

# **ELEC 221 Lecture 17**

## **The sampling theorem**

Thursday 3 November 2022

# Announcements

- Midterms available for pickup at my office
- Assignment 5 available; due 11:59 Friday Nov. 11 (**no extensions**; solutions to be posted immediately after for studying)

**Important:** on Zoom for the next week.

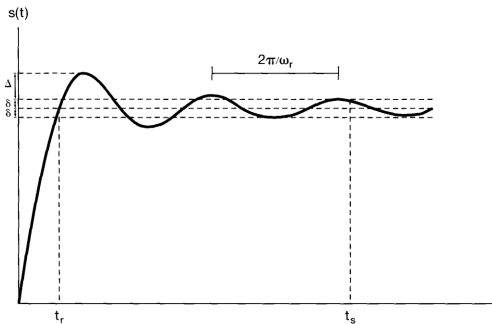
- Nov. 8 class
- Office hours this Friday and next Friday
- Still available by appointment

Links will be distributed on Canvas.

## Last time

We introduced the **step response** of filters.

$$s(t) = \int_{-\infty}^t h(\tau) d\tau$$



**Figure 6.17** Step response of a continuous-time lowpass filter, indicating the rise time  $t_r$ , overshoot  $\Delta$ , ringing frequency  $\omega_r$ , and settling time  $t_s$ —i.e., the time at which the step response settles to within  $\pm\delta$  of its final value.

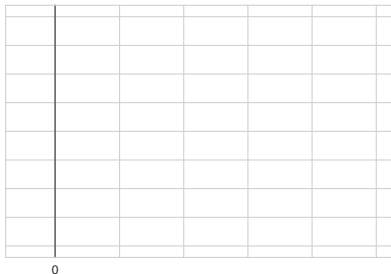
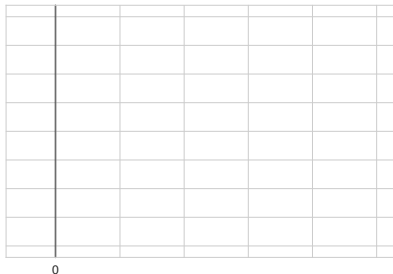
## Last time

$$\tau \frac{dy(t)}{dt} + y(t) = x(t), \quad H(j\omega) = \frac{1}{1 + j\omega\tau}$$

The impulse and step response of the system are

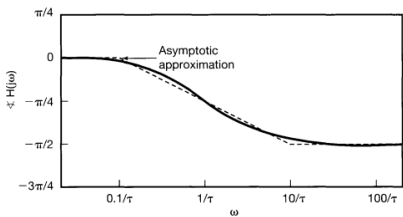
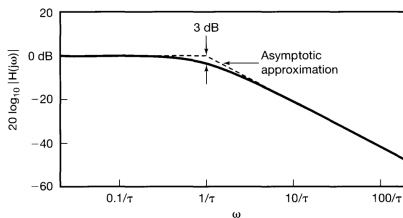
$$h(t) = \frac{1}{\tau} e^{-t/\tau} u(t), \quad s(t) = (1 - e^{-t/\tau}) u(t)$$

$\tau$  is the **time constant** of the system.



# Last time

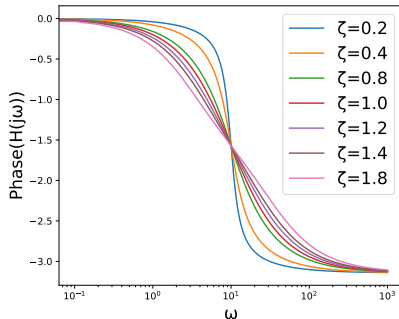
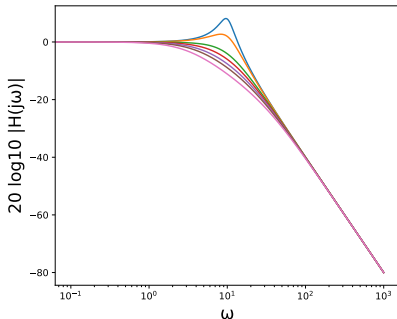
We drew some simple Bode plots.



## Last time

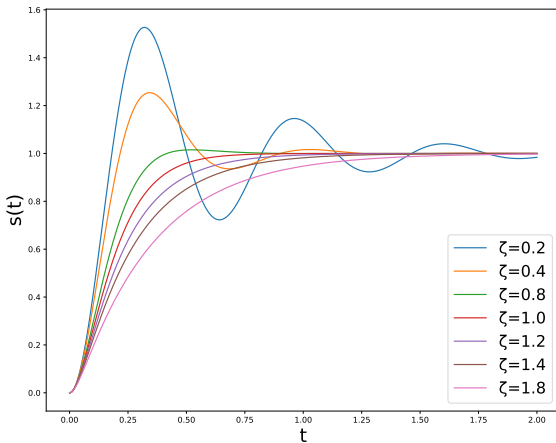
We looked at systems described by second-order ODEs.

$$\frac{d^2y(t)}{dt^2} + 2\zeta\omega_n\frac{dy(t)}{dt} + \omega_n^2y(t) = \omega_n^2x(t)$$

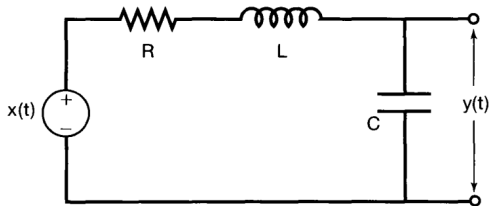


## Last time

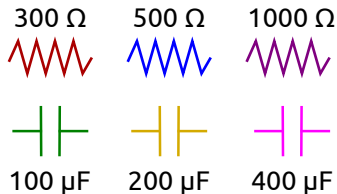
$\zeta$  is the damping ratio (can be under, over, or critically damped).



## Last time



Suppose  $L = 6\text{H}$ . We have a box of capacitors and resistors:



What is the best choice to ensure step response doesn't oscillate?



$$x(t) = LC \frac{d^2 y(t)}{dt^2} + RC \frac{dy(t)}{dt} + y(t)$$

Solution: compute the frequency response

$$\begin{aligned} H(j\omega) &= \frac{1}{LC(j\omega)^2 + RCj\omega + 1} \\ &= \frac{1}{(j\omega/(1/\sqrt{LC}))^2 + 2(R/2)\sqrt{C/L}j\omega + 1} \end{aligned}$$

Find that  $\zeta = (R/2)\sqrt{C/L}$

If  $\zeta = (R/2)\sqrt{C/L}$ , and  $L = 6H$ , we want

$$\begin{aligned}\frac{R}{2}\sqrt{\frac{C}{L}} &\geq 1 \\ R^2 C &\geq 4L = 24\end{aligned}$$

Best choice is  $R = 500\Omega$ , and  $C = 100\mu F$  ( $R^2 C = 25$ )

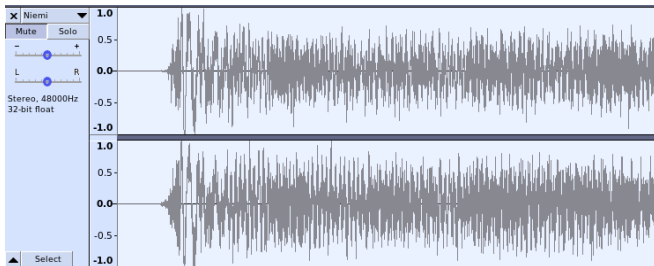
- state the sampling theorem
- define the Nyquist sampling rate and determine if a sampling rate is sufficient to reconstruct a signal
- construct systems of filters to interpolate a signal from its samples
- describe the phenomenon of aliasing

## Lecture 04 Demos

```
import numpy as np
import matplotlib.pyplot as plt
from IPython.display import Audio
```

### Demo 1: fun with square waves

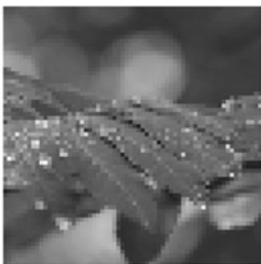
```
tone = 65 # A frequency in Hz
duration = 2 # The length of the audio signal (in seconds)
sample_rate = 48000 # The number of samples per second to take
t_range = np.linspace(0, duration, sample_rate * duration) # Range of time
```



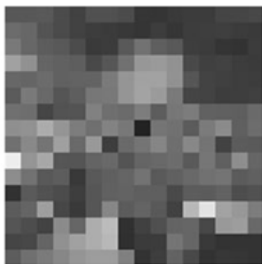
# Sampling



256 x 256



64 x 64



16 x 16

Image credit: [https://what-when-how.com/introduction-to-video-and-image-processing/  
image-acquisition-introduction-to-video-and-image-processing-part-2/](https://what-when-how.com/introduction-to-video-and-image-processing/image-acquisition-introduction-to-video-and-image-processing-part-2/)

# Sampling



History of frame rate in film:

<https://www.youtube.com/watch?v=mjYjFEp9Yx0>

Image credit: <https://www.mediacollege.com/video/frame-rate/img/frame-rates.jpg>

We saw that the discrete Fourier transform was a set of equally-spaced samples of the discrete-time Fourier transform.

## The discrete Fourier transform

What if we sample this signal at particular values of  $k\omega = k2\pi/N$ ?

$$X(e^{jk2\pi/N}) = \sum_{n=0}^{N-1} x[n]e^{-jk2\pi n/N} = N\tilde{X}[k]$$
$$\frac{1}{N}X(e^{jk2\pi/N}) = \tilde{X}[k]$$

The discrete Fourier transform is a set of equally spaced samples of the full discrete-time Fourier transform.

**Key point 1:** Any signal  $x[n]$  can be uniquely specified by a finite set of samples from its DTFT (i.e., its DFT).

## The unit impulse as a sampler

Multiplying the signal by a shifted impulse picks out the value of the signal at that point:

$$x[n] \cdot \delta[n-k] = x[k] \cdot \delta[n-k]$$

This allows us to write any signal as a **superposition of weighted impulses**.

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



In continuous time:

$$x(t)\delta(t - t_0) = x(t_0)\delta(t - t_0)$$

What if we have more than one?

$$p(t) = \sum_{n=-\infty}^{\infty} \delta(t - nT)$$

where  $\omega_s = 2\pi/T$

What does the following signal look like?

$$x_p(t) = x(t)p(t)$$

The combined signal is

$$x_p(t) = \sum_{n=-\infty}^{\infty} x(nT)\delta(t - nT)$$

This is all time domain; what happens in the frequency domain?

By the multiplication property,

$$x_p(t) = x(t)p(t) \quad \leftrightarrow \quad X_p(j\omega) = \frac{1}{2\pi} \int_{-\infty}^{\infty} X(j\theta)P(j(\omega - \theta))d\theta$$

But what is  $P(j\omega)$ ? We haven't evaluated this yet...

We have a periodic impulse train. Recall what Fourier transforms of periodic signals looked like:

$$X(j\omega) = 2\pi\delta(\omega - \omega_0) \quad \xleftrightarrow{\mathcal{F}} \quad x(t) = e^{j\omega_0 t}$$

$$X(j\omega) = \sum_{k=-\infty}^{\infty} 2\pi a_k \delta(\omega - k\omega_0) \quad \xleftrightarrow{\mathcal{F}} \quad x(t) = \sum_{k=-\infty}^{\infty} a_k e^{j\omega_0 k t}$$

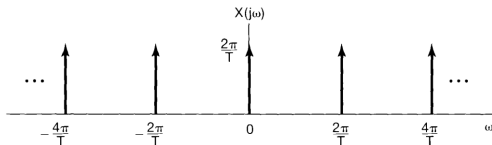
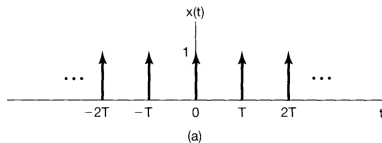
## Impulse train sampling

We need to find the Fourier series coefficients of the periodic impulse train.

$$\begin{aligned} p(t) &= \sum_{n=-\infty}^{\infty} \delta(t - nT) \\ a_k &= \frac{1}{T} \int_{-T/2}^{T/2} p(t) e^{-jk\omega t} dt \\ &= \frac{1}{T} \int_{-T/2}^{T/2} \delta(t) e^{-jk\omega t} dt \\ &= \frac{1}{T} \end{aligned}$$

# Impulse train sampling

$$\begin{aligned}P(j\omega) &= \sum_{k=-\infty}^{\infty} 2\pi a_k \delta(\omega - k\omega_s) \\&= \frac{2\pi}{T} \sum_{k=-\infty}^{\infty} \delta(\omega - k\omega_s)\end{aligned}$$



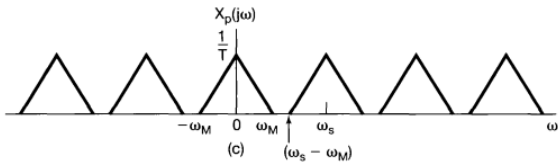
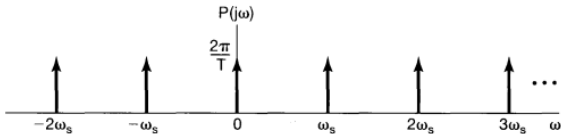
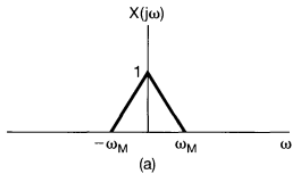
$$X(j\omega)$$

$$P(j\omega) = \frac{2\pi}{T} \sum_{k=-\infty}^{\infty} \delta(\omega - k\omega_s)$$

$$\begin{aligned} X_p(j\omega) &= \frac{1}{2\pi} \int_{-\infty}^{\infty} X(j\theta) P(j(\omega - \theta)) d\theta \\ &= \frac{1}{2\pi} \int_{-\infty}^{\infty} X(j\theta) \left( \frac{2\pi}{T} \sum_{k=-\infty}^{\infty} \delta(\omega - k\omega_s - \theta) \right) d\theta \\ &= \frac{1}{T} \sum_{k=-\infty}^{\infty} X(j(\omega - k\omega_s)) \end{aligned}$$

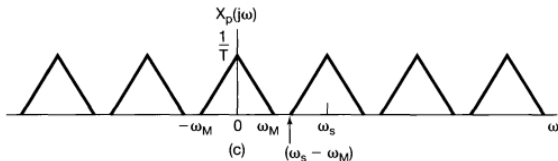


# Impulse train sampling



# Impulse train sampling

Suppose we have sampled...



How do we recover our original signal from this spectrum?

Image credit: Oppenheim 7.1

## The sampling theorem

“Let  $x(t)$  be a **band-limited** signal with  $X(j\omega) = 0$  for  $|\omega| > \omega_M$ . Then  $x(t)$  is uniquely determined by its samples  $x(nT)$ ,  $n = 0, \pm 1, \pm 2, \dots$ , if

$$\omega_s > 2\omega_M, \quad \omega_s = \frac{2\pi}{T}$$

Given these samples, we can reconstruct  $x(t)$  by generating a periodic impulse train in which successive impulses have amplitudes that are successive sample values. This impulse train is then processed through an ideal lowpass filter with gain  $T$  and cutoff frequency greater than  $\omega_M$  and less than  $\omega_s - \omega_M$ . The resulting output signal will exactly equal  $x(t)$ .”

# The sampling theorem

Let's show this graphically:

## The Nyquist rate

The sampling frequency is key:

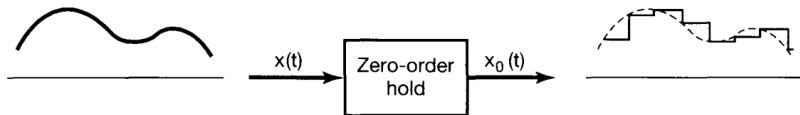
- $\omega_s = 2\omega_M$  is referred to as the **Nyquist rate**
- $\omega_M = \omega_s/2$  is referred to as the **Nyquist frequency**

*Exercise:* suppose we perform impulse-train sampling with period  $T = 10^{-4}$ . If a signal  $x(t)$  has  $X(j\omega) = 0$  for  $|\omega| > 15000\pi$ , can we reconstruct it exactly from the samples?

$$\begin{aligned}\omega_s &> 30000\pi \\ T = 10^{-4} &\rightarrow \omega \approx 62800 < 30000\pi\end{aligned}$$

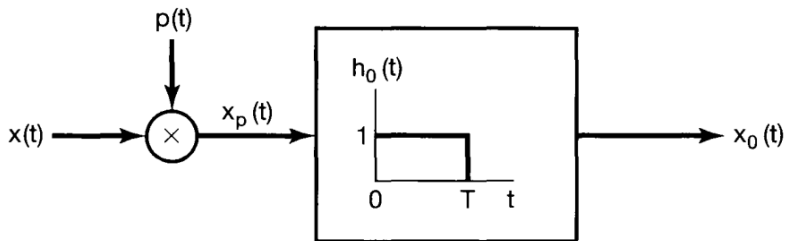
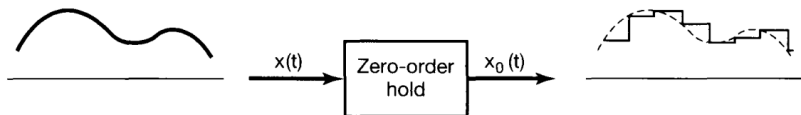
## Sampling in practice: zero-order hold

In reality we cannot generate perfect narrow, large-amplitude impulses. Instead:

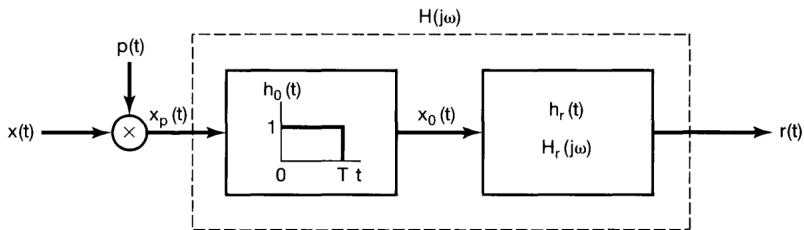
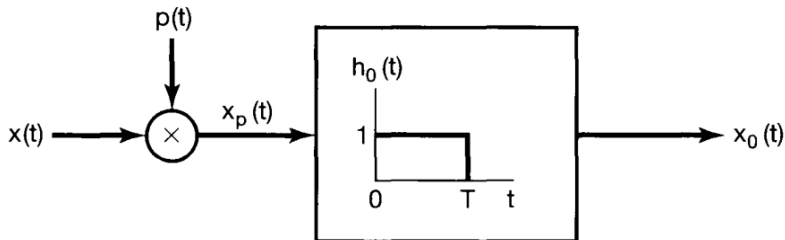


Can we still reconstruct our signal?

## Sampling in practice: zero-order hold

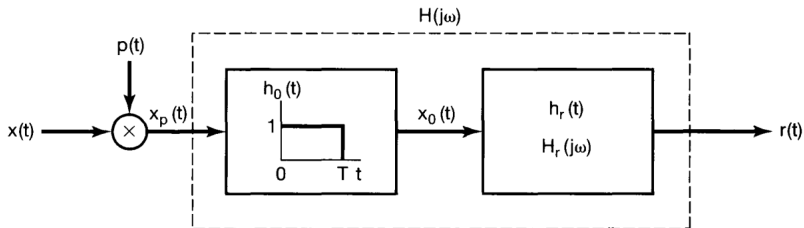


## Sampling in practice: zero-order hold





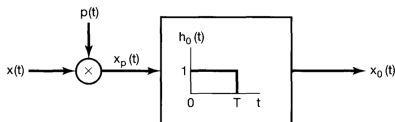
## Sampling in practice: zero-order hold



To obtain  $r(t) = x(t)$ , need  $H_r(j\omega)H_0(j\omega) = H(j\omega)$  for ideal lowpass filter.

But what is  $H_0(j\omega)$ ?

## Sampling in practice: zero-order hold



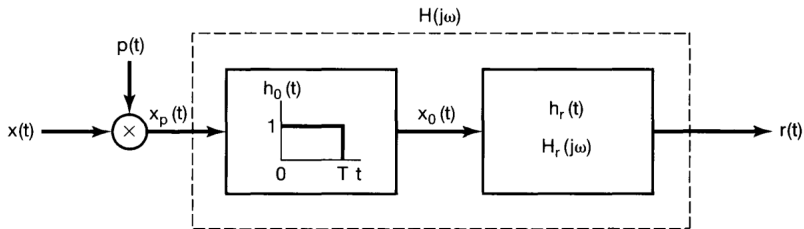
Square pulse between  $-T_1$  and  $T_1$ :

$$X(j\omega) = 2 \frac{\sin(\omega T_1)}{\omega}$$

Use properties of the Fourier transform to obtain

$$H_0(j\omega) = e^{-j\omega T/2} \left( 2 \frac{\sin(\omega T/2)}{\omega} \right)$$

## Sampling in practice: zero-order hold



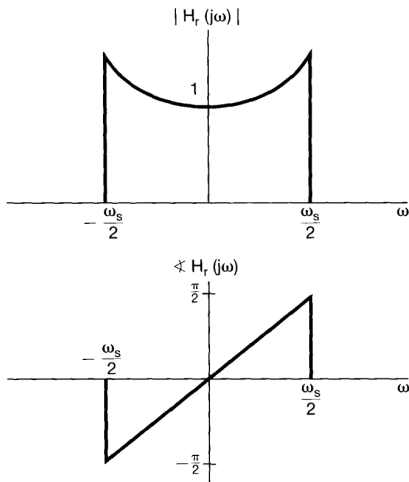
If  $H_r(j\omega)H_0(j\omega) = H(j\omega)$  (ideal lowpass filter) and

$$H_0(j\omega) = e^{-j\omega T/2} \left( 2 \frac{\sin(\omega T/2)}{\omega} \right)$$

then we need

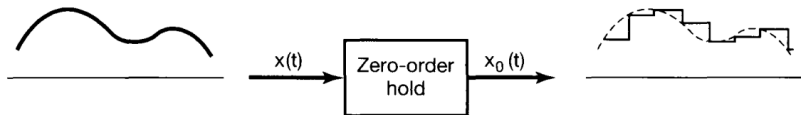
$$H_r(j\omega) = \frac{e^{j\omega T/2}}{2 \frac{\sin(\omega T/2)}{\omega}} H(j\omega)$$

# Sampling in practice: zero-order hold



# Interpolation

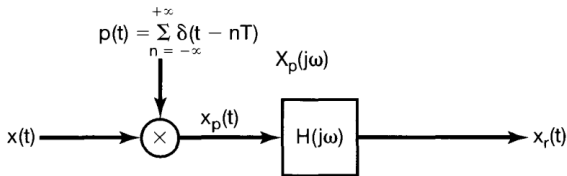
In some cases, the ZOH actually provides a good enough interpolation:



But we can do a lot better using, e.g., linear (first-order hold) or higher-order polynomial reconstruction methods.

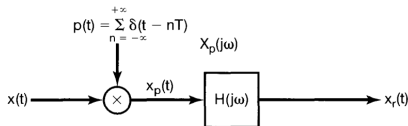
Image credit: Oppenheim 7.1

# Band-limited interpolation



$$\begin{aligned}x_p(t) &= \sum_{n=-\infty}^{\infty} x(nT)\delta(t - nT) \\x_r(t) &= x_p(t) * h(t) \\&= \sum_{n=-\infty}^{\infty} x(nT)h(t - nT)\end{aligned}$$

# Band-limited interpolation



For lowpass filter with cutoff  $\omega_c$  and gain  $T$ ,

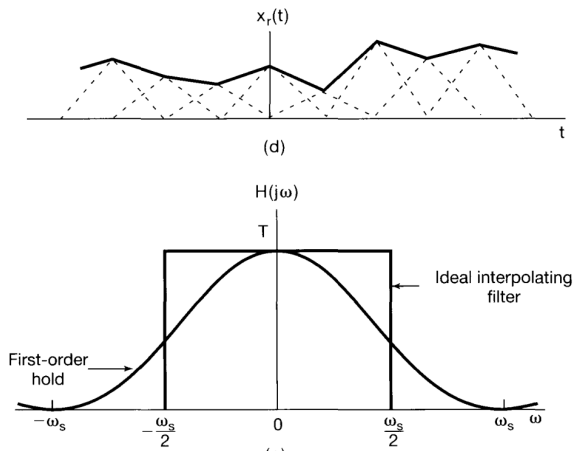
$$h(t) = \frac{\omega_c T \sin(\omega_c t)}{\pi \omega_c t}$$

Then

$$\begin{aligned} x_r(t) &= \sum_{n=-\infty}^{\infty} x(nT) h(t - nT) \\ &= \sum_{n=-\infty}^{\infty} x(nT) \frac{\omega_c T \sin(\omega_c (t - nT))}{\pi \omega_c (t - nT)} \end{aligned}$$

# Band-limited interpolation

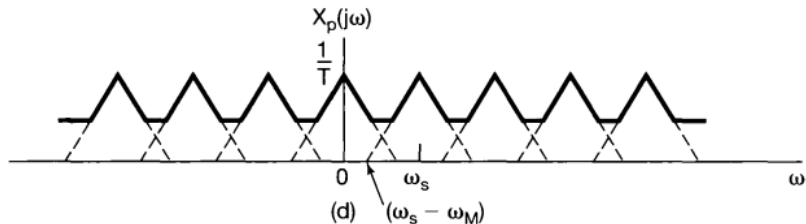
Sometimes zero- or first-order are good enough; increasing the order will improve interpolation at the cost of complexity.





# Aliasing

What happens when you don't sample at a high enough rate?



# Aliasing

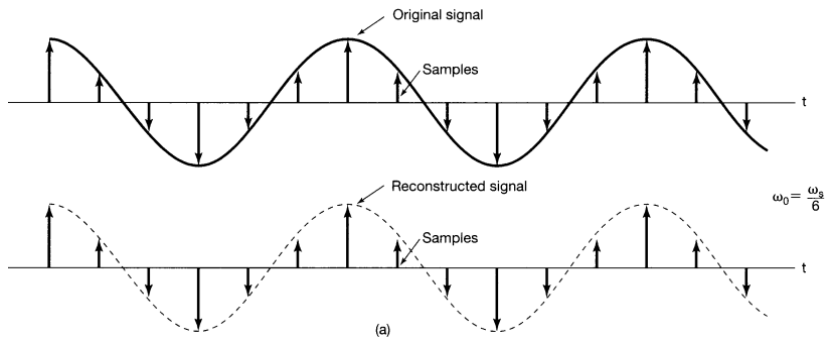


Image credit: Oppenheim 7.3

# Aliasing

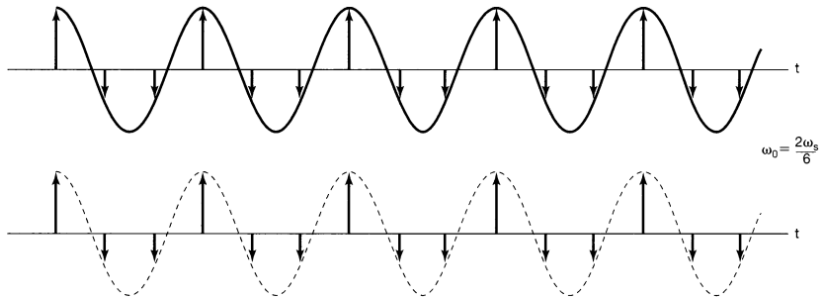


Image credit: Oppenheim 7.3

# Aliasing

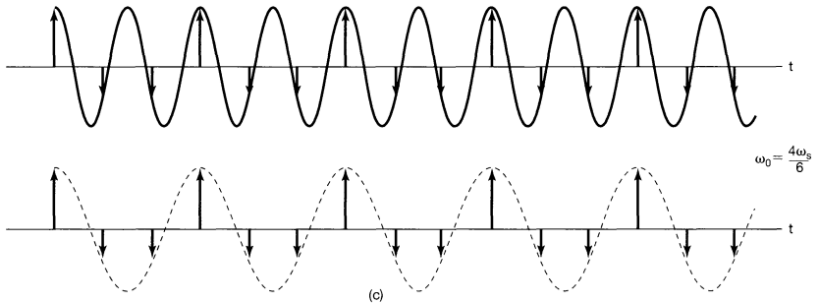


Image credit: Oppenheim 7.3

# Aliasing

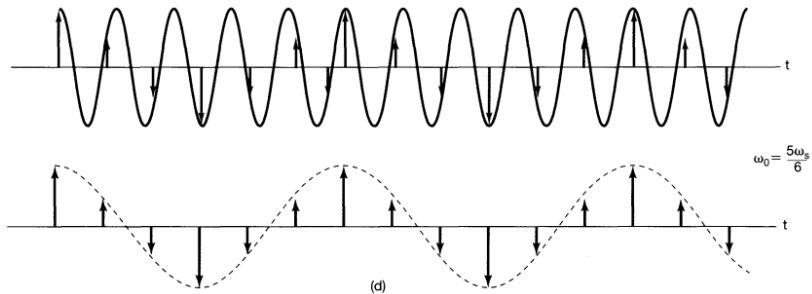
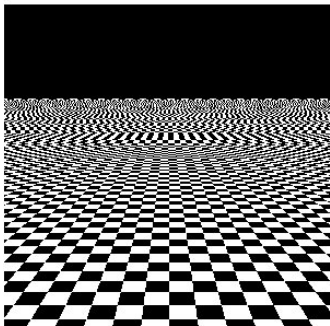


Image credit: Oppenheim 7.3



## Real-world examples



Fun on your own: read up about Moiré patterns, and various **anti-aliasing** techniques that are used in music/images/games!

Image credit: <https://textureingraphics.wordpress.com/what-is-texture-mapping/anti-aliasing-problem-and-mipmapping/>

[//textureingraphics.wordpress.com/what-is-texture-mapping/anti-aliasing-problem-and-mipmapping/](https://textureingraphics.wordpress.com/what-is-texture-mapping/anti-aliasing-problem-and-mipmapping/)

Learning outcomes:

- state the sampling theorem
- define the Nyquist sampling rate and determine if a sampling rate is sufficient to reconstruct a signal
- construct systems of filters to interpolate a signal from its samples
- describe the phenomenon of aliasing

Oppenheim practice problems: 7.1-7.7, 7.21, 7.25



## For next time

### Content:

- DT processing of CT signals
- Sampling in discrete time
- Decimation/interpolation

### Action items:

1. Assignment 5 due 11:59pm Friday 11 Nov
2. Midterm 2 Monday 14 Nov during tutorial

### Recommended reading:

- From this class: Oppenheim 7.1-7.3
- For next class: Oppenheim 7.4-7.6