

**EE 451**

**Solution of Difference Equations in the Time Domain**

The difference equation

\[ \sum_{k=0}^{N} d_k y[n - k] = \sum_{k=0}^{M} p_k x[n - k] \]

has the solution

\[ y[n] = \left( \alpha_1 \lambda_1^n + \cdots + \alpha_M \lambda_M^n \right) \mu[n] + \gamma_0 \delta[n] + \cdots + \gamma_{M-N-1} \delta[n - (M - N - 1)] + y_p[n] \]

where the \( \lambda_k \)'s are the roots of the characteristic polynomial of the system:

\[ d_0 \lambda^N + d_1 \lambda^{N-1} + \cdots + d_{N-1} \lambda + d_N = 0. \]

(For repeated roots, use \( \lambda_k^n, n \lambda_k^n, n^2 \lambda_k^n \), etc. For example, if \( \lambda_1 = \lambda_2 \), you would use \( \alpha_1 \lambda_1^n + \alpha_2 n \lambda_1^n \) instead of \( \alpha_1 \lambda_1^n + \alpha_2 \lambda_1^n \).

\( y_p[n] \) is the particular (steady-state) solution which depends on the input:

\[
\begin{array}{c|c}
\hline
x[n] & y_p[n] \\
\hline
\alpha_1 \lambda_1^n & \beta_1 \cos[\omega_0 n] + \beta_2 \sin[\omega_0 n] \\
\alpha_2 \lambda_2^n & \\
\alpha_3 \lambda_3^n & \\
\alpha_4 \lambda_4^n & \\
\vdots & \\
\alpha_M \lambda_M^n & \\
A \cos[\omega_0 n] + B \sin[\omega_0 n] & \\
A a^n \mu[n] & \\
A a^2 \mu[n] & \\
A a^3 \mu[n] & \\
A a^4 \mu[n] & \\
\vdots & \\
A a^M \mu[n] & \\
\end{array}
\]

If the characteristic polynomial has a root at the value of an input exponential (e.g., \( \lambda_1 = \frac{1}{2} \) and \( x[n] = \left( \frac{1}{2} \right)^n \)) you would use \( \beta n \lambda_k^n \) for \( y_p[n] \) (e.g., \( y_p[n] = \beta n \left( \frac{1}{2} \right)^n \)).

There are \( M - N \) \( \gamma_i \)'s. If \( M \leq N \) there are no \( \gamma_i \)'s.

To find the impulse response, there is no \( y_p[n] \), and there are \( M - N + 1 \) \( \gamma_i \)'s. If \( M < N \) there are no \( \gamma_i \)'s for the impulse response.

Solve for the unknowns by finding \( y[0], y[1], y[2], \cdots \) until you get as many equations as you have unknowns. Solve these equations for the unknowns.

You can solve the difference equation for the case \( x[n] = 0 \), subject to the initial conditions of the system. This is the natural response or zero-input response, \( y_z[n] \), of the system. You can further solve for the case \( y[n] = 0 \) for \( n < 0 \) for the actual input \( x[n] \). This is the forced response or zero-state response, \( y_SS[n] \), of the system. The total response is the sum of the two: \( y[n] = y_z[n] + y_SS[n] \).

A system is **BIBO stable** if, for every bounded input, the output is bounded. If the input \( x[n] \) is bounded, then the particular solution \( y_p[n] \) will be bounded. Thus, the only possible unbounded terms in Eq. (1) are the \( \lambda_k^n \)'s. These terms are bounded (\( \lambda_k^n \to 0 \) as \( n \to \infty \)) if \( |\lambda_k| < 1 \). Hence, the system is BIBO stable if \( |\lambda_k| < 1 \) for all the \( k \)'s. (\( \lambda_k^n \) doesn’t blow up if \( |\lambda_k| = 1 \). However, if \( |\lambda_k| = 1 \), the input \( x[n] = \lambda_k^n \) will produce the output \( y[n] = \alpha_1 \lambda_k^n + \beta n \lambda_k^n \), and the \( \beta n \lambda_k^n \) term will blow up as \( n \to \infty \).)