatrix} \begin{bmatrix} h_0 \\ h_1 \end{bmatrix} = \begin{bmatrix} r_{dx}[0] \\ r_{dx}[1] \end{bmatrix}$$

The measured signal is given by

$$x[n] = d[n] + w[n]\,.$$

The auto-correlation function is then given by

$$r_x[k] = E\{(d[n] + w[n])(d[n-k] + w[n-k])\} = r_d[k] + r_w[k]$$

(since $d[n]$ and $w[n]$ are uncorrelated). Now, $w[n]$ is white noise, so $r_w[k] = \sigma_w^2\delta[k] = \delta[k]$. The desired signal is the AR(1) process:

$$d[n] - 0.9d[n-1] = e[n]\,.$$

To compute the autocorrelation function we use the Yule-Walker equations, which on matrix form become

$$\begin{bmatrix} 1 & -0.9 \\ -0.9 & 1 \end{bmatrix} \begin{bmatrix} r_d[0] \\ r_d[1] \end{bmatrix} = \begin{bmatrix} 1 \\ 0 \end{bmatrix} \sigma_e^2.$$

Solving this gives

$$\begin{bmatrix} r_d[0] \\ r_d[1] \end{bmatrix} \approx \begin{bmatrix} 5.26 \\ 4.74 \end{bmatrix} \sigma_e^2.$$

Inserting the autocorrelation functions into the W-H equations now gives

$$\left(\begin{bmatrix} 5.26 & 4.74 \\ 4.74 & 5.26 \end{bmatrix} \sigma_e^2 + \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \right) \begin{bmatrix} h_0 \\ h_1 \end{bmatrix} = \begin{bmatrix} 5.26 \\ 4.74 \end{bmatrix} \sigma_e^2\,,$$

where we have also used that $r_{dx}[k] = r_d[k]$, since $d[n]$ and $w[n]$ are uncorrelated. It is possible, but a bit cumbersome, to solve this 2×2 system for a general σ_e^2. Instead we take the extreme cases $\sigma_e^2 \to 0$ and $\sigma_e^2 \to \infty$ first, and then the intermediate case $\sigma_e^2 = 1$. For σ_e^2 very small we have

$$\begin{bmatrix} h_0 \\ h_1 \end{bmatrix} \to \begin{bmatrix} 5.26 \\ 4.74 \end{bmatrix} \sigma_e^2 \to \begin{bmatrix} 0 \\ 0 \end{bmatrix}\,.$$

Thus, when the SNR is very low, the estimation is $\hat{d}[n] = 0$. For very large σ_e^2 we have instead

$$\begin{bmatrix} 5.26 & 4.74 \\ 4.74 & 5.26 \end{bmatrix} \begin{bmatrix} h_0 \\ h_1 \end{bmatrix} = \begin{bmatrix} 5.26 \\ 4.74 \end{bmatrix},$$

which gives the optimal FIR coefficients as

$$\begin{bmatrix} h_0 \\ h_1 \end{bmatrix} = \begin{bmatrix} 1 \\ 0 \end{bmatrix}$$

Thus, in the high SNR case we take $\hat{d}[n] = x[n]$, i.e. no filtering is necessary. For the intermediate case $\sigma_e^2 = 1$ we get

$$\begin{bmatrix} 6.26 & 4.74 \\ 4.74 & 6.26 \end{bmatrix} \begin{bmatrix} h_0 \\ h_1 \end{bmatrix} = \begin{bmatrix} 5.26 \\ 4.74 \end{bmatrix},$$

which leads to

$$\begin{bmatrix} h_0 \\ h_1 \end{bmatrix} = \begin{bmatrix} 0.63 \\ 0.28 \end{bmatrix}.$$

The estimate is then $\hat{d}[n] = 0.62x[n] + 0.28x[n-1]$, corresponding to the transfer function

$$H(e^{j\omega}) = 0.62 + 0.28e^{-j\omega}.$$

This is a low-pass filter with DC gain $H(e^{j0}) = 0.62 + 0.28 = 0.9$, and 3 dB cutoff frequency $f_c \approx 0.25$ "per samples". The filter is an optimal compromise for removing as much noise (which has a flat spectrum) as possible, without distorting the signal (which is low-pass) too much.

References

[1] M. Hayes, *Statistical Digital Signal Processing*. John Wiley & Sons, New York 1996.

[2] S. L. Miller and D. G. Childers, *Probability and Random Processes With Applications to Signal Processing and Communications*. Elsevier Academic Press, Burlington, MA, 2004.

[3] S. L. Miller and D. G. Childers, *Probability and Random Processes With Applications to Signal Processing and Communications*, Second Edition. Elsevier Academic Press, Waltham, MA, 2012.

[4] A. V. Oppenheim, A. S. Willsky and H. Nawab, *Signals and Systems*, Second Edition. Prentice-Hall, Upper Saddly River, NJ, 1997.

[5] P. Stoica and R. Moses, *Spectral Analysis of Signals*. Prentice-Hall, Upper Saddly River, NJ, 2005.