## Sampling and Discrete-Time Signals

- MATLAB, and other digital processing systems, can not process continuous-time signals.
- Instead, MATLAB requires the continuous-time signal to be converted into a discrete-time signal.
- The conversion process is called sampling.
- To sample a continuous-time signal, we evaluate it at a discrete set of times $t_n = nT_s$, where
  - $n$ is a integer,
  - $T_s$ is called the sampling period (time between samples),
  - $f_s = 1/T_s$ is the sampling rate (samples per second).

## Sampling and Discrete-Time Signals

- Sampling results in a sequence of samples

$$x(nT_s) = A \cdot \cos(2\pi f n T_s + \phi).$$

- Note that the independent variable is now $n$, not $t$.
- To emphasize that this is a discrete-time signal, we write

$$x[n] = A \cdot \cos(2\pi f n T_s + \phi).$$

- Sampling is a straightforward operation.
- We will see that the sampling rate $f_s$ must be chosen with care!

# Sampled Signals in MATLAB

- Note that we have worked with sampled signals whenever we have used MATLAB.
- For example, we use the following MATLAB fragment to generate a sinusoidal signal:

```
fs = 100;
tt = 0:1/fs:3;
xx = 5*cos(2*pi*2*tt + pi/4);
```

- The resulting signal `xx` is a discrete-time signal:
  - The vector `xx` contains the samples, and
  - the vector `tt` specifies the sampling instances: $0, 1/f_s, 2/f_s, \ldots, 3$.
- We will now turn our attention to the impact of the sampling rate $f_s$.

# Example: Three Sinuoids

- **Objective:** In MATLAB, compute sampled versions of three sinusoids:
  1. $x(t) = \cos(2\pi t + \pi/4)$
  2. $x(t) = \cos(2\pi 9t - \pi/4)$
  3. $x(t) = \cos(2\pi 11t + \pi/4)$
- The sampling rate for all three signals is $f_s = 10$.

# MATLAB code

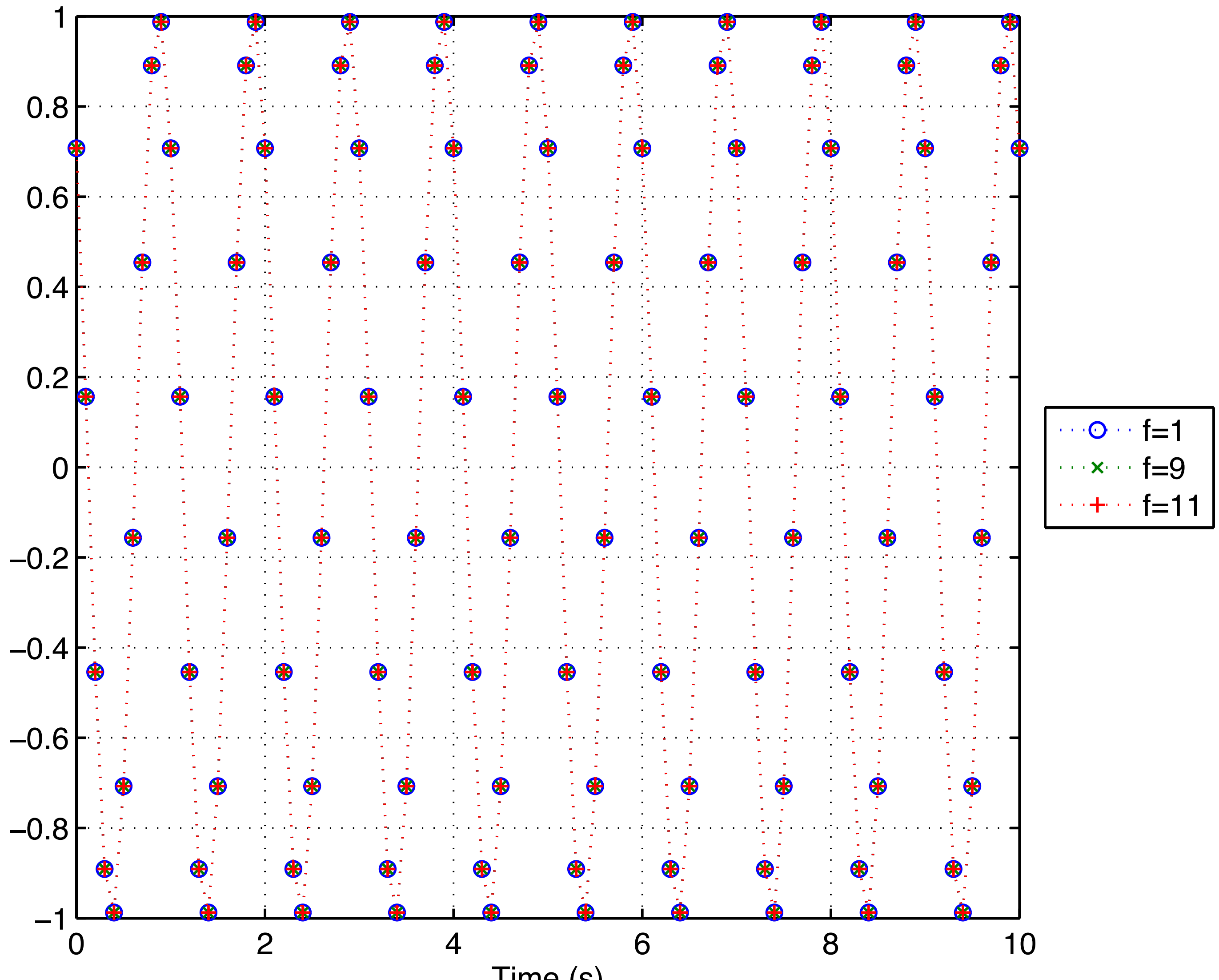
```
% plot_SamplingDemo - Sample three sinusoidal signals to
%                      demonstrate the impact of sampling

%% set parameters
fs  = 10;
dur = 10;

%% generate signals
tt  = 0:1/fs:dur;
xx1 = cos(2*pi*tt+pi/4);
xx2 = cos(2*pi*9*tt-pi/4);
xx3 = cos(2*pi*11*tt+pi/4);

%% plot
plot(tt,xx1,':o',tt,xx2,':x',tt,xx3,':+');
xlabel('Time␣(s)')
grid
legend('f=1','f=9','f=11','Location','EastOutside')
```

# Resulting Plot



# What happened?

- The samples for all three signals are identical: how is that possible?
- Is there a "bug" in the MATLAB code?
  - No, the code is correct.
- **Suspicion:** The problem is related to our choice of sampling rate.
  - To test this suspicion, repeat the experiment with a different sampling rate.
  - We also reduce the duration to keep the number of samples constant - that keeps the plots reasonable.

# MATLAB code

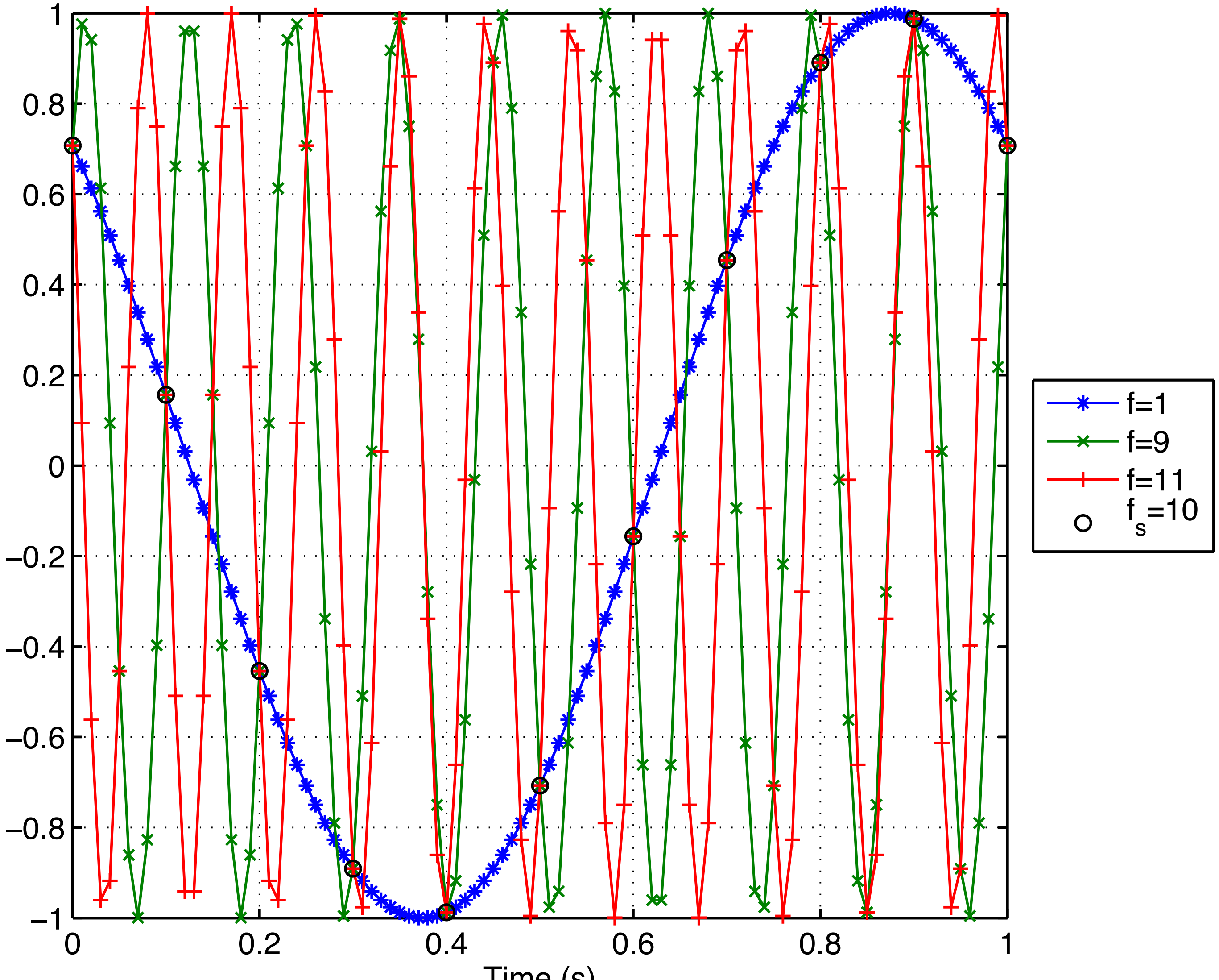
```
% plot_SamplingDemoHigh - Sample three sinusoidal signals to
%                         demonstrate the impact of sampling

%% set parameters
fs  = 100;
dur = 1;

%% generate signals
tt  = 0:1/fs:dur;
xx1 = cos(2*pi*tt+pi/4);
xx2 = cos(2*pi*9*tt-pi/4);
xx3 = cos(2*pi*11*tt+pi/4);

%% plots
plot(tt,xx1,'-*',tt,xx2,'-x',tt,xx3,'-+',...
    tt(1:10:end), xx1(1:10:end),'ok');
grid
xlabel('Time␣(s)')
legend('f=1','f=9','f=11','f_s=10','Location','EastOutside')
```

# Resulting Plot



# The Influence of the Sampling Rate

- Now the three sinusoids are clearly distinguishable and lead to different samples.
- Since the only parameter we changed is the sampling rate $f_s$, it must be responsible for the ambiguity in the first plot.
- Notice also that every 10-th sample (marked with a black circle) is identical for all three sinusoids.
  - Since the sampling rate was 10 times higher for the second plot, this explains the first plot.
- It is useful to investigate the effect of sampling mathematically, to understand better what impact it has.
  - To do so, we focus on sampling sinusoidal signals.

# Sampling a Sinusoidal Signal

- A continuous-time sinusoid is given by

$$x(t) = A\cos(2\pi f t + \phi).$$

- When this signal is sampled at rate $f_s$, we obtain the discrete-time signal

$$x[n] = A\cos(2\pi f n / f_s + \phi).$$

- It is useful to define the normalized frequency $\hat{f}_d = \frac{f}{f_s}$, so that

$$x[n] = A\cos(2\pi \hat{f}_d n + \phi).$$

# Three Cases

- We will distinguish between three cases:
  1. $0 \leq \hat{f}_d \leq 1/2$ (Oversampling, this is what we want!)
  2. $1/2 < \hat{f}_d \leq 1$ (Undersampling, folding)
  3. $1 < \hat{f}_d \leq 3/2$ (Undersampling, aliasing)
- This captures the three situations addressed by the first example:
  1. $f = 1, f_s = 10 \Rightarrow \hat{f}_d = 1/10$
  2. $f = 9, f_s = 10 \Rightarrow \hat{f}_d = 9/10$
  3. $f = 11, f_s = 10 \Rightarrow \hat{f}_d = 11/10$
- We will see that all three cases lead to identical samples.

# Oversampling

- When the sampling rate is such that $0 \leq \hat{f}_d \leq 1/2$, then the samples of the sinusoidal signal are given by

$$x[n] = A\cos(2\pi\hat{f}_d n + \phi).$$

- This cannot be simplified further.
- It provides our base-line.
- Oversampling is the desired behaviour!

# Undersampling, Aliasing

- When the sampling rate is such that $1 < \hat{f}_d \leq 3/2$, then we define the apparent frequency $\hat{f}_a = \hat{f}_d - 1$.
- Notice that $0 < \hat{f}_a \leq 1/2$ and $\hat{f}_d = \hat{f}_a + 1$.
  - For $f = 11$, $f_s = 10 \Rightarrow \hat{f}_d = 11/10 \Rightarrow \hat{f}_a = 1/10$.
- The samples of the sinusoidal signal are given by

$$x[n] = A\cos(2\pi\hat{f}_d n + \phi) = A\cos(2\pi(1 + \hat{f}_a)n + \phi).$$

- Expanding the terms inside the cosine,

$$x[n] = A\cos(2\pi\hat{f}_a n + 2\pi n + \phi) = A\cos(2\pi\hat{f}_a n + \phi)$$

- **Interpretation:** The samples are identical to those from a sinusoid with frequency $f = \hat{f}_a \cdot f_s$ and phase $\phi$.

# Undersampling, Folding

- When the sampling rate is such that $1/2 < \hat{f}_d \leq 1$, then we introduce the apparent frequency $\hat{f}_a = 1 - \hat{f}_d$; again $0 < \hat{f}_a \leq 1/2$; also $\hat{f}_d = 1 - \hat{f}_a$.
  - For $f = 9$, $f_s = 10 \Rightarrow \hat{f}_d = 9/10 \Rightarrow \hat{f}_a = 1/10$.
- The samples of the sinusoidal signal are given by

$$x[n] = A\cos(2\pi\hat{f}_d n + \phi) = A\cos(2\pi(1 - \hat{f}_a)n + \phi).$$

- Expanding the terms inside the cosine,

$$x[n] = A\cos(-2\pi\hat{f}_a n + 2\pi n + \phi) = A\cos(-2\pi\hat{f}_a n + \phi)$$

- Because of the symmetry of the cosine, this equals

$$x[n] = A\cos(2\pi\hat{f}_a n - \phi).$$

- **Interpretation:** The samples are identical to those from a sinusoid with frequency $f = \hat{f}_a \cdot f_s$ and phase $-\phi$ (phase reversal).

## Sampling Higher-Frequency Sinusoids

- For sinusoids of even higher frequencies $f$, either folding or aliasing occurs.
- As before, let $\hat{f}_d$ be the normalized frequency $f/f_s$.
- Decompose $\hat{f}_d$ into an integer part $N$ and fractional part $f_p$.
  - **Example:** If $\hat{f}_d$ is 5.7 then $N$ equals 5 and $f_p$ is 0.7.
  - Notice that $0 \leq f_p < 1$, always.
- **Phase Reversal** occurs when the phase of the sampled sinusoid is the negative of the phase of the continuous-time sinusoid.
- We distinguish between
  - **Folding** occurs when $f_p > 1/2$. Then the apparent frequency $\hat{f}_a$ equals $1 - f_p$ and phase reversal occurs.
  - **Aliasing** occurs when $f_p \leq 1/2$. Then the apparent frequency is $\hat{f}_a = f_p$; no phase reversal occurs.

# Examples

- For the three sinusoids considered earlier:
  1. $f = 1, \phi = \pi/4, f_s = 10 \Rightarrow \hat{f}_d = 1/10$
  2. $f = 9, \phi = -\pi/4, f_s = 10 \Rightarrow \hat{f}_d = 9/10$
  3. $f = 11, \phi = \pi/4, f_s = 10 \Rightarrow \hat{f}_d = 11/10$
- The first case, represents oversampling: The apparent frequency $\hat{f}_a = \hat{f}_d$ and no phase reversal occurs.
- The second case, represents folding: The apparent $\hat{f}_a$ equals $1 - \hat{f}_d$ and phase reversal occurs.
- In the final example, the fractional part of $\hat{f}_d = 1/10$. Hence, this case represents alising; no phase reversal occurs.

## Exercise

The discrete-time sinusoidal signal

$$x[n] = 5\cos(2\pi 0.2n - \frac{\pi}{4}).$$

was obtained by sampling a continuous-time sinusoid of the form

$$x(t) = A\cos(2\pi f t + \phi)$$

at the sampling rate $f_s = 8000\ Hz$.

1. Provide three different sets of paramters $A$, $f$, and $\phi$ for the continuous-time sinusoid that all yield the discrete-time sinusoid above when sampled at the indicated rate. The parameter $f$ must satisfy $0 < f < 12000\ Hz$ in all three cases.
2. For each case indicate if the signal is undersampled or oversampled and if aliasing or folding occurred.