Outline

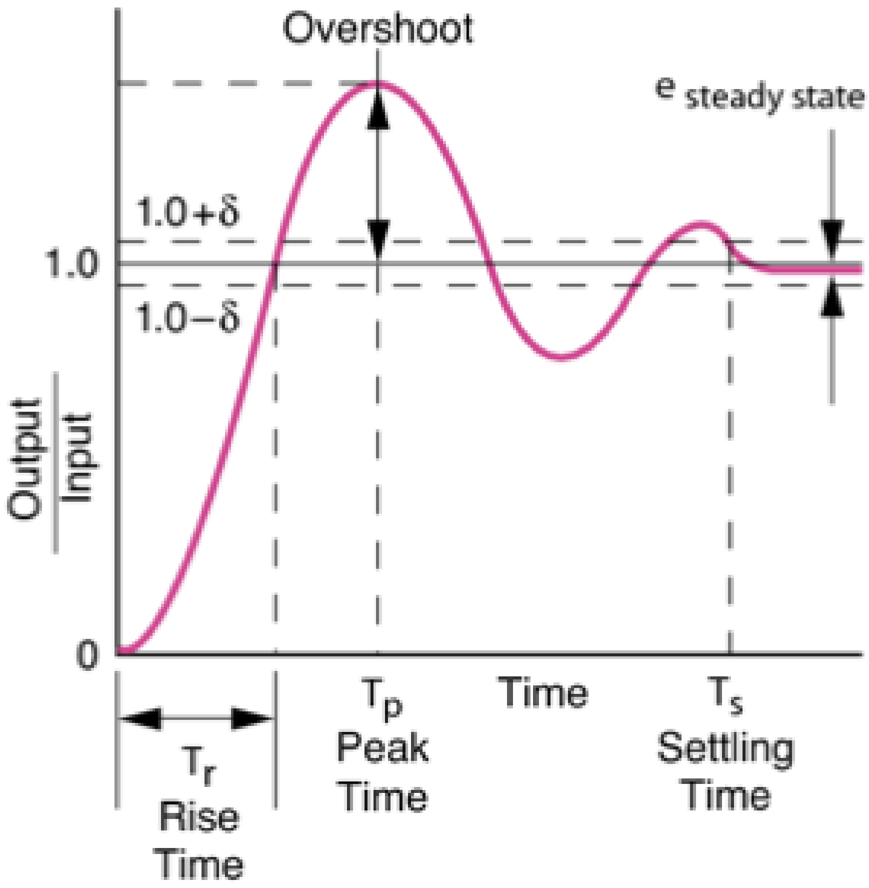
Sampling time T_s based on the time response

Rise time

Time needed to reach the steady state for the first time.

Practical rule

A good rule of thumb is $T_s = \frac{T_{rise}}{10}$.



Sampling time T_s based on the frequency response

Practical rule

A good rule of thumb is $T_s = \frac{1}{2{,}2 Bandwidth}$.

Signal vs System

It is very important to understand that although the previous rule of thumb for a system seems the same as the rule of thumb for a signal, they differ! The rule of thumb for a signal uses the bandwidth of the signal and the rule of thumb for a system uses the bandwidth of the system, which can be found in the bodeplot or using MATLAB.

Outline

Outline

MATLAB

MATLAB Commands for calculations

MATLABs Control System Toolbox offers extensive support for discretization and re-sampling of linear systems:

- "c2d(system,sampling time,method)" used for discretization
- "d2c(system,sampling time,method)" used for transforming a discrete-time system to a continuous-time system (not used in the examples).

These commands can also have extra options. It is necessary to specify the options in an additional command: "c2dOptions('OptionName',OptionValue)".

MATLAB

Available options

- 'Method'
- 'PrewarpFrequency'
- 'FractDelayApproxOrder'

Available methods

- zero-order hold equivalent: "zoh"
- first-order hold equivalent: "foh"
- impulse invariant rule: "impuls"
- zero-pole matching equivalent: "matched"
- bilinear: "tustin"

MATLAB

MATLAB Commands for visualisation

- MATLAB can draw the bode plots of the continuous system $\rightarrow$ command: "bode(system)";
- MATLAB can carry out a graphic showing the impulse response of the continuous system and the discretized system $\rightarrow$ command: "impulse(H,'b',Hd,'r')" in which 'b' and 'r' stand for blue and red. By using colours, the step responses of both systems can easily be distinguished;
- MATLAB can carry out a graphic showing the step response of the continuous system and the discretized system $\rightarrow$ command: "step(H,'b',Hd,'r')".

Outline

Exercise 1 with MATLAB

Exercise 1

We will now discuss 4 methods applied on the same continuous system: $H(s) = \frac{s+1}{s^2+s+1}$

- The sampling time can be determined by the rule of thumb previously explained: $T_s = 2.2\text{Bandwidth}$;
- MATLAB can calculate the bandwidth of a given continuous system, using the command: "bandwidth(system)".

This results in a sampling time of 0.25033 seconds for the given system

Zero-order hold equivalent

Example

Given:

$H(s) = \frac{s+1}{s^2+s+1}$

Sampling time = 0.25033sec

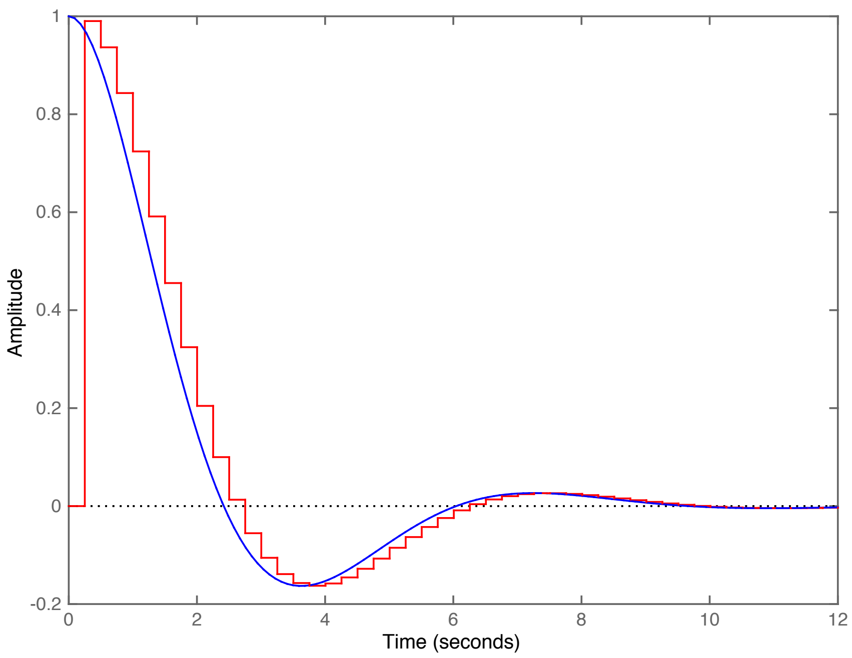
MATLAB commands:

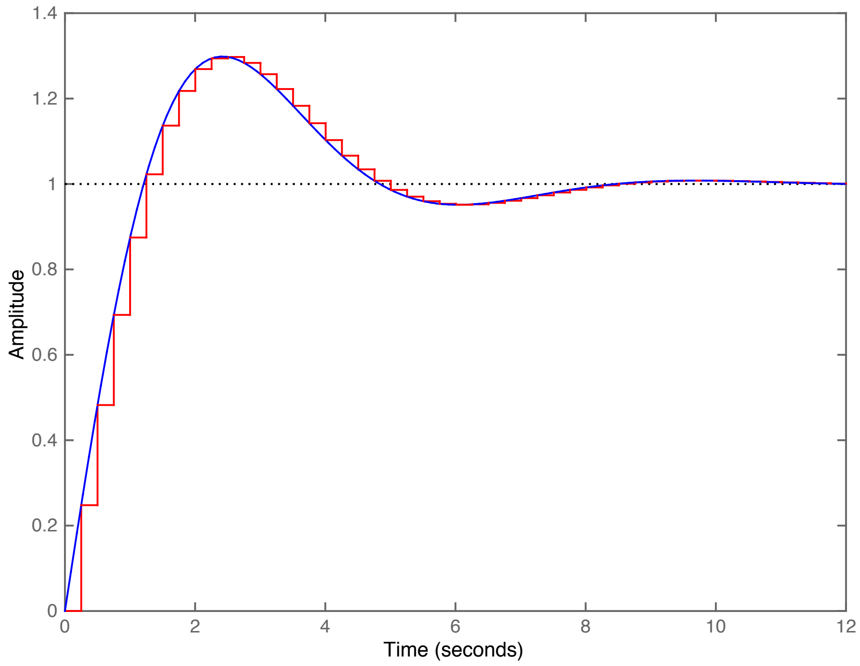
H = tf([1 1],[1 1 1]) we define the system H by its transfer function

Hd = c2d(H,0.25033,'zoh') we calculate the discrete-time model

Result: $H_{zoh}(z) = \frac{0.2479z-0.1927}{z^2-1.723z+0.7785}$

Zero-order hold equivalent (impulse and step response)





First-order hold equivalent

Example

Given:

$H(s) = \frac{s+1}{s^2+s+1}$

Sampling time = 0.25033sec

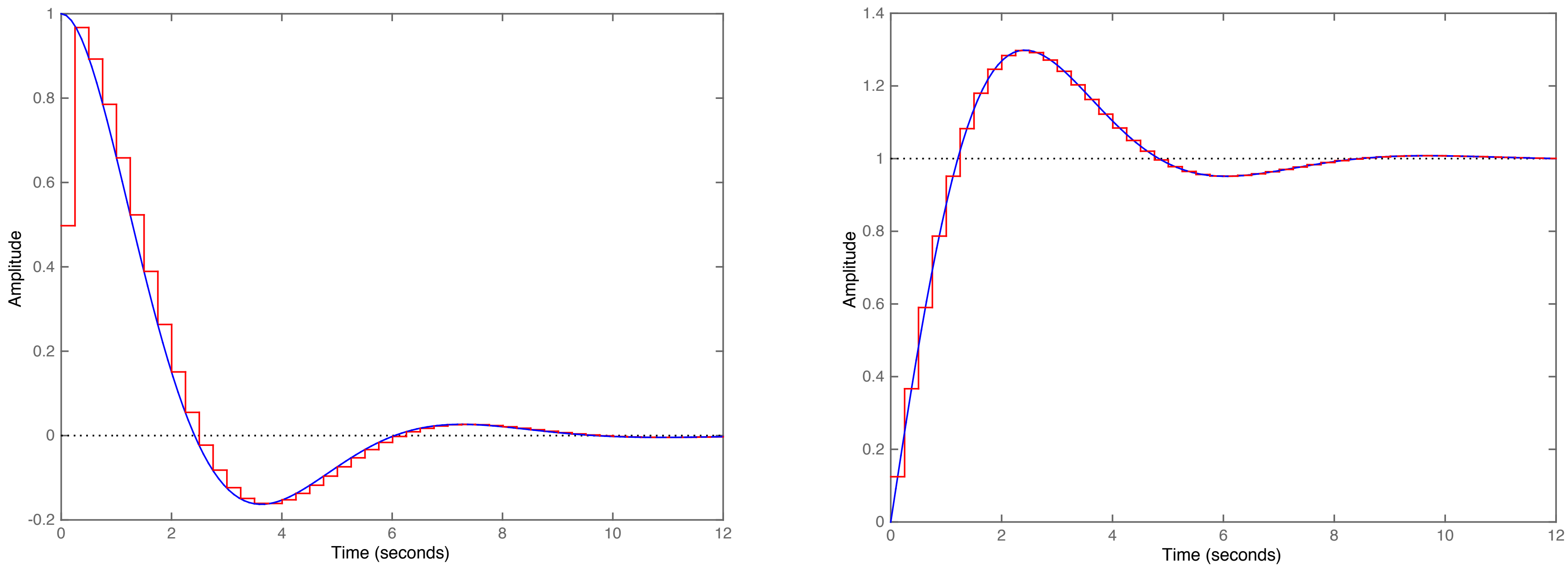
MATLAB commands:

H = tf([1 1],[1 1 1]) we define the system H by its transfer function

Hd = c2d(H,0.25033,'foh') we calculate the discrete-time model

Result: $H_{foh}(z) = \frac{0.1245z^2+0.02752z-0.09691}{z^2-1.723z+0.7785}$

First-order hold equivalent (impulse and step response)



Impulse invariant rule

Example

Given:

$H(s) = \frac{s+1}{s^2+s+1}$

Sampling time = 0.25033sec

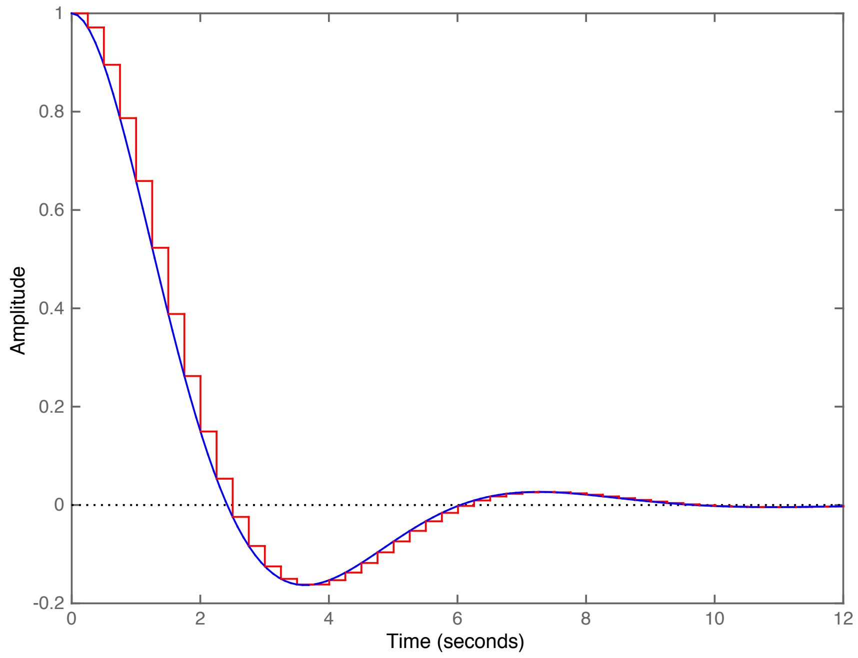
MATLAB commands:

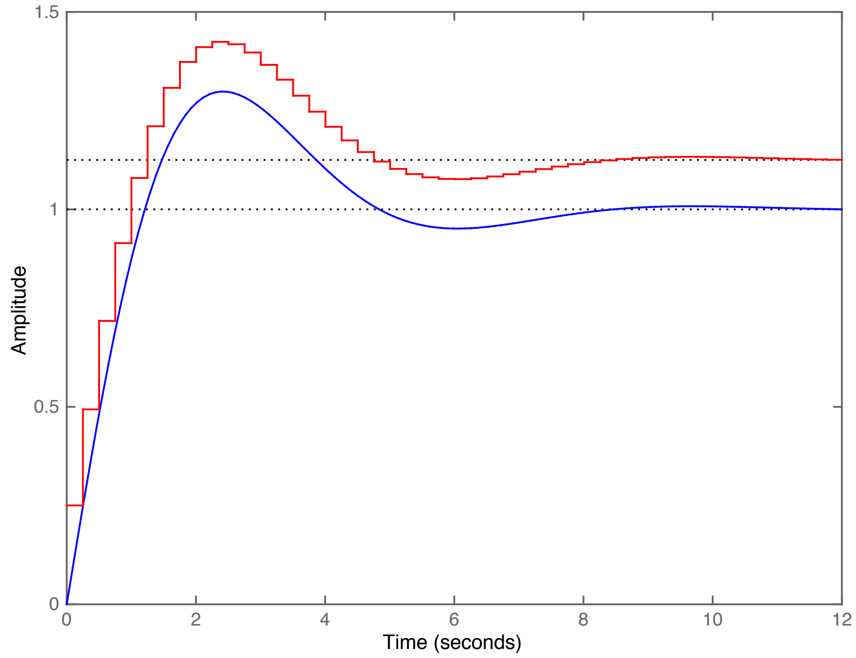
H = tf([1 1],[1 1 1]) we define the system H by its transfer function

Hd = c2d(H,0.25033,'impuls') we calculate the discrete-time model

Result: $H_{impulse}(z) = \frac{0.2503z^2-0.1883z}{z^2-1.723z+0.7785}$

Impuls invariant rule (impulse and step response)





Zero-pole equivalent

Example

Given:

$H(s) = \frac{s+1}{s^2+s+1}$

Sampling time $=$ 0.25033sec

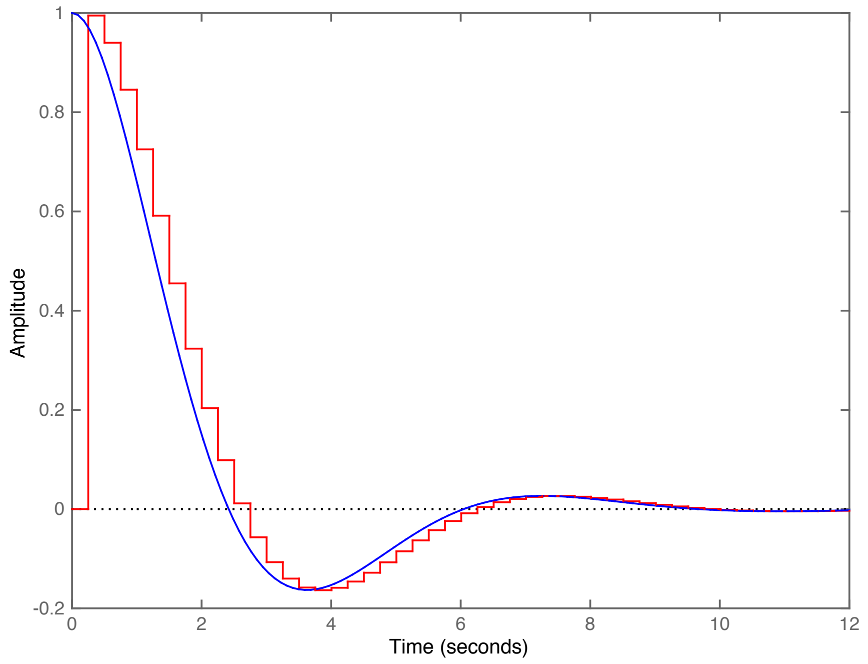
MATLAB commands:

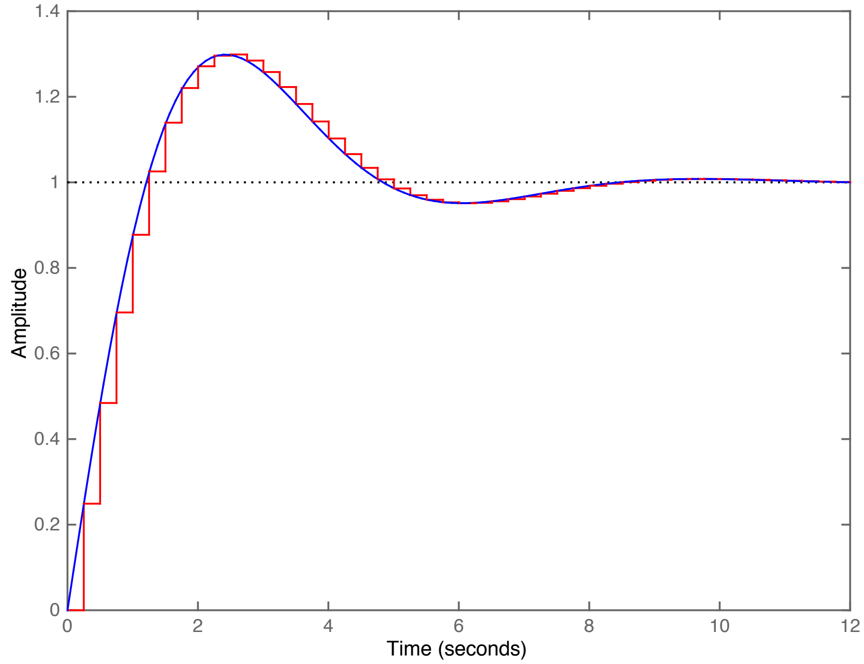
H $=$ tf([1 1],[1 1 1]) we define the system H by its transfer function

Hd $=$ c2d(H,0.25033,'matched') we calculate the discrete-time model

Result: $H_{matched}(z) = \frac{0.249z-0.1939}{z^2-1.723z+0.7785}$

Zero-pole equivalent (impulse and step response)





Exercise 2 with MATLAB

Exercise 2

We will now apply the bilinear rule with and without prewarping on the same continuous system: $H(s) = \frac{s^2+0.5s+9}{s^2+5s+1}$

Criteria:

1. The sampling time is chosen to be 0.5 seconds;
2. The discrete system must have the same behaviour as the continuous one at 3rad/s.

The last criteria applies to this specific exercise. It is possible that the parity of the magnitude, of the discrete and the continuous system, at a different frequency (e.g. the peak frequency) is required.

Tustin rule

Example

Given:

$H(s) = \frac{s^2+0.5s+9}{s^2+5s+1}$

Sampling time = 0.5sec

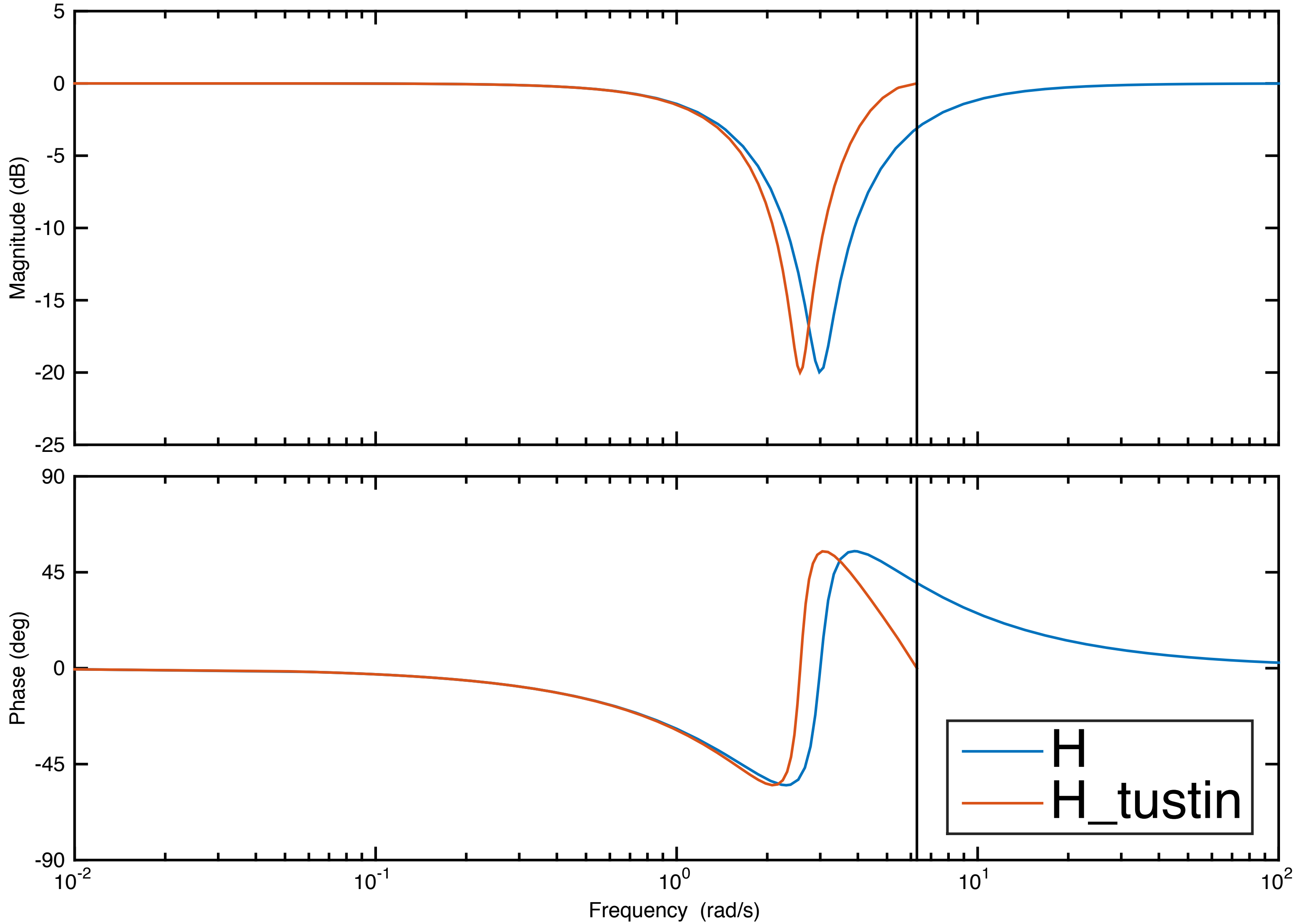
MATLAB commands:

H = tf([1,0.5,9],[1,5,9]) definition of the system H

Hdt = c2d(H,0.5,'tustin') we calculate the discrete-time model

Result: $H_{Tustin}(z) = \frac{0.6z^2-0.3111z+0.5111}{z^2-0.3111z+0.1111}$

Tustin rule (bode plot)



Tustin rule with prewarping

Example

Given:

$H(s) = \frac{s^2+0.5s+9}{s^2+5s+1}$

Sampling time $= 0.5$sec

MATLAB commands:

H = tf([1,0.5,9],[1,5,9]) definition of the system H

damp(H) results in 3.0Hz

discopts = c2dOptions('Method','tustin','PrewarpFrequency',3.0) at the prewarpfrequency, the discrete-time model will have the same behavior as the continuous-time one

Hdtp = c2d(H,0.5,discopts) we calculate the discrete-time model

Result: $H_{TustinPrewarp}(z) = \frac{0.5915z^2-0.07726z+0.5007}{z^2-0.07726z+0.09215}$

Tustin rule with prewarping (bode plot)