COM303: Digital Signal Processing

Lecture 17: Sampling and applications

overview

- raw sampling and aliasing
- DT processing of CT signals

Sinc Sampling

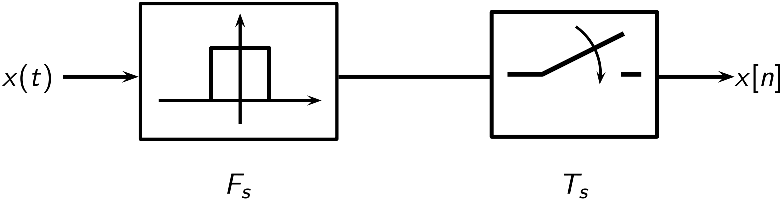
$$x[n] = \left\langle \operatorname{sinc}\left(\frac{t - nT_s}{T_s}\right), x(t) \right\rangle$$

Sinc Sampling

$$x[n] = (\mathrm{sinc}_{T_s} * x)(nT_s)$$

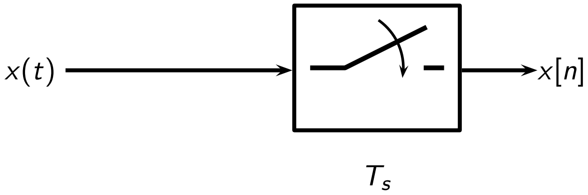
Sinc Sampling

$$x[n] = (\mathrm{sinc}_{T_s} * x)(nT_s)$$



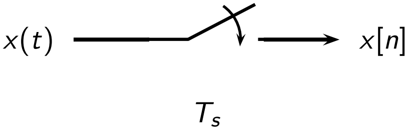
Sinc Sampling for F_s-BL signals

$$x[n] = (\mathrm{sinc}_{T_s} * x)(nT_s) = T_s\, x(nT_s)$$



"Raw" sampling - can we always do that?

$$x[n] = x(nT_s)$$



Remember the wagonwheel effect?

The continuous-time complex exponential

$$x(t) = e^{j2\pi f_0 t}$$

- always periodic, period $t_0 = 1/f_0$
- all angular speeds are allowed
- $\text{FT}\left\{e^{j2\pi f_0 t}\right\} = \delta(f - f_0)$
- highest (and only) frequency is f_0

The continuous-time complex exponential

$$x(t) = e^{j2\pi f_0 t}$$

- always periodic, period $t_0 = 1/f_0$
- all angular speeds are allowed
- $\mathrm{FT}\left\{e^{j2\pi f_0 t}\right\} = \delta(f - f_0)$
- highest (and only) frequency is f_0

The continuous-time complex exponential

$$x(t) = e^{j2\pi f_0 t}$$

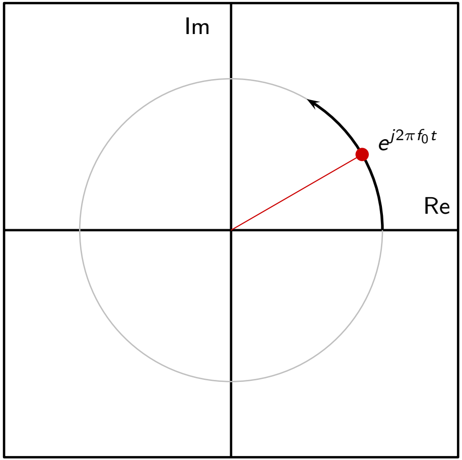
- always periodic, period $t_0 = 1/f_0$
- all angular speeds are allowed
- $\mathrm{FT}\left\{e^{j2\pi f_0 t}\right\} = \delta(f - f_0)$
- highest (and only) frequency is f_0

The continuous-time complex exponential

$$x(t) = e^{j2\pi f_0 t}$$

- always periodic, period $t_0 = 1/f_0$
- all angular speeds are allowed
- $\text{FT}\left\{e^{j2\pi f_0 t}\right\} = \delta(f - f_0)$
- highest (and only) frequency is f_0

The continuous-time complex exponential



Raw samples of the continuous-time complex exponential

$$x[n] = e^{j2\pi f_0 n T_s}$$

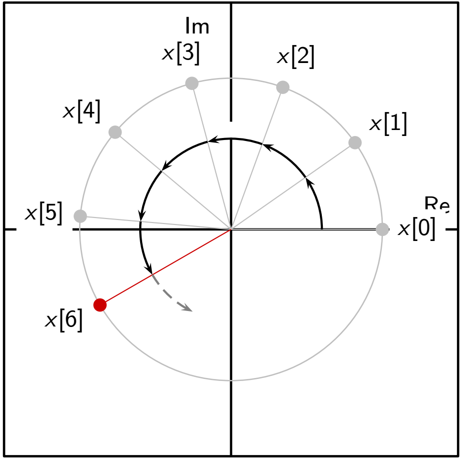
- raw samples are snapshots at regular intervals of the rotating point
- resulting digital frequency is $\omega_0 = 2\pi f_0 T_s = 2\pi(f_0/F_s)$

Raw samples of the continuous-time complex exponential

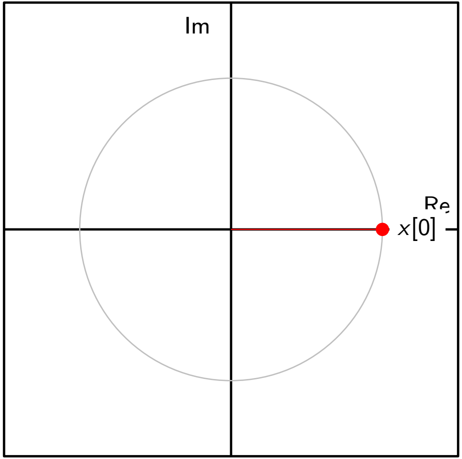
$$x[n] = e^{j2\pi f_0 n T_s}$$

- raw samples are snapshots at regular intervals of the rotating point
- resulting digital frequency is $\omega_0 = 2\pi f_0 T_s = 2\pi(f_0/F_s)$

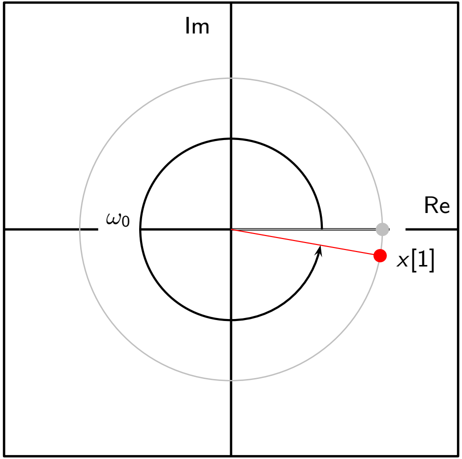
Easy: $f_0 < F_s/2 \quad \Rightarrow \quad \omega_0 < \pi$



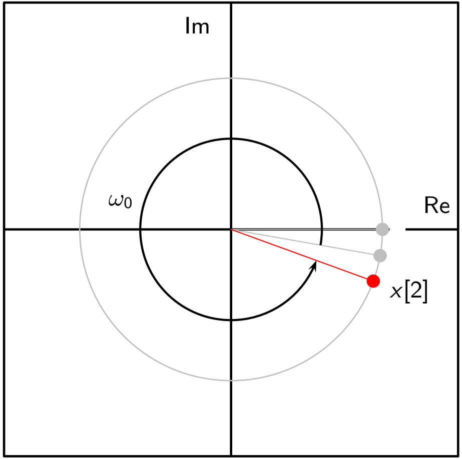
Tricky: $F_s/2 < f_0 < F_s \quad \Rightarrow \quad \pi < \omega_0 < 2\pi$



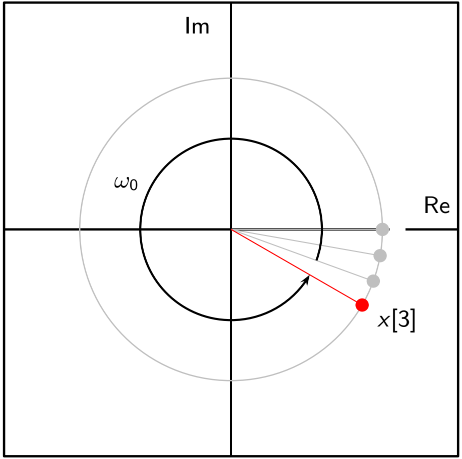
Tricky: $F_s/2 < f_0 < F_s \quad \Rightarrow \quad \pi < \omega_0 < 2\pi$



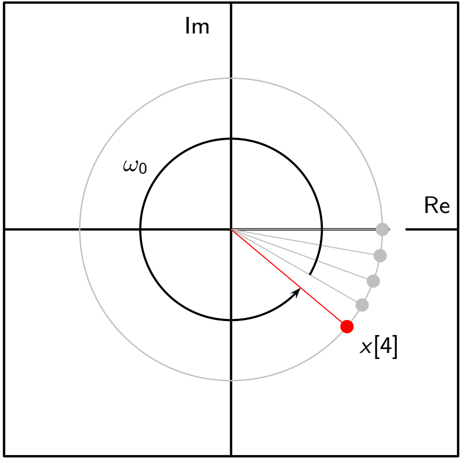
Tricky: $F_s/2 < f_0 < F_s \quad \Rightarrow \quad \pi < \omega_0 < 2\pi$



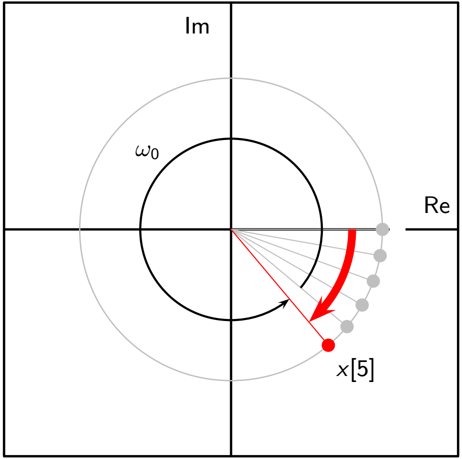
Tricky: $F_s/2 < f_0 < F_s \quad \Rightarrow \quad \pi < \omega_0 < 2\pi$



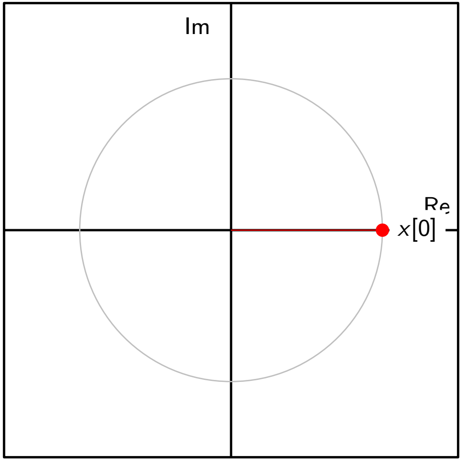
Tricky: $F_s/2 < f_0 < F_s \quad \Rightarrow \quad \pi < \omega_0 < 2\pi$



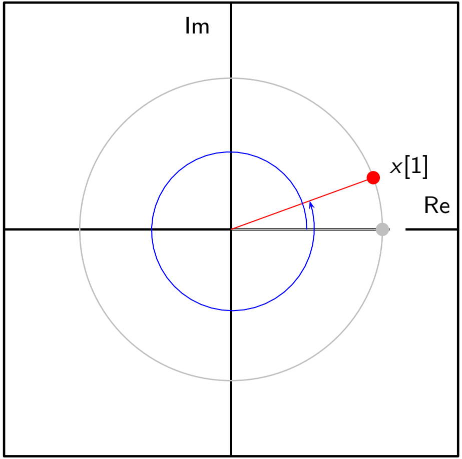
Tricky: $F_s/2 < f_0 < F_s \quad \Rightarrow \quad \pi < \omega_0 < 2\pi$



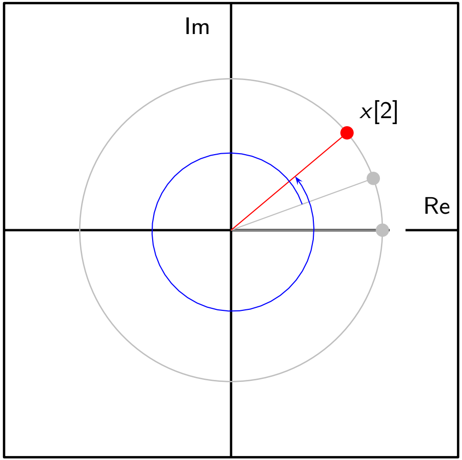
Trouble: $f_0 > F_s \quad \Rightarrow \quad \omega_0 > 2\pi$



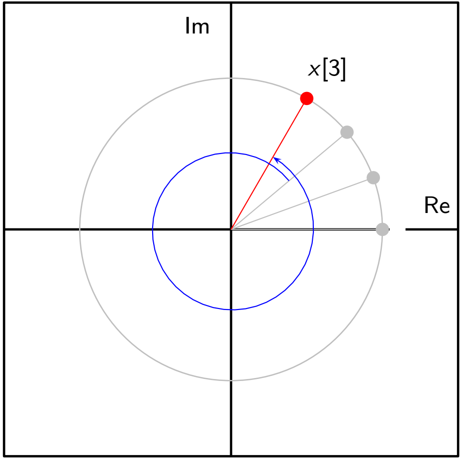
Trouble: $f_0 > F_s \quad \Rightarrow \quad \omega_0 > 2\pi$



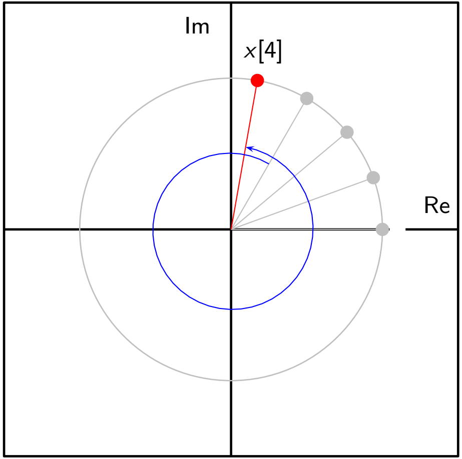
Trouble: $f_0 > F_s \quad \Rightarrow \quad \omega_0 > 2\pi$



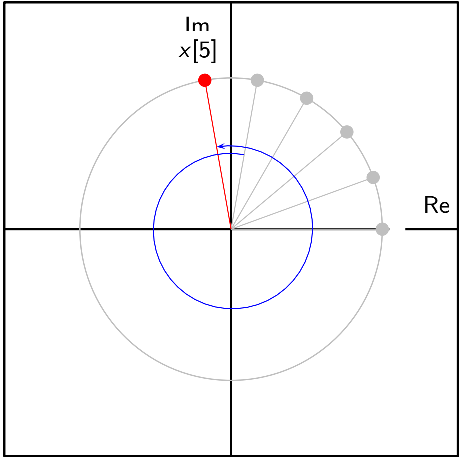
Trouble: $f_0 > F_s \quad \Rightarrow \quad \omega_0 > 2\pi$



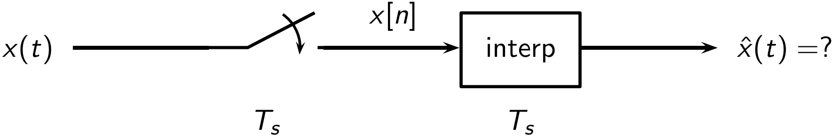
Trouble: $f_0 > F_s \quad \Rightarrow \quad \omega_0 > 2\pi$



Trouble: $f_0 > F_s \quad \Rightarrow \quad \omega_0 > 2\pi$



Aliasing



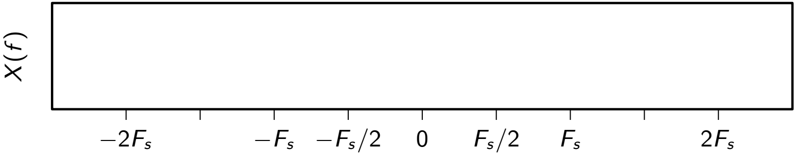
Aliasing

$$\text{pick } T_s;\ F_s = 1/T_s$$

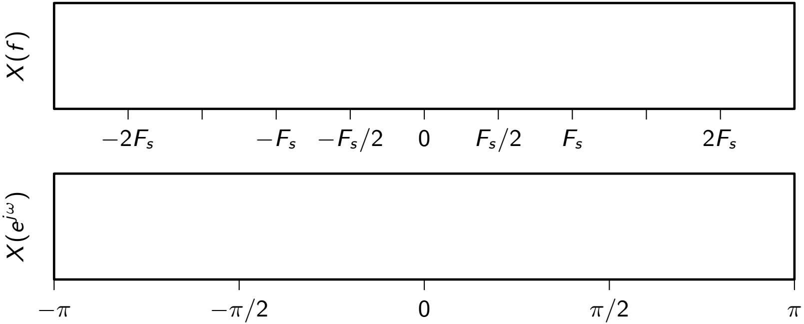
$$\text{input: } x(t) = e^{j2\pi f_0 t}$$
$$\text{digital frequency: } \omega_0 = 2\pi f_0/F_s$$

	digital frequency	$\hat{x}(t)$	
$f_0 < F_s/2$	$0 < \omega_0 < \pi$	$e^{j2\pi f_0 t}$	
$f_0 = F_s/2$	$\omega_0 = \pi$	$e^{j2\pi f_0 t}$	
$F_s/2 < f_0 < F_s$	$\pi < \omega_0 < 2\pi$	$e^{j2\pi f_1 t}$,	$f_1 = f_0 - F_s < 0$
$f_0 > F_s$	$\omega_0 > 2\pi$	$e^{j2\pi f_2 t}$,	$f_2 = f_0 \mod F_s$

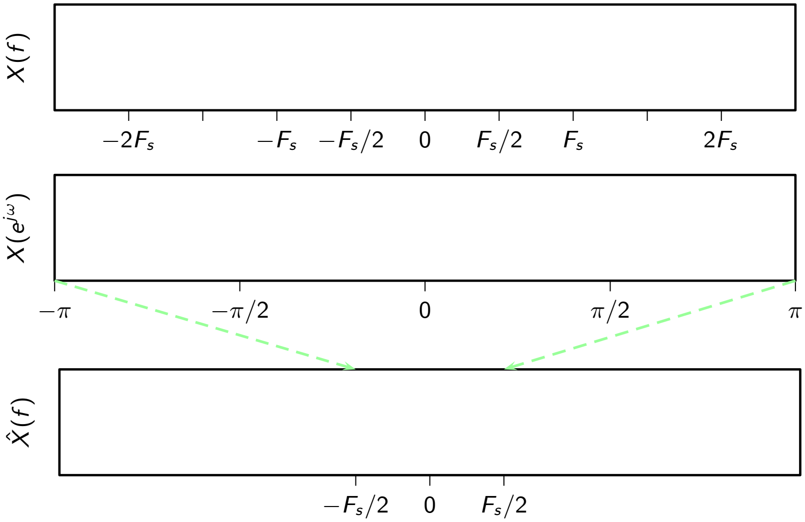
Aliasing of sinusoids: increasing the input frequency



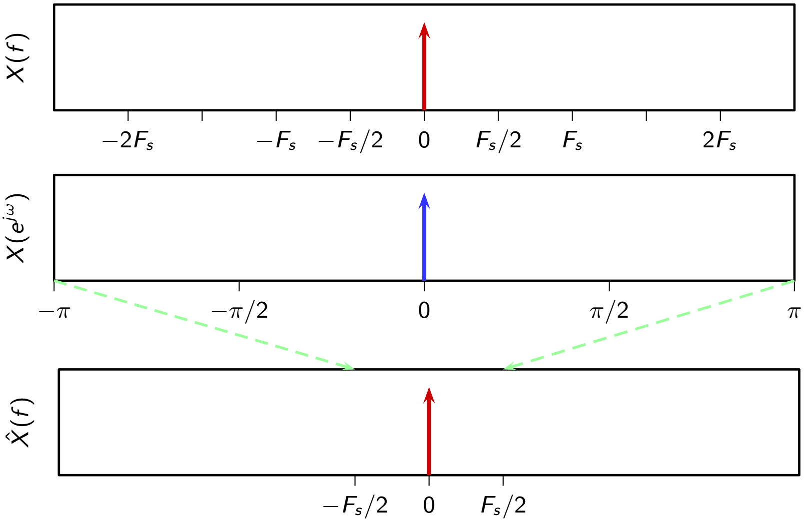
Aliasing of sinusoids: increasing the input frequency



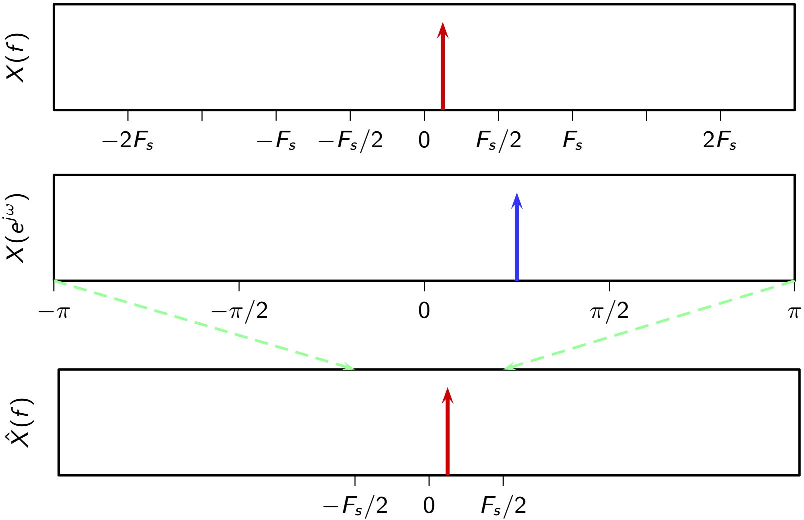
Aliasing of sinusoids: increasing the input frequency



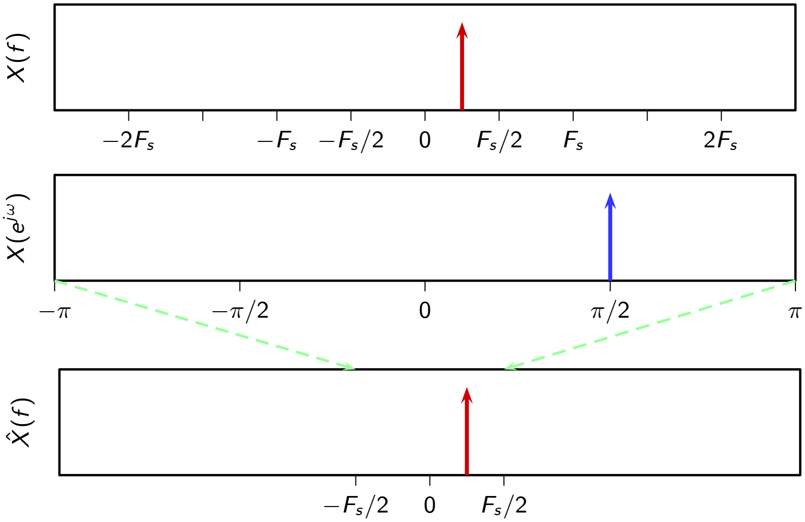
Aliasing of sinusoids: increasing the input frequency



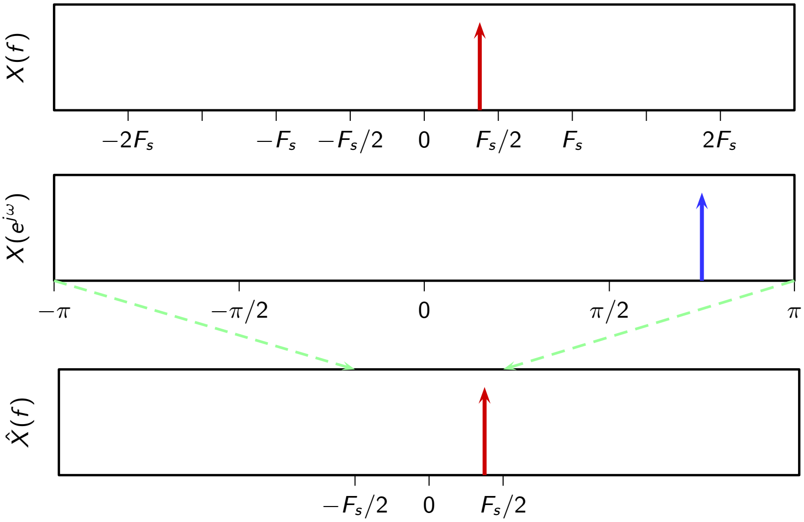
Aliasing of sinusoids: increasing the input frequency



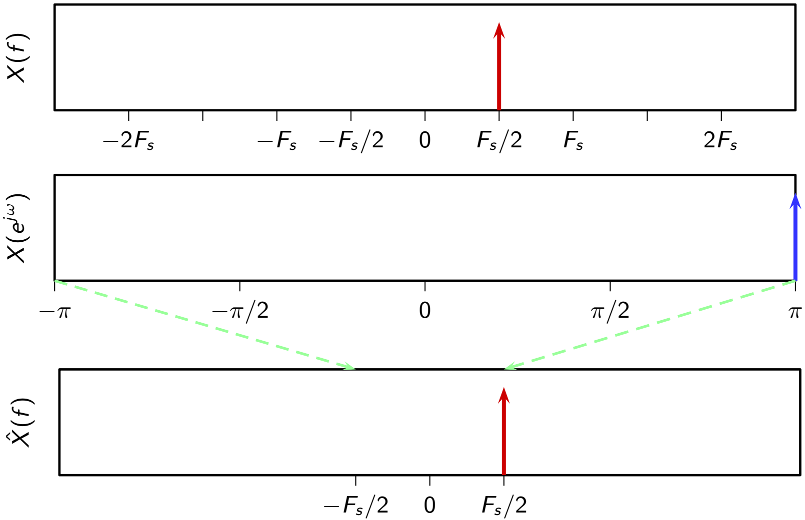
Aliasing of sinusoids: increasing the input frequency



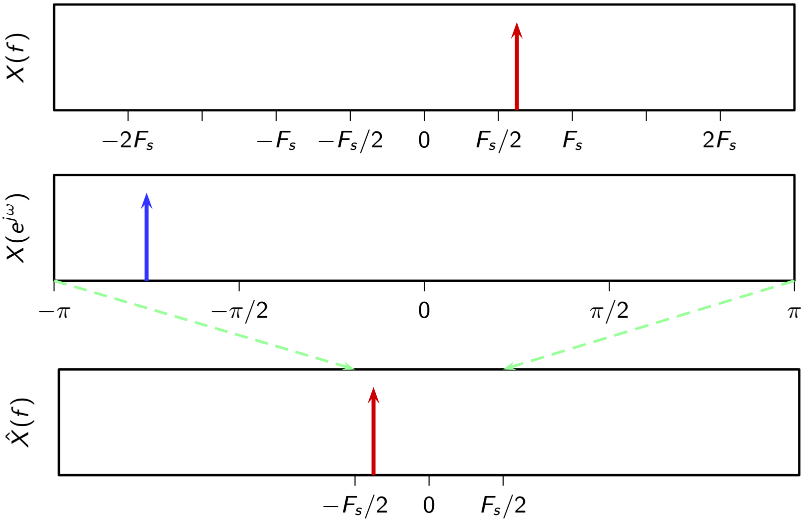
Aliasing of sinusoids: increasing the input frequency



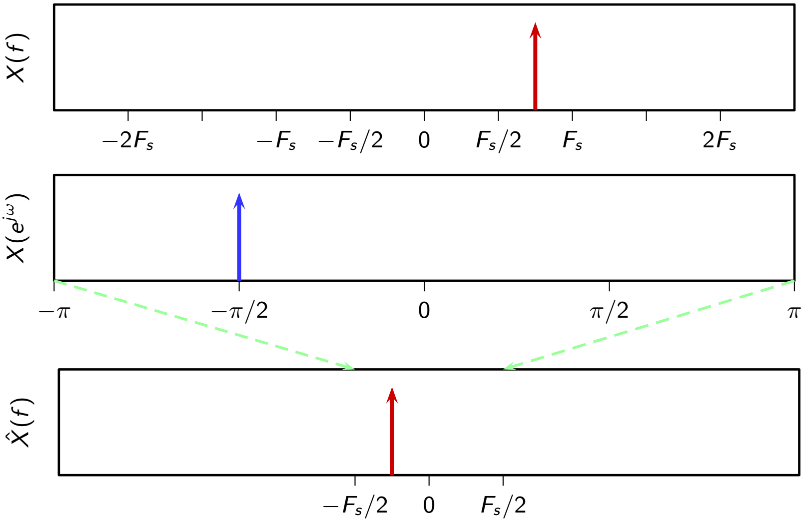
Aliasing of sinusoids: increasing the input frequency



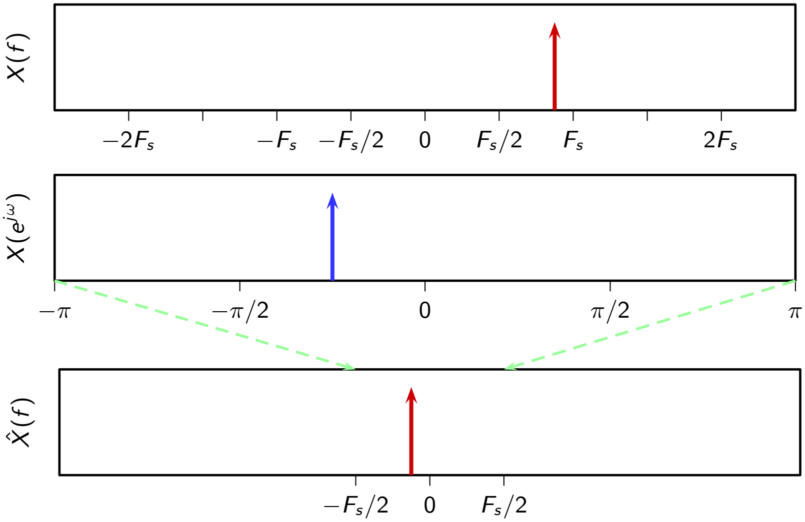
Aliasing of sinusoids: increasing the input frequency



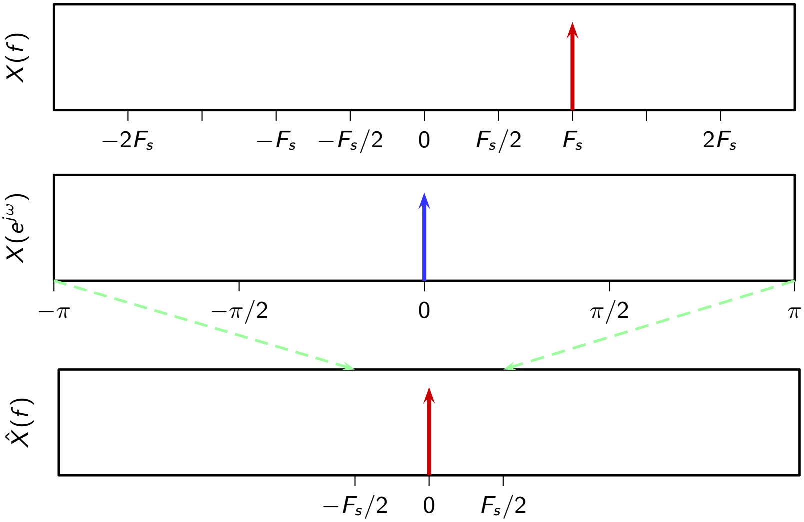
Aliasing of sinusoids: increasing the input frequency



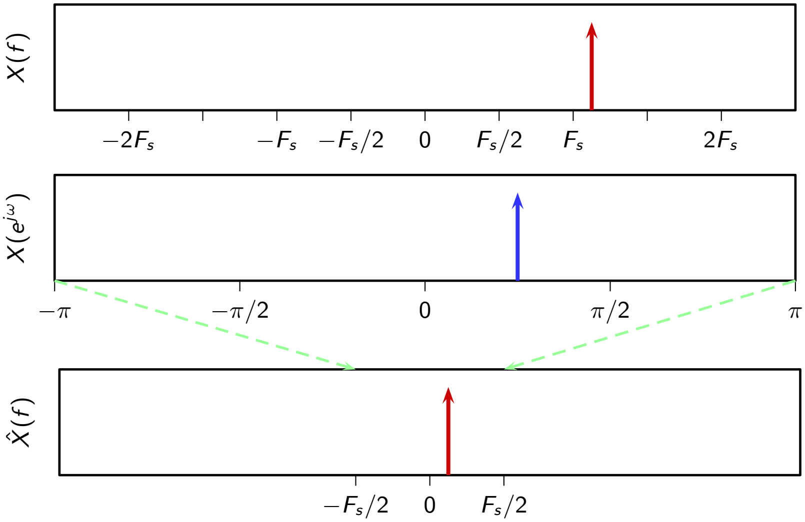
Aliasing of sinusoids: increasing the input frequency



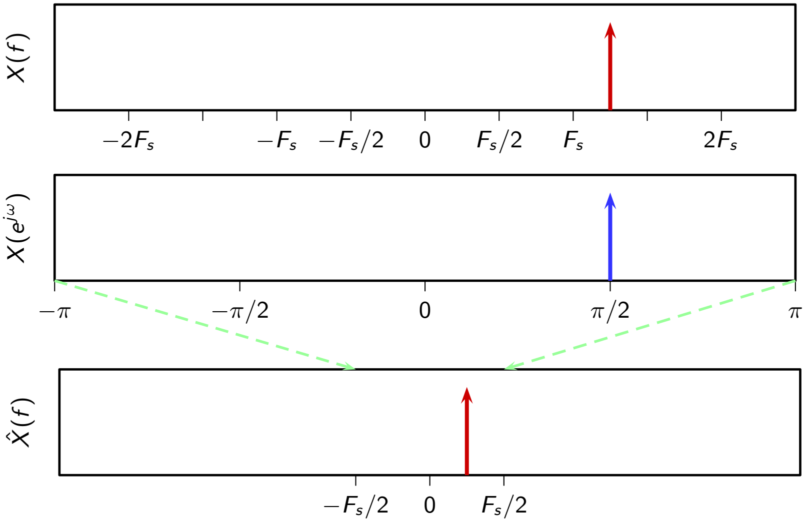
Aliasing of sinusoids: increasing the input frequency



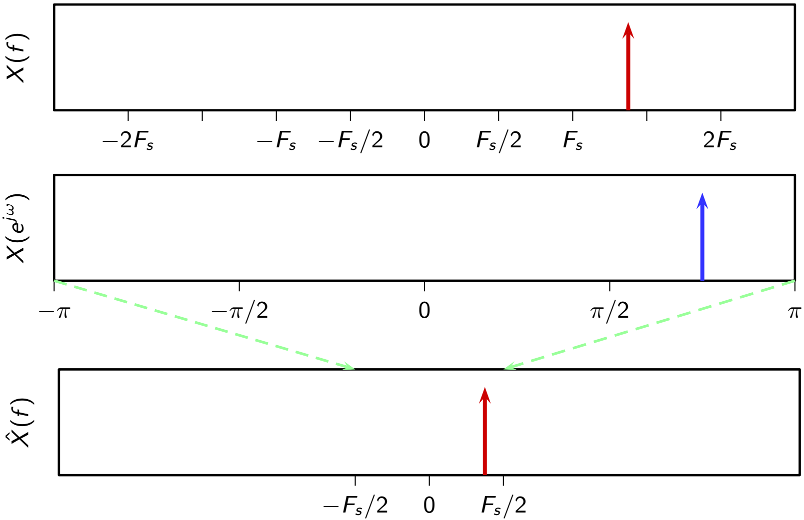
Aliasing of sinusoids: increasing the input frequency



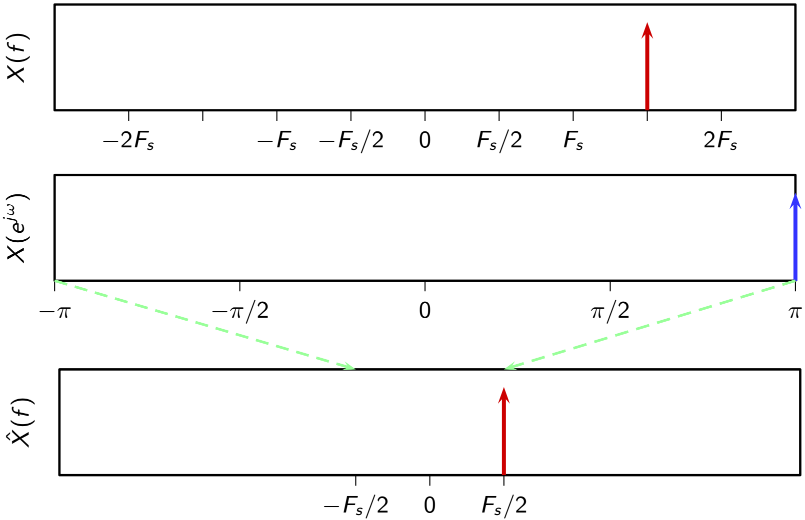
Aliasing of sinusoids: increasing the input frequency



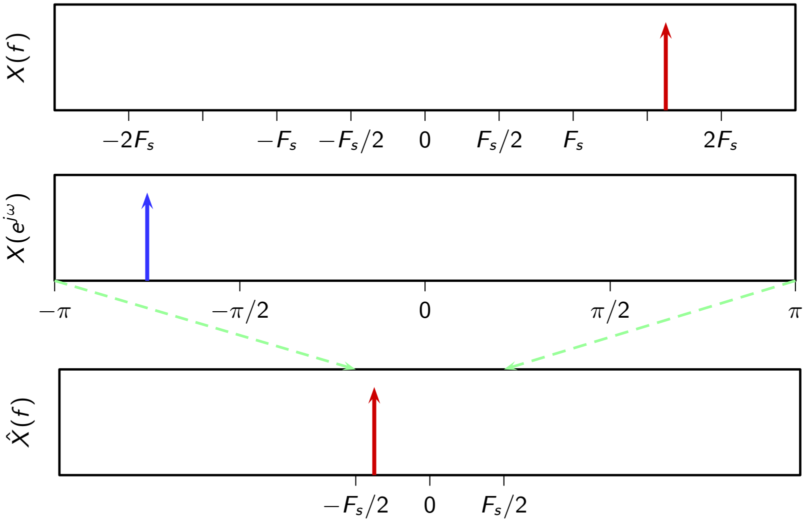
Aliasing of sinusoids: increasing the input frequency



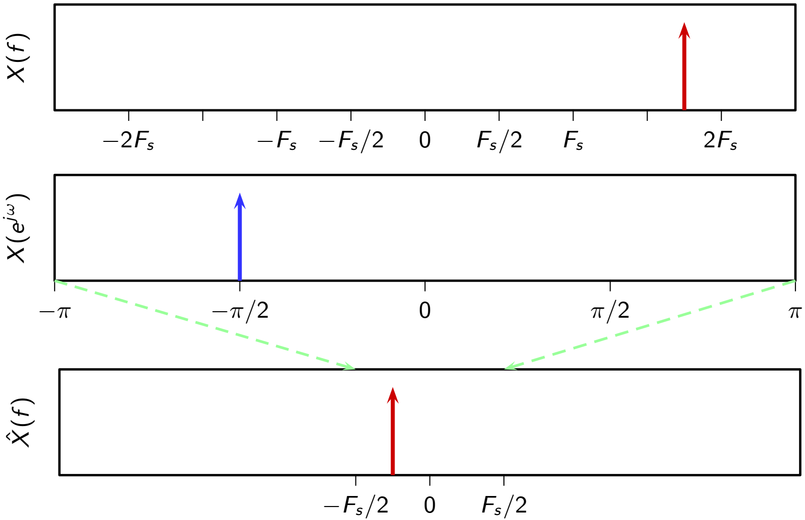
Aliasing of sinusoids: increasing the input frequency



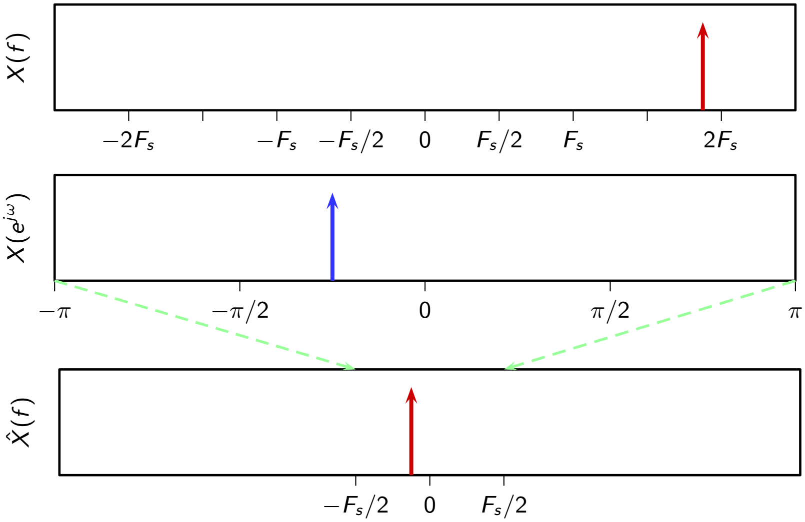
Aliasing of sinusoids: increasing the input frequency



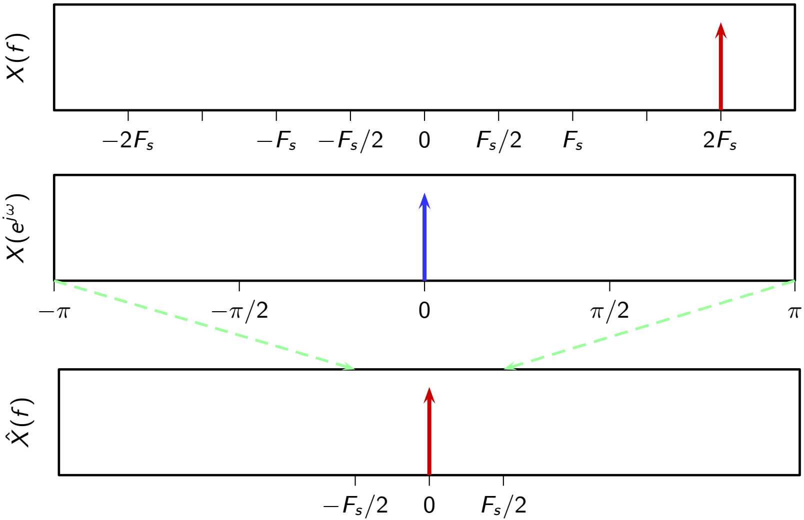
Aliasing of sinusoids: increasing the input frequency



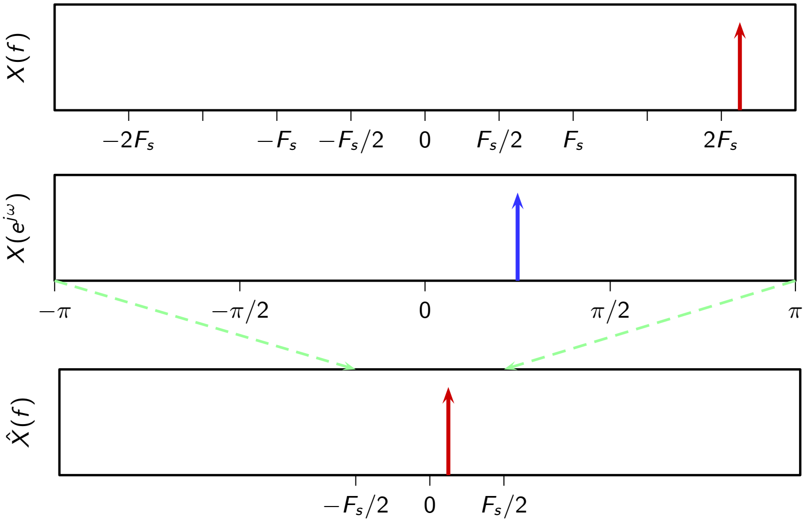
Aliasing of sinusoids: increasing the input frequency



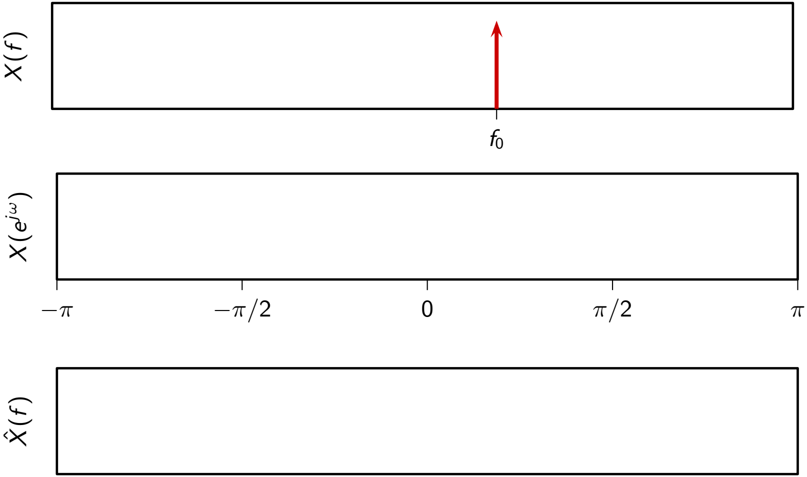
Aliasing of sinusoids: increasing the input frequency



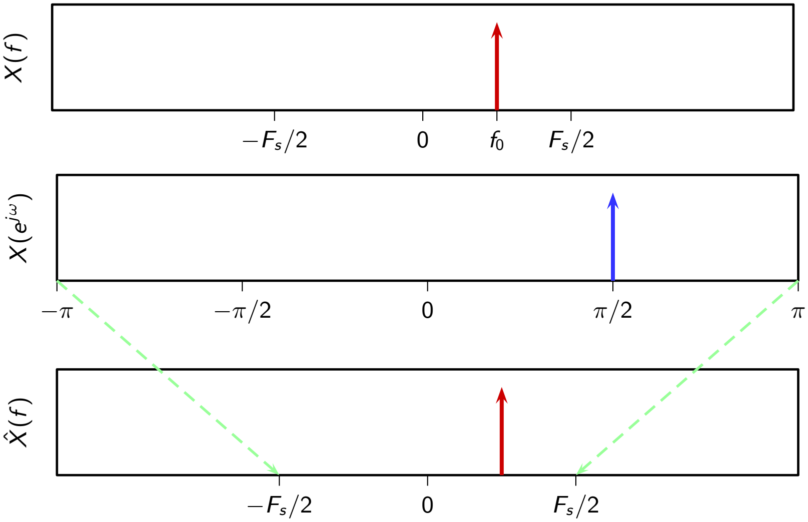
Aliasing of sinusoids: increasing the input frequency



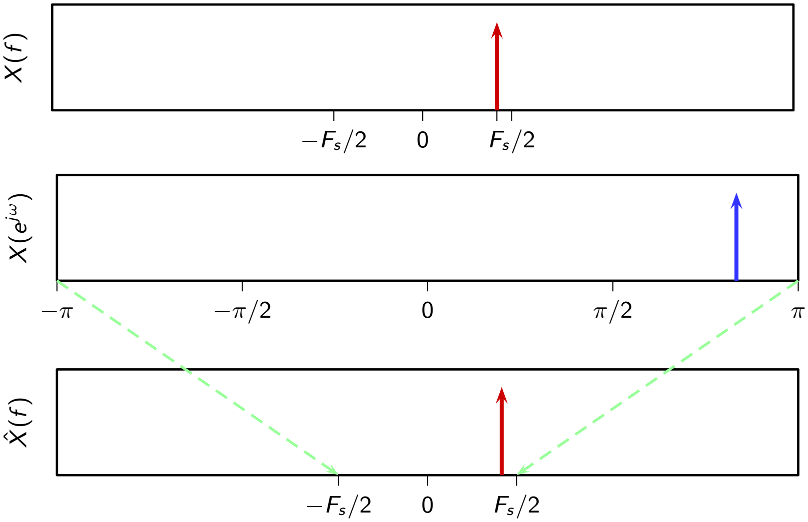
Aliasing of sinusoids: decreasing the sampling frequency



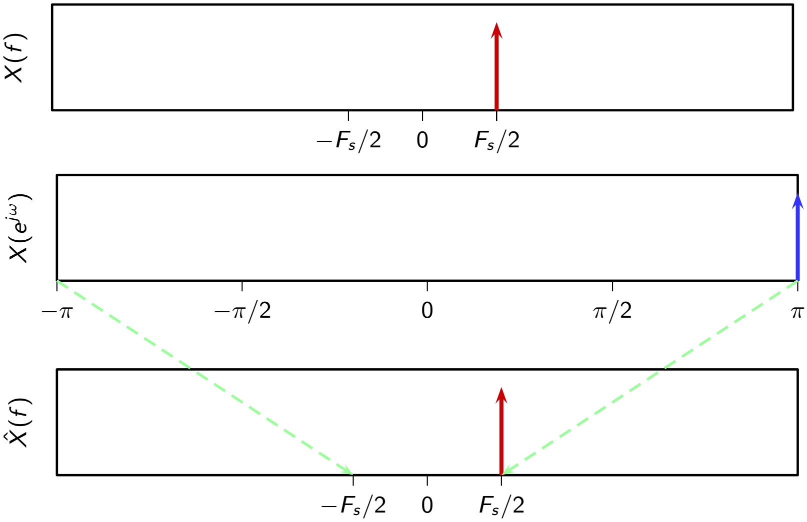
Aliasing of sinusoids: decreasing the sampling frequency



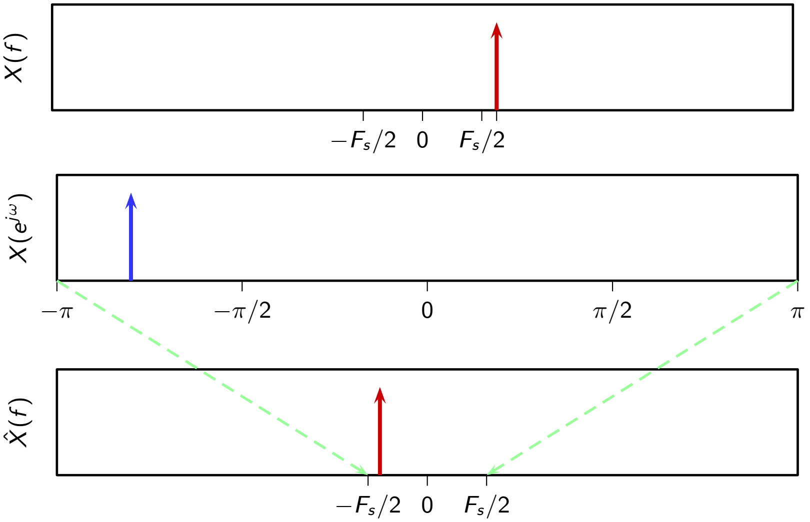
Aliasing of sinusoids: decreasing the sampling frequency



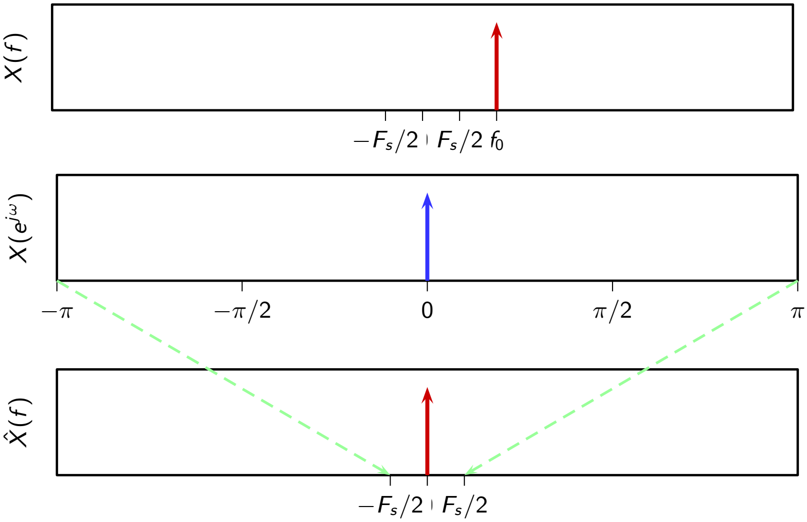
Aliasing of sinusoids: decreasing the sampling frequency



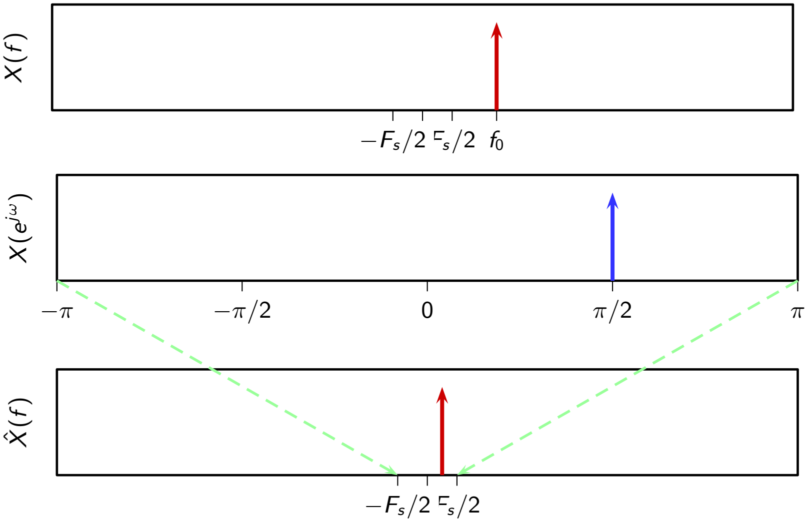
Aliasing of sinusoids: decreasing the sampling frequency



Aliasing of sinusoids: decreasing the sampling frequency



Aliasing of sinusoids: decreasing the sampling frequency

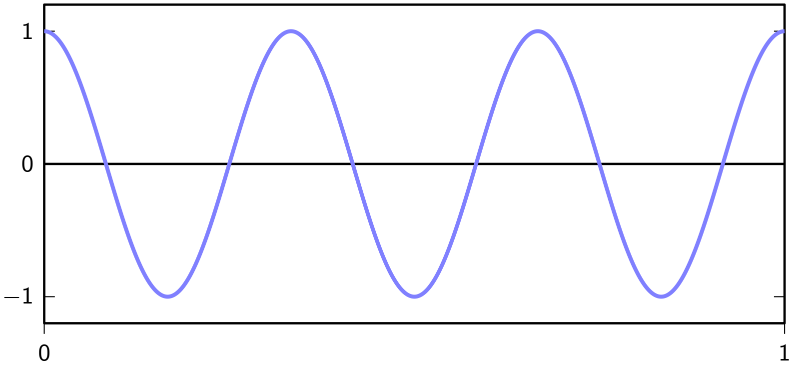


Sampling a Sinusoid

sampling frequency	digital frequency	interpolation
$F_s > 2f_0$	$0 < \omega_0 < \pi$	OK: $\hat{f}_0 = f_0$
$F_s = 2f_0$	$\omega_0 = \pi$	OK (max frequency $\hat{f}_0 = F_s$)
$f_0 < F_s < 2f_0$	$\pi < \omega_0 < 2\pi$	negative frequency: $\hat{f}_0 = f_0 - F_s$
$F_s < f_0$	$\omega_0 > 2\pi$	full aliasing: $\hat{f}_0 = f_0 \mod F_s$

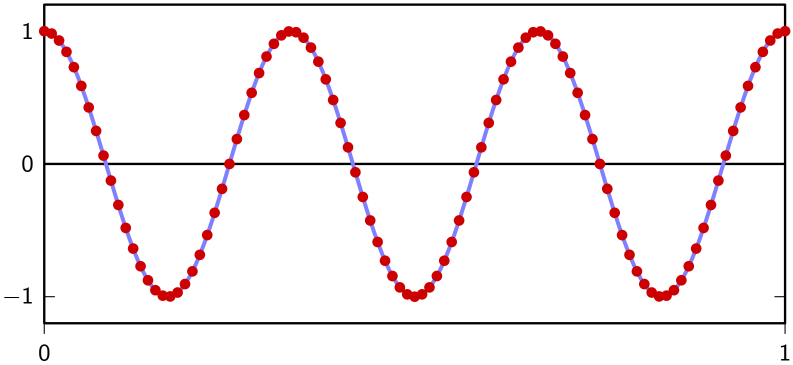
Aliasing: Sampling a Sinusoid

$$x(t) = \cos(6\pi t) \qquad (f_0 = 3\text{Hz})$$



Aliasing: Sampling a Sinusoid

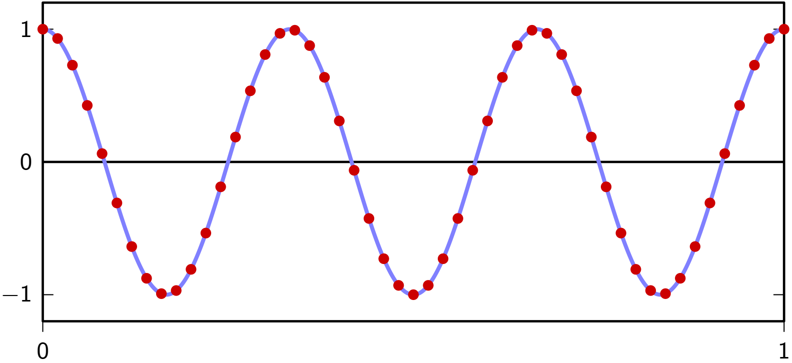
$$x(t) = \cos(6\pi t) \qquad (f_0 = 3\text{Hz})$$



$F_s = 100\text{Hz}$

Aliasing: Sampling a Sinusoid

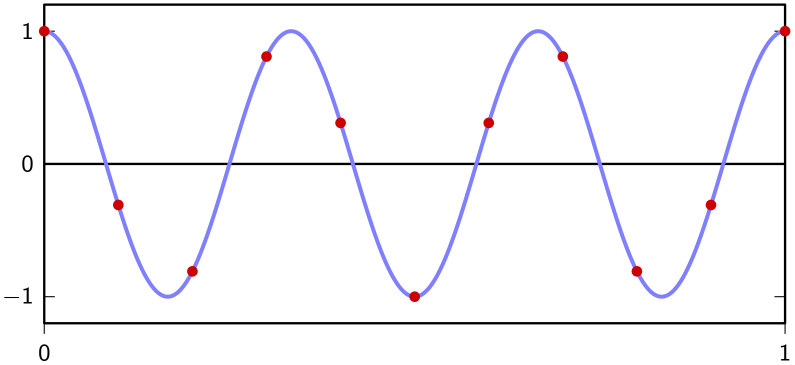
$$x(t) = \cos(6\pi t) \qquad (f_0 = 3\text{Hz})$$



$F_s = 50\text{Hz}$

Aliasing: Sampling a Sinusoid

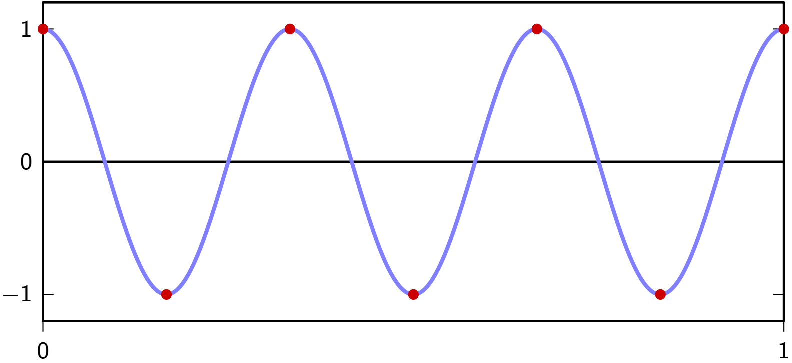
$$x(t) = \cos(6\pi t) \qquad (f_0 = 3\text{Hz})$$



$F_s = 10\text{Hz}$

Aliasing: Sampling a Sinusoid

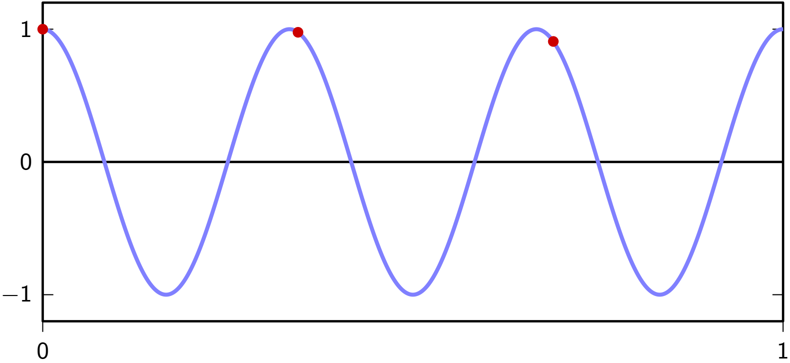
$$x(t) = \cos(6\pi t) \qquad (f_0 = 3\text{Hz})$$



$F_s = 6\text{Hz}$

Aliasing: Sampling a Sinusoid

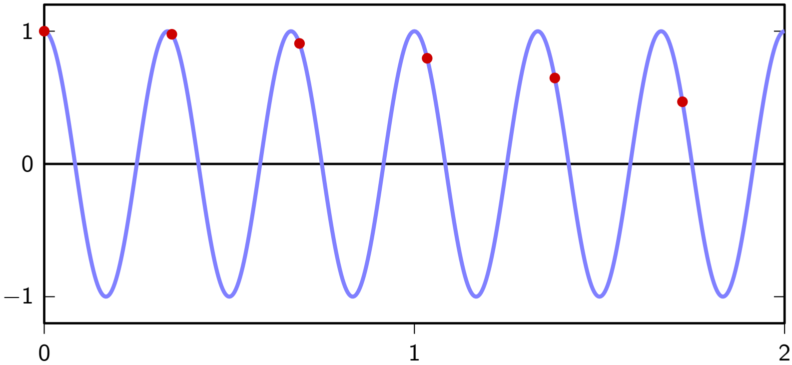
$$x(t) = \cos(6\pi t) \qquad (f_0 = 3\text{Hz})$$



$F_s = 2.9\text{Hz}$

Aliasing: Sampling a Sinusoid

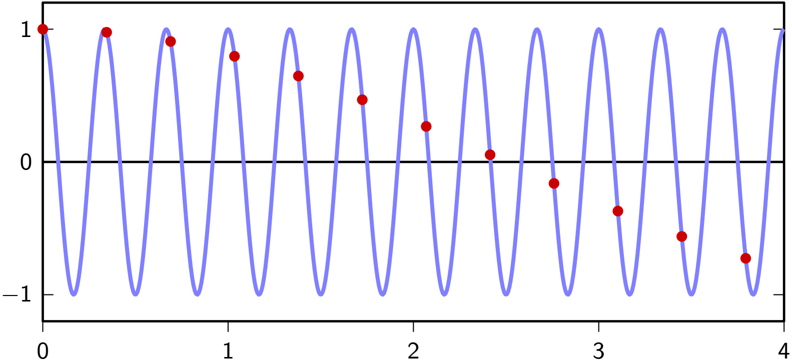
$$x(t) = \cos(6\pi t) \qquad (f_0 = 3\text{Hz})$$



$F_s = 2.9\text{Hz}$

Aliasing: Sampling a Sinusoid

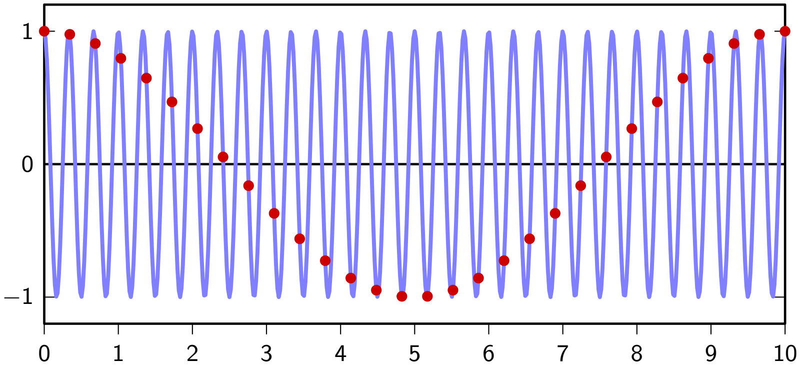
$$x(t) = \cos(6\pi t) \qquad (f_0 = 3\text{Hz})$$



$F_s = 2.9\text{Hz}$

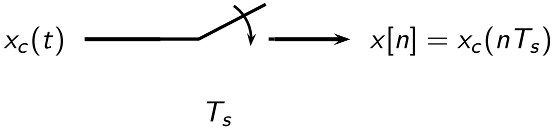
Aliasing: Sampling a Sinusoid

$$x(t) = \cos(6\pi t) \qquad (f_0 = 3\text{Hz})$$

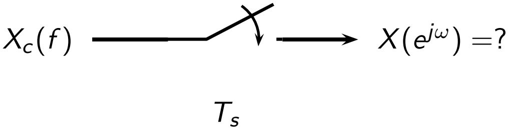


$F_s = 2.9\text{Hz}$

Raw-sampling an arbitrary signal

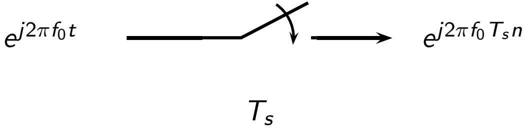


Raw-sampling an arbitrary signal



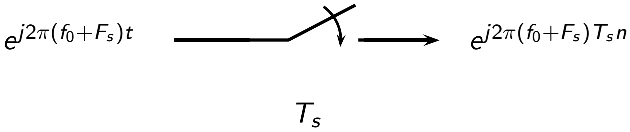
Key idea

- pick T_s (and set $F_s = 1/T_s$)
- pick $f_0 < F_s/2$



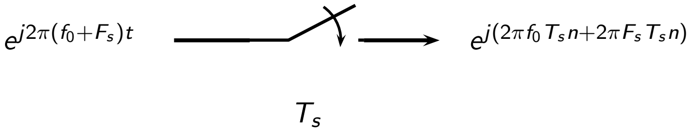
Key idea

- pick T_s (and set $F_s = 1/T_s$)
- pick $f_0 < F_s/2$



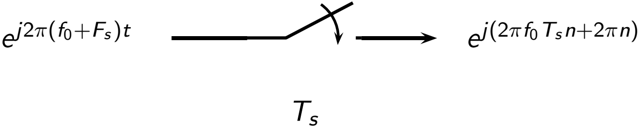
Key idea

- pick T_s (and set $F_s = 1/T_s$)
- pick $f_0 < F_s/2$



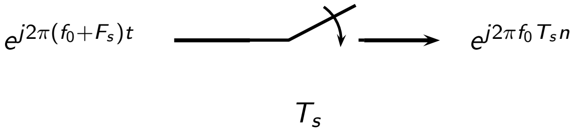
Key idea

- pick T_s (and set $F_s = 1/T_s$)
- pick $f_0 < F_s/2$



Key idea

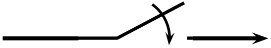
- pick T_s (and set $F_s = 1/T_s$)
- pick $f_0 < F_s/2$



Key idea

- pick T_s (and set $F_s = 1/T_s$)
- pick $f_0 < F_s/2$

$$Ae^{j2\pi f_0 t} + Be^{j2\pi (f_0 + F_s)t}$$



$$(A+B)e^{j2\pi f_0 T_s n}$$

Spectrum of raw-sampled signals

outline: start with the inverse Fourier Transform

$$x[n] = x_c(nT_s) = \int_{-\infty}^{\infty} X_c(f) e^{j2\pi f T_s n} df$$

and manipulate the integral until it looks like

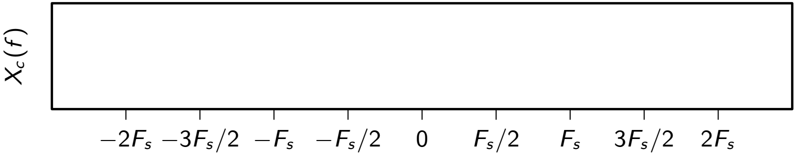
$$x[n] = \frac{1}{2\pi} \int_{-\pi}^{\pi} f(\omega) e^{j\omega n} d\omega$$

Spectrum of raw-sampled signals

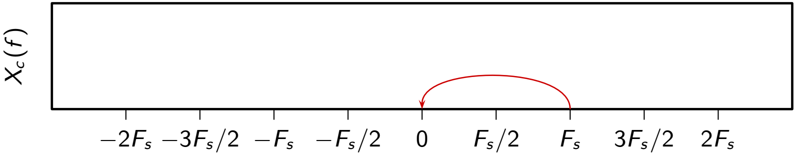
frequencies F_s Hz apart will be aliased, so split the integration interval

$$
\begin{aligned}
x[n] &= \int_{-\infty}^{\infty} X_c(f) e^{j2\pi f\, T_s n} df \\
&= \sum_{k=-\infty}^{\infty} \int_{kF_s - \frac{F_s}{2}}^{kF_s + \frac{F_s}{2}} X_c(f) e^{j2\pi f\, T_s n} df
\end{aligned}
$$

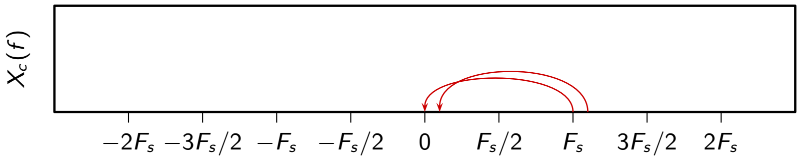
Spectrum of raw-sampled signals



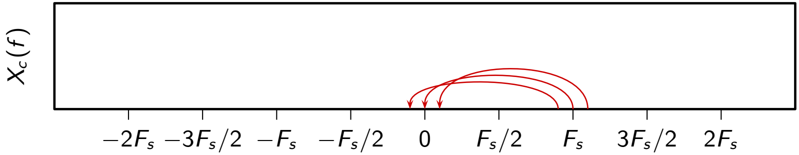
Spectrum of raw-sampled signals



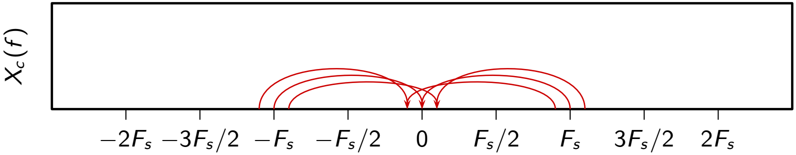
Spectrum of raw-sampled signals



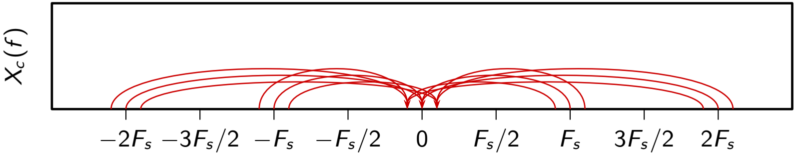
Spectrum of raw-sampled signals



Spectrum of raw-sampled signals



Spectrum of raw-sampled signals



Spectrum of raw-sampled signals

$$x[n] = \sum_{k=-\infty}^{\infty} \int_{kF_s - \frac{F_s}{2}}^{kF_s + \frac{F_s}{2}} X_c(f) e^{j2\pi f\, T_s n} df$$

operate the change of variable $f \to f + kF_s$:

- integration limits become $\pm F_s/2$
- $e^{j2\pi(f - kF_s)T_s n} = e^{j2\pi f\, T_s n}$

Spectrum of raw-sampled signals

$$x[n] = \sum_{k=-\infty}^{\infty} \int_{kF_s-\frac{F_s}{2}}^{kF_s+\frac{F_s}{2}} X_c(f) e^{j2\pi f T_s n} df$$

$$= \sum_{k=-\infty}^{\infty} \int_{-F_s/2}^{F_s/2} X_c(f - kF_s) e^{j2\pi f T_s n} df$$

$$= \int_{-F_s/2}^{F_s/2} \left[\sum_{k=-\infty}^{\infty} X_c(f - kF_s) \right] e^{j2\pi f T_s n} df$$

Spectrum of raw-sampled signals

F_s-periodization of the spectrum; define:

$$\tilde{X}_c(f) = \sum_{k=-\infty}^{\infty} X_c(f - kF_s)$$

then:

$$x[n] = \int_{-F_s/2}^{F_s/2} \tilde{X}_c(f) e^{j2\pi f\, T_s n} df$$

Spectrum of raw-sampled signals

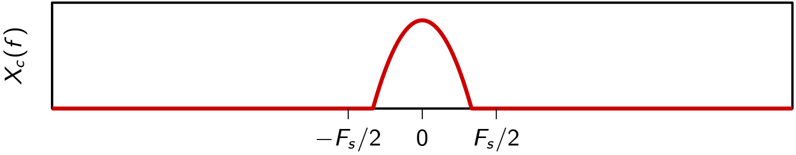
set $\omega = 2\pi f\, T_s$, so that $f = \frac{\omega}{2\pi} F_s$

$$x[n] = \frac{1}{2\pi} \int_{-\pi}^{\pi} F_s\, \tilde{X}_c\left(\frac{\omega}{2\pi} F_s\right) e^{j\omega n} d\omega$$

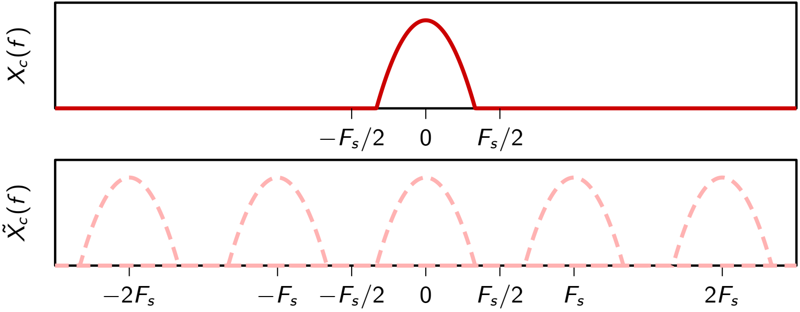
$$= \text{IDTFT}\left\{F_s \tilde{X}_c\left(\frac{\omega}{2\pi} F_s\right)\right\}$$

$$X(e^{j\omega}) = F_s \tilde{X}_c\left(\frac{\omega}{2\pi} F_s\right) = F_s \sum_{k=-\infty}^{\infty} X_c\left(\frac{\omega}{2\pi} F_s - kF_s\right)$$

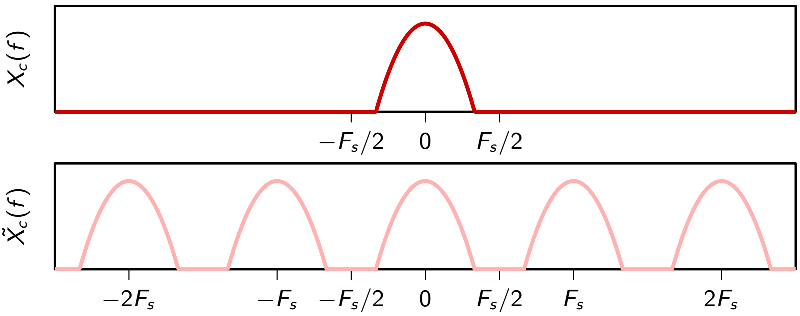
Example: signal bandlimited to f_0 and $F_s > 2f_0$



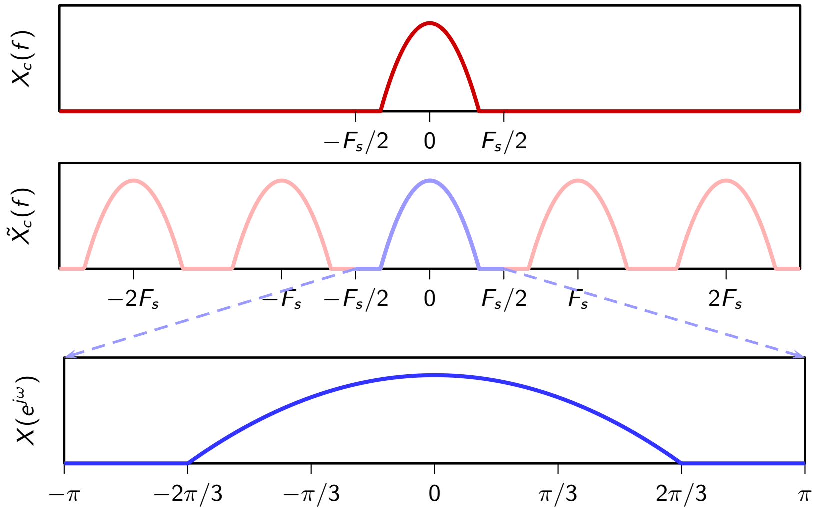
Example: signal bandlimited to f_0 and $F_s > 2f_0$



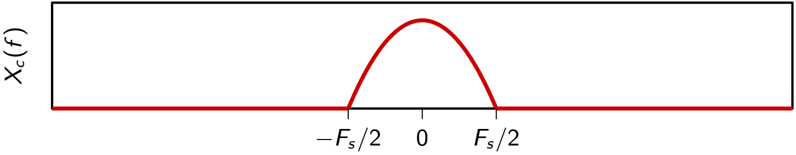
Example: signal bandlimited to f_0 and $F_s > 2f_0$



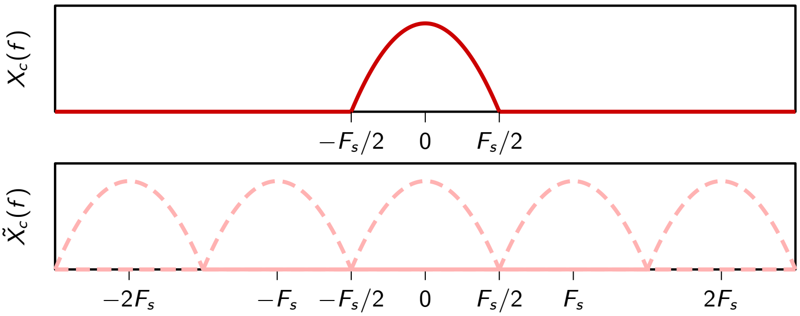
Example: signal bandlimited to f_0 and $F_s > 2f_0$



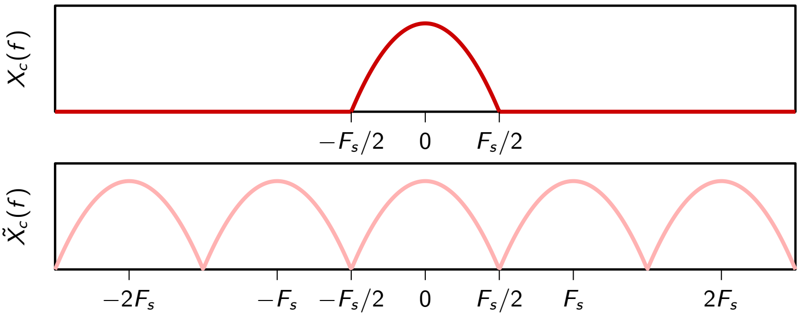
Example: signal bandlimited to f_0 and $F_s = 2f_0$



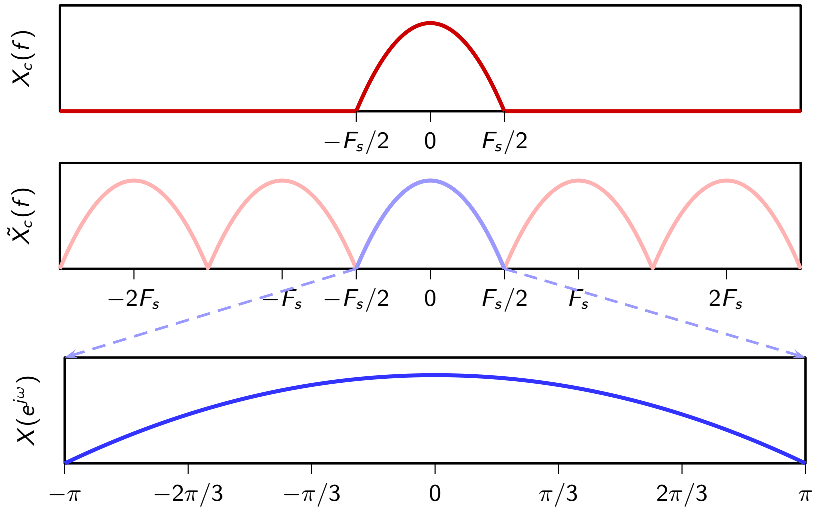
Example: signal bandlimited to f_0 and $F_s = 2f_0$



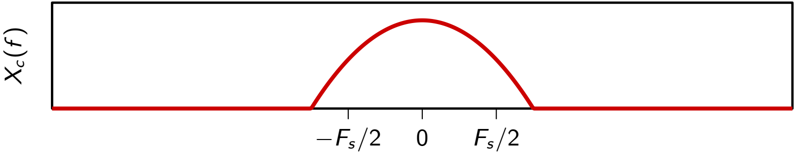
Example: signal bandlimited to f_0 and $F_s = 2f_0$



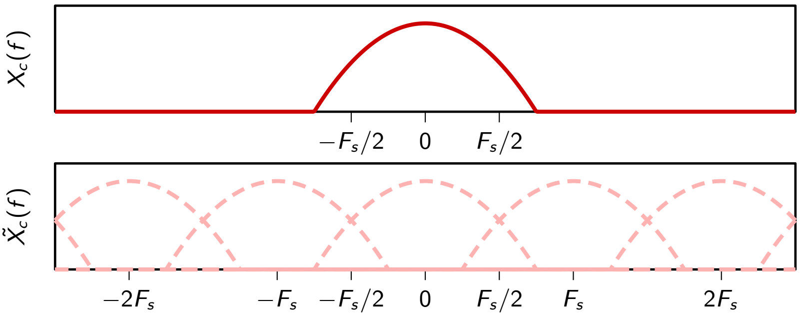
Example: signal bandlimited to f_0 and $F_s = 2f_0$



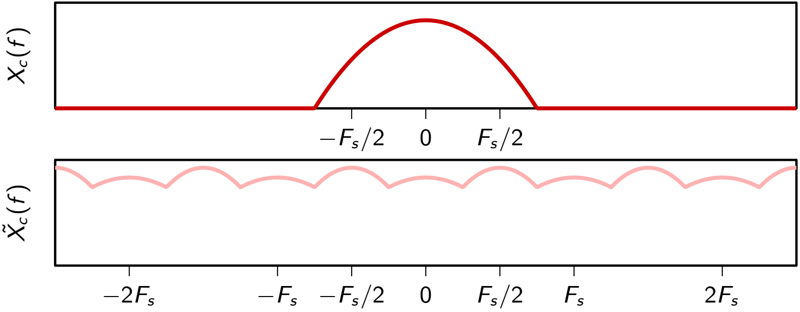
Example: signal bandlimited to f_0 and $F_s < 2f_0$



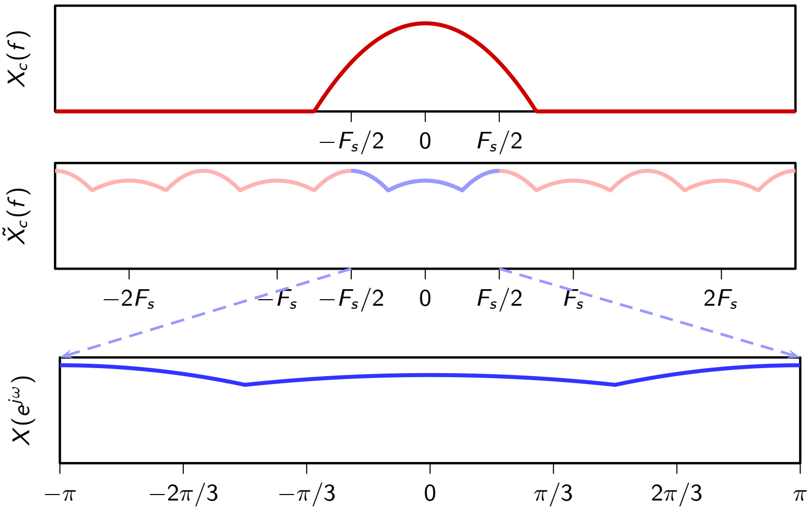
Example: signal bandlimited to f_0 and $F_s < 2f_0$



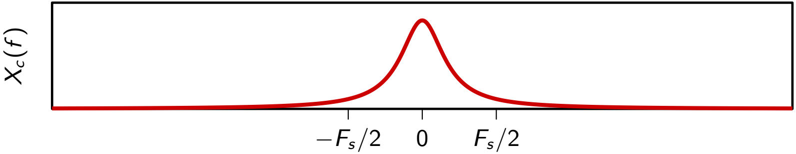
Example: signal bandlimited to f_0 and $F_s < 2f_0$



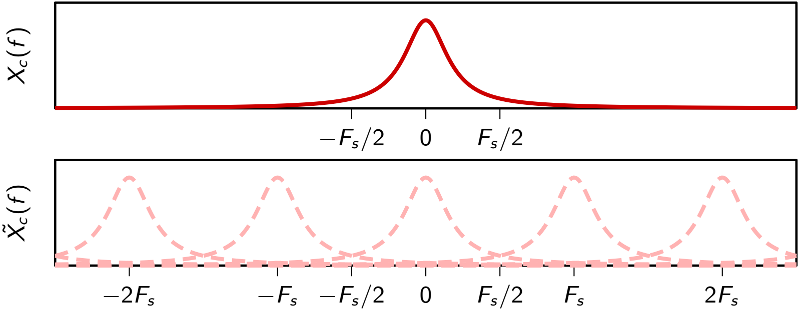
Example: signal bandlimited to f_0 and $F_s < 2f_0$



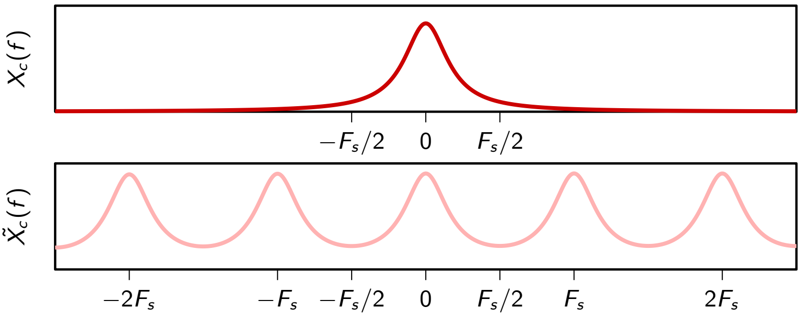
Example: non-bandlimited signal



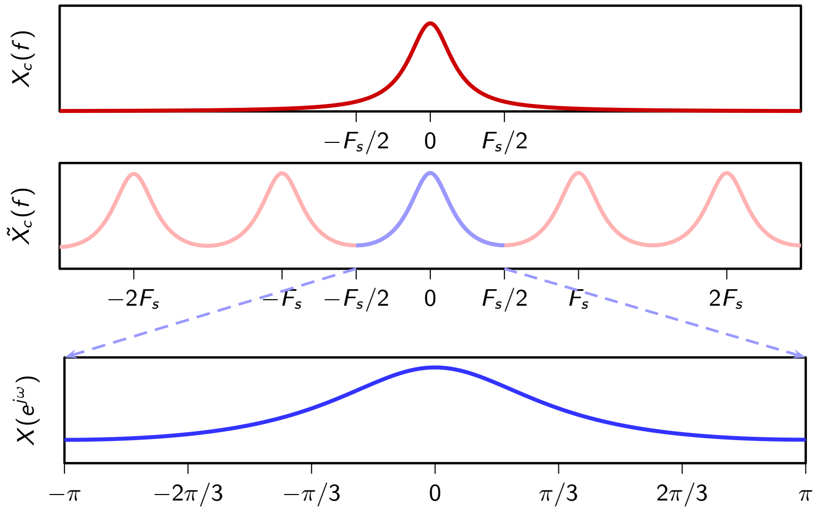
Example: non-bandlimited signal



Example: non-bandlimited signal



Example: non-bandlimited signal



Sampling strategies

given a sampling frequency F_s

- if the signal is bandlimited to $F_s/2$ or less, raw sampling is fine (i.e. equivalent to sinc sampling up to a scaling factor T_s)
- if the signal is not bandlimited, two choices:
  - bandlimit via a lowpass filter *in the continuous-time domain* before sampling (i.e. sinc sampling)
  - or, raw sample the signal and incur aliasing
- aliasing introduces errors we cannot control, so the sensible choice is to bandlimit in continuous time
- bandlimiting is also optimal wrt least squares approximation!

Sampling strategies

given a sampling frequency F_s

- if the signal is bandlimited to $F_s/2$ or less, raw sampling is fine (i.e. equivalent to sinc sampling up to a scaling factor T_s)
- if the signal is not bandlimited, two choices:
  - bandlimit via a lowpass filter *in the continuous-time domain* before sampling (i.e. sinc sampling)
  - or, raw sample the signal and incur aliasing
- aliasing introduces errors we cannot control, so the sensible choice is to bandlimit in continuous time
- bandlimiting is also optimal wrt least squares approximation!

Sampling strategies

given a sampling frequency F_s

- if the signal is bandlimited to $F_s/2$ or less, raw sampling is fine (i.e. equivalent to sinc sampling up to a scaling factor T_s)
- if the signal is not bandlimited, two choices:
  - bandlimit via a lowpass filter *in the continuous-time domain* before sampling (i.e. sinc sampling)
  - or, raw sample the signal and incur aliasing
- aliasing introduces errors we cannot control, so the sensible choice is to bandlimit in continuous time
- bandlimiting is also optimal wrt least squares approximation!

Sampling strategies

given a sampling frequency F_s

- if the signal is bandlimited to $F_s/2$ or less, raw sampling is fine (i.e. equivalent to sinc sampling up to a scaling factor T_s)
- if the signal is not bandlimited, two choices:
  - bandlimit via a lowpass filter *in the continuous-time domain* before sampling (i.e. sinc sampling)
  - or, raw sample the signal and incur aliasing
- aliasing introduces errors we cannot control, so the sensible choice is to bandlimit in continuous time
- bandlimiting is also optimal wrt least squares approximation!

Sampling strategies

given a sampling frequency F_s

- if the signal is bandlimited to $F_s/2$ or less, raw sampling is fine (i.e. equivalent to sinc sampling up to a scaling factor T_s)
- if the signal is not bandlimited, two choices:
  - bandlimit via a lowpass filter *in the continuous-time domain* before sampling (i.e. sinc sampling)
  - or, raw sample the signal and incur aliasing
- aliasing introduces errors we cannot control, so the sensible choice is to bandlimit in continuous time
- bandlimiting is also optimal wrt least squares approximation!

Sampling strategies

given a sampling frequency F_s

- if the signal is bandlimited to $F_s/2$ or less, raw sampling is fine (i.e. equivalent to sinc sampling up to a scaling factor T_s)
- if the signal is not bandlimited, two choices:
  - bandlimit via a lowpass filter *in the continuous-time domain* before sampling (i.e. sinc sampling)
  - or, raw sample the signal and incur aliasing
- aliasing introduces errors we cannot control, so the sensible choice is to bandlimit in continuous time
- bandlimiting is also optimal wrt least squares approximation!

Sinc Sampling and Interpolation

$$\hat{x}[n] = \left\langle \operatorname{sinc}\left(\frac{t - nT_s}{T_s}\right), x_c(t) \right\rangle \;=\; (\operatorname{sinc}_{T_s} * x_c)(nT_s)$$

Sinc Sampling and Interpolation

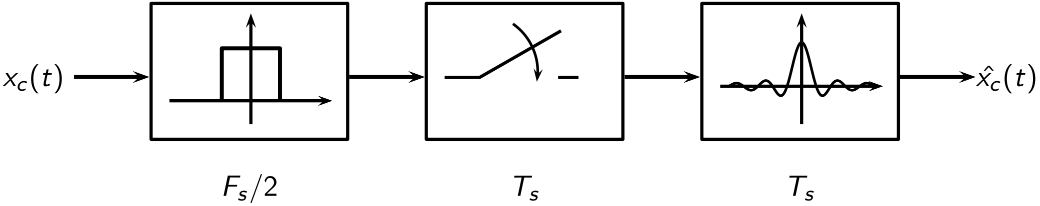
$$\hat{x}[n] = \left\langle \operatorname{sinc}\left(\frac{t - nT_s}{T_s}\right), x_c(t) \right\rangle \; = \; (\operatorname{sinc}_{T_s} * x_c)(nT_s)$$

$$\hat{x}_c(t) = \sum_n x[n] \operatorname{sinc}\left(\frac{t - nT_s}{T_s}\right)$$

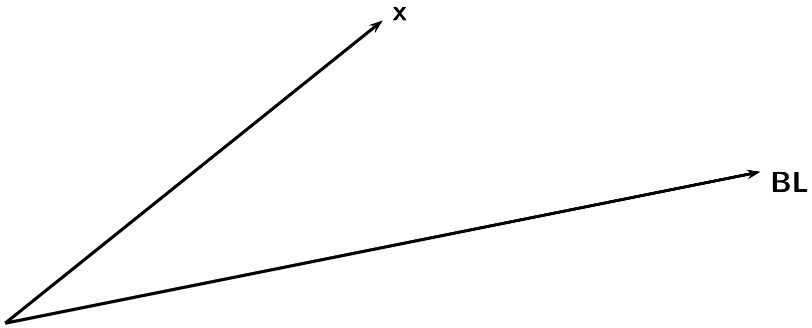
Sinc Sampling and Interpolation

$$\hat{x}[n] = \left\langle \operatorname{sinc}\left(\frac{t - nT_s}{T_s}\right), x_c(t) \right\rangle = (\operatorname{sinc}_{T_s} * x_c)(nT_s)$$

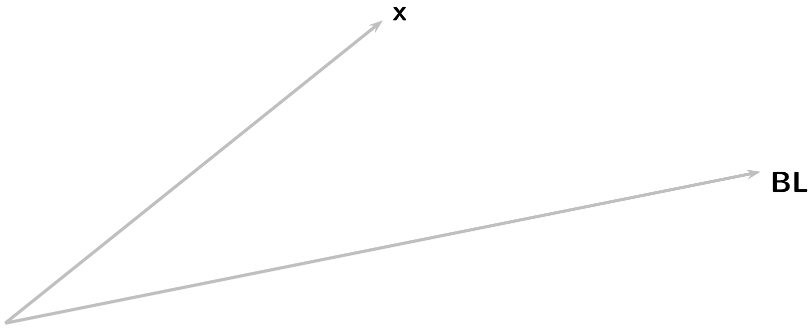
$$\hat{x}_c(t) = \sum_n x[n] \operatorname{sinc}\left(\frac{t - nT_s}{T_s}\right)$$



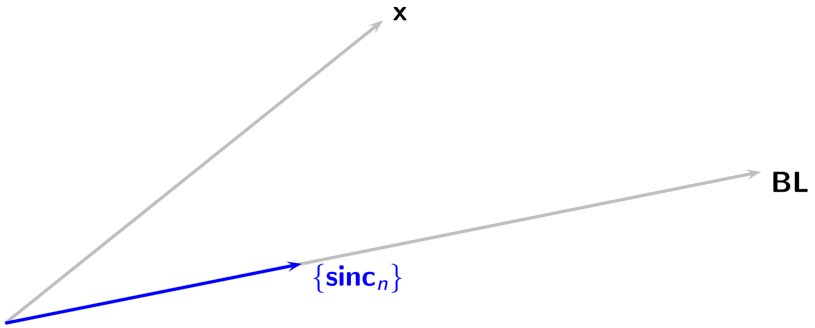
Least squares approximation with sinc sampling and interpolation



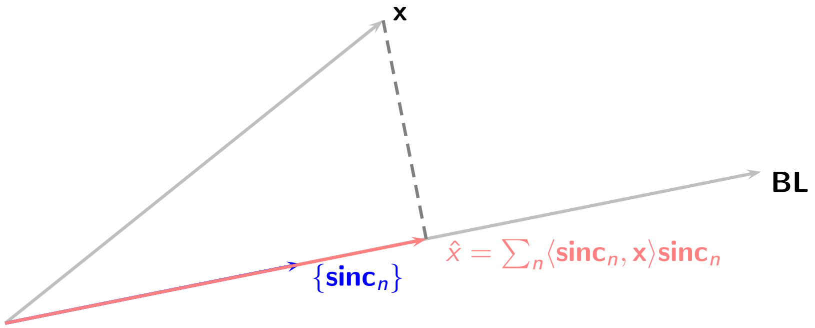
Least squares approximation with sinc sampling and interpolation



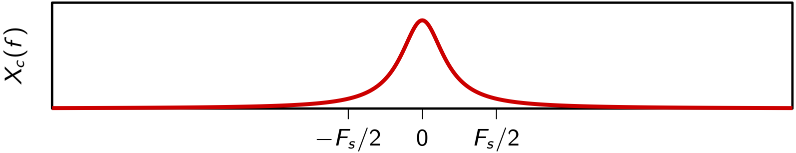
Least squares approximation with sinc sampling and interpolation



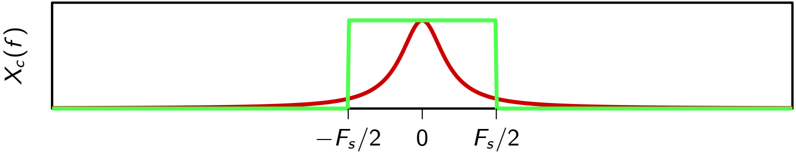
Least squares approximation with sinc sampling and interpolation



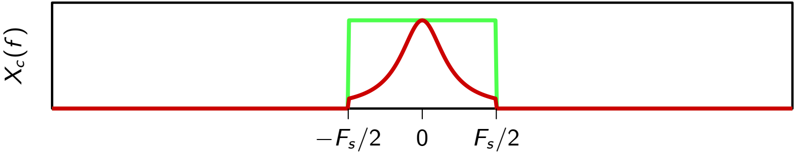
Least squares approximation with sinc sampling and interpolation



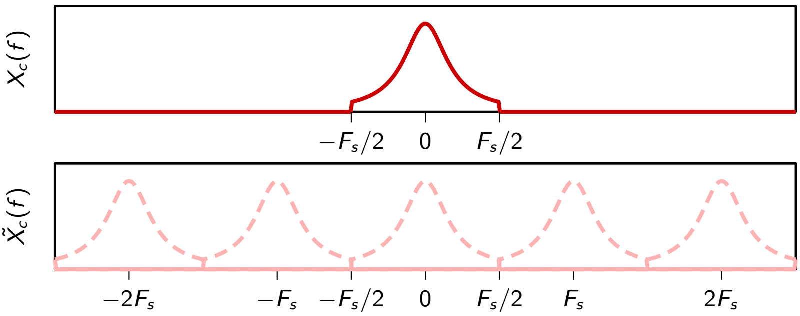
Least squares approximation with sinc sampling and interpolation



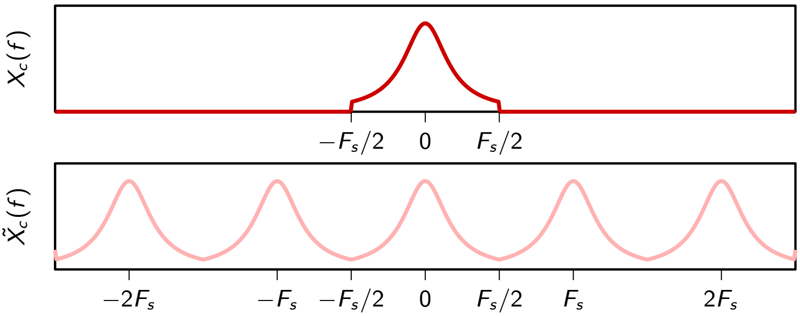
Least squares approximation with sinc sampling and interpolation



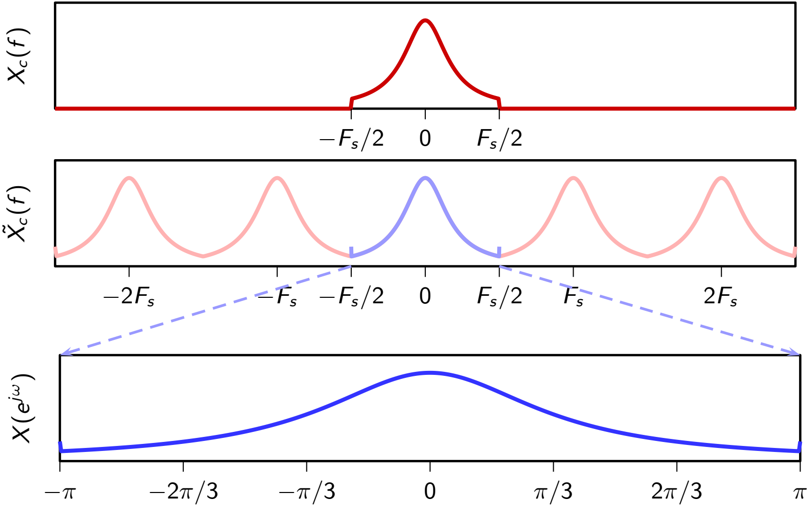
Least squares approximation with sinc sampling and interpolation



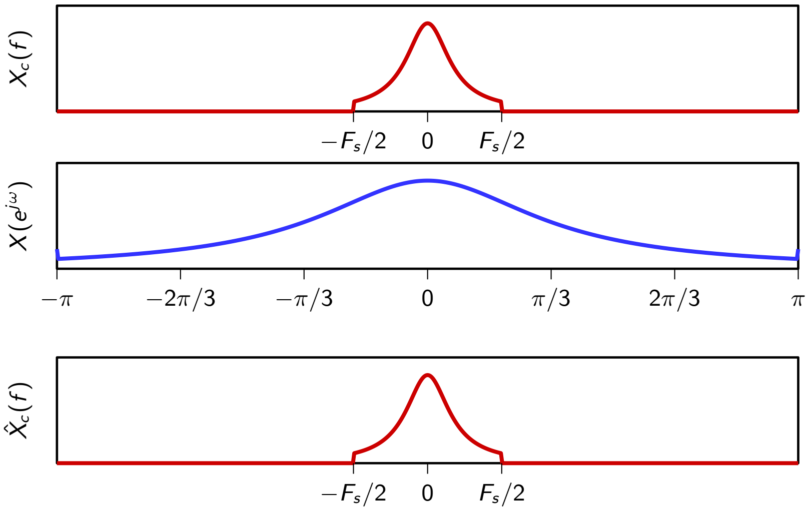
Least squares approximation with sinc sampling and interpolation



Least squares approximation with sinc sampling and interpolation



Least squares approximation with sinc sampling and interpolation



Sinc Sampling and Interpolation

$$\hat{x}[n] = \left\langle \text{sinc}\left(\frac{t - nT_s}{T_s}\right), x_c(t) \right\rangle \; = \; (\text{sinc}_{T_s} * x_c)(nT_s)$$

$$X(e^{j\omega}) = X_c\left(\frac{\omega}{2\pi} F_s\right) \quad \text{prolonged by periodicity outside of } [-\pi, \pi]$$

Alternatively:

$$X(e^{j\omega}) = X_c\left(\left(\frac{\omega}{\pi} - 2\left\lfloor \frac{\omega - \pi}{2\pi} \right\rfloor - 2\right) F_s\right)$$

processing of analog signals

Overview:

- Impulse invariance
- Duality
- Examples

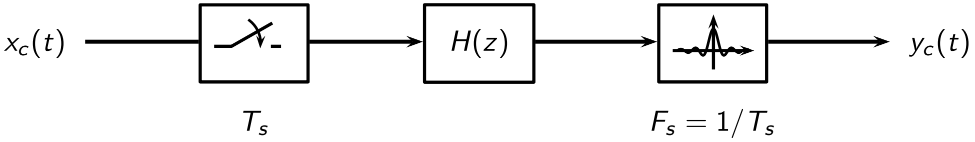
Overview:

- Impulse invariance
- Duality
- Examples

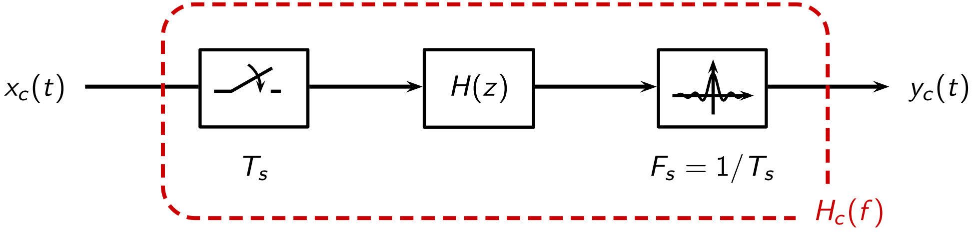
Overview:

- Impulse invariance
- Duality
- Examples

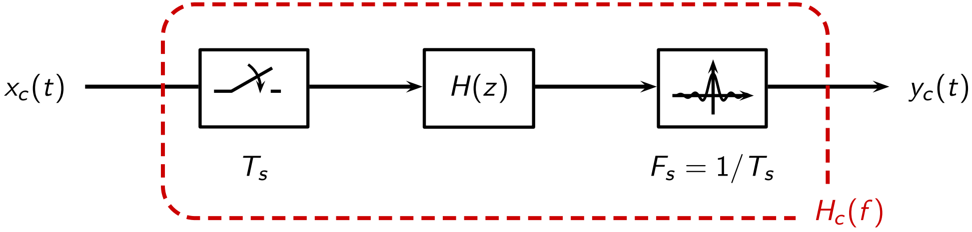
Basic setup



Basic setup

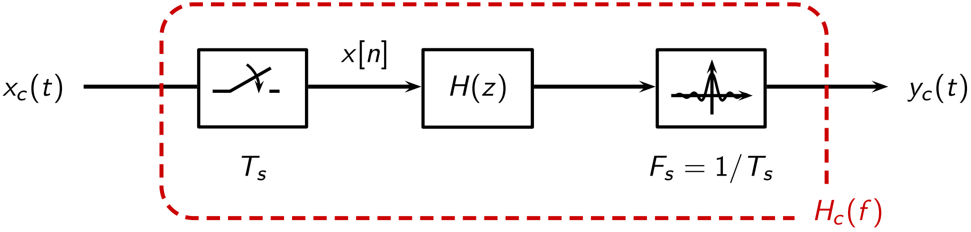


Basic setup



assume $x_c(t)$ is F_s-bandlimited and $T_s = 1/F_s$:

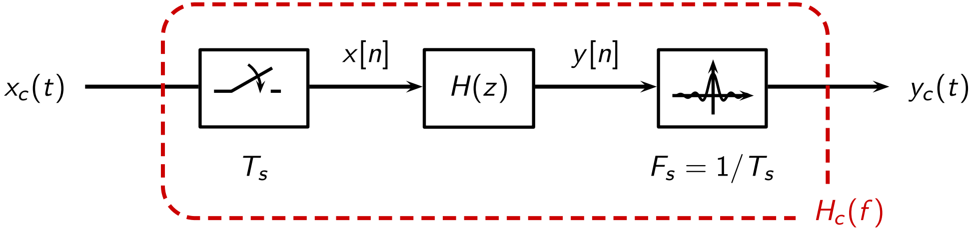
Basic setup



assume $x_c(t)$ is F_s-bandlimited and $T_s = 1/F_s$:

- $X(e^{j\omega}) = F_s X_c\left(\frac{\omega}{2\pi} F_s\right)$

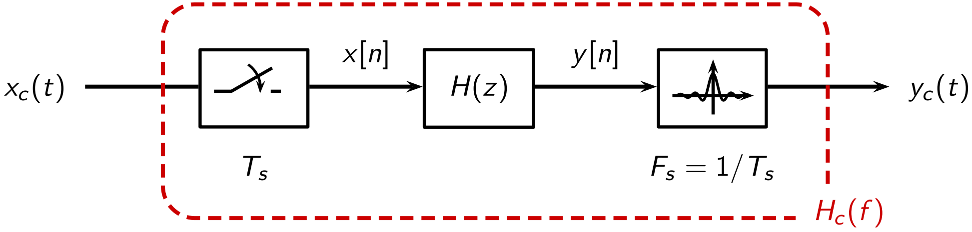
Basic setup



assume $x_c(t)$ is F_s-bandlimited and $T_s = 1/F_s$:

- $X(e^{j\omega}) = F_s X_c\left(\frac{\omega}{2\pi} F_s\right)$
- $Y(e^{j\omega}) = X(e^{j\omega})\, H(e^{j\omega})$

Basic setup



assume $x_c(t)$ is F_s-bandlimited and $T_s = 1/F_s$:

- $X(e^{j\omega}) = F_s X_c\left(\frac{\omega}{2\pi} F_s\right)$
- $Y(e^{j\omega}) = X(e^{j\omega})\, H(e^{j\omega})$
- $Y_c(f) = (1/F_s)\, Y(e^{j2\pi f/F_s})$

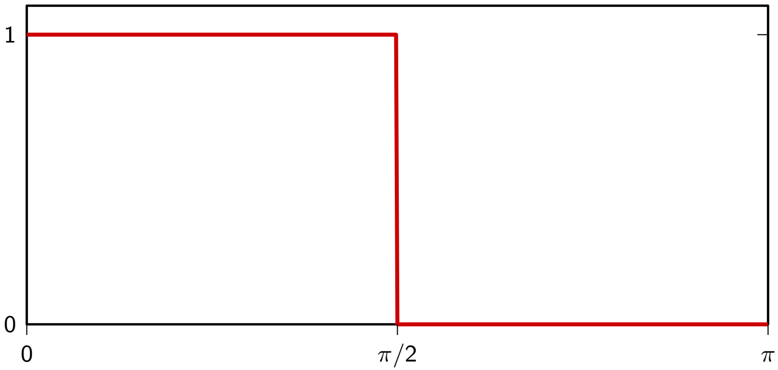
Impulse invariance

$$Y_c(f) = X_c(f)\, H(e^{j2\pi f/F_s})$$

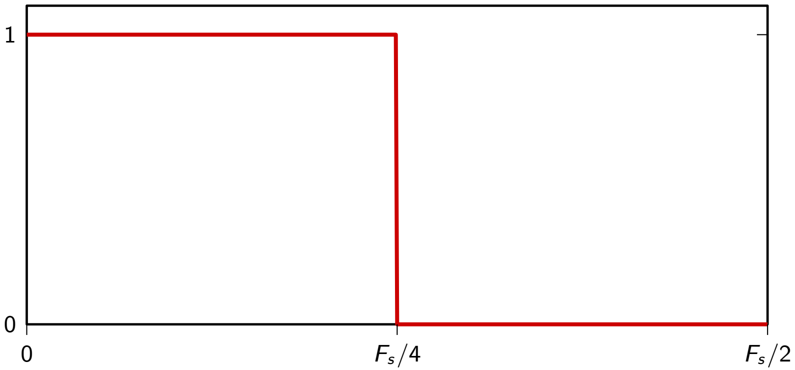
Impulse invariance

$$H_c(f) = H(e^{j2\pi f/F_s})$$

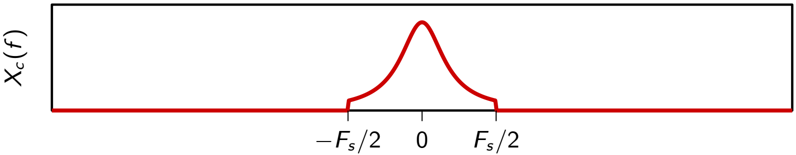
Impulse invariance



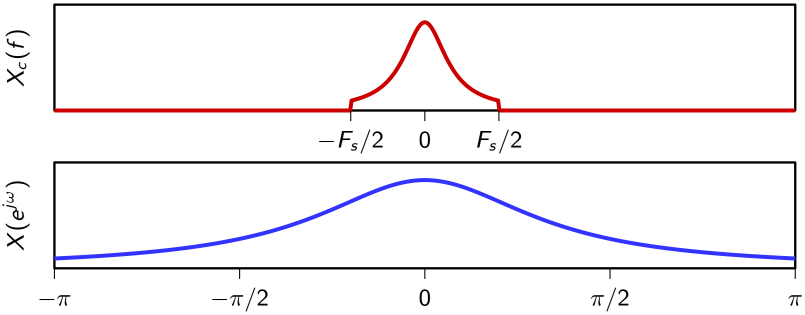
Impulse invariance



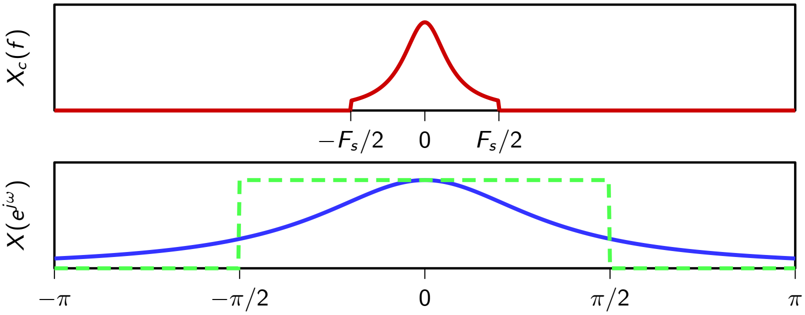
DT processing of CT signals



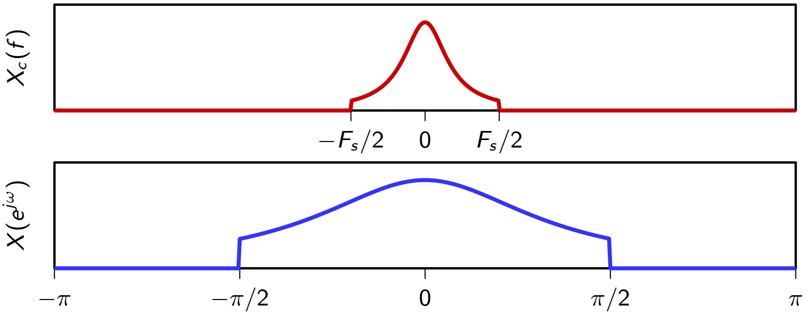
DT processing of CT signals



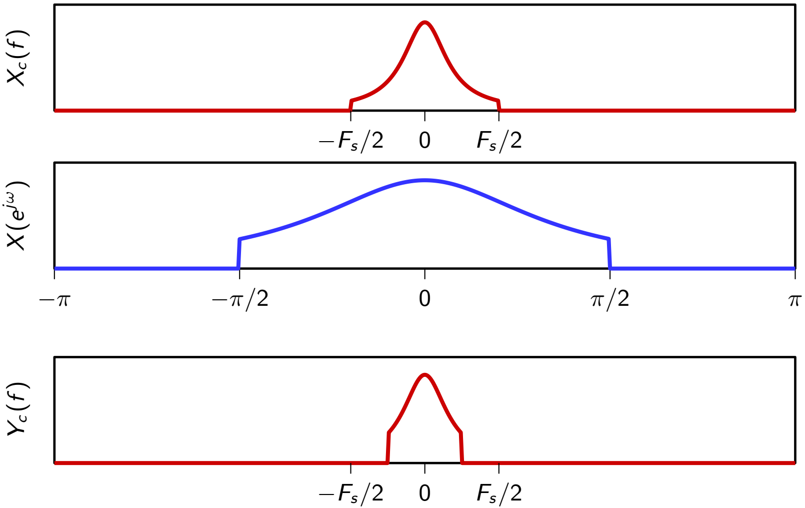
DT processing of CT signals



DT processing of CT signals



DT processing of CT signals



Example

quickly design a discrete-time filter to isolate a band of frequencies between 4000 and 5000Hz; input signals are bandlimited to 7KHz.

Example

- 7KHz band limit $\Rightarrow$ we can use any sampling frequency above 14KHz
- pick $F_s = 16KHz$
- we need a bandpass with a 1000Hz bandwidth
- start with some lowpass with cutoff $f_c = 500$Hz
- modulate it to center it on $f_0 = 4500$Hz

Example

- 7KHz band limit $\Rightarrow$ we can use any sampling frequency above 14KHz
- pick $F_s = 16KHz$
- we need a bandpass with a 1000Hz bandwidth
- start with some lowpass with cutoff $f_c = 500$Hz
- modulate it to center it on $f_0 = 4500$Hz

Example

- 7KHz band limit $\Rightarrow$ we can use any sampling frequency above 14KHz
- pick $F_s = 16KHz$
- we need a bandpass with a 1000Hz bandwidth
- start with some lowpass with cutoff $f_c = 500$Hz
- modulate it to center it on $f_0 = 4500$Hz

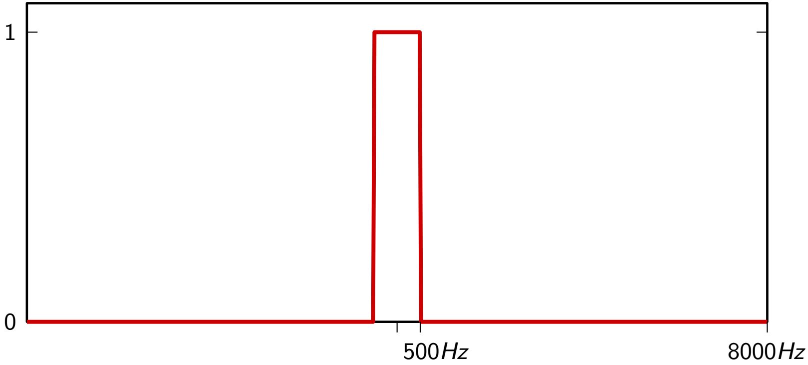
Example

- 7KHz band limit $\Rightarrow$ we can use any sampling frequency above 14KHz
- pick $F_s = 16KHz$
- we need a bandpass with a 1000Hz bandwidth
- start with some lowpass with cutoff $f_c = 500$Hz
- modulate it to center it on $f_0 = 4500$Hz

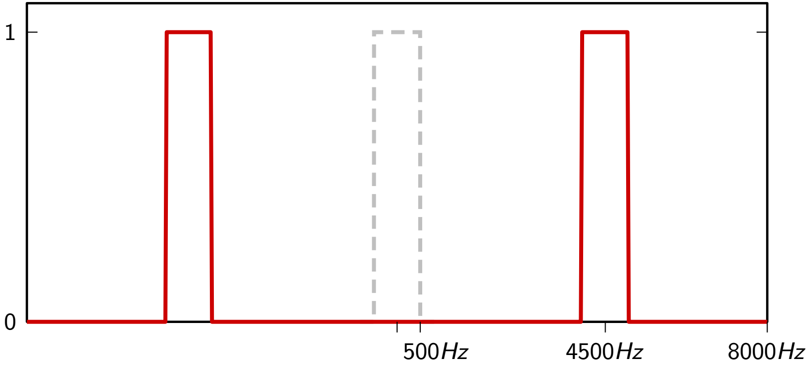
Example

- 7KHz band limit $\Rightarrow$ we can use any sampling frequency above 14KHz
- pick $F_s = 16KHz$
- we need a bandpass with a 1000Hz bandwidth
- start with some lowpass with cutoff $f_c = 500$Hz
- modulate it to center it on $f_0 = 4500$Hz

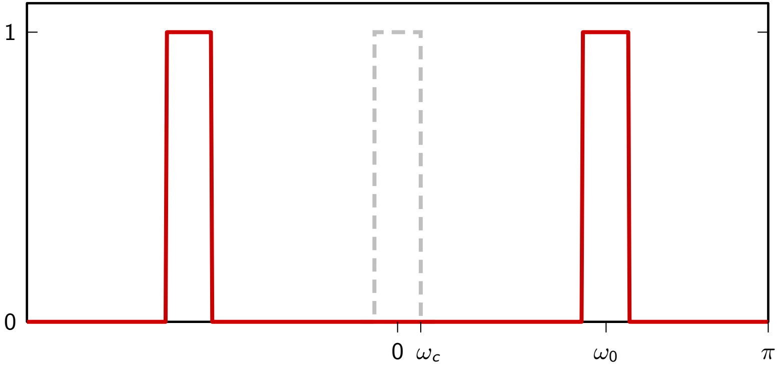
Impulse invariance



Impulse invariance



Impulse invariance



Example

- $\omega_c = 2\pi \frac{f_c}{F_s} = 2\pi \frac{500}{16000} = 0.0625\pi$
- $\omega_0 = 2\pi \frac{4500}{16000} = 0.5625\pi$
- modulate the impulse response of an ideal lowpass with cutoss ω_c by $2\cos\omega_0 n$
- truncate the impulse response with an appropriate window

Example

- $\omega_c = 2\pi \frac{f_c}{F_s} = 2\pi \frac{500}{16000} = 0.0625\pi$
- $\omega_0 = 2\pi \frac{4500}{16000} = 0.5625\pi$
- modulate the impulse response of an ideal lowpass with cutoss ω_c by $2 \cos \omega_0 n$
- truncate the impulse response with an appropriate window

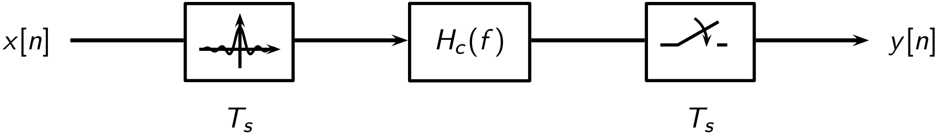
Example

- $\omega_c = 2\pi \frac{f_c}{F_s} = 2\pi \frac{500}{16000} = 0.0625\pi$
- $\omega_0 = 2\pi \frac{4500}{16000} = 0.5625\pi$
- modulate the impulse response of an ideal lowpass with cutoss ω_c by $2\cos\omega_0 n$
- truncate the impulse response with an appropriate window

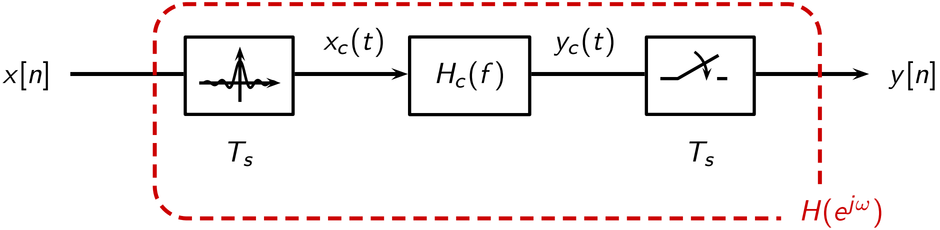
Example

- $\omega_c = 2\pi \frac{f_c}{F_s} = 2\pi \frac{500}{16000} = 0.0625\pi$
- $\omega_0 = 2\pi \frac{4500}{16000} = 0.5625\pi$
- modulate the impulse response of an ideal lowpass with cutoss ω_c by $2 \cos \omega_0 n$
- truncate the impulse response with an appropriate window

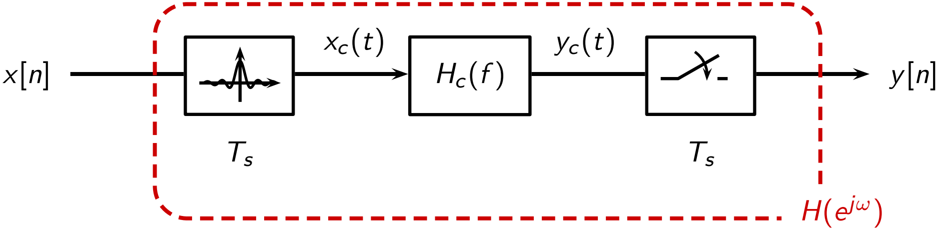
Duality



Duality

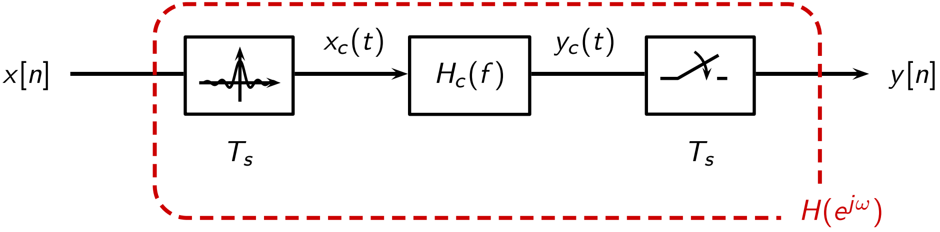


Duality



we can pick any T_s so pick $T_s = 1$:

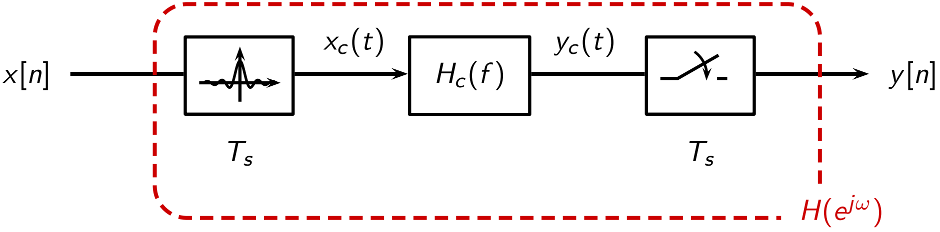
Duality



we can pick any T_s so pick $T_s = 1$:

- $X_c(f) = X(e^{j2\pi f})$

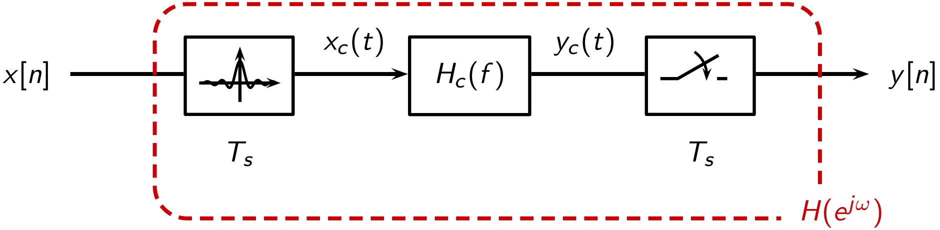
Duality



we can pick any T_s so pick $T_s = 1$:

- $X_c(f) = X(e^{j2\pi f})$
- $Y_c(f) = X_c(f)H_c(f)$

Duality



we can pick any T_s so pick $T_s = 1$:

- $X_c(f) = X(e^{j2\pi f})$
- $Y_c(f) = X_c(f)H_c(f)$
- LTI systems cannot change the bandwidth $\Rightarrow Y(e^{j\omega}) = Y_c(\frac{\omega}{2\pi})$

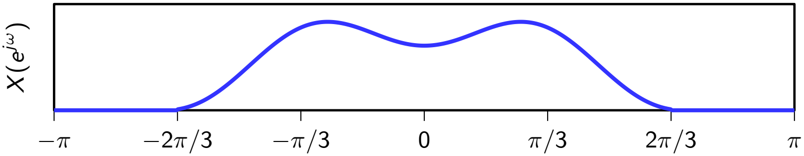
Duality

$$Y(e^{j\omega}) = X(e^{j\omega})\, H_c\left(\frac{\omega}{2\pi}\right)$$

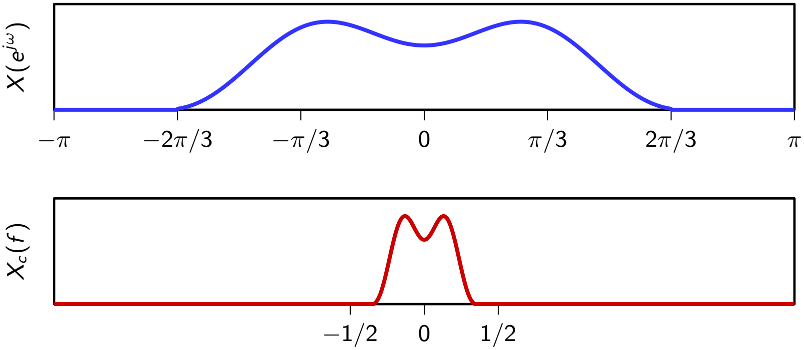
Duality

$$H(e^{j\omega}) = H_c\left(\frac{\omega}{2\pi}\right)$$

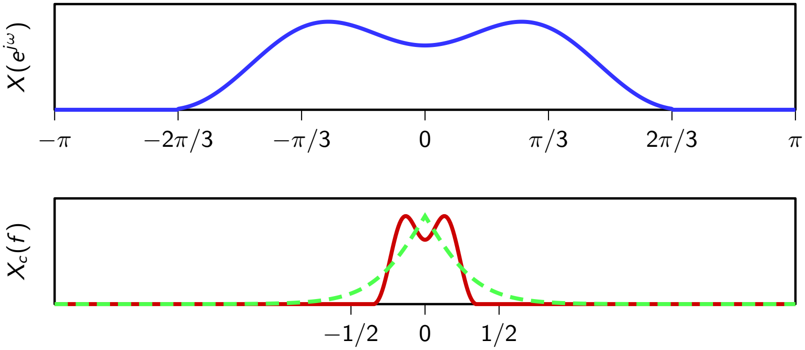
CT processing of DT signals



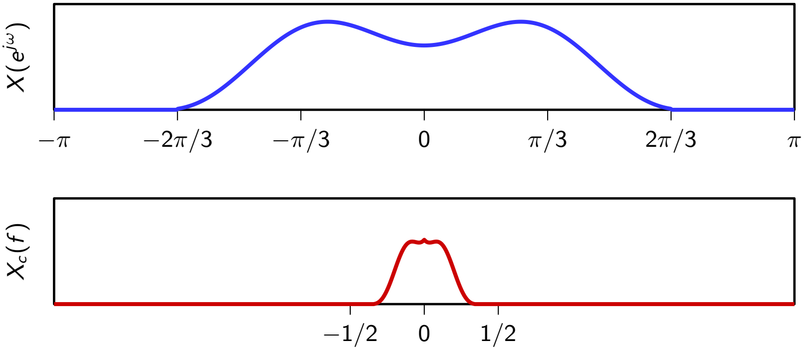
CT processing of DT signals



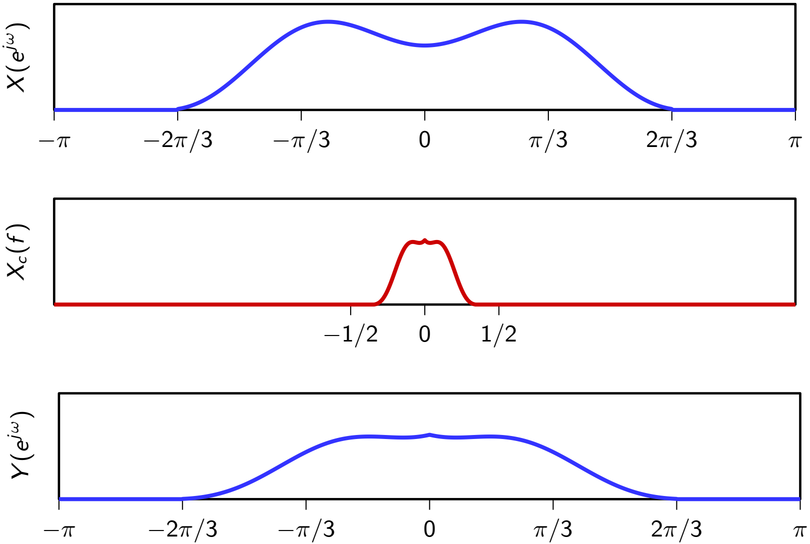
CT processing of DT signals



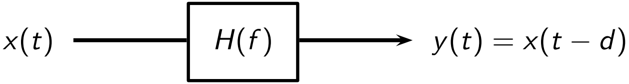
CT processing of DT signals



CT processing of DT signals



Delays in continuous time

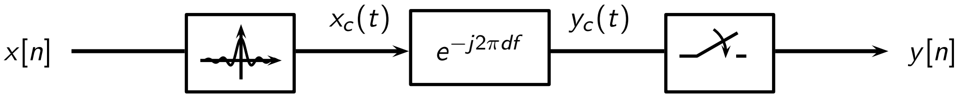


Delays in continuous time

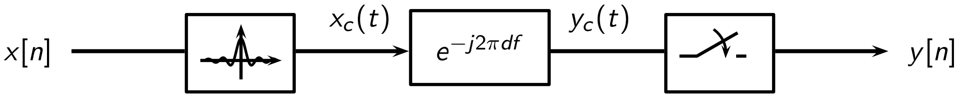
$$x(t) \longrightarrow \boxed{H(f)} \longrightarrow y(t) = x(t-d)$$

- in continuous time, delays are well defined for all $d \in \mathbb{R}$
- $H(f) = e^{-j2\pi df}$

Interpretation of fractional delay by duality

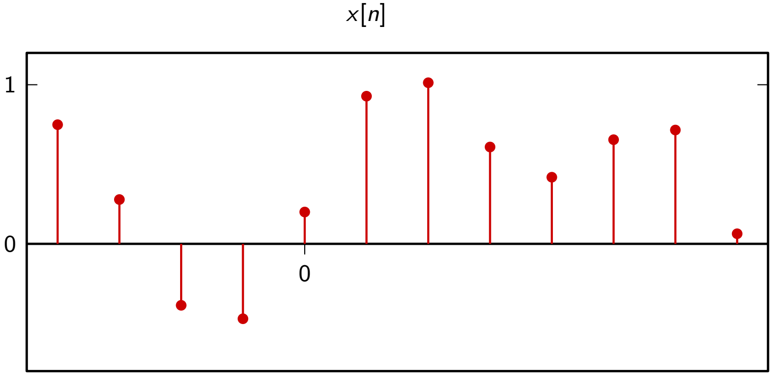


Interpretation of fractional delay by duality

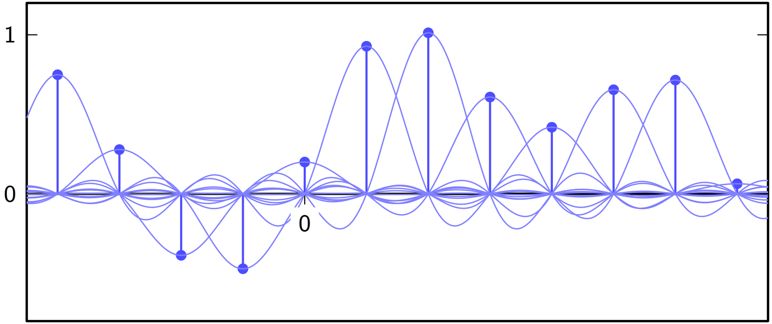


- chain interpolates $x[n]$, delays the result by d and resamples
- equivalent filter $H(e^{j\omega}) = H_c(\omega/(2\pi)) = e^{-j\omega d}$
- that's how a discrete-time fractional delay works internally

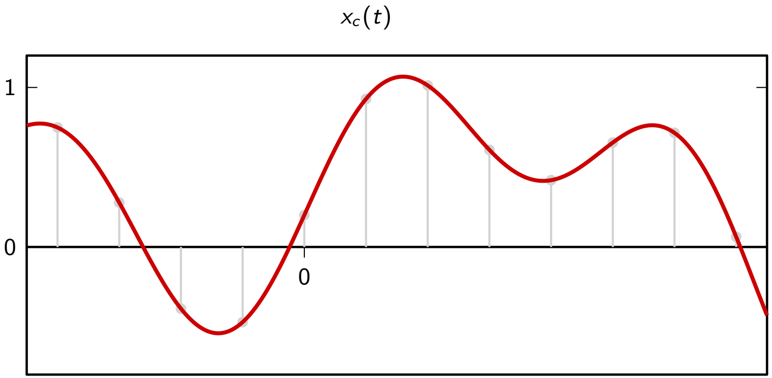
Example: fractional delay



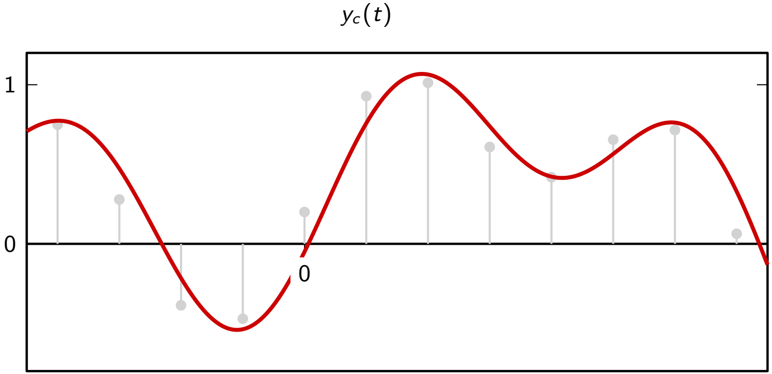
Example: fractional delay



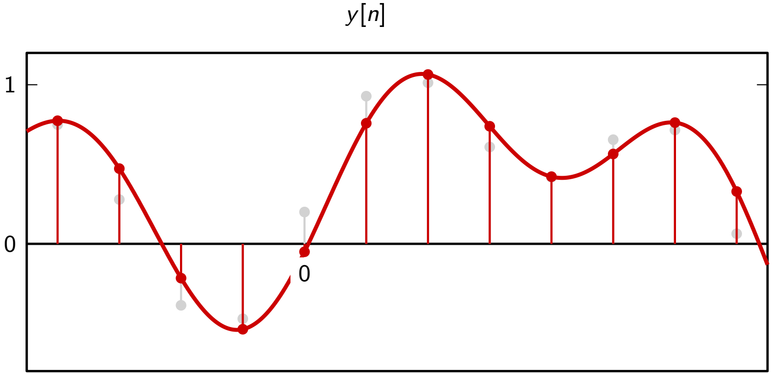
Example: fractional delay



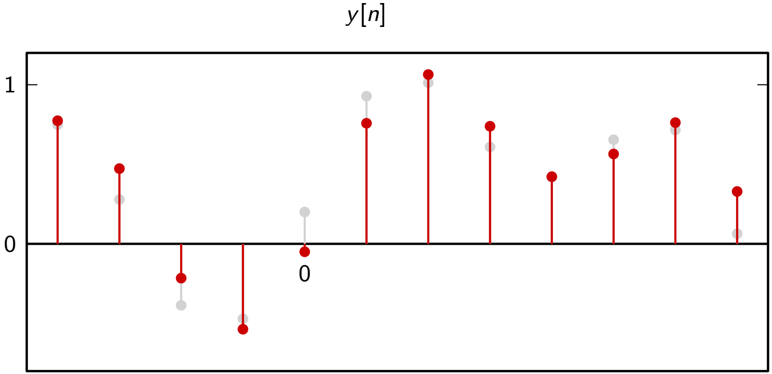
Example: fractional delay



Example: fractional delay



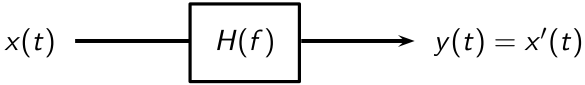
Example: fractional delay



Example: fractional delay

- $h[n] = \text{sinc}(n - d)$
- to delay a discrete-time signal by a fraction of a sample we need an ideal filter!
- efficient approximations exist (e.g. cubic local interpolation)

Differentiation in continuous time

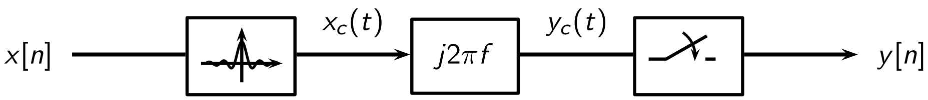


Differentiation in continuous time

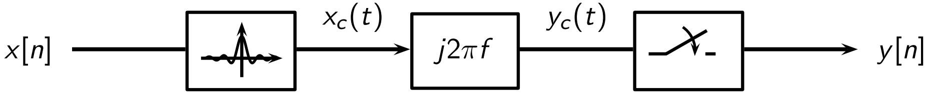
$$x(t) \longrightarrow \boxed{H(f)} \longrightarrow y(t) = x'(t)$$

- easy to show that FT $\{x'(t)\} = j2\pi f\, X(f)$
- $H(f) = j2\pi f$

By duality



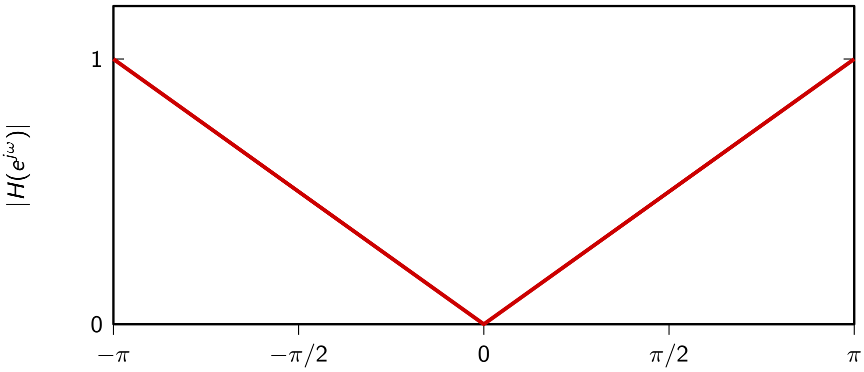
By duality



- chain interpolates $x[n]$, differentiates the result by d and resamples
- equivalent filter $H(e^{j\omega}) = H_c(\omega/(2\pi)) = j\omega$
- equivalent filter defines a digital differentiator

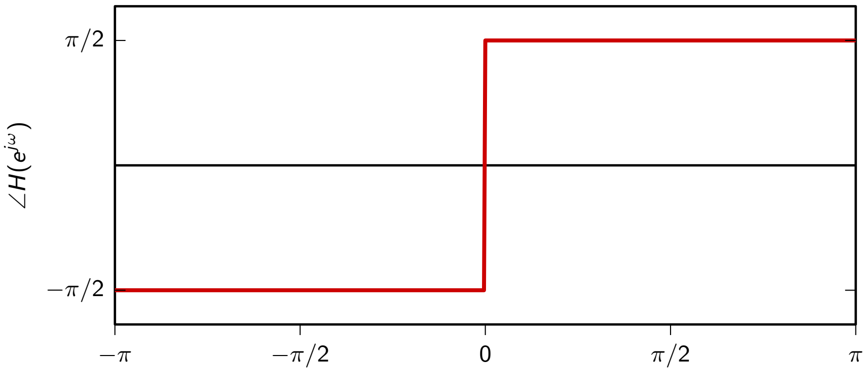
Digital differentiator, magnitude response

$$H(e^{j\omega}) = j\omega$$



Digital differentiator, phase response

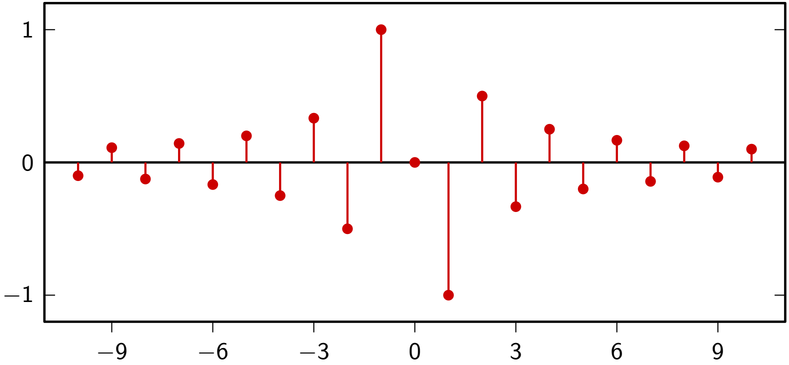
$$H(e^{j\omega}) = j\omega$$



Digital differentiator, impulse response

$$
\begin{aligned}
h[n] &= \frac{1}{2\pi}\int_{-\pi}^{\pi} j\omega e^{j\omega n}\, d\omega \\
&= \ldots \textit{(integration by parts)} \ldots \\
&= \begin{cases} 0 & n = 0 \\ \dfrac{(-1)^n}{n} & n \neq 0 \end{cases}
\end{aligned}
$$

Digital differentiator, impulse response



Digital differentiator

- the digital differentiator is again an ideal filter!
- many approximations exist, with different properties

Wrap up

- Continuous-time processing of discrete-time sequences
- Discrete-time processing of continuous-time signals
- Jumping back and forth using sampling and interpolation
- In practice: Many applications of processing continuous-time signals in discrete time!