Discrete-Time Signals and Systems

Textbook pages 118-130

Dr D. Laurenson

3rd September 2020

1 Introduction

In its basic form, correlation is a function that expresses the similarity of two signals, allowing for scaling and time shifts. Mathematically, the computation is very similar in form to convolution, however the purpose of computing the result is very different.

Consider a radar system where the received signal is:

$$y(n) = \alpha x(n-D) + w(n) \tag{6.1}$$

where $x(n)$ is the transmitted signal, $w(n)$ is random noise, and D is the time delay related to the distance to the target. The received signal is assumed to be a linear combination of the transmitted signal and a noise term. Whilst it would be possible to use a single pulse for $x(n)$, making detection in $y(n)$ very straightforward, it can be very helpful to make $x(n)$ more complicated. This can help to ensure that multiple radars do not detect each other's signals, and can also be used to make detection by the target more difficult. However, making the signal more complicated, whilst maintaining the same transmit energy, means that the received signal is likely to be dominated by noise.

2 Correlation

In order to detect the presence of the transmitted signal in the received radar return, correlation can be used. Correlation is a technique that determines the relationship between two signals, allowing for a displacement in time. Crosscorrelation between two signals is defined as:

$$r_{xy}(l) = \sum_{n=-\infty}^{\infty} x(n)y(n-l)\ ;\ l \in \mathbb{Z} \tag{6.3}$$

or

$$r_{xy}(l) = \sum_{n=-\infty}^{\infty} x(n+l)y(n)\ ;\ l \in \mathbb{Z} \tag{6.4}$$

This finds the similarity between $x(n)$ and $y(n)$ delayed by l samples. Where the similarity is large, $r_{xy}(l)$ will be large. Where the signals are not similar, $r_{xy}(l)$ will be close to 0. Note that we consider only real sequences at this point; the definition of correlation for complex valued sequences includes a complex conjugation operation. Although this is not required in this lecture course, for completeness:

$$r_{xy}(l) = \sum_{n=-\infty}^{\infty} x(n)y^*(n-l)\ ;\ l \in \mathbb{Z}$$

where * denotes complex conjugation.

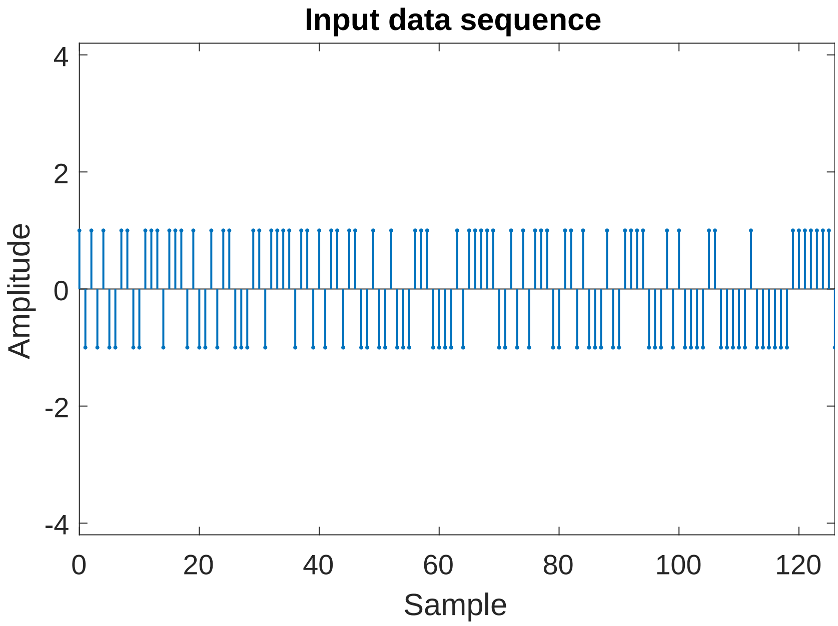
Crosscorrelation example

Since, $y(n) = \alpha x(n-D) + w(n)$, then

$$\begin{aligned} r_{xy}(l) &= \sum_{n=-\infty}^{\infty} x(n)y(n-l)\ ;\ l \in \mathbb{Z} \\ &= \alpha \sum_{n=-\infty}^{\infty} x(n)x(n-D-l) + \sum_{n=-\infty}^{\infty} x(n)w(n-l) \end{aligned}$$

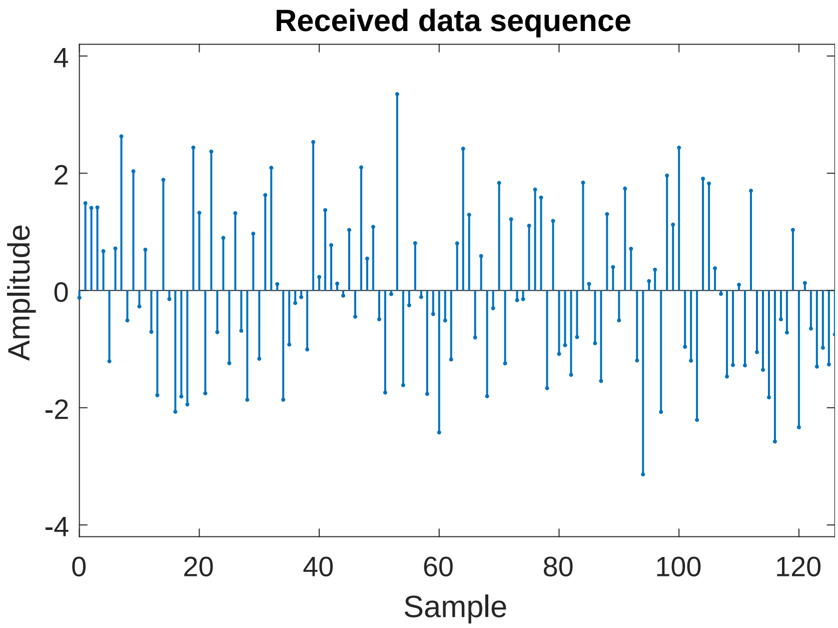
When $l = -D$, the first term is a maximum, and a peak will be observed in $r_{xy}(l)$. As the noise is not related to the transmitted signal, the second term will be approximately constant. Note that we will consider the second term in more detail when we explore the topic of signals in noise.

$x(n)$ is specified by the system designer. For example:



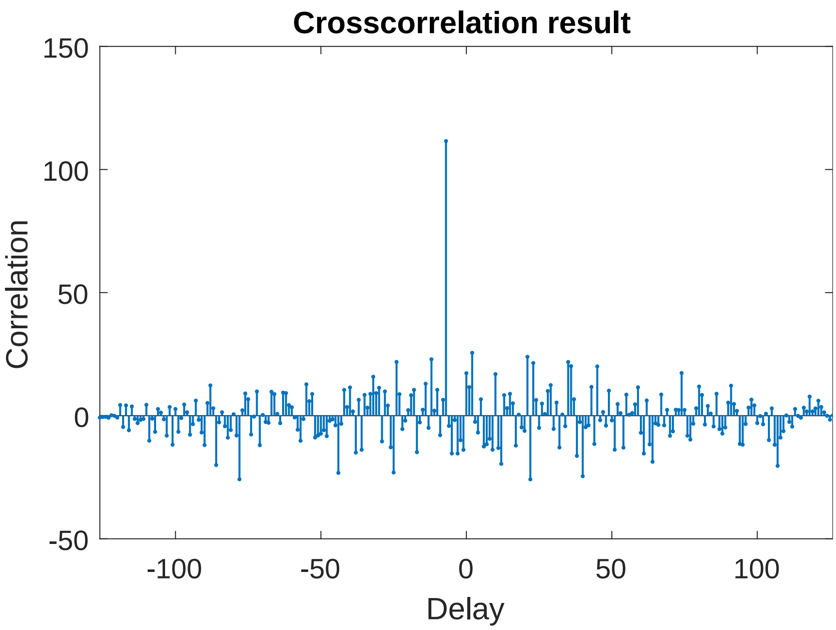
In this case, the signal is generated using a deterministic algorithm. The particular sequence shown here is an m-sequence of length 127 samples. (Note that m-sequences are not in the syllabus for this course, but interested readers can find out more about these at: http://dx.doi.org/10.1007/11863854_1)

The received signal, $y(n)$, is corrupted by noise, as well as an unknown delay:



For this particular example, the average noise power is set to be the same as the signal power at the receiver. It is obvious that the noise is sufficiently large that the transmitted sequence cannot be observed directly. Similarly, any system that does not know the sequence in advance cannot detect its presence in the received signal. For covert observation, this is clearly an advantage of this technique.

Next, a crosscorrelation is performed between the transmitted sequence and the received data. Crosscorrelation indicates the signal strength and delay



The height of the peak is dependent upon the scaling parameter, α, that is a function of the distance and target size. The location of the peak is dependent upon the delay with which the signal is received, D. In this particular example, $\alpha = 1$ and $D = 7$.

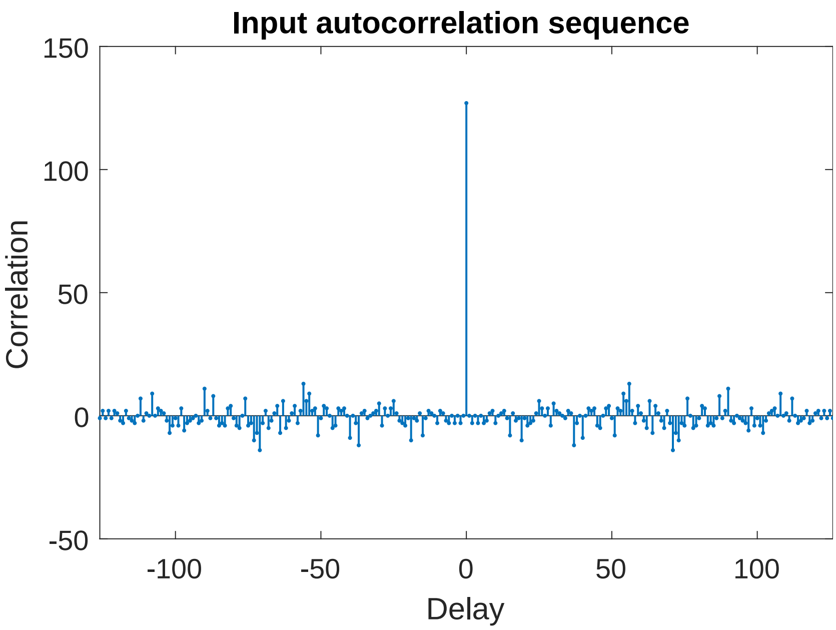
Using the definition of crosscorrelation, but making $y(n) = x(n)$, defines the autocorrelation:

$$r_{xx}(l) = \sum_{n=-\infty}^{\infty} x(n)x(n-l) \; ; \; l \in \mathbb{Z} \tag{6.9}$$

or

$$r_{xx}(l) = \sum_{n=-\infty}^{\infty} x(n+l)x(n) \; ; \; l \in \mathbb{Z} \tag{6.10}$$

It determines the self-similarity of a signal at different time shifts, l. The autocorrelation of a sequence is an important property when designing signals to be used as inputs to systems. For the radar example, it is important that the autocorrelation of the signal has one strong peak, and at other time shifts the autocorrelation is of small amplitude. The example sequence does have a large peak relative to other delays, however it is possible to design sequences with smaller amplitudes for all delays except for $l = 0$.



To observe the mechanics of the process, consider a second example, which is given in the textbook: Determine the crosscorrelation of $x(n) = \{\ldots, 0, 0, 2, -1, 3, 7, \underset{\uparrow}{1}, 2, -3, 0, 0, \ldots\}$ and $y(n) = \{\ldots, 0, 0, 1, -1, 2, -2, \underset{\uparrow}{4}, 1, -2, 5, 0, 0, \ldots\}$. Note that the arrow indicates the sample corresponding to $n = 0$.

To find $r_{xy}(0)$, we determine $x(n)y(n) = \{\ldots, 0, 0, 2, 1, 6, -14, \underset{\uparrow}{4}, 2, 6, 0, 0, \ldots\}$, so the sum makes $r_{xy}(0) = 7$.

For $r_{xy}(1)$, a shift of y gives $x(n)y(n-1) = \{\ldots, 0, 0, 0, -1, -3, 14, \underset{\uparrow}{-2}, 8, -3, 0, 0, \ldots\}$, so $r_{xy}(1) = 13$.

For the remaining delays, the table below gives the details.

l	$x(n)y(n-l)$	$r_{xy}(l)$
-7	$\{\ldots, 0, 0, 10, 0, 0, 0, \underset{\uparrow}{0}, 0, 0, 0, 0, \ldots\}$	10
-6	$\{\ldots, 0, 0, -4, -5, 0, 0, \underset{\uparrow}{0}, 0, 0, 0, 0, \ldots\}$	-9
-5	$\{\ldots, 0, 0, 2, 2, 15, 0, \underset{\uparrow}{0}, 0, 0, 0, 0, \ldots\}$	19
-4	$\{\ldots, 0, 0, 8, -1, -6, 35, \underset{\uparrow}{0}, 0, 0, 0, 0, \ldots\}$	36
-3	$\{\ldots, 0, 0, -4, -4, 3, -14, \underset{\uparrow}{5}, 0, 0, 0, 0, \ldots\}$	14
-2	$\{\ldots, 0, 0, 4, 2, 12, 7, \underset{\uparrow}{-2}, 10, 0, 0, 0, \ldots\}$	33
-1	$\{\ldots, 0, 0, -2, -2, -6, 28, \underset{\uparrow}{1}, -4, -15, 0, 0, \ldots\}$	0
0	$\{\ldots, 0, 0, 2, 1, 6, -14, \underset{\uparrow}{4}, 2, 6, 0, 0, \ldots\}$	7
1	$\{\ldots, 0, 0, 0, -1, -3, 14, \underset{\uparrow}{-2}, 8, -3, 0, 0, \ldots\}$	13
2	$\{\ldots, 0, 0, 0, 0, 3, -7, \underset{\uparrow}{2}, -4, -12, 0, 0, \ldots\}$	-18
3	$\{\ldots, 0, 0, 0, 0, 0, 7, \underset{\uparrow}{-1}, 4, 6, 0, 0, \ldots\}$	16
4	$\{\ldots, 0, 0, 0, 0, 0, 0, \underset{\uparrow}{1}, -2, -6, 0, 0, \ldots\}$	-7
5	$\{\ldots, 0, 0, 0, 0, 0, 0, \underset{\uparrow}{0}, 2, 3, 0, 0, \ldots\}$	5
6	$\{\ldots, 0, 0, 0, 0, 0, 0, \underset{\uparrow}{0}, 0, -3, 0, 0, \ldots\}$	-3

The final result is then

$$\begin{aligned} r_{xy}(l) &= \{\ldots, 0, 0, 10, -9, 19, 36, 14, 33, 0, \underset{\uparrow}{7}, 13, \\ &\qquad -18, 16, -7, 5, -3, 0, 0, \ldots\} \end{aligned}$$

Note that:

- the sequence is not symmetric;
- the largest value is at $l = -4$; and
- the sequences are strongly related at multiple delays.

From the crosscorrelation result, there is an indication of strong correlation at delays $l = -4$ and $l = -2$. With such a small sequence, though, the interpretation of the results is tricky. With longer sequences, as more terms contribute to the overall result, more significance can be given to its interpretation.

1 Finite length sequences

Due to practical computing considerations, when calculating correlations, the sequences must be finite in length, even though they may be drawn from a system with a potentially unlimited number of samples. This results in a modification of the correlation definition: If $x(n)$ and $y(n)$ are of finite length, with N samples between 0 and $N-1$, the crosscorrelation can be written:

$$r_{xy}(l) = \sum_{n=i}^{N-|k|-1} x(n)y(n-l) \tag{6.11}$$

where $i = l$ and $k = 0$ when $l \geq 0$, and $i = 0$ and $k = l$ when $l < 0$.

Note the correct lower limit of the summation - please correct your copy of the textbook on page 122.

Properties

There are a number of important properties relating to correlations. The first is that, for crosscorrelation, the order of the subscripts is important. In particular, $r_{xy}(l) = r_{yx}(-l)$, so care must be taken to observe the order of the subscripts, as well as the sign of l. However, this relationship also implies that $r_{xx}(l) = r_{xx}(-l)$, and is therefore symmetric.

The process of computing a correlation is very similar to the process of computing a convolution, even though the two operations have very different applications. It turns out that there are very efficient methods for computing convolutions, thus it is helpful to be able to use these for computing correlations. The equation for a convolution is:

$$x(n) * y(n) = \sum_{i=-\infty}^{\infty} x(i)y(n-i)$$

The similarity in form between this and (6.3) is apparent. Correlation can then be re-defined as:

$$r_{xy}(l) = \sum_{n=-\infty}^{\infty} x(n)y(n-l)$$

Define the time reverse of $y(l)$

$$\begin{aligned}
y'(l) &= y(-l) \\
\Rightarrow y'(n-l) &= y(l-n) \\
y'(l-n) &= y(n-l) \\
\Rightarrow r_{xy}(l) &= \sum_{n=-\infty}^{\infty} x(n)y'(l-n) \\
&= x(l) * y'(l) \\
\Rightarrow r_{xy}(l) &= x(l) * y(-l) \qquad (6.8)
\end{aligned}$$

Thus, if we have hardware that can efficiently compute a convolution, we can use the same hardware to compute a cross-correlation.

Consider, now, an autocorrelation, $r_{xx}(l)$. From the definition, $r_{xx}(0) = E_x$ is the energy of the sequence, $x(n)$. Using this, it is possible to define two important properties:

$$|r_{xy}(l)| \le \sqrt{E_x E_y} \qquad (6.15)$$

and

$$|r_{xx}(l)| \le E_x \qquad (6.16)$$

In other words, there are limits to the maximum value that a crosscorrelation and an autocorrelation can attain. These bounds can be used to answer the question: "What value of a correlation function constitutes a strong correlation?". It also allows normalised correlations to be defined:

$$\rho_{xy}(l) = \frac{r_{xy}(l)}{\sqrt{r_{xx}(0)r_{yy}(0)}} \qquad (6.18)$$

and

$$\rho_{xx}(l) = \frac{r_{xx}(l)}{r_{xx}(0)} \qquad (6.17)$$

Where ρ is close to the value 1, or -1, then the correlation is said to be strong; where its value is close to 0, the two sequences being correlated are not strongly related at that displacement.

Example 2.6.2

Compute the autocorrelation of

$$x(n) = a^n u(n)\;; 0 < a < 1$$

where $u(n)$ is the unit step function:

$$u(n) = \begin{cases} 1 & n \geq 0 \\ 0 & n < 0 \end{cases}$$

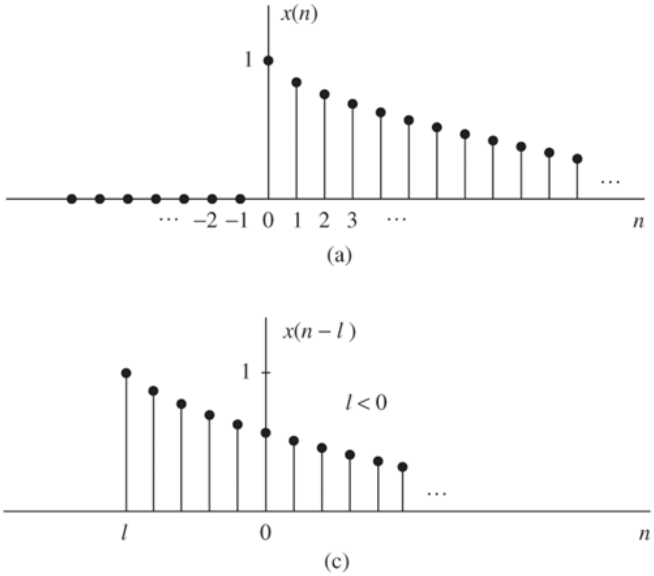


Figure 2.6.2 Computation of the autocorrelation of the signal $x(n) = a^n$, $0 < a < 1$. [1]

If $l < 0$, then

$$\begin{aligned} r_{xx}(l) &= \sum_{n=-\infty}^{\infty} x(n)x(n-l) = \sum_{n=0}^{\infty} a^n a^{n-l} \\ &= a^{-l} \sum_{n=0}^{\infty} (a^2)^n \end{aligned}$$

As $|a| < 1$, the geometric series gives

$$r_{xx}(l) = \frac{1}{1-a^2} a^{-l}\;; l < 0$$

[1] Selected figures taken from “Digital Signal Processing, New International Edition/4th”, Proakis & Manolakis, ©Pearson Education Limited, 2014. ISBN: 978-1-29202-573-5

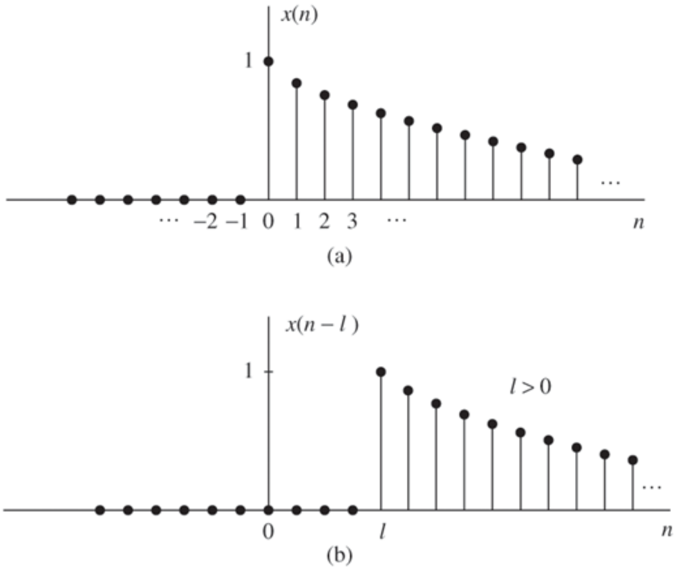


Figure 2.6.2 Computation of the autocorrelation of the signal $x(n) = a^n$, $0 < a < 1$.

If $l \geq 0$, then

$$r_{xx}(l) = \sum_{n=l}^{\infty} a^n a^{n-l} = \sum_{m=0}^{\infty} a^{m+l} a^m = a^l \sum_{m=0}^{\infty} (a^2)^m$$

Note the correction to the summation limits at the top of page 124 of the textbook.

This gives

$$r_{xx}(l) = \frac{1}{1-a^2} a^l \; ; l \geq 0$$

Combining the two terms:

$$r_{xx}(l) = \frac{1}{1-a^2} a^{-l} \; ; l < 0$$
$$r_{xx}(l) = \frac{1}{1-a^2} a^l \; ; l \geq 0$$
$$\Rightarrow r_{xx}(l) = \frac{1}{1-a^2} a^{|l|}$$
$$\Rightarrow \rho_{xx}(l) = a^{|l|}$$

Thus, as expected, the autocorrelation is symmetric.

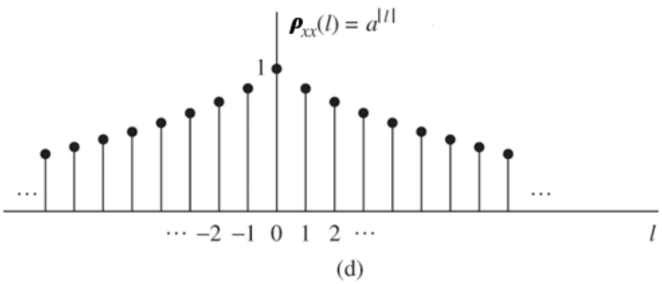


Figure 2.6.2 Computation of the autocorrelation of the signal $x(n) = a^n$, $0 < a < 1$.

Note that this figure plots $\rho_{xx}(l)$ and not $r_{xx}(l)$ as indicated in the textbook. As can be seen, the resulting autocorrelation is symmetric, and indicates that the strongest correlation is at delay 0. As the delay increases, the correlation between the signal and itself decreases.

Periodic Signals

The definition for correlation, presented above, is applicable when analysing energy signals. (An energy signal is defined as one with finite energy. In contrast, a power signal has finite power, and thus infinite energy). When correlation is applied to power signals, the equations need to be modified to give a finite result.

For power signals, correlation is redefined as:

$$r_{xy}(l) = \lim_{M\to\infty} \frac{1}{2M+1} \sum_{n=-M}^{M} x(n)y(n-l) \tag{6.22}$$

For the case of periodic signals, with period N, only one period needs to be considered:

$$r_{xy}(l) = \frac{1}{N} \sum_{n=0}^{N-1} x(n)y(n-l) \tag{6.24}$$

And similarly for autocorrelation:

$$r_{xx}(l) = \lim_{M\to\infty} \frac{1}{2M+1} \sum_{n=-M}^{M} x(n)x(n-l) \tag{6.23}$$

and

$$r_{xx}(l) = \frac{1}{N} \sum_{n=0}^{N-1} x(n)x(n-l) \tag{6.25}$$

Periodic signals

Where the input signal is periodic, the correlation functions are also periodic. Thus, for signals with a period of N,

$$r_{xy}(l) = r_{xy}(l-N)$$

and

$$r_{xx}(l) = r_{xx}(l-N)$$

Identifying periodicities

In practice, the period of an observed signal is often unknown. In such cases, autocorrelation can be used to identify the period from the observed signal provided that the duration of the signal is much larger than the period. If $y(n) = x(n) + w(n)$, where $x(n)$ is a periodic signal, with an unknown period N, and $w(n)$ is a random noise term. If M samples of $y(n)$ are observed, where $M \gg N$, then

$$r_{yy}(l) = \frac{1}{M} \sum_{n=i}^{M-|k|-1} y(n)y(n-l)$$

where $i = l$, $k = 0$ when $l \geq 0$, and $i = 0$, $k = l$ when $l < 0$.

Substituting in for $y(n)$, the following is obtained:

$$\begin{aligned}
r_{yy}(l) &= \frac{1}{M} \sum_{n=i}^{M-|k|-1} [x(n)+w(n)][x(n-l)+w(n-l)] \\
&= \frac{1}{M} \sum_{n=i}^{M-|k|-1} x(n)x(n-l) + \frac{1}{M} \sum_{n=i}^{M-|k|-1} x(n)w(n-l) + \\
&\quad \frac{1}{M} \sum_{n=i}^{M-|k|-1} w(n)x(n-l) + \frac{1}{M} \sum_{n=i}^{M-|k|-1} w(n)w(n-l) \\
&= r_{xx}(l) + r_{xw}(l) + r_{wx}(l) + r_{ww}(l)
\end{aligned}$$

The first term represents the periodic component of $x(n)$.

Due to the finite length of $y(n)$, the periodic repetitions of $r_{yy}(l)$ diminish as l increases.

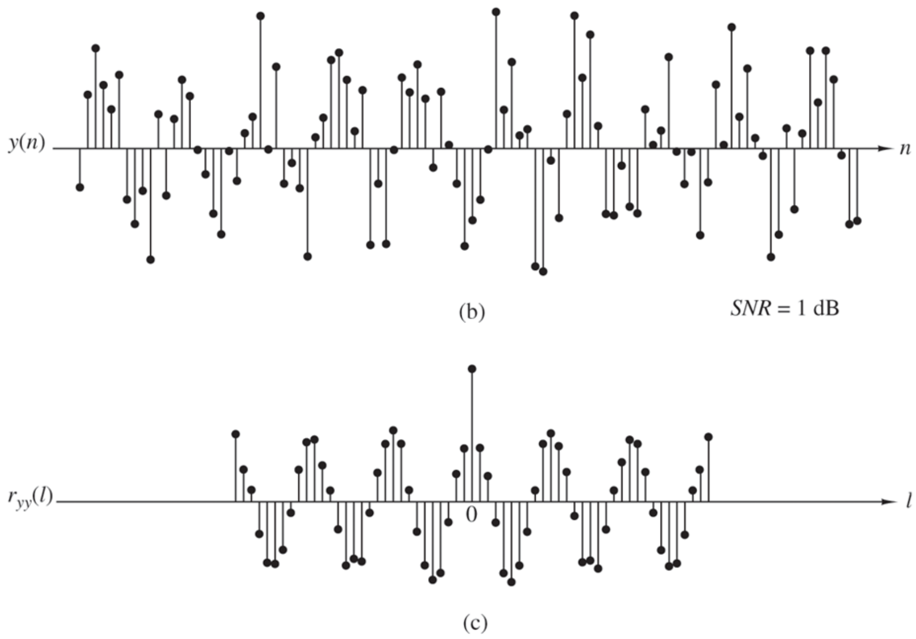


Figure 2.6.4 Use of autocorrelation to detect the presence of a periodic signal corrupted by noise.

The periodic nature of the $x(n)$ component is evident from the regular peaks separated by 10 samples. The much larger sample at $l = 0$ is due to the autocorrelation of the noise term, $r_{ww}(l)$, and indicates that the noise has a finite, non-zero power. Thus, although the periodic nature of the signal is not immediately evident from the time domain representation, it is clear that periodicity exists when the autocorrelation is examined.

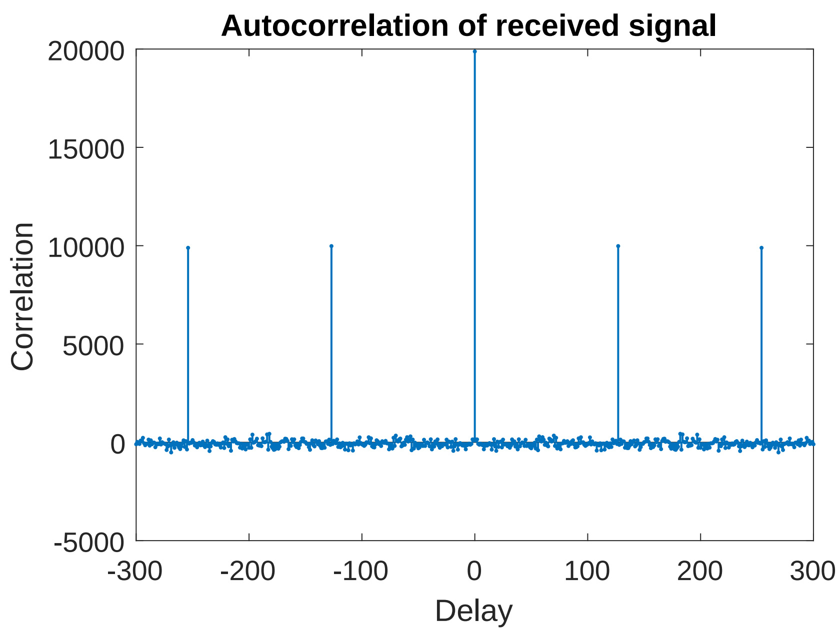
Radar system example

Extending the example of the radar system from previously, assume that the transmitted signal $x(n)$ is repeated multiple times. As before:

$$y(n) = \alpha x(n - D) + w(n) \tag{6.1}$$

Without knowing the transmitted signal, the autocorrelation of the received signal can be computed as

$$r_{yy}(l) = \frac{1}{M} \sum_{n=i}^{M-|k|-1} y(n)y(n-l)$$



The periodic nature of the radar signal can thus be identified by another receiver, without any knowledge of the transmitted waveform. It is possible, once the period is determined, to estimate the transmitted waveform by averaging samples of the received waveform separated by the period. In a practical radar system, such a straightforward method of detecting this property is not desirable, so other techniques are used to avoid this situation.

Calculation of $r_{xy}(l)$

Given its importance to signal analysis, the computational effort required to compute $r_{xy}(l)$, and similarly $r_{yx}(l)$ needs to be considered[1]. If computed directly, this can be time consuming, particularly if the sequences $x(n)$ and $y(n)$ are large, as the sum of products must be computed for each value of l required. As the computation of correlation is closely related to that of convolution, one approach to reduce computation would be to perform the correlation in the frequency domain.

We begin by considering finite length sequences, $x(n)$, of length M, and $y(n)$ of length L.

$$\begin{aligned}
r_{xy}(l) &= \sum_{n=0}^{N-1} x(n)y(n-l) \\
&= \sum_{n=0}^{N-1} \frac{1}{N} \sum_{k=0}^{N-1} X(k)e^{j2\pi kn/N} y(n-l) \\
&= \frac{1}{N} \sum_{k=0}^{N-1} X(k) \sum_{n=0}^{N-1} e^{j2\pi kn/N} y(n-l) \\
&= \frac{1}{N} \sum_{k=0}^{N-1} X(k) \sum_{u=-l}^{N-1-l} y(u)e^{j2\pi ku/N} e^{j2\pi kl/N}
\end{aligned}$$

At this point it is clear that if the length of the sequence $y(n)$ has more than $N-l$ samples for a given value of l, there will be loss of information, which implies an incorrect result. To proceed, we need to define the length of the sequence $y(n)$ as L, and the maximum value of l as $\max(l)$. $\max(l) = M-1$ where M is the length of $x(n)$. Then we impose the criterion that $N \geq L+M-1$. Provided this holds, then:

$$r_{xy}(l) = \frac{1}{N} \sum_{k=0}^{N-1} X(k)Y^*(k)e^{j2\pi kl/N}$$

Thus, the crosscorrelation is the inverse transform of the product of $X(k)$ and $Y^*(k)$, which corresponds to the convolution of $x(n)$ and $y(-n)$ as we derived previously.

We will use the computationally efficient form of the discrete Fourier transform, the FFT, to compute the N-point transforms, $X(k)$ and $Y(k)$, where $N \geq M+L-1$. The best efficiency is achieved when N is a power of 2.

Let $x(n) = \{1,2,3,4,5,6\}$, and $y(n) = \{1,0,1\}$. $N = M+L-1 = 6+3-1 = 8$. Compute the 8-point FFT of $x(n)$ and $y(n)$:

$$\begin{aligned}
X(k) &= \{21, -9.66-3j, 3-4j, 1.66+3j, -3, \ldots\} \\
Y(k) &= \{2, 1-j, 0, 1+j, 2, 1-j, 0, 1+j\}
\end{aligned}$$

Then compute $Z(k) = X(k)Y^*(k)$, and use the inverse FFT to compute the values of $r_{xy}(n)$:

$$\begin{aligned}
Z(k) &= \{42, -6.66-12.66j, 0, 4.66+1.34j, -6, \ldots\} \\
z(n) &= \{4, 6, 8, 10, 5, 6, 1, 2\} \\
r_{xy}(n) &= \{1, 2, \underset{\uparrow}{4}, 6, 8, 10, 5, 6\}
\end{aligned}$$

$z(0)$ to $z(5)$ give the values of $r_{xy}(0)$ to $r_{xy}(5)$ in sequence. The values for $r_{xy}(-1)$ and $r_{xy}(-2)$ are given by $z(7)$ and $z(6)$ respectively.

In this example, the computational cost of computing the correlation directly (18 multiplications if the multiplication with 0 is included) is lower than that using the FFT approach (32 complex multiplies). However, in practice, the sequences we will be working with will be much longer. In this situation, the FFT method is significantly better than direct computation.

The Matlab code for this is:

```
x=[1 2 3 4 5 6];  % Define inputs x(n) and y(n)
y=[1 0 1];
X=fft(x,8);       % Compute 8-point DFT of both
Y=fft(y,8);
Z=X.*conj(Y);     % Calculate Z(k)=X(k)Y*(k)
z=ifft(Z);        % Inverse transform
```

[1] Please note that this material is not covered in the course textbook.

The equivalent code in Python is:

```
import numpy as np

# Define inputs x(n) and y(n)
x=[1 2 3 4 5 6]
y=[1 0 1]

# Compute 8-point DFT of both
X=np.fft.fft(x,8)
Y=np.fft.fft(y,8)

# Calculate Z(k)=X(k)Y*(k)
Z=np.multiply(X,np.conj(Y))

# Inverse transform
z=np.fft.ifft(Z)
```

Where the sequences are short, it can often be simpler to compute the correlation directly, however for longer sequences using this method can be much more computationally efficient.