title of your presentation

July 15, 2015

Outline

Discretization of Continuous-time systems

Use of discretization

Many systems in the real world are continous systems: chemical reactions, rocket trajectories, power plants, ice cap melting... Computers, however, are mainly digital. If we want to simulate the continuous system with a digital device, we need a method to convert the continuous model into a discrete one. This conversion is called "discretization" or "sampling". Discretization also comes in handy when a continuous filter with usefull properties has been designed and a discrete filter with the same properties is required.

Discretization

Problem statement

While converting, some information of the continuous model will be lost due to the different nature of the systems. It is important that the loss of information is minimized. Each discretization method has its own qualities and they will all lead to different discrete representations of the same continuous system.

Discretization methods discussed in this lecture

- Numerical Integration
- Zero-pole equivalent
- Hold equivalents

Outline

Outline

Numerical Integration

General approach

The system transfer function H(s) is first represented by a differential equation. Next a difference equation, whose solution is an approximation of this differential equation, is derived:

$H(s) = \frac{a}{s+a}$ is equivalent to the differential equation

$$\frac{\mathrm{d}}{\mathrm{d}t}\big(u(t)\big) + au(t) = ae(t)$$

solving this equation results in the following integral

$$u(t) = \int_0^t \big(-au(\tau) + ae(\tau)\big)\mathrm{d}\tau$$

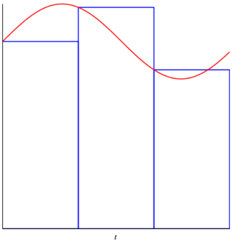
$$u(kT) = u(kT-T) + \begin{cases} \text{area of } -au(t) + ae(t) \\ \text{over } kT - T \leq \tau < kT \end{cases} \quad (8.1)$$

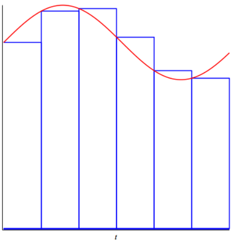
Where T is the sampling time. The transfer function H(s) above will be used for all the numerical integration methods.

Forward rectangular rule (=Forward Euler)

General approach

The area is approximated by the rectangle looking **forward** from $(kT - T)$ toward kT with an amplitude equal to the value of the function at $(kT - T)$. A smaller step-size T leads to a more accurate approximation, as shown in the figures.





Forward rectangular rule

Mathematical approach

The general approach (formula (8.1)) applied on the forward rectangle rule, results in an equation u_1:

$$\begin{aligned} u_1(kT) &= u_1(kT - T) + T\big(- au_1(kT - T) \\ &\quad + ae(kT - T)\big) \\ &= (1 - aT)u_1(kT - T) + aTe(kT - T) \end{aligned}$$

Forward rectangular rule

Mathematical approach

In this case, the transfer function is:

$$H_F(z) = \frac{a}{(z-1)/T+a}$$

Which can also be derived using the following substitution in the given transfer function:

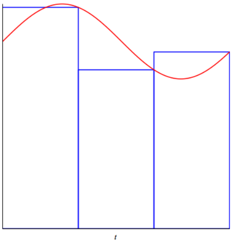
$$s \leftarrow \frac{z-1}{T} \ (8.2)$$

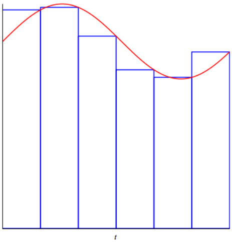
This is extremely useful while making exercises.

Backward rectangular rule (=Backward Euler)

General approach

The area is approximated by the rectangle looking **backward** from kT toward $(kT - T)$ with an amplitude equal to the value of the function at kT.





Backward rectangular rule

Mathematical approach

The general approach (formula (8.1)) applied on the forward rectangle rule, results in an equation u_2:

$$\begin{aligned} u_2(kT) &= u_2(kT - T) + T\big(- au_2(kT) + ae(kT)\big) \\ &= \frac{u_2(kT - T)}{1 + aT} + \frac{aT}{1 + aT} e(kT) \end{aligned}$$

Backward rectangular rule

Mathematical approach

In this case, the transfer function is:

$$H_B(z) = \frac{a}{(z-1)/Tz + a}$$

Which can also be derived using the following substitution in the given transfer function:

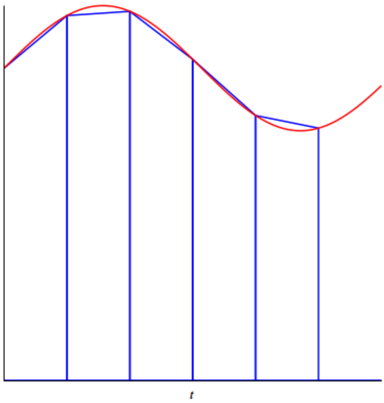
$$s \leftarrow \frac{z-1}{Tz} \quad (8.3)$$

Again this is extremely useful while making exercises.

Trapezoidal rule (= bilinear or Tusting rule)

General approach

This method makes use of the area of the **trapezoid** formed by the average of the selected rectangles used in the forward en backward rectangle rule.Thus the amplitude equal to the value of the function at $(kT - T)$ and the amplitude equal to the value of the function at (kT) are connected by a line as shown in the illustration.



Trapezoidal rule

Mathematical approach

The general approach (formula (8.1)) applied on the forward rectangle rule, results in an equation u_3:

$$\begin{aligned} u_3(kT) &= u_3(kT - T) + T/2\big(-au_3(kT - T) \\ &\quad + ae(kT - T) - au_3(kT) + ae(kT)\big) \\ &= \frac{1 - (aT/2)}{1 + (aT/2)} u_3(kT - T) \\ &\quad + \frac{aT/2}{1 + (aT/2)} \big(e_3(kT - T) + e_3(kT)\big) \end{aligned}$$

Trapezoidal rule

Mathematical approach

In this case, the transfer function is:

$$H_T(z) = \frac{a}{\frac{2}{T}\frac{z-1}{z+1}a}$$

Which can also be derived using the following substitution in the given transfer function:

$$s \leftarrow \frac{2}{T}\frac{z-1}{z+1} \ (8.4)$$

This is extremely useful while making exercises.

Trapezoidal rule

Example

Given:

$$H(s) = \frac{s+1}{0.1s+1}$$

We now apply substitution (8.4):

$$H(z) = \frac{(2+T)(T-2)z^{-1}}{(0.2+T)+(T-0.2)z^{-1}}$$

Using T=0.25s, this results in:

$$H(z) = \frac{5(z-0.7778)}{z+0.1111}$$

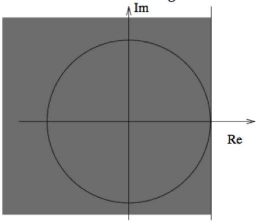
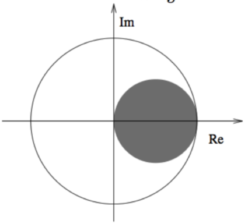
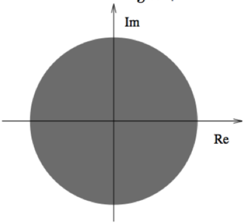
Stability of the numerical integration methods

Stability

As already mentioned, a continuous system is stable when its poles have a negative real part in the s-plane and a discrete system is stable when its poles lie within the unit circle of the z-plane.Subsequently the $(s = j\omega)$-axis is the boundary between poles of stable and unstable continuous systems. Each of the discretization methods can be considered as a map from the s-plane to the z-plane. It is interesting to know how the $j\omega$-axis is mapped by every rule and where the stable part of the s-plane appears in the z-plane. This can be realized by solving formulas (8.2-8.3) to z and replacing s by $j\omega$.

Graphical representation

■ = projection of the left half-plane

Euler	backward rect.	bilinear transf.
$s \longrightarrow \frac{z-1}{T_s}$	$s \longrightarrow \frac{z-1}{zT_s}$	$s \longrightarrow \frac{2}{T_s}\frac{z-1}{z+1}$
		

Bilinear rule with prewarping

Distortion

The bilinear rule maps the stable region of the s-plane into the stable region of the z-plane and the entire $j\omega$-axis is compressed into the 2Π-length of the unit circle, causing a great deal of distortion.

When the actual frequency of ω is input to the discrete-time filter designed by use of the bilinear transform, it is desired to know at what frequency, ω_d, for the continuous-time filter that this ω is mapped to.

Bilinear rule with prewarping

Trapezoid rule with prewarping

By extending the trapezoidal rule one step we can correct the distortion of the real frequencies mapped by the rule.

$$
\begin{aligned}
H_d(z) &= H(\frac{2}{T}\frac{z-1}{z+1}) \\
H_d(e^{j\omega T}) &= H(\frac{2}{T}\frac{e^{j\omega T}-1}{e^{j\omega T}+1}) \\
&= H(\frac{2}{T}\frac{e^{j\omega T/2}(e^{j\omega T/2}-e^{-j\omega T/2})}{e^{j\omega T/2}(e^{j\omega T/2}+e^{-j\omega T/2})}) \\
&= H(j\frac{2}{T}\frac{sin(\omega T/2)}{cos(\omega T/2)})
\end{aligned}
$$

Bilinear rule with prewarping

Trapezoid rule with prewarping

The discrete-time system has the same behaviour at frequency ω as the continuous-time system at frequency $\omega_a = 2/T * tan(\omega T/2)$. This results in following substitution formula:

$$s \leftarrow \frac{\omega_0}{tan\left(\frac{\omega_0 * T}{2}\right)} \frac{z-1}{z+1} \quad (8.4)$$

Outline

Zero-pole equivalent

General approach

The map $z = e^{sT}$ is applied to the poles as well as to the zeros of the continuous system. The following rules must be followed:

1. If $s = -a$ is a pole of H(s), then $z = e^{-aT}$;
2. All finite zeros $s = -b$ are mapped by $z = e^{-bT}$;
3. Zeros at ∞ are mapped to $z = -1$;
4. The gain of the digital filter must match the gain of H(s) at the band center or a similar critical point.

Zero-pole equivalent

Example

Given:

$$H(s) = \frac{s+1}{0.1s+1}$$

Pole at $s = -10$ and zero at $s = -1$.
New discrete transfer function with equivalent poles and zeros:

$$H(z) = K\frac{z-e^{-T}}{z-e^{-10T}}$$

K is chosen so that $\mid H(z) \mid_{z=1} = \mid H(s) \mid_{s=0} \rightarrow K = 4.150$
Using $T = 0.25$, this results in:

$$H(z) = 4.150\frac{z-0.7788}{z-0.0821}$$

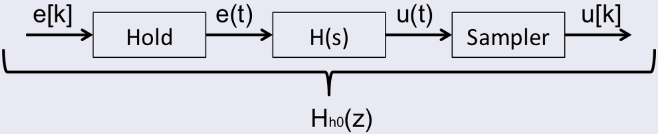
Outline

Hold equivalent

General approach

This method uses a discrete system consisting of 3 subsystems, each with its own purpose.

1. Hold: approximating $e_h(t)$ from the samples $e[k]$
2. H(s): putting the $e_h(t)$ through the given transfer function H(s) of the continuous system, resulting in u(t)
3. Sampler: sampling u(t)



There are many techniques for holding a sequence of samples.

Impuls-invariant method

Impuls-invariant transformation

This method converts a continuous system into a discrete one by matching the impulse response using these transformations.

$\hat{H}_i(p)$	$H_i(z),\ a=e^{bT}$
$\frac{c}{p-b}$	$\frac{Tc}{1-az^{-1}}$
$\frac{c}{(p-b)^2}$	$T^2\frac{caz^{-1}}{(1-az^{-1})^2}$
$\frac{c}{(p-b)^3}$	$\frac{T^3c}{2}\cdot\frac{az^{-1}}{(1-az^{-1})^2}+\frac{2a^2z^{-2}}{(1-az^{-1})^3}=\frac{T^3caz^{-1}(1+az^{-1})}{2(1-az^{-1})^3}$
$\frac{c}{(p-b)^4}$	$\frac{T^4c}{6}\cdot\frac{az^{-1}}{(1-az^{-1})^2}+\frac{6a^2z^{-2}}{(1-az^{-1})^3}+\frac{6a^3z^{-3}}{(1-az^{-1})^4}$ $=\frac{T^4caz^{-1}(1+4az^{-1}+a^2z^{-2})}{6(1-az^{-1})^4}$

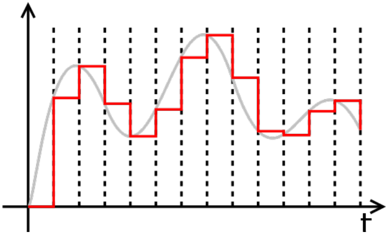
Zero-order hold equivalent (ZOH)

Practical rule

The zero-order hold equivalent transfer function $H_{zoh}(z)$ can be found by computing the following:

$$H_{zoh}(z) = (1 - z^{-1})\mathcal{Z}\{\frac{H(s)}{s}\}$$

This rule is also called the Step-invariant method because it matches the step response of the continuous and the discrete system.



Zero-order hold equivalent

Example

$$H(s) = \frac{0.1}{s+0.1}$$

Step 1: multiply by 1/s and perform partial fraction expansion

$$\frac{H(s)}{s} = \frac{0.1}{s*(s+0.1)} = \frac{1}{s} - \frac{1}{s+0.1}$$

Step 2: perform z-transformation

$$\mathcal{Z}\{\frac{H(s)}{s}\} = \frac{1}{1-z^{-1}} - \frac{1}{1-e^{-0.1T}*z^{-1}}$$

Step3: simplify and multiply by $(1-z^1)$

$$H_{ho}(z) = \frac{1-e^{-0.1T}}{z-e^{0.1T}}$$

Zero-order hold equivalent

Step Invariant Transformation

Zero-order hold equivalent of continuous systems with common transfer functions can easily be found by these transformations.

$f(t)$	$F(s)$	$f(k)$	$F(z)$
$u(t)$, unit step	$\frac{1}{s}$	$u(k)$, unit step	$\frac{z}{z-1}$
$tu(t)$	$\frac{1}{s^2}$	$kTu(k)$	$\frac{Tz}{(z-1)^2}$
$e^{-\mathrm{at}}u(t)$	$\frac{1}{s+a}$	$(e^{-\mathrm{aT}})^{\mathrm{k}}u(k)$	$\frac{z}{z-e^{-\mathrm{aT}}}$
$te^{-\mathrm{at}}u(t)$	$\frac{1}{(s+a)^2}$	$kT(e^{-\mathrm{aT}})^{\mathrm{k}}u(k)$	$\frac{Tze^{-\mathrm{aT}}}{(z-e^{-\mathrm{aT}})^2}$
$\sin(\omega t)u(t)$	$\frac{\omega}{s^2+\omega^2}$	$\sin(k\omega T)u(k)$	$\frac{z\sin\omega T}{z^2-2z\cos\omega T+1}$
$\cos(\omega t)u(t)$	$\frac{s}{s^2+\omega^2}$	$\cos(k\omega T)u(k)$	$\frac{z(z-\cos\omega T)}{z^2-2z\cos\omega T+1}$

Non-causal first-order (= Triangle-hold equivalent (TRI))

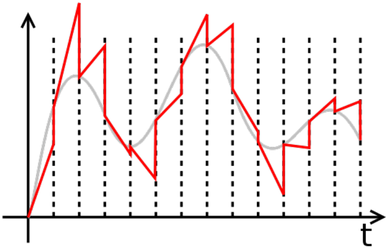
Practical rule

The non-causal first-order hold equivalent transfer function $H_{tri}(z)$ can be found by computing the following:

$$H_{tri}(z) = \frac{(z-1)^2}{Tz} \mathcal{Z}\left\{\frac{H(s)}{s^2}\right\}$$

Non-causal vs causal

A causal first-order hold equivalent introduces a time-delay resulting in a less accurate approximation.



Triangle-hold equivalent

Example

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

Step 1: multiply by $1/s^2$ and perform partial fraction expansion

$$H(s) = \frac{1}{s^4}$$

Step 2: perform z-transformation

$$\mathcal{Z}\{\frac{H(s)}{s^2}\} = \frac{T^3}{6}\frac{(z^2+4z+1)z}{(z-1)^4}$$

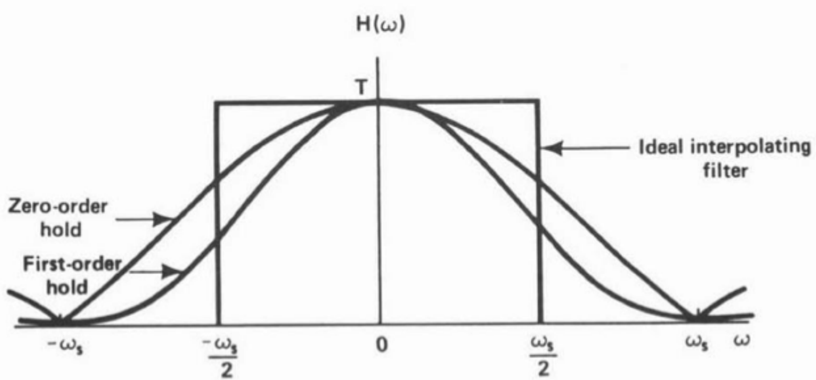
Step 3: simplify and multiply by $\frac{(z-1)^2}{T*z}$

$$H_{tri}(z) = \frac{T^2}{6}\frac{z^2+4z+1}{(z-1)^2}$$

Effect of zero- and first-order hold equivalents

Effects on the spectrum

Some information will be permanently lost.



Outline

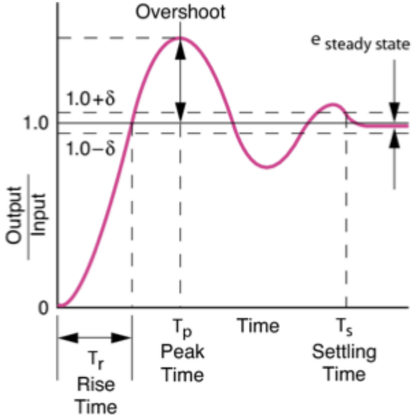
Sampling time Ts based on the time response

Rise time

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

Practical rule

A good rule of thumb is $Ts = T_{rising}/10$ seconds/sample.



Sampling time Ts based on the frequency response

Nyquist-Shannon sampling theorem

If a function $x(t)$ contains no frequencies higher than B hertz, it is completely determined by giving its ordinates at a series of points spaced $1/(2B)$ seconds apart. A sufficient sample-rate is therefore **2B samples/second**. This is called the **"Nyquist sampling rate"**. Equivalently, for a given sample rate f_s, perfect reconstruction is guaranteed possible for a bandlimit $Bf_s/2$. A higher sampling rate creates a margin of error.

Practical rule

A good rule of thumb is $Ts = 1/(2,2B)$ seconds/sample.

Outline

Matlab

Matlab Commands

Matlabs Control System Toolbox offers extensive support for discretization and resampling of linear systems:

- "c2d(system,sampling time,method)" used for discretization
- "d2c(system,sampling time,method)" used for reconstruction
- "d2d(system,sampling time,method)" used for resampling

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"
- impuls invariant rule: "impuls"
- zero-pole matching equivalent: "matched"
- bilinear: "tustin"

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: $\text{Ts} = 2.2\text{Bandwidth};$
- 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

Commands in Matlab:
H = tf([1 1],[1 1 1])
Hd = c2d(H,Ts,'zoh')

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

First-order hold equivalent

Example

Given:
$H(s) = \frac{s+1}{s^2+s+1}$
Sampling time = 0.25033sec

Commands in Matlab:
H = tf([1 1],[1 1 1]);
Hd = c2d(H,Ts,'foh')

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

Impuls invariant rule

Example

Given:
$H(s) = \frac{s+1}{s^2+s+1}$
Sampling time = 0.25033sec

Commands in Matlab:
H = tf([1 1],[1 1 1]);
Hd = c2d(H,Ts,'impuls')

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

Zero-pole equivalent

Example

Given:

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

Sampling time $= 0.25033$sec

Commands in Matlab:

H = tf([1 1],[1 1 1]);

Hd = c2d(H,Ts,'matched')

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

Exercise 2 with Matlab

Exercise 2

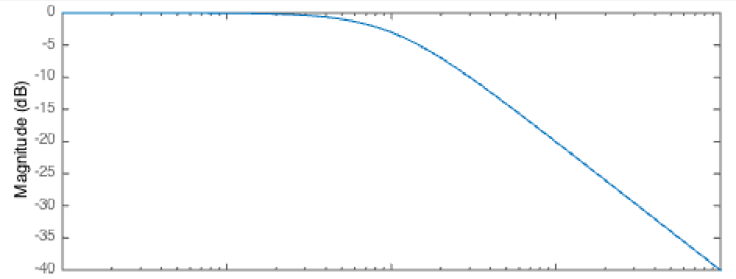
We will use the transfer function of a low pass filter to indicate the difference between the bilinear rule without and the bilinear rule with prewarping: $H(s) = \frac{1}{s+1}$

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

Exercise 2 with Matlab

Exercise 2

This results in a sampling time of 0.45563 seconds and the following magnitude bode plot with a cutoff frequency of 0rad/s:



Tusting rule

Example

Given:
$H(s) = \frac{1}{s+1}$
sampling rate $= 0.45563$sec

Commands in Matlab:
H = tf([1],[1 1])
Hd = c2d(H,Ts,'tustin')

Result: $Hd_{Tustin}(z) = \frac{0.1855z-0.1855}{z-0.6289}$

Tustin rule with prewarping

Example

Given:

$H(s) = e^{-s}\frac{s-2}{s^2+3s+20}$

sampling rate $= 0.25033\text{sec}$

Commands in Matlab:

H = tf([1],[1 1])

opt = c2dOptions('Method', 'tustin', 'PrewarpFrequency', 0)

Hd = c2d(G,Ts,opt)

Result: $Hd_{TustinPrewarp}(z) =$

Matlab

Matlab Commands for visualisation

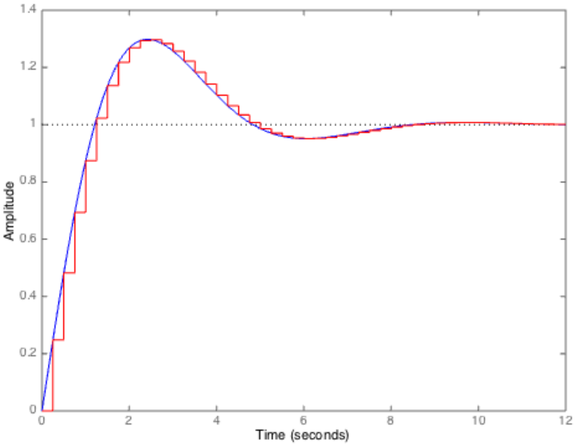
It is possible to compare the step response of the continuous system and the discretized system in a graphic

Command: "step(H,'b',Hd,'r')"

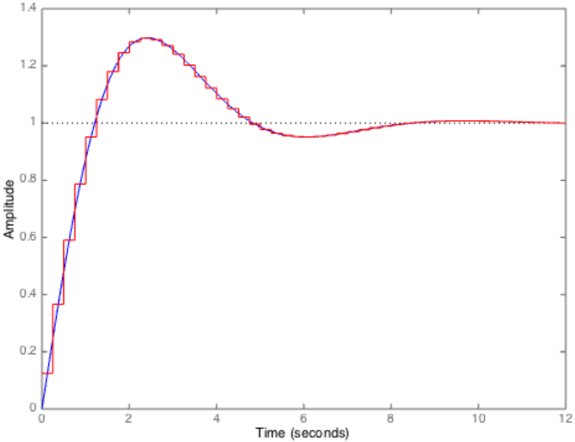
'b' and 'r' stand for blue and red, by using colors, the step responses of both systems can easily be distinguished.

The following figures are a graphic representation of the exercises in Matlab.

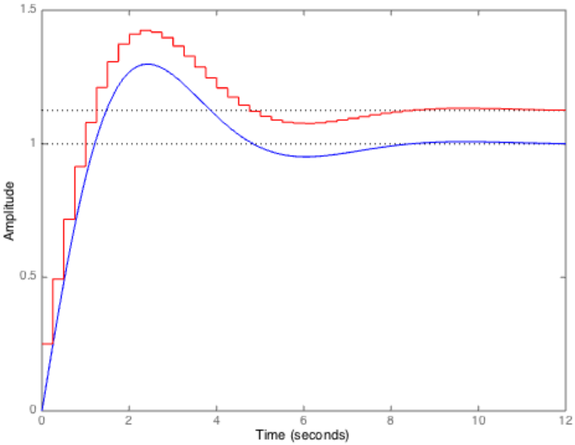
Zero-order hold equivalent



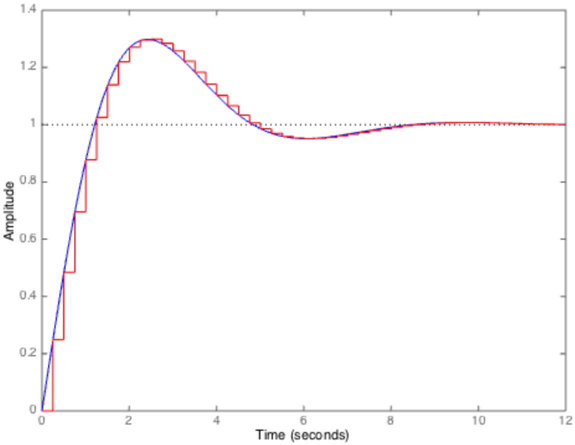
First-order hold equivalent



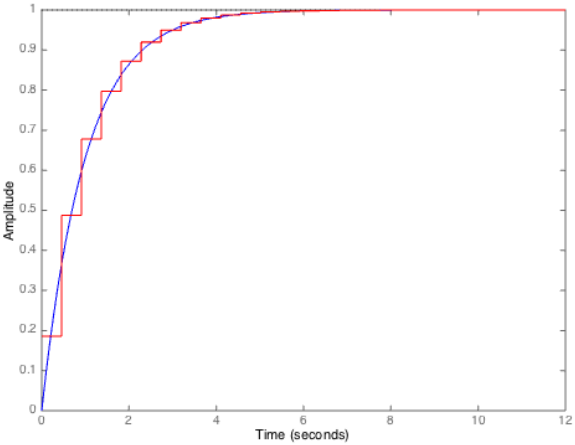
Impuls invariant rule



Zero-pole equivalent



Tustin rule



Tustin rule with prewarping