ECE503: Digital Signal Processing Lecture 2

D. Richard Brown III

WPI

23-January-2012

Lecture 2 Topics

1. Examples of discrete-time systems.
2. Qualitative properties of discrete-time systems.
3. Time-domain mathematical descriptions of systems:
   - Input-output difference equation
   - Transfer function
   - Impulse response
4. Solving for the output of a discrete-time system given an arbitrary input and initial conditions
5. Frequency response of a discrete-time system
6. Phase and group delay
7. Simple filtering

Examples of Discrete-Time Systems

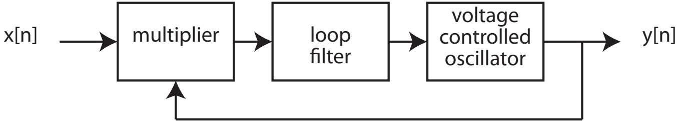
Lots of good examples in Chapter 4 of your textbook.

Some other examples:

- The systems you analyzed in Mitra problem 2.4, all of which were described by an input-output difference equation

$$y[n] = \sum_{k=0}^{M-1} b_k x[n-k] - \sum_{k=1}^{N-1} a_k y[n-k]$$

- Phase locked loop



- Peak tracker: $y[n] = \max(\{x[n-L+1], \dots, x[n]\})$.

Qualitative Properties of Systems

1. **SISO**, MISO, SIMO, or MIMO
2. **Discrete-time** or continuous-time
3. **Linear** or nonlinear
4. Shift-invariant (aka **time-invariant**) or shift-variant (aka time-varying)
5. Memoryless or **dynamic**
6. Causal, non-causal, or anti-causal
7. Stable or unstable
8. Passive, lossless, or active

These properties are covered pretty well in your textbook.
Our focus is going to be primarily on **SISO discrete-time linear time-invariant** (LTI) **dynamic** systems for two reasons:

- Lots of real-world systems are LTI (or approximately LTI if operated in a linear region).
- There are an abundance of analysis techniques for LTI systems.

Common Mathematical Descriptions of Systems

- Input-output differential/difference equation
- Impulse/step/ramp response
- Frequency response (Fourier series, Fourier transform, DFT, DTFT, ...)
- Transfer function (Laplace/z)
- State-space (ECE504)

These descriptions are related but not equivalent, in general.

Input-Output Description: Capabilities and Limitations

Example (causal discrete-time system):

$$y[k] = f(y[k-1], y[k-2], \ldots, x[k], x[k-1], \ldots)$$

+ Can describe memoryless or dynamic systems.
+ Can describe causal or non-causal systems.
+ Can describe linear or non-linear systems.
+ Can describe time-invariant or time-varying systems.
+ Can describe relaxed or non-relaxed systems (non-zero initial conditions).

– No explicit access to internal behavior of systems.
– Difficult to analyze directly.

Transfer Function Description: Capabilities and Limitations

Example:

$$H(z) = \frac{N(z)}{D(z)} = \frac{b_m z^m + b_{m-1} z^{m-1} + \cdots + b_1 z + b_0}{z^n + a_{n-1} z^{n-1} + \cdots + a_1 z + a_0}$$

+ Can describe memoryless or dynamic systems.
+ Can describe causal and non-causal systems (ROC).
– Can't describe non-linear systems. Only linear systems.
– Can't describe time-varying systems. Only time-invariant systems.
– No explicit access to internal behavior of system.
– Can't describe systems with non-zero initial conditions. Implicitly assumes that system is relaxed.
+ Abundance of analysis techniques. Systems are usually analyzed with **basic algebra**, not calculus.

Impulse Response Description: Capabilities and Limitations

Definition

The **impulse response** of a system is the output of the system given an input $x[n] = \delta[n]$ assuming relaxed initial conditions. We denote the impulse response of the system $\mathcal{H}$ as $h[n] : \mathbb{Z} \mapsto \mathbb{R}$.

Example: If $y[n] = x[n] + 0.5x[n-1]$ then $h[n] = \delta[n] + 0.5\delta[n-1]$.

+ Can describe memoryless and dynamic systems.
+ Can describe causal and non-causal systems.
– Nonlinear systems have an impulse response, but it isn't useful.
+ Can describe time-invariant and time-varying systems.
– No explicit access to internal behavior of system.
– Can't describe systems with non-zero initial conditions. Implicitly assumes that system is relaxed.

Impulse Response Description: Useful for Linear Systems

The primarily utility of the impulse response is that we can compute the output of a discrete-time **linear** system with arbitrary input sequence $\{x[n]\}$ by convolving $\{x[n]\}$ with the impulse response $h[n]$.

This doesn't work for nonlinear systems.

Example: Suppose system $\mathcal{H}_1$ has an input/output description

$$y[n] = x[n]$$

and system $\mathcal{H}_2$ has an input/output description

$$y[n] = x^2[n].$$

- What is $y[n]$ given $x[n] = -\mu[n]$?
- What are the impulse responses $h_1[n]$ and $h_2[n]$?

Impulse Response of an LTI System

If a system $\mathcal{H}$ is LTI, its relaxed behavior is **fully characterized** by its impulse response $h[n]$.

Time-invariance implies that if we apply a delayed impulse $\delta[n-k]$ to the input of the system $\mathcal{H}$, we will get a delayed output $h[n-k]$.

Given an arbitrary input sequence $\{x[n]\}$, note that this sequence can be written as

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

Linearity and time-invariance implies

$$y[n] = \mathcal{H}\left\{\sum_{k=-\infty}^{\infty} x[k]\delta[n-k]\right\} = \sum_{k=-\infty}^{\infty} x[k]h[n-k] = \sum_{k=-\infty}^{\infty} x[n-k]h[k]$$

where the final equality is obtained by change of variable. This is the standard discrete-time convolution sum (Matlab `conv` function).

Impulse Response of an LTV System

If a system $\mathcal{H}$ is LTV, it may have a different impulse response if the impulse is applied to the input at different times. Example:

$$y[n] = nx[n]$$

- Applying an input $x[n] = \delta[n]$ results in what output?
- Applying an input $x[n] = \delta[n-1]$ results in what output?

If we denote $h[n,k]$ as the response of the system $\mathcal{H}$ at time n to an impulse at time k, we can derive the convolution sum for an LTV system as

$$y[n] = \mathcal{H}\left\{\sum_{k=-\infty}^{\infty} x[k]\delta[n-k]\right\} = \sum_{k=-\infty}^{\infty} x[k]h[n,k]$$

I don't know of any Matlab function that will compute this directly. Note $h[n,k] = h[n-k]$ if the system is LTI.

Convolution Matrix of an LTI System (1 of 2)

Suppose you want to convolve two finite-length sequences: $\{a[0],\dots,a[M-1]\}$ and $\{b[0],\dots,b[N-1]\}$. The result $\{c[n]\} = \{a[n]\} \circledast \{b[n]\}$ will have $M+N-1$ elements and can be computed as

$$\begin{aligned}
c[0] &= a[0]b[0] \\
c[1] &= a[1]b[0] + a[0]b[1] \\
c[2] &= a[2]b[0] + a[1]b[1] + a[0]b[2] \\
\vdots &= \vdots \\
c[M+N-3] &= a[M-1]b[N-2] + a[M-2]b[N-1] \\
c[M+N-2] &= a[M-1]b[N-1]
\end{aligned}$$

If you know a little linear algebra, you can write this convolution as the product of a **convolution matrix** and a vector.

Convolution Matrix of an LTI System (2 of 2)

To illustrate the idea, suppose we have $\{a[0], a[1]\}$ and $\{b[0], b[1], b[2]\}$.

$$\begin{array}{rcllll} c[0] & = & a[0]b[0] & & & \\ c[1] & = & a[1]b[0] & +a[0]b[1] & & \\ c[2] & = & & a[1]b[1] & +a[0]b[2] & \\ c[3] & = & & & a[1]b[2] & \end{array}$$

This is the same as

$$\begin{bmatrix} c[0] \\ c[1] \\ c[2] \\ c[3] \end{bmatrix} = \underbrace{\begin{bmatrix} a[0] & 0 & 0 \\ a[1] & a[0] & 0 \\ 0 & a[1] & a[0] \\ 0 & 0 & a[1] \end{bmatrix}}_{\text{convolution matrix}} \begin{bmatrix} b[0] \\ b[1] \\ b[2] \end{bmatrix} = \underbrace{\begin{bmatrix} b[0] & 0 \\ b[1] & b[0] \\ b[2] & b[1] \\ 0 & b[2] \end{bmatrix}}_{\text{convolution matrix}} \begin{bmatrix} a[0] \\ a[1] \end{bmatrix}$$

The convolution matrix has a Toeplitz structure and can be generated in Matlab with the `convmtx` command.

Solving LTI Systems Described by Difference Equations

Most LTI systems can be described by finite-dimensional constant-coefficient difference equations:

$$y[n] = \sum_{k=0}^{M-1} b_k x[n-k] - \sum_{k=1}^{N-1} a_k y[n-k]$$

Your textbook describes two methods to calculate $\{y[n]\}$ given $\{x[n]\}$ and the initial conditions $y[-1], \dots, y[-N+1]$:

1. Complementary + particular solution
2. Zero-input response + zero-state response

Both give the same answer and use similar methods (root finding, solving simultaneous equations, ...).

I personally prefer the zero-input response + zero-state response method because it explicitly separates the effects of the initial conditions and the input. The zero-state response describes the behavior of the system when it is relaxed, which is useful for computing the impulse/step responses.

Solving LTI Systems Described by Difference Equations

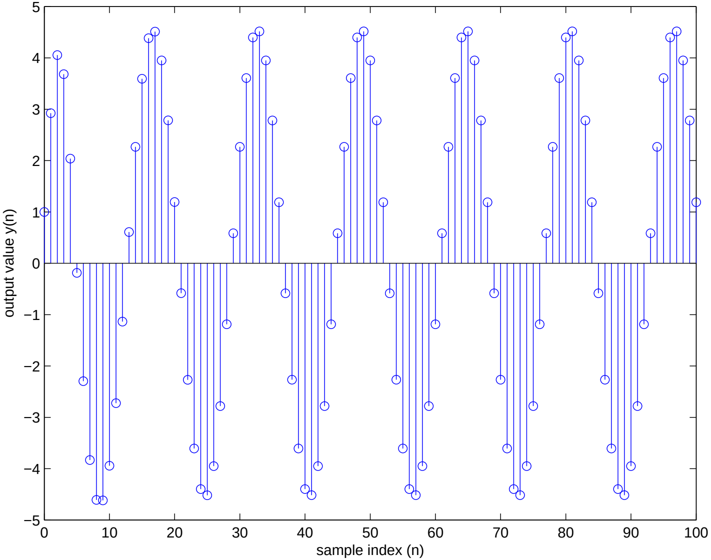
Matlab can also numerically solve LTI systems described by finite-dimensional constant-coefficient difference equations

$$y[n] = \sum_{k=0}^{M-1} b_k x[n-k] - \sum_{k=1}^{N-1} a_k y[n-k]$$

Example

```
a = [1 -1 0.5];          % vector containing a0, a1, a2
b = [1 1];               % vector containing b0, b1
n = 0:100;               % sample indices
x = cos(pi/8*n);         % input function
zi = [0 0];              % initial conditions (relaxed here)
y = filter(b,a,x,zi);    % compute output
stem(0:length(y)-1,y);   % plot
xlabel('sample index (n)');
ylabel('output value y(n)');
```

Also check out Matlab functions `impulse` and `step`.


5
4
3
2
1
0
−1
−2
−3
−4
−5
output value y(n)
0
10
20
30
40
50
60
70
80
90
100
sample index (n)

Impulse Response to Frequency Response (1 of 2)

Suppose we apply an input sequence $x[n] = e^{j\omega_0 n}$ for all $n \in \mathbb{Z}$ to a discrete-time LTI system $\mathcal{H}$ with impulse response $h[n]$. We can compute the output via the usual convolution

$$\begin{aligned} y[n] &= \sum_{k=-\infty}^{\infty} h[k]x[n-k] \\ &= \sum_{k=-\infty}^{\infty} h[k]e^{j\omega_0(n-k)} \\ &= \left(\sum_{k=-\infty}^{\infty} h[k]e^{-j\omega_0 k}\right) e^{j\omega_0 n} \\ &= H(\omega_0)e^{j\omega_0 n} \end{aligned}$$

where we have assumed the sum converges (it is sufficient for the impulse response to be absolutely summable). The final equality is from the definition of the DTFT.

Impulse Response to Frequency Response (2 of 2)

So, given an input $x[n] = e^{j\omega_0 n}$ for all $n \in \mathbb{Z}$, we get an output

$$y[n] = H(\omega_0)e^{j\omega_0 n} = |H(\omega_0)|e^{j(\omega_0 n + \angle H(\omega_0))}.$$

Remarks:

1. $H(\omega_0)$ is just a complex number. It has a magnitude and a phase.
2. The output is a complex exponential at the same frequency as the input.
3. The only things the system has changed is the phase and amplitude of the complex exponential.
4. We say that exponential sequences $e^{j\omega_0 n}$ are **eigenfunctions** of LTI systems.

Given a discrete-time LTI system with impulse response $h[n]$, we say the **frequency response** of this system is

$$H(\omega) = \sum_{k=-\infty}^{\infty} h[k]e^{-j\omega k} = \text{DTFT}(\{h[n]\})$$

You can get the impulse response from the frequency response via the IDTFT.

Convolution Theorem

Textbook pp. 108-109 proves

$$\mathsf{DTFT}(h[n] \circledast x[n]) = H(\omega)X(\omega)$$

if the DTFTs both exist. This then implies that, for an LTI system $\mathcal{H}$ with frequency response $H(\omega)$,

$$Y(\omega) = H(\omega)X(\omega).$$

It can sometimes be easier to compute the output of a system by converting everything to frequency domain first, computing this product, and then doing an IDTFT to get $\{y[n]\}$.

This result also implies

$$H(\omega) = \frac{Y(\omega)}{X(\omega)}$$

which will be useful for deriving an expression for $H(\omega)$ when the system is specified by a constant-coefficient difference equation.

Difference Equation to Frequency Response

For LTI systems described by finite-dimensional constant-coefficient difference equations

$$y[n] = \sum_{k=0}^{M-1} b_k x[n-k] - \sum_{k=1}^{N-1} a_k y[n-k]$$

we can set $a_0 = 1$, rearrange the terms, and take the DTFT of both sides to write

$$\begin{aligned} \sum_{k=0}^{N-1} a_k y[n-k] &= \sum_{k=0}^{M-1} b_k x[n-k] \\ \sum_{k=0}^{N-1} a_k Y(\omega) e^{-j\omega k} &= \sum_{k=0}^{M-1} b_k X(\omega) e^{-j\omega k} \\ \frac{Y(\omega)}{X(\omega)} &= \frac{\sum_{k=0}^{M-1} b_k e^{-j\omega k}}{\sum_{k=0}^{N-1} a_k e^{-j\omega k}} = H(\omega) \end{aligned}$$

The `freqz` function in Matlab is handy for computing $H(\omega)$ at various values of $\omega \in [-\pi, \pi)$. You just pass in vectors `b`, `a`, and `w`.

Response of LTI Systems to Sinusoidal Inputs (1 of 2)

Suppose we have an input sequence $x[n] = A\cos(\omega_0 n + \phi)$ for all $n \in \mathbb{Z}$. We can use Euler's identity to write

$$\begin{aligned} A\cos(\omega_0 n + \phi) &= \frac{A}{2}\left(e^{j(\omega_0 n+\phi)} + e^{-j(\omega_0 n+\phi)}\right) \\ &= \frac{A}{2}e^{j\phi}e^{j\omega_0 n} + \frac{A}{2}e^{-j\phi}e^{-j\omega_0 n} \end{aligned}$$

Passing this signal through an LTI system with impulse response $h[n]$ results in

$$\begin{aligned} y[n] &= \frac{A}{2}e^{j\phi}H(\omega_0)e^{j\omega_0 n} + \frac{A}{2}e^{-j\phi}H(-\omega_0)e^{-j\omega_0 n} \\ &= \frac{A}{2}e^{j\phi+\angle H(\omega_0)}|H(\omega_0)|e^{j\omega_0 n} + \frac{A}{2}e^{-j\phi+\angle H(-\omega_0)}|H(-\omega_0)|e^{-j\omega_0 n} \end{aligned}$$

We can simplify this a bit with an additional assumption...

Response of LTI Systems to Sinusoidal Inputs (2 of 2)

Let's assume the impulse response $h[n]$ is real-valued. This implies $|H(-\omega_0)| = |H(-\omega_0)|$ and $\angle H(-\omega_0) = -\angle H(\omega_0)$.

Then

$$\begin{aligned} y[n] &= \frac{A}{2} e^{j\phi + \angle H(\omega_0)} |H(\omega_0)| e^{j\omega_0 n} + \frac{A}{2} e^{-j\phi + \angle H(-\omega_0)} |H(-\omega_0)| e^{-j\omega_0 n} \\ &= |H(\omega_0)| \frac{A}{2} \left(e^{j\phi + \angle H(\omega_0)} e^{j\omega_0 n} + e^{-j\phi - \angle H(\omega_0)} e^{-j\omega_0 n} \right) \\ &= |H(\omega_0)| A \cos\left(\omega_0 n + \phi + \angle H(\omega_0)\right) \end{aligned}$$

Hence, given an input sequence $x[n] = A\cos(\omega_0 n + \phi)$ for all $n \in \mathbb{Z}$, the output sequence is the same sinusoidal sequence with two differences:

- Amplitude scaled by $|H(\omega_0)|$
- Phase shifted by $\angle H(\omega_0)$

The frequency of the output is identical to the frequency of the input. No new frequencies are generated.

Phase Delay

We know, given an LTI system $\mathcal{H}$ and an input sequence $x[n] = A\cos(\omega_0 n + \phi)$ for all $n \in \mathbb{Z}$, the output sequence

$$y[n] = |H(\omega_0)| A \cos\left(\omega_0 n + \phi + \angle H(\omega_0)\right)$$

Denote $\theta(\omega_0) = \angle H(\omega_0)$. Then

$$y[n] = |H(\omega_0)| A \cos\left(\omega_0 (n + \theta(\omega_0)/\omega_0) + \phi\right)$$
$$y[n] = |H(\omega_0)| A \cos\left(\omega_0 (n - \tau_p(\omega_0)) + \phi\right)$$

where $\tau_p := -\theta(\omega_0)/\omega_0$ is called the **phase delay** of the LTI system $\mathcal{H}$ at frequency ω_0.

What are the units of $\tau_p(\omega_0)$?

What does it physically mean if $\tau_p(\omega_0) = 7$?

See Matlab function `phasedelay`.

Linear Phase Systems

Definition

A **linear phase system** $\mathcal{H}$ is a system with phase response $\theta(\omega) = \angle H(\omega) = -c\omega$ for all ω and any constant c.

For example, suppose we have an LTI system $\mathcal{H}$ with impulse response

$$h[n] = \{1, 2, 1\}.$$

We can compute the frequency response

$$H(\omega) = \sum_{n=-\infty}^{\infty} h[n]e^{-j\omega n} = 1 + 2e^{-j\omega} + 1e^{-j2\omega} = (2\cos(\omega) + 2)e^{-j\omega}$$

We see that $\theta(\omega) = \angle H(\omega) = -\omega$. Is this a linear phase system?

Note the phase delay of a linear phase system is $\tau_p(\omega) = -\theta(\omega)/\omega = c$. In other words, all frequencies are delayed by the same amount of time.

Effect of Nonlinear Phase on Narrowband Signals (1 of 3)

Suppose we have an LTI system $\mathcal{H}$ and a **narrowband** input sequence $x[n] = A[n]\cos(\omega_0 n + \phi)$. The narrowband assumption means that $X(\omega)$ is nonzero only around $\omega = \pm\omega_0$.

To analyze how an LTI system $\mathcal{H}$ affects this narrowband signal, we take a Taylor series approximation of the phase response of $\mathcal{H}$ for values of ω close to $\pm\omega_0$. For values of ω close to ω_0, we have

$$\angle H(\omega) \approx \theta(\omega_0) + (\omega - \omega_0)\left[\frac{d\theta(\omega)}{d\omega}\right]_{\omega=\omega_0} = \theta(\omega_0) - (\omega - \omega_0)\tau_g(\omega_0).$$

Similarly, for values of ω close to $-\omega_0$, we have

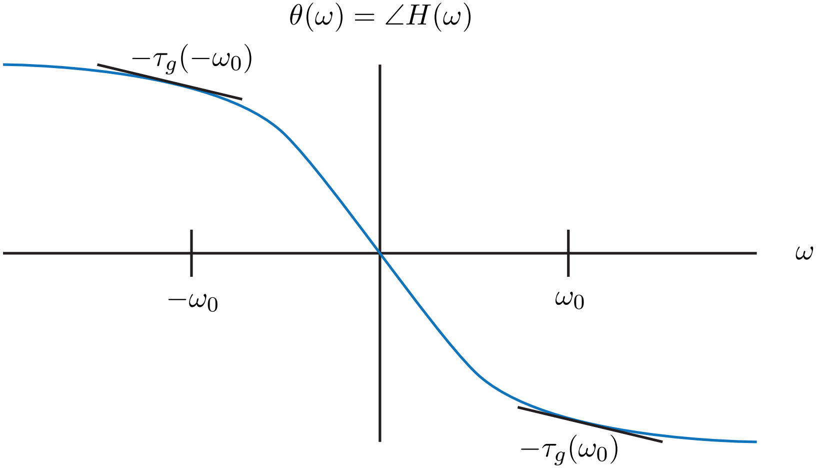
$$\angle H(\omega) \approx \theta(-\omega_0) + (\omega + \omega_0)\left[\frac{d\theta(\omega)}{d\omega}\right]_{\omega=-\omega_0} = -\theta(\omega_0) - (\omega + \omega_0)\tau_g(-\omega_0).$$

where

$$\tau_g(x) := -\left[\frac{d\theta(\omega)}{d\omega}\right]_{\omega=x}$$

is called the "group delay" of $\mathcal{H}$ at normalized frequency x. Units?

Group Delay


$\theta(\omega) = \angle H(\omega)$
$-\tau_g(-\omega_0)$
ω
$-\omega_0$
ω_0
$-\tau_g(\omega_0)$

Effect of Nonlinear Phase on Narrowband Signals (2 of 3)

If we further assume the magnitude response of the LTI system $\mathcal{H}$ to be constant over the bandwidth of the input, i.e. $|H(\omega)| = a$ for ω close to $\pm\omega_0$, then

$$H(\omega) \approx \begin{cases} ae^{j(\theta(\omega_0)-(\omega-\omega_0)\tau_g(\omega_0))} & \omega \approx \omega_0 \\ ae^{j(-\theta(\omega_0)-(\omega+\omega_0)\tau_g(-\omega_0))} & \omega \approx -\omega_0 \end{cases}$$

Since $\tau_g(-\omega_0) = \tau_g(\omega_0)$, we can write

$$H(\omega) \approx \begin{cases} ae^{j(\theta(\omega_0)+\omega_0\tau_g(\omega_0))}e^{-j\omega\tau_g(\omega_0)} & \omega \approx \omega_0 \\ ae^{-j(\theta(\omega_0)+\omega_0\tau_g(\omega_0))}e^{-j\omega\tau_g(\omega_0)} & \omega \approx -\omega_0 \end{cases}$$

We can think of this as the cascade of three systems: $H_1(\omega) = a$

$$H_2(\omega) = \begin{cases} e^{j(\theta(\omega_0)+\omega_0\tau_g(\omega_0))} & \omega \geq 0 \\ e^{-j(\theta(\omega_0)+\omega_0\tau_g(\omega_0))} & \omega < 0 \end{cases}$$

and $H_3(\omega) = e^{-j\omega\tau_g(\omega_0)}$.

Interlude: A System Like $\mathcal{H}_2$

Suppose you have an LTI system with frequency response

$$H(\omega) = \begin{cases} e^{-j\theta_0} & \omega \geq 0 \\ e^{j\theta_0} & \omega < 0. \end{cases}$$

Is this a linear phase system?

Given an input of $x[n] = A[n]\cos(\omega_0 n + \phi)$ and assuming $A(\omega) \approx 0$ for all $\omega > \omega_0$, we can compute the output of this system as follows.

First we compute

$$X(\omega) = \frac{1}{2}A(\omega - \omega_0)e^{j\phi} + \frac{1}{2}A(\omega + \omega_0)e^{-j\phi}.$$

Then we compute the output $Y(\omega) = H(\omega)X(\omega)$ as

$$Y(\omega) = \frac{1}{2}A(\omega - \omega_0)e^{j(\phi-\theta_0)} + \frac{1}{2}A(\omega + \omega_0)e^{-j(\phi-\theta_0)}.$$

Hence

$$y[n] = A[n]\cos(\omega_0 n + \phi - \theta_0).$$

Effect of Nonlinear Phase on Narrowband Signals (3 of 3)

Given the input $x[n] = A[n]\cos(\omega_0 n + \phi)$, the output of $\mathcal{H}_1$ is simply

$$y_1[n] = ax[n] = aA[n]\cos(\omega_0 n + \phi).$$

This is then processed by $\mathcal{H}_2$. Note that $\mathcal{H}_2$ is the same system we just saw with $\theta_0 = -\theta(\omega_0) - \omega_0\tau_g(\omega_0)$. Hence the output of $\mathcal{H}_2$ is

$$y_2[n] = aA[n]\cos(\omega_0 n + \phi + \theta(\omega_0) + \omega_0\tau_g(\omega_0)).$$

This is then processed by $\mathcal{H}_3$. Recognizing $\mathcal{H}_3$ is a linear phase system, the output of $\mathcal{H}_3$ (and the overall output of $\mathcal{H}$) is

$$\begin{aligned} y[n] &= aA[n-\tau_g(\omega_0)]\cos(\omega_0(n-\tau_g(\omega_0)) + \phi + \theta(\omega_0) + \omega_0\tau_g(\omega_0)) \\ &= aA[n-\tau_g(\omega_0)]\cos(\omega_0 n + \phi + \theta(\omega_0)) \\ &= aA[n-\tau_g(\omega_0)]\cos(\omega_0(n-\tau_p(\omega_0)) + \phi) \end{aligned}$$

Group Delay

Remarks:

- Group delay specifies the delay (in samples) of the lowpass "envelope" signal $A[n]$ when it is modulated at frequency ω_0 and sent through the LTI system $\mathcal{H}$.
- Phase delay specifies the delay (in samples) of the "carrier" $\cos(\omega_0 n + \phi)$ when it is sent through the LTI system $\mathcal{H}$.
- For a linear phase system, $\tau_g(\omega) = \tau_p(\omega) = c$, i.e. the group delay is the same as the phase delay.
- Group delay is also a measure of the deviation from phase linearity of a system, i.e. if the group delay varies wildly, then the system has highly nonlinear phase.
- See Matlab function `grpdelay`.

Simple Filtering (1 of 2)

Problem: We have an input signal

$$x[n] = c_0 \cos(\omega_0 n + \phi_0) + c_1 \cos(\omega_1 n + \phi_1).$$

We want to design an LTI system $\mathcal{H}$ that blocks the signal at ω_0 and passes the signal at ω_1.

Approach: Assume a real-valued symmetric impulse response

$$h[n] = \{\alpha_0, \alpha_1, \alpha_0\}.$$

We want to find values for α_0 and α_1 so that $|H(\omega_0)| = 0$ and $|H(\omega_1)| = 1$. Two equations and two unknowns.

Simple Filtering (2 of 2)

To compute the values of α_0 and α_1 that achieve the desired goal, we first compute the frequency response

$$H(\omega) = \sum_{n=-\infty}^{\infty} h[n]e^{-j\omega n} = \alpha_0 + \alpha_1 e^{-j\omega} + \alpha_0 e^{-j2\omega} = (2\alpha_0 \cos(\omega) + \alpha_1)e^{-j\omega}$$

Note $|H(\omega)| = |2\alpha_0 \cos(\omega) + \alpha_1|$. Hence, we can achieve the desired goal of $|H(\omega_0)| = 0$ and $|H(\omega_1)| = 1$ if

$$\begin{aligned} 2\alpha_0 \cos(\omega_0) + \alpha_1 &= 0 \\ 2\alpha_0 \cos(\omega_1) + \alpha_1 &= 1. \end{aligned}$$

These simultaneous equations are not difficult to solve for α_0:

$$2(\cos(\omega_1) - \cos(\omega_0))\alpha_0 = 1 \qquad \Leftrightarrow \qquad \alpha_0 = \frac{1}{2(\cos(\omega_1) - \cos(\omega_0))}$$

Then plug this result back into one of the equations above to get α_1.

Conclusions

1. This concludes Chapter 4. You are responsible for all of the material in this chapter, even if it wasn't covered in lecture.
2. Please read Chapter 5 before the next lecture and have some questions prepared.
3. The next lecture is on Monday 30-Jan-2012 at 6pm.