You can repeat Example 7.1 using a different MATLAB random number generator called *rand*, which follows another stochastic process called *uniform probability distribution*. Definition and discussion of *probability distributions* of random processes is beyond the scope of this book, but can be found in a textbook on probability, such as that by Popoulis [1].

7.2 Filtering of Random Signals

When random signals are passed through a deterministic system, their statistical properties are modified. A deterministic system to which random signals are input, so that the output is a random signal with desired statistical properties is called a *filter*. Filters can be linear or nonlinear, time-invariant or time varying. However, for simplicity we will usually consider linear, time-invariant filters. Linear, time-invariant filters are commonly employed in control systems to reduce the effect of measurement noise on the control system. In such systems, the output is usually a superposition of a deterministic signal and a random measurement noise. Often, the measurement noise has a predominantly *high-frequency* content (i.e. its power spectral density has *more peaks* at *higher* frequencies). To filter-out the high-frequency noise, a special filter called a *low-pass* filter is employed, which blocks *all* signal frequencies *above* a specified *cut-off frequency*, ω_0. The output of a low-pass filter thus contains only lower frequencies, i.e. $\omega < \omega_0$, which implies a *smoothening* of the input signal fed to the filter. Sometimes, it is desired to block both high- and low-frequency contents of a noisy signal. This is achieved by passing the signal through a *band-pass* filter, which allows only a specified band of frequencies, $\omega_1 < \omega < \omega_2$, to pass through as the output signal. Similarly, a *high-pass* filter blocks all frequencies *below* a specified *cut-off frequency*, ω_0, and has an output containing only the *higher* frequencies, i.e. $\omega > \omega_0$.

Since it is impossible for a filter to *perfectly block* the undesirable signals, it is desired that the magnitude of signal above or below a given frequency decays rapidly with frequency. Such a decay of signal magnitude with frequency is called *attenuation*, or *roll-off*. The output of a filter not only has a frequency content different from the input signal, but also certain other characteristics of the filter, such as a phase-shift or a change in magnitude. In other words, the signal passing through a filter is also *distorted* by the filter, which is undesirable. It is inevitable that a filter would produce an output signal based upon its characteristics, described by the transfer-function, frequency response, impulse response, or a state-space representation of the filter. However, a filter can be designed to achieve a desired set of performance objectives – such as cut-off frequencies, desired *attenuation (roll-off)* of signals above or below the cut-off frequencies, etc., with the *minimum* possible *distortion* of the signals passing through the filter. Usually, it is observed that a greater attenuation of noise also leads to a greater distortion of the filtered signal. It is beyond the scope of this book to discuss the many different approaches followed in filter design, and you may refer to Parks and Burrus [2] for details. It suffices here to state that the numerator and denominator polynomials of the filter's transfer function, or coefficient matrices of the filter's state-space model, can be selected by a design process to achieve the conflicting requirements of maximum noise attenuation and minimum signal distortion.

Example 7.2

Consider a single-input, single-output filter with the following transfer function:

$$G(s) = \omega_0/(s + \omega_0) \tag{7.15}$$

This is the simplest possible low-pass filter with cut-off frequency, ω_0. Let us pass the following signal – which is a deterministic system's output ($\sin(10t)$) corrupted by a normally distributed random noise – through this low-pass filter:

$$u(t) = \sin(10t) + 0.2^{*}randn(t) \tag{7.16}$$

where *randn(t)* denotes the random noise generated by the MATLAB random number generator with a normal distribution (see Example 7.1). The random input signal is generated as follows:

```
>> t=0:0.001:1;randn('seed',0);u=sin(10*t)+0.2*randn(size(t)); <enter>
```

To block the high-frequency random noise, a cut-off frequency $\omega_0 = 10$ rad/s is initially selected for the low-pass filter, and the Bode plot of the filter is obtained as follows:

```
>> n = 10; d = [1 10]; sys=tf(n,d); [mag, phase,w] = bode(sys); <enter>
```

The filter is re-designed with cut-off frequency values, $\omega_0 = 40$ and 100 rad/s. The Bode plots of the low-pass filter with cut-off frequency values $\omega_0 = 10$, 40, and 100 rad/s, are plotted in Figure 7.2. Note that the magnitude of the filter decays with frequency at $\omega \geq \omega_0$, providing *noise attenuation* for frequencies above ω_0. The filter does not *totally* block the frequencies above the cut-off frequency, but decreases the magnitude (*attenuates*) almost linearly with the logarithmic frequency. The ideal *roll-off* (i.e. slope of decreasing magnitude with frequency) for filtering noise is 20 dB per 10 divisions of logarithmic frequency scale in rad/s (called *20 dB per decade*). The smaller the value of the cut-off frequency, the earlier would noise attenuation begin, and smaller will be the noise content of the filtered signal. It can be seen in the magnitude plot of Figure 7.2 that the filter with $\omega_0 = 10$ rad/s achieves a roll-off of 20 dB per decade at frequencies above 20 rad/s, while the other two filters achieve the same roll-off at much higher frequencies. The phase plot in Figure 7.2 shows that the low-pass filter *decreases* the phase of the input signal. Hence, a first order low-pass filter is also said to act as a *phase-lag* device. However, the phase of filter with $\omega_0 = 10$ and 40 rad/s begins to change at frequencies below 10 rad/s, which implies that the deterministic part of the random input signal, $\sin(10t)$, would be appreciably *distorted* (i.e. changed in wave-form) by these two filters, which is undesirable. The phase of the filter with $\omega_0 = 100$ rad/s is relatively unchanged until 10 rad/s, indicating little distortion of the deterministic signal.

A simulation of the filter's output is obtained using Control System Toolbox (CST) function *lsim* as follows:

```
>>[y,t,X]=lsim(sys,u,t); <enter>
```

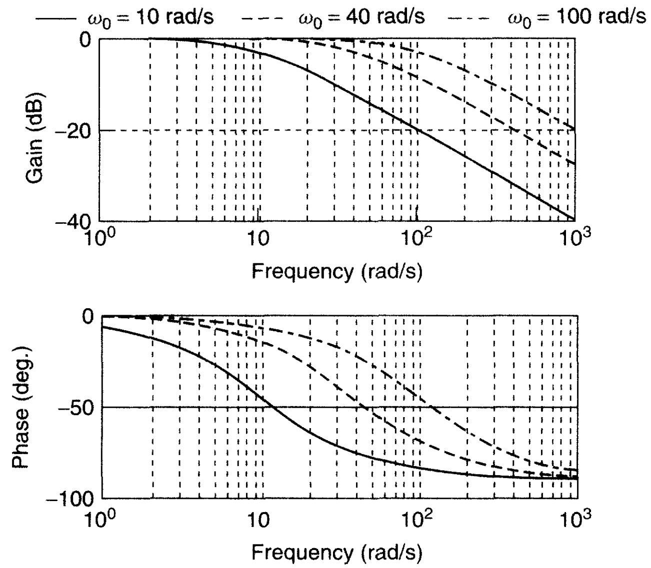

Figure 7.2 Bode plots of a first-order, low-pass filter with cut-off frequency, $\omega_0 = 10$, 40, and 100 rad/s

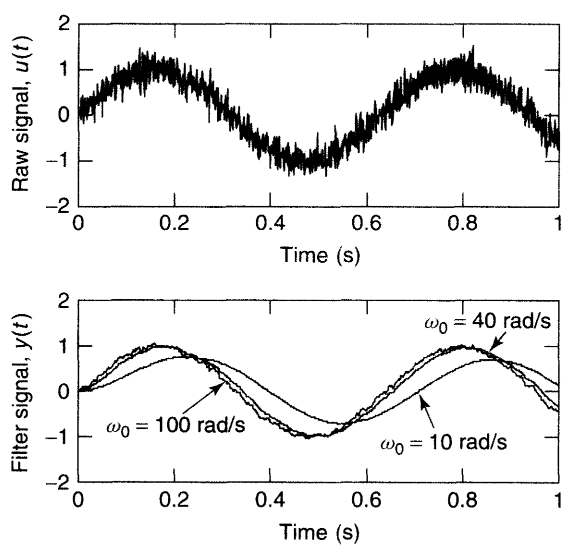

Figure 7.3 Raw signal and filtered signals after passing through a first-order, low-pass filter with cut-off frequency, 10, 40, and 100 rad/s, respectively

The input, $u(t)$, (i.e. raw signal) and the output, $y(t)$ (i.e. the filtered signal) for $\omega_0 = 10$, 40, and 100 rad/s are plotted in Figure 7.3. Note in Figure 7.3 that the filtered signals are smoother than the raw signal, but have an appreciable *distortion* in the wave-form, compared to the desired noise-free signal $\sin(10t)$). Among the three filter designs, for the filter with $\omega_0 = 10$ rad/s the distortion is maximum, but the filtered signal is the smoothest. For the filter with $\omega_0 = 100$ rad/s, the waveform distortion is minimum, but the filtered signal is the roughest indicating that the filter has allowed a lot of noise to pass through. An intermediate value of the cut-off frequency, $\omega_0 = 40$ rad/s, provides a good compromise between the conflicting requirements of smaller signal distortion and greater noise attenuation.

Example 7.3

Let us consider filtering a random signal which consists of a linear superposition of a deterministic signal, $\sin(20t) + \sin(50t)$, with a random noise given by

$$u(t) = \sin(20t) + \sin(50t) + 0.5^{*}randn(t) \tag{7.17}$$

where *randn(t)* denotes the random noise generated by the MATLAB's normal distribution random number generator. To filter the noise with least possible distortion, a sophisticated low-pass filter, called *elliptic filter*, is used with the following state-space representation:

$$\mathbf{A} = \begin{bmatrix} -153.34 & -94.989 & 0 & 0 \\ 94.989 & 0 & 0 & 0 \\ -153.34 & 104940 & -62.975 & -138.33 \\ 0 & 0 & 138.33 & 0 \end{bmatrix}$$

$$\mathbf{B} = \begin{bmatrix} 120 \\ 0 \\ 120 \\ 0 \end{bmatrix}$$

$$\mathbf{C} = [-1.2726 \times 10^{-5} \quad 8.7089 \times 10^{-3} \quad -5.2265 \times 10^{-6} \quad 1.0191 \times 10^{-3}]$$

$$\mathbf{D} = 9.9592 \times 10^{-6}$$

The Bode plot of the fourth order elliptic filter, shown in Figure 7.4, is obtained as follows:

```
>>sfilt=ss(A,B,C,D); bode(sfilt) <enter>
```

Figure 7.4 shows a *cut-off frequency* of 120 rad/s, a roll-off of about 100 dB per decade between the cut-off frequency and 1000 rad/s, and *ripples* in the magnitude for frequency greater than 1000 rad/s. The *passband*, i.e. the band of frequencies which the filter lets pass ($\omega \leq 120$ rad/s), is seen to be *flat* in magnitude, which implies a negligible magnitude distortion of the deterministic part of the filtered signal. The *stopband*, i.e. the band of frequencies which the filter is supposed to block ($\omega > 120$ rad/s), is seen to have ripples at -100 dB magnitude. As this magnitude is very small, there is expected to be a negligible influence of the ripples on the filtered signal. The phase plot shows a gradual 180° phase change in the passband, and rapid phase changes in the stopband above 1000 rad/s.

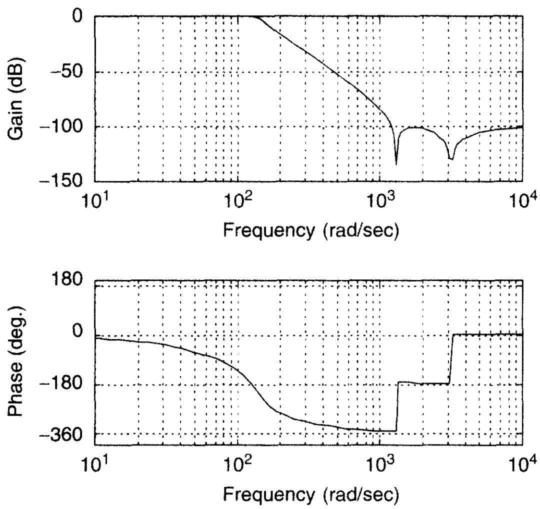

Figure 7.4 Bode plot of a fourth-order, low-pass elliptic filter with cut-off frequency 120 rad/s (Example 7.3)

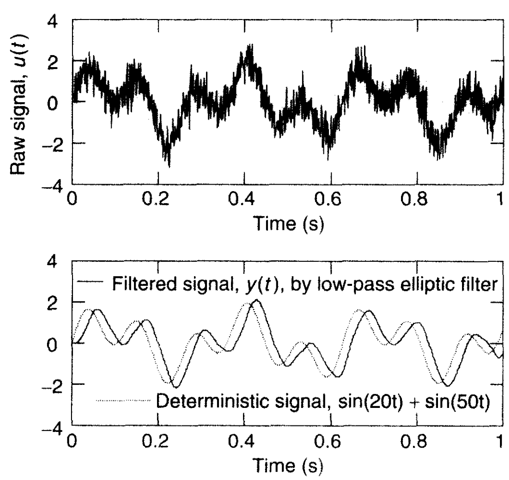

Figure 7.5 Raw random plus deterministic signal, and filtered signal after passing through a fourth-order, low-pass elliptic filter with cut-off frequency 120 rad/s

The filter output is simulated using MATLAB command *lsim* as follows:

```
>>[y,t,X] = lsim(sfilt,u,t); <enter>
```

Figure 7.5 shows time-plots of the raw signal, $u(t)$, and the filtered signal, $y(t)$, compared with the desired deterministic output, $\sin(20t) + \sin(50t)$. The filtered signal is seen to be smooth, and with only a phase-lag as the signal distortion, when compared with the desired deterministic signal.