Continuous to Discrete Transfer Function Transformation Using the Euler Methods

MATLAB Implementation

Tamas Kis | kis@stanford.edu

TAMAS KIS
https://github.com/tamaskis

LICENSE

Contents

c2d_euler

Transforms a continuous transfer function to a discrete transfer function using the forward and backward Euler methods.

Syntax

```
Hz = c2d_euler(Hs,T,'forward')
Hz = c2d_euler(Hs,T,'backward')
```

Description

`Hz = c2d_euler(Hs,T,'forward')` returns the discrete transfer function `Hz` obtained by applying the forward Euler (i.e. forward difference) transformation to a continuous transfer function `Hs`, where `T` is the sampling period.

`Hz = c2d_euler(Hs,T,'backward')` returns the discrete transfer function `Hz` obtained by applying the backward Euler (i.e. backward difference) transformation to a continuous transfer function `Hs`, where `T` is the sampling period.

Examples

Example 1

Consider the following continuous transfer function:

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

Find the corresponding discrete transfer function $H(z)$ using a sampling period of $T = 0.25$ s. Use both the forward Euler and backward Euler transformations.

■ *SOLUTION*

Defining $H(s)$ as a transfer function object,

```
s = tf('s');
Hs = (s+1)/(0.1*s+1);
```

Applying the forward Euler transformation,

```
Hz_forward = c2d_euler(Hs,T,'forward')
```

This results in

```
Hz_forward =

  20 z - 15
  ---------
   2 z + 3
```

```
Sample time: 0.25 seconds
Discrete-time transfer function.
```

Applying the backward Euler transformation,

```
Hz_backward = c2d_euler(Hs,T,'backward')
```

This results in

```
Hz_backward =

  25 z - 20
  ---------
   7 z - 2

Sample time: 0.25 seconds
Discrete-time transfer function.
```

Links

MATLAB® Central's File Exchange:
https://www.mathworks.com/matlabcentral/fileexchange/90267-euler-c2d-transformation-c2d_euler

GitHub®:
https://github.com/tamaskis/c2d_euler-MATLAB

Continuous to Discrete Transformations Using the Euler Methods

Consider a continuous transfer function $H(s)$ that we wish to transform to a discrete transfer function $H(z)$ using a sampling period of T and some approximation method (i.e. some approximate transformation). Both the **forward Euler** and **backward Euler** are transformations based on evaluating $H(s)$ at some value of s representing an *approximate* transformation between s and z [1].

$$\textbf{Forward Euler:} \quad s \leftarrow \frac{z-1}{T} \qquad \therefore H(z) = H(s)|_{s=\frac{z-1}{T}}$$

$$\textbf{Backward Euler:} \quad s \leftarrow \frac{z-1}{Tz} \qquad \therefore H(z) = H(s)|_{s=\frac{z-1}{Tz}}$$

References for Code

Syms to TF Conversion (`syms2tf.m`) [4]:

- Conversion of a symbolic function object to a transfer function object.

SYM2TF: a matlab function which converts symbolic math rationals to transfer function object [3]:

- Conversion of a symbolic function object to a transfer function object.

How can I convert a transfer function object from the Control System Toolbox into a symbolic object for use with the Symbolic Math Toolbox? [2]:

- Conversion of a transfer function object to a symbolic function object.

References

[1] Gene F. Franklin, J. David Powell, and Michael Workman. *Digital Control of Dynamic Systems*. 3rd. Half Mooon Bay, CA: Ellis-Kagle Press, 1998.

[2] *How can I convert a transfer function object from the Control System Toolbox into a symbolic object for use with the Symbolic Math Toolbox?* MATLAB Answers. https://www.mathworks.com/matlabcentral/answers/96275-how-can-i-convert-a-transfer-function-object-from-the-control-system-toolbox-into-a-symbolic-object. (accessed: April 10, 2021).

[3] Ichiro Maruta. *SYM2TF: a matlab function which converts symbolic math rationals to transfer function object*. GitHub Gist. https://gist.github.com/maruta/1035254. (accessed: April 10, 2021).

[4] Crystal Nassouri. *Syms to TF Conversion*. MATLAB Central File Exchange. https://www.mathworks.com/matlabcentral/fileexchange/27302-syms-to-tf-conversion. (accessed: April 10, 2021).