Correlation

Correlation is a fundamental mathematical tool in signal processing used to measure the similarity between two signals as a function of a time shift applied to one of them. It is widely used in various applications such as signal detection, pattern recognition, and system identification.

Types of Correlation

1. **Autocorrelation**

- Measures the similarity of a signal with a delayed version of itself.
- Useful for identifying repeating patterns or determining signal periodicity.
- **Definition:**

$$R_x(\tau) = \int_{-\infty}^{\infty} x(t)x(t+\tau)\,dt$$

- For discrete signals:

$$R_x[k] = \sum_{n=-\infty}^{\infty} x[n]x[n+k]$$

2. **Cross-Correlation**

- Measures the similarity between two different signals as one is shifted relative to the other.
- Useful in detecting whether one signal is present in another (e.g., template matching).
- **Definition:**

$$R_{xy}(\tau) = \int_{-\infty}^{\infty} x(t)y(t+\tau)\,dt$$

- For discrete signals:

$$R_{xy}[k] = \sum_{n=-\infty}^{\infty} x[n]y[n+k]$$

Properties of Correlation

1. **Symmetry:**
   - $R_x(-\tau) = R_x(\tau)$ for autocorrelation.
   - $R_{xy}(-\tau) = R_{yx}(\tau)$ for cross-correlation.
2. **Maximum Value:**
   - The autocorrelation function achieves its maximum at $\tau = 0$.
   - For normalized signals, $R_x(0) = 1$.
3. **Linearity:**
   - Correlation is linear with respect to the input signals.
4. **Energy Interpretation:**
   - The value of autocorrelation at $\tau = 0$ corresponds to the total energy of the signal for finite-energy signals:

$$R_x(0) = \int_{-\infty}^{\infty} |x(t)|^2 \, dt$$

■ 3.11 INTRODUCTION TO CORRELATION

The correlation is another mathematical operation to measure the degree of similarity of any two signals/images. Correlation is used in RADAR, digital communication, remote sensing engineering, etc.

Let us explain correlation with respect to the following example. The signal sequences $x(n)$ and $y(n)$ are the transmitted and received signals respectively.

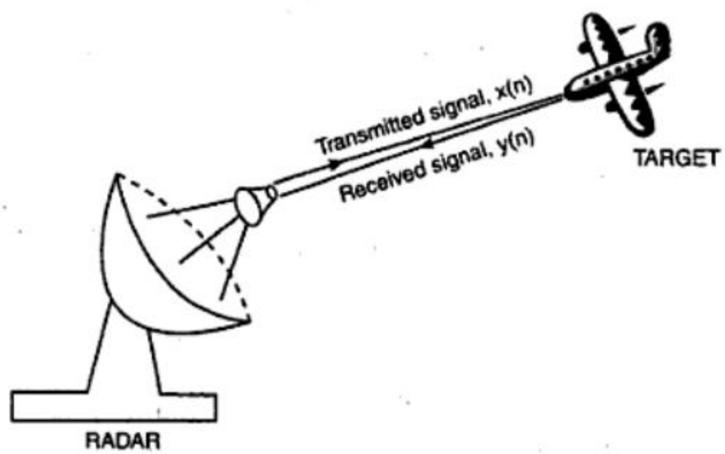

Fig. 3.12 Target Detection Using RADAR

If the target is present in the space during RADAR search, then the received signal $y(n)$ is the delayed input signal $x(n-D)$, i.e.,

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

where α = attenuation of signal $x(n)$

$w(n)$ = additive noise pick up along with the signal at RADAR

D = delay factor

The delay is directly proportional to the distance between the RADAR and the target. In practice, the received signal $x(n-D)$ is heavily corrupted by the additive noise to the point where a visual inspection of $y(n)$ does not reveal the presence or absence of the desired signal. Correlation helps us to extract this important information from $y(n)$.

3.11.1 Cross-correlation

Let us consider two different signal sequences $x(n)$ and $y(n)$ which has finite energy. The cross-correlation of $x(n)$ and $y(n)$ is given by,

$$\gamma_{xy}(m) = \sum_{n=-\infty}^{\infty} x(n)\,y(n-m),\ m = 0, \pm 1, \pm 2, \pm 3, \ldots \tag{3.44}$$

or equivalently

$$\gamma_{xy}(m) = \sum_{n=-\infty}^{\infty} x(n+m)\,y(n),\ m = 0, \pm 1, \pm 2, \pm 3, \ldots \tag{3.45}$$

where, m = lag parameter

The subscript parameter '*xy*' in γ_{xy} indicate the direction in which the sequence is shifted by m. In equation (3.44), the signal $x(n)$ is unshifted while $y(n)$ is shifted by 'm' units to the right (m is positive). In equation (3.45) the signal $y(n)$ is unshifted while $x(n)$ is shifted by 'm' units to the left (m is negative). Both equations (3.44) and (3.45) are identical, that is, both relations yields identical cross-correlation sequence.

Reversing the roll of $x(n)$ and $y(n)$ in equations (3.44) and (3.45) result in equations (3.46) and (3.47) respectively.

$$\gamma_{yx}(m) = \sum_{n=-\infty}^{\infty} y(n)\,x(n-m),\ m = 0, \pm 1, \pm 2, \pm 3, \ldots \tag{3.46}$$

or equivalently

$$\gamma_{yx}(m) = \sum_{n=-\infty}^{\infty} x(n)\,y(n+m),\ m = 0, \pm 1, \pm 2, \pm 3, \ldots \tag{3.47}$$

On comparing equation (3.44) with (3.45) or (3.46) with (3.47), we conclude that

$$\gamma_{xy} = \gamma_{yx}(-m) \tag{3.48}$$

It is clear from equation (3.48) that γ_{yx} is the folded version of γ_{xy}.

If the length of the sequence $x(n)$ is N_1 and the length of the sequence $y(n)$ is N_2 then total length of correlation sequence is $N_1 + N_2 - 1$.

The major computational difference between convolution and correlation is that in case of convolution, one of the sequence is folded, then shifted, then multiplied by the other sequence to form the product sequence for that shift and the product terms are added. In case of correlation, except folding all other process remain the same, that is, one of the sequence is shifted, then multiplied by the other sequence to form the product sequence for that shift and the product terms are added. Mathematically, the correlation and convolution can be related as

$$\gamma_{xy} = x(m) * y(-m) \tag{3.49}$$

SOLVED PROBLEM

Problem 3.27 Determine the cross-correlation sequence of the sequences

$$x(n) = \{1, 2, \underset{\uparrow}{3}, 4, 5\};\ y(n) = \{5, 6, \underset{\uparrow}{7}, 8, 9\}$$

Solution

Let us consider equation (3.44)

$$\gamma_{xy}(m) = \sum_{n=-\infty}^{\infty} x(n)\, y(n-m)$$

For the given problem equation (1) reduces to

$$\gamma_{xy}(m) = \sum_{n=-2}^{2} x(n)\, y(n-m)$$

When $m = 0$

$$\gamma_{xy}(0) = \sum_{n=-2}^{2} x(n)\, y(n)$$

For $m = 0$, the cross-correlation is the product of $x(n)$ and $y(n)$ and sum of all the products, that is,

$$\gamma_{xy}(0) = x(-2)\,y(-2) + x(-1)\,y(-1) + x(0)\,y(0) + x(1)\,y(1) + x(2)\,y(2)$$
$$\gamma_{xy}(0) = 1\times5 + 2\times6 + 3\times7 + 4\times8 + 5\times9 = 115$$

When $m = 1$

$$\gamma_{xy}(1) = \sum_{n=-2}^{2} x(n)\, y(n-1)$$
$$\gamma_{xy}(1) = x(-2)\,y(-3) + x(-1)\,y(-2) + x(0)\,y(-1) + x(1)\,y(0) + x(2)\,y(1)$$
$$\gamma_{xy}(1) = 0 + 2\times5 + 3\times6 + 4\times7 + 5\times8 + 0 = 96$$

When $m = 2$

$$\gamma_{xy}(2) = \sum_{n=-2}^{2} x(n)\, y(n-2)$$
$$\gamma_{xy}(2) = x(-2)\,y(-4) + x(-1)\,y(-3) + x(0)\,y(-2) + x(1)\,y(-1) + x(2)\,y(0)$$
$$\gamma_{xy}(2) = 0 + 0 + 3\times5 + 4\times6 + 5\times7 + 0 + 0 = 74$$

When $m = 3$

$$\gamma_{xy}(3) = \sum_{n=-2}^{2} x(n)\, y(n-3)$$

$$\gamma_{xy}(3) = x(-2)\, y(-5) + x(-1)\, y(-4) + x(0)\, y(-3) + x(1)\, y(-2) + x(2)\, y(-1)$$

$$\gamma_{xy}(3) = 0+0+0+4\times5+5\times6+0+0 = 50$$

When $m = 4$

$$\gamma_{xy}(4) = \sum_{n=-2}^{2} x(n)\, y(n-4)$$

$$\gamma_{xy}(4) = x(-2)\, y(-6) + x(-1)\, y(-5) + x(0)\, y(-4) + x(1)\, y(-3) + x(2)\, y(-2)$$

$$\gamma_{xy}(4) = 0+0+0+0+5\times5+0 = 25$$

When $m = 5$

$$\gamma_{xy}(5) = \sum_{n=-2}^{2} x(n)\, y(n-5)$$

$$\gamma_{xy}(5) = x(-2)\, y(-7) + x(-1)\, y(-6) + x(0)\, y(-5) + x(1)\, y(-4) + x(2)\, y(-3)$$

$$\gamma_{xy}(5) = 0$$

$$\gamma_{xy}(m \geq 5) = 0$$

When $m = -1$

$$\gamma_{xy}(-1) = \sum_{n=-2}^{2} x(n)\, y(n+1)$$

$$\gamma_{xy}(-1) = x(-2)\, y(-1) + x(-1)\, y(0) + x(0)\, y(1) + x(1)\, y(2) + x(2)\, y(3)$$

$$\gamma_{xy}(-1) = 1\times6+2\times7+3\times8+4\times9+0 = 80$$

When $m = -2$

$$\gamma_{xy}(-2) = \sum_{n=-2}^{2} x(n)\, y(n+2)$$

$$\gamma_{xy}(-2) = x(-2)\, y(0) + x(-1)\, y(1) + x(0)\, y(2) + x(1)\, y(3) + x(2)\, y(4)$$

$$\gamma_{xy}(-2) = 1\times7+2\times8+3\times9+0+0 = 50$$

When $m = -3$

$$\gamma_{xy}(-3) = \sum_{n=-2}^{2} x(n)\, y(n+3)$$

$$\gamma_{xy}(-3) = x(-2)\, y(1) + x(-1)\, y(2) + x(0)\, y(3) + x(1)\, y(4) + x(2)\, y(5)$$

$$\gamma_{xy}(-3) = 1\times8+2\times9+0+0+0 = 26$$

When $m = -4$

$$\gamma_{xy}(-4) = \sum_{n=-2}^{2} x(n)\, y(n+4)$$

$$\gamma_{xy}(-4) = x(-2)\, y(2) + x(-1)\, y(3) + x(0)\, y(4) + x(1)\, y(5) + x(2)\, y(6)$$

$$\gamma_{xy}(-4) = 1\times9+0+0+0+0 = 9$$

When $m=-5$

$$\gamma_{xy}(-5)=\sum_{n=-2}^{2}x(n)\,y(n+5)$$
$$\gamma_{xy}(-5)=x(-2)\,y(3)+x(-1)\,y(4)+x(0)\,y(5)+x(1)\,y(6)+x(2)\,y(7)$$
$$\gamma_{xy}(-5)=0$$
$$\gamma_{xy}(m\le -5)=0$$

Therefore, the cross-correlation sequence of $x(n)$ and $y(n)$ is

$$\gamma_{xy}(m)=\{9,26,50,80,115,96,74,50,25\}$$

3.11.2 Autocorrelation

If $y(n)=x(n)$ then equation (3.43) and (3.46) reduces to

$$\gamma_{xx}(l)=\sum_{n=-\infty}^{\infty}x(n)\,x(n-l) \tag{3.50}$$

or equivalently

$$\gamma_{xx}(l)=\sum_{n=-\infty}^{\infty}x(n+l)\,x(n) \tag{3.51}$$

Equations (3.50) and (3.51) are the autocorrelation equations of the sequence.

3.11.3 Properties of Cross-correlation and Autocorrelation

Let us consider two data sequences $x(n)$ and $y(n)$ whose linear combination is given by

$$Z(n)=A\,x(n)+B\,y(n-l)$$

Where, A and B are scalar

l is the shift

The energy of the sequence $Z(n)$ is given by

$$E=\sum_{n=-\infty}^{\infty}Z^2(n)$$
$$E=\sum_{n=-\infty}^{\infty}[A\,x(n)+B\,y(n-l)]^2$$

On simplification,

$$E=A^2\sum_{n=-\infty}^{\infty}x^2(n)+B^2\sum_{n=-\infty}^{\infty}y^2(n-l)+2AB\sum_{n=-\infty}^{\infty}x(n)\,y(n-l)$$
$$E=A^2\gamma_{xx}(0)+B^2\gamma_{yy}(0)+2AB\gamma_{xy}(l)$$

If the energy of signals $x(n)$ and $y(n)$ are finite, then the energy of $Z(n)$ must also be finite, that is,

$$E=A^2\gamma_{xx}(0)+B^2\gamma_{yy}(0)+2AB\gamma_{xy}(l)\ge 0$$

On simplification, by inequality

$$\left|\gamma_{xy}(l)\right| \le \sqrt{\gamma_{xx}(0)+\gamma_{yy}(0)} \quad \text{(Proof is left to the reader)}$$

If $x(n) = y(n)$, then

$$\left|\gamma_{xy}(l)\right| \le \sqrt{\gamma_{xx}(0)} = E_x$$

The autocorrelation of sequence attains maximum energy condition at zero lag (that is, $l = 0$).

Problem 3.28 Find the cross-correlation of two finite length sequences $x(n) = \{1, 2, 3, 4\}$ and $y(n) = \{5, 6, 7, 8\}$.

Solution

$x(l)=\{1,2,3,4\}$; $y(-l)=\{8,7,6,5\}$

By definition, $\gamma_{xy}(l)=x(l)*y(-l)$

x(l) \ y(−l)	8	7	6	5
1	8	7	6	5
2	16	14	12	10
3	24	21	18	15
4	32	28	24	20

$$\gamma_{xy}(l)=\{8, 16+7,\ 24+14+6,\ 32+21+12+5, 28+18+10,\ 24+15,\ 20\}$$
$$\gamma_{xy}(l)=\{8, 23, 44, 70, 56, 39, 20\}$$

Applications of Correlation in Signal Processing

1. **Signal Detection and Synchronization**
   - Used in radar, sonar, and communication systems to detect signals and align timing.
2. **Feature Extraction and Pattern Matching**
   - Helps to identify patterns or templates within larger datasets.
3. **System Identification**
   - Assists in modeling unknown systems by analyzing input and output signals.
4. **Spectral Analysis**
   - The Fourier transform of the autocorrelation function gives the power spectral density (Wiener-Khinchin theorem).
5. **Image Processing**
   - Cross-correlation is used for template matching to locate objects within images.

Examples:

- Example 1 demonstrates autocorrelation of a sinusoidal signal.
- Example 2 calculates the cross-correlation between two shifted sinusoidal signals.
- Example 3 adds noise to a signal and computes cross-correlation with the original.

```
import numpy as np
import matplotlib.pyplot as plt
from scipy.signal import correlate, correlation_lags

# Define a function for autocorrelation
def compute_autocorrelation(signal):
    # Using numpy's correlate function
    auto_corr = correlate(signal, signal, mode='full', method='auto')
    lags = correlation_lags(len(signal), len(signal), mode='full')
    return auto_corr, lags

# Define a function for cross-correlation
def compute_cross_correlation(signal1, signal2):
    # Using numpy's correlate function
    cross_corr = correlate(signal1, signal2, mode='full', method='auto')
    lags = correlation_lags(len(signal1), len(signal2), mode='full')
    return cross_corr, lags

# Example 1: Autocorrelation of a sinusoidal signal
fs = 1000  # Sampling frequency in Hz
t = np.linspace(0, 1, fs, endpoint=False)  # Time vector
freq = 5  # Frequency of the sine wave
sin_signal = np.sin(2 * np.pi * freq * t)

# Compute and plot autocorrelation
auto_corr, lags = compute_autocorrelation(sin_signal)
plt.figure(figsize=(12, 6))
plt.plot(lags, auto_corr)
plt.title("Autocorrelation of a Sinusoidal Signal")
plt.xlabel("Lag")
plt.ylabel("Autocorrelation")
plt.grid()
plt.show()

# Example 2: Cross-correlation between two signals
signal1 = np.sin(2 * np.pi * freq * t)  # Original sinusoidal signal
signal2 = np.roll(signal1, 100)  # Shifted version of the sinusoidal signal

# Compute and plot cross-correlation
cross_corr, lags = compute_cross_correlation(signal1, signal2)
plt.figure(figsize=(12, 6))
```

```
plt.plot(lags, cross_corr)
plt.title("Cross-Correlation between Two Signals")
plt.xlabel("Lag")
plt.ylabel("Cross-Correlation")
plt.grid()
plt.show()

# Example 3: Cross-correlation with noise
noise = np.random.normal(0, 0.5, fs)  # Additive Gaussian noise
noisy_signal = signal1 + noise

# Compute and plot cross-correlation
cross_corr_noise, lags = compute_cross_correlation(signal1, noisy_signal)
plt.figure(figsize=(12, 6))
plt.plot(lags, cross_corr_noise)
plt.title("Cross-Correlation with Noisy Signal")
plt.xlabel("Lag")
plt.ylabel("Cross-Correlation")
plt.grid()
plt.show()
```