Interpolatory framework for mixed continuous-sampled-data models interconnection analysis

C. Poussot-Vassal

ONERA
THE FRENCH AEROSPACE LAB

2019, Toulouse, France
(coll. w. P-L. Garoche, T. Loquen, C. Pagetti and P. Vuillemin)

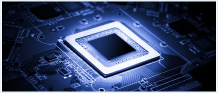
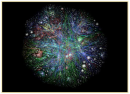
FEANICSES

Outlines

Real-time control systems

- Most control systems are real-time systems
- Many hard real-time systems are control systems

Makes digital control a central challenge in control applications

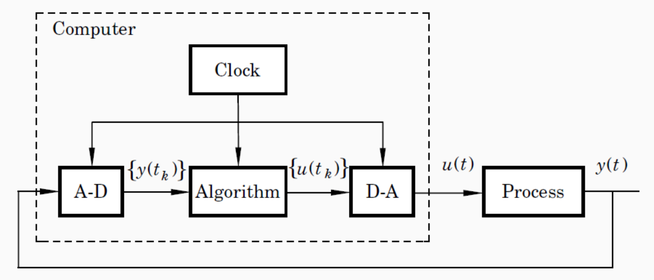
In general cases,

- the **controlled system** is a continuous-time dynamical one
- the **controller** is a sampled-time dynamical one

One challenge aims at **stabilizing / analyzing** a **continuous** dynamical system with a **sampled-time** controller

Digital Control Systems

- Process: plant, system to be controlled and which some "states" can measured
  Assumptions: linear invariant, continuous-time, controllable, observable, *etc.*
- Computer (controller): chip, microcontroller, *etc.*
  Assumptions: linear invariant, discrete-time (constant clock h), bits limitations
- A/D and D/A: analog digital interfaces
  Assumptions: piece-wise linear, discrete-time (constant clock h), bits limitations



K-E Arzen's team in Lund University, *"http://www.control.lth.se/Education/EngineeringProgram/FRTN01.html"*, Lund, Sweden.

Today's presentation

Suggest an hybrid (continuous / sampled-time) analysis framework

- Using irrational and descriptor modelling...
- ... solved with interpolatory methods

Summary

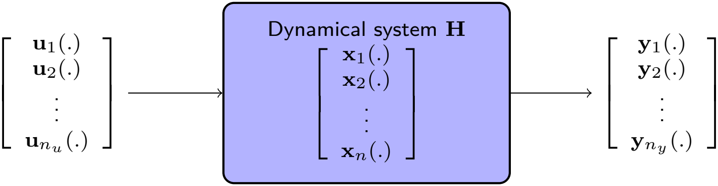
- Continuous to sampled-time dynamical modelling
- Continuous-sampled-time interconnection
- Illustration

Outlines

Continuous and sampled-time dynamical models

Let us consider $\mathbf{H}$, a n_u inputs, n_y outputs linear dynamical system described by the **complex-valued function of order** n (n large or ∞) equipped with realisation $\mathcal{S}$

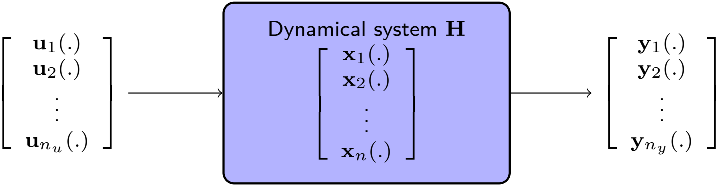
$$\mathcal{S}:\left\{\begin{array}{rcl} E\dot{\mathbf{x}}(t) & = & A\mathbf{x}(t)+B\mathbf{u}(t)\in\mathbb{R}^n \\ \mathbf{y}(t) & = & C\mathbf{x}(t)+D\mathbf{u}(t)\in\mathbb{R}^{n_y}\end{array}\right. ,\ \mathbf{y}(s)=\mathbf{H}(s)\mathbf{u}(s)\in\mathbb{C}^{n_y}$$



Continuous and sampled-time dynamical models

Let us consider $\mathbf{H}$, a n_u inputs, n_y outputs linear dynamical system described by the **complex-valued function of order n** (n large or ∞) equipped with realisation $\mathcal{S}$

$$\mathcal{S}:\left\{\begin{array}{rcl} E\dot{\mathbf{x}}(t) & = & A\mathbf{x}(t)+B\mathbf{u}(t)\in\mathbb{R}^n \\ \mathbf{y}(t) & = & C\mathbf{x}(t)+D\mathbf{u}(t)\in\mathbb{R}^{n_y}\end{array}\right. ,\ \mathbf{y}(s)=\mathbf{H}(s)\mathbf{u}(s)\in\mathbb{C}^{n_y}$$



... when sampled with constant value h

$$\mathcal{S}_s:\left\{\begin{array}{rcl} E_s\mathbf{x_s}(t_k+h) & = & A_s\mathbf{x_s}(t_k)+B_s\mathbf{u}(t_k)\in\mathbb{R}^n \\ \mathbf{y}(t_k) & = & C_s\mathbf{x_s}(t_k)+D_s\mathbf{u}(t_k)\in\mathbb{R}^{n_y}\end{array}\right. ,\ \mathbf{y}(z)=\mathbf{H}_s(z)\mathbf{u}(z)\in\mathbb{C}^{n_y}$$

Some standard z transform

Zero order hold (perfect holder)

$$z = \frac{1 - e^{sh}}{s} \quad , \quad s = \frac{z-1}{h}$$

Bilinear transform (approximation)

$$z = e^{sh} = \frac{e^{sh/2}}{e^{-sh/2}} \approx \frac{1+sh/2}{1-sh/2} \quad , \quad s = \frac{1}{h}\ln z \approx \frac{2}{h}\frac{z-1}{z+1}$$

Pre-wrapped bilinear transform (approximation)

$$z \approx \frac{1 + s\tan(\omega_1 h/2)/\omega_1}{1 - s\tan(\omega_1 h/2)/\omega_1} \quad , \quad s \approx \frac{\omega_1}{\tan(\omega_1 h/2)}\frac{z-1}{z+1}$$

$$\mathbf{H}_s(e^{\imath\omega_1 h}) = \mathbf{H}(\imath\omega_1)$$

All Mobius transformations... how accurate is it?

Some standard z transform

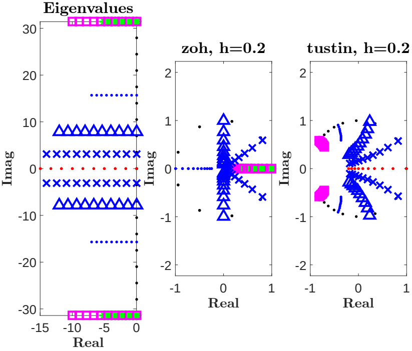
Singularities mapping

- $\mathbb{C}_-$ becomes $\mathbb{D}$ and $\mathbb{C}_+$ becomes $\overline{\mathbb{D}}$
- $\imath\mathbb{R}$ becomes $\partial\mathbb{D}$
- ... but

Some standard z transform

Singularities mapping

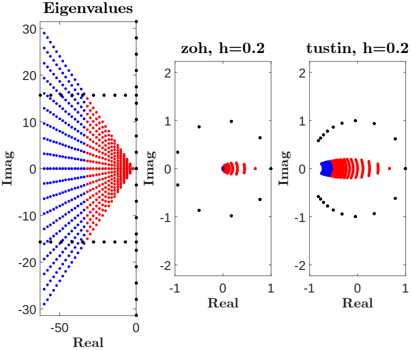
- $\mathbb{C}_-$ becomes $\mathbb{D}$ and $\mathbb{C}_+$ becomes $\overline{\mathbb{D}}$
- $\imath\mathbb{R}$ becomes $\partial\mathbb{D}$
- ... but



Some standard z transform

Singularities mapping

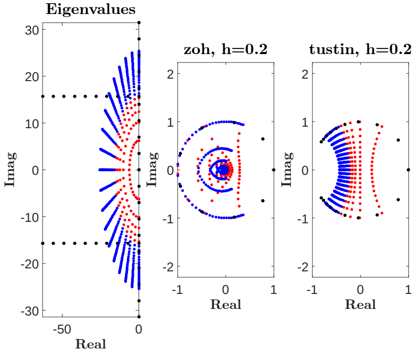
- $\mathbb{C}_-$ becomes $\mathbb{D}$ and $\mathbb{C}_+$ becomes $\overline{\mathbb{D}}$
- $\imath\mathbb{R}$ becomes $\partial\mathbb{D}$
- ... but



Some standard z transform

Singularities mapping

- $\mathbb{C}_-$ becomes $\mathbb{D}$ and $\mathbb{C}_+$ becomes $\overline{\mathbb{D}}$
- $\imath\mathbb{R}$ becomes $\partial\mathbb{D}$
- ... but



The exact $z \leftrightarrow s$ correspondance

$s = \imath\omega$ **and** $z = e^{sh}$**, thus** $s = \frac{1}{h}\ln(z)$

Transfer function

$$\mathbf{H}_s(z) = C_s\left(zE_s - A_s\right)^{-1}B_s + D_s$$

$$\mathbf{H}_s(s) = C_s\left(e^{sh}E_s - A_s\right)^{-1}B_s + D_s$$

and spectrum is now **repeated every** π/h

- $\mathbf{H}(s) \in \mathcal{RH}_\infty$, $\#\left(\Lambda(A,E)\right) = n$ and $\Lambda(A,E) \in \mathbb{C}_-$
  $\mathbf{H}_c : \mathbb{C} \to \mathbb{C}^{n_y \times n_u}$
- $\mathbf{H}_s(z) \in \mathcal{R}h_\infty$, $\#\left(\Lambda(A_s,E_s)\right) = n$ and $\Lambda(A_s,E_s) \in \mathbb{D} \cup \partial\mathbb{D}$
  $\mathbf{H}_s : \mathbb{C}([0\ w_N]) \to \mathbb{C}^{n_y \times n_u}$
- $\mathbf{H}_s(s)$, $\notin \mathcal{H}_\infty$, $\notin \mathcal{L}_\infty$ **and** $\#\left(\Lambda(A_s, e^{sh}E_s)\right) = \infty$
  $\mathbf{H}_s : \mathbb{C}([0\ w_N]) \to \mathbb{C}^{n_y \times n_u}$

The exact $z \leftrightarrow s$ correspondance

$s = \imath\omega$ **and** $z = e^{sh}$**, thus** $s = \frac{1}{h}\ln(z)$

Transfer function

$$\mathbf{H}_s(z) = C_s\left(zE_s - A_s\right)^{-1}B_s + D_s$$

$$\mathbf{H}_s(s) = C_s\left(e^{sh}E_s - A_s\right)^{-1}B_s + D_s$$

and spectrum is now **repeated every** π/h

- $\mathbf{H}(s) \in \mathcal{RH}_\infty$, $\#\left(\Lambda(A,E)\right) = n$ and $\Lambda(A,E) \in \mathbb{C}_-$
  $\mathbf{H}_c : \mathbb{C} \rightarrow \mathbb{C}^{n_y \times n_u}$
- $\mathbf{H}_s(z) \in \mathcal{R}h_\infty$, $\#\left(\Lambda(A_s,E_s)\right) = n$ and $\Lambda(A_s,E_s) \in \mathbb{D} \cup \partial\mathbb{D}$
  $\mathbf{H}_s : \mathbb{C}([0\ w_N]) \rightarrow \mathbb{C}^{n_y \times n_u}$
- $\mathbf{H}_s(s)$, $\notin \mathcal{H}_\infty$, $\notin \mathcal{L}_\infty$ **and** $\#\left(\Lambda(A_s, e^{sh}E_s)\right) = \infty$
  $\mathbf{H}_s : \mathbb{C}([0\ w_N]) \rightarrow \mathbb{C}^{n_y \times n_u}$

The exact $z \leftrightarrow s$ correspondance

$s = \imath\omega$ **and** $z = e^{sh}$**, thus** $s = \frac{1}{h}\ln(z)$

Transfer function

$$\mathbf{H}_s(z) = C_s\Big(zE_s - A_s\Big)^{-1}B_s + D_s$$

$$\mathbf{H}_s(s) = C_s\Big(e^{sh}E_s - A_s\Big)^{-1}B_s + D_s$$

and spectrum is now **repeated every** π/h

- $\mathbf{H}(s) \in \mathcal{RH}_\infty$, $\#\Big(\Lambda(A,E)\Big) = n$ and $\Lambda(A,E) \in \mathbb{C}_-$
  $\mathbf{H}_c : \mathbb{C} \rightarrow \mathbb{C}^{n_y \times n_u}$
- $\mathbf{H}_s(z) \in \mathcal{R}h_\infty$, $\#\Big(\Lambda(A_s,E_s)\Big) = n$ and $\Lambda(A_s,E_s) \in \mathbb{D} \cup \partial\mathbb{D}$
  $\mathbf{H}_s : \mathbb{C}([0\ w_N]) \rightarrow \mathbb{C}^{n_y \times n_u}$
- $\mathbf{H}_s(s)$, $\notin \mathcal{H}_\infty$, $\notin \mathcal{L}_\infty$ **and** $\#\Big(\Lambda(A_s, e^{sh}E_s)\Big) = \infty$
  $\mathbf{H}_s : \mathbb{C}([0\ w_N]) \rightarrow \mathbb{C}^{n_y \times n_u}$

The exact $z \leftrightarrow s$ correspondance

$s = \imath\omega$ **and** $z = e^{sh}$**, thus** $s = \frac{1}{h}\ln(z)$

Transfer function

$$\mathbf{H}_s(z) = C_s\left(zE_s - A_s\right)^{-1}B_s + D_s$$

$$\mathbf{H}_s(s) = C_s\left(e^{sh}E_s - A_s\right)^{-1}B_s + D_s$$

and spectrum is now **repeated every** π/h

Note that

$$\mathbf{y}(s) = \mathbf{H}_s(s)\mathbf{u}(s)$$

leads to a delayed algebraic

$$\mathcal{S}_d : \begin{cases} 0 & = & A_s\mathbf{x_s}(t) - E_s\mathbf{x_s}(t+h) + B_s\mathbf{u}(t) \\ \mathbf{y}(t) & = & C_s\mathbf{x_s}(t) + D_s\mathbf{u}(t) \end{cases}$$

The exact $z \leftrightarrow s$ correspondance

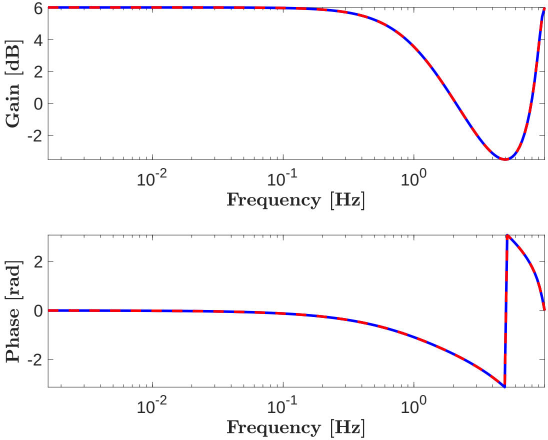
Example #1

$$\mathbf{H}_s(z) = \frac{1}{z-0.5} \text{ and } \mathbf{H}_s(s) = \frac{1}{e^{sh}-0.5}$$

```
Ad  = .5;
h   = .1;
wn  = pi/h;
W   = logspace(-2,log10(2*wn),200);
Hs  = @(s) 1/(exp(s*h)-Ad);
Hd  = ss(Ad,1,1,0,h);
Fd  = freqresp(Hd,W);
figure
subplot(211); mor.bode(Hs,'b-',{Fd},'r--',W)
subplot(212); mor.bode(Hs,'b-',{Fd},'r--',W,'phase')
```

The exact $z \leftrightarrow s$ correspondance

Example #1



The exact $z \leftrightarrow s$ correspondance

Example #2 (trying to convince)

- $h = 0.2$, $\omega_n \approx 31 rad/s$, $\omega_N \approx 15,7 rad/s$ ($f_N = 1/h/2 = 2.5$Hz)
- Zero order hold, '`zoh`' vs. Bilinear transform, '`tustin`' vs. exact $z = e^{sh}$ ...

$$\mathbf{H}(s) = \frac{1}{(s^2/10 + .1s/\sqrt{10} + 1)(s^2/25 + .1s/5 + 1)}$$

$$\mathbf{H}_z(z) = \frac{0.0154z^3 + 0.1518z^2 + 0.1468z + 0.01396}{z^4 - 2.594z^3 + 3.454z^2 - 2.382z + 0.8494}$$

$$\mathbf{H}_t(z) = \frac{0.01699z^4 + 0.06798z^3 + 0.102z^2 + 0.06798z + 0.01699}{z^4 - 2.744z^3 + 3.703z^2 - 2.558z + 0.8715}$$

The exact $z \leftrightarrow s$ correspondance

Example #2 (trying to convince)

- $h = 0.2$, $\omega_n \approx 31 rad/s$, $\omega_N \approx 15,7 rad/s$ ($f_N = 1/h/2 = 2.5$Hz)
- Zero order hold, `'zoh'` vs. Bilinear transform, `'tustin'` vs. exact $z = e^{sh}$ ...

$$\mathbf{H}(s) = \frac{1}{(s^2/10 + .1s/\sqrt{10} + 1)(s^2/25 + .1s/5 + 1)}$$

$$\mathbf{H}_z(z) = \frac{0.0154z^3 + 0.1518z^2 + 0.1468z + 0.01396}{z^4 - 2.594z^3 + 3.454z^2 - 2.382z + 0.8494}$$

$$\mathbf{H}_t(z) = \frac{0.01699z^4 + 0.06798z^3 + 0.102z^2 + 0.06798z + 0.01699}{z^4 - 2.744z^3 + 3.703z^2 - 2.558z + 0.8715}$$

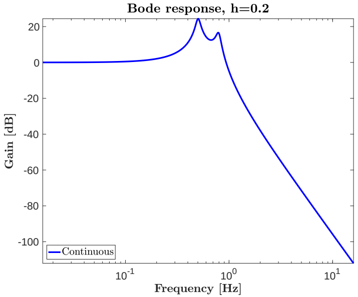
$$\mathbf{H}_e(z) = \frac{1}{\left((\ln(z)/h)^2/10 + .1\ln(z)/h/\sqrt{10} + 1\right)\left((\ln(z)/h)^2/25 + .1\ln(z)/h/5 + 1\right)}$$

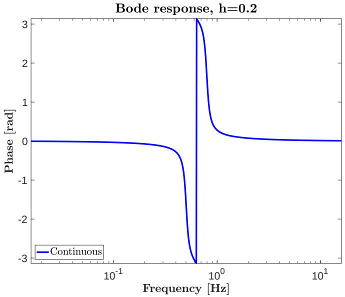
Irrational form !

The exact $z \leftrightarrow s$ correspondance

Example #2 (trying to convince)

- $h = 0.2$, $\omega_n \approx 31 rad/s$, $\omega_N \approx 15,7 rad/s$ ($f_N = 1/h/2 = 2.5$Hz)
- Zero order hold, 'zoh' vs. Bilinear transform, 'tustin' vs. exact $z = e^{sh}$ ...

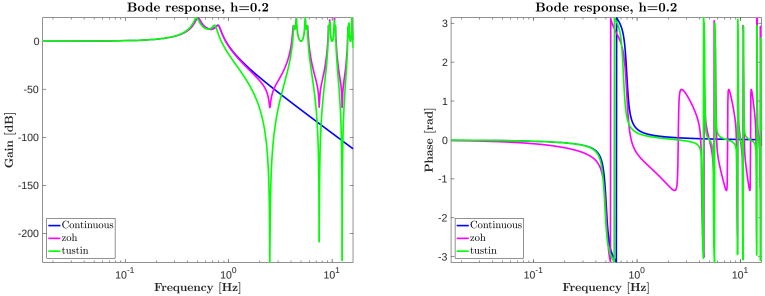




The exact $z \leftrightarrow s$ correspondance

Example #2 (trying to convince)

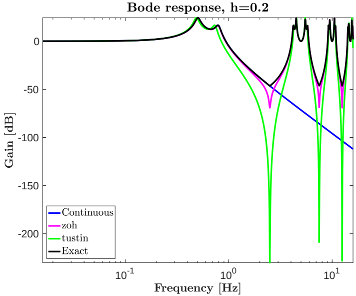
- $h = 0.2$, $\omega_n \approx 31 rad/s$, $\omega_N \approx 15,7 rad/s$ ($f_N = 1/h/2 = 2.5$Hz)
- Zero order hold, `'zoh'` vs. Bilinear transform, `'tustin'` vs. exact $z = e^{sh}$ ...

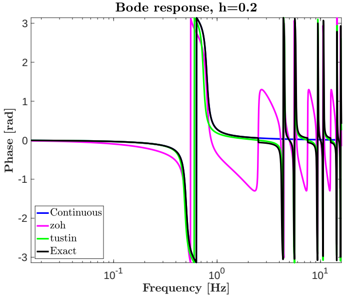


The exact $z \leftrightarrow s$ correspondance

Example #2 (trying to convince)

- $h = 0.2$, $\omega_n \approx 31 rad/s$, $\omega_N \approx 15,7 rad/s$ ($f_N = 1/h/2 = 2.5$Hz)
- Zero order hold, `'zoh'` vs. Bilinear transform, `'tustin'` vs. exact $z = e^{sh}$ ...

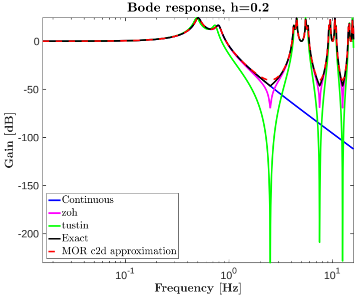


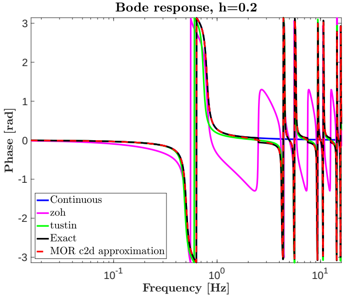


The exact $z \leftrightarrow s$ correspondance

Example #2 (trying to convince)

- $h = 0.2$, $\omega_n \approx 31 rad/s$, $\omega_N \approx 15,7 rad/s$ ($f_N = 1/h/2 = 2.5$Hz)
- Zero order hold, '`zoh`' vs. Bilinear transform, '`tustin`' vs. exact $z = e^{sh}$ ...





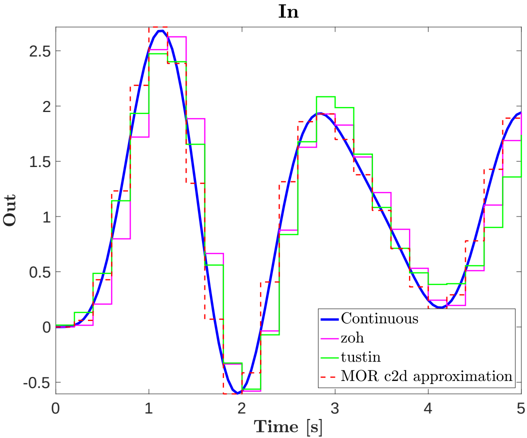
$$\mathbf{H}_r(z) = \frac{0.05964z^3 + 0.2134z^2 + 0.0552z - 0.0001461}{z^4 - 2.594z^3 + 3.454z^2 - 2.382z + 0.8494}$$

Rational approximation

The exact $z \leftrightarrow s$ correspondance

Example #2 (trying to convince)

- $h = 0.2$, $\omega_n \approx 31 rad/s$, $\omega_N \approx 15,7 rad/s$ ($f_N = 1/h/2 = 2.5$Hz)
- Zero order hold, '`zoh`' vs. Bilinear transform, '`tustin`' vs. exact $z = e^{sh}$ ...



Outlines

Mixed interconnection

Continuous-time plant

Let us consider $\mathbf{H}$, a n_u inputs, n_y outputs linear dynamical system described by the **complex-valued function of order** n (n large or ∞) equipped with realisation $\mathcal{S}$

$$\mathcal{S} : \begin{cases} E\dot{\mathbf{x}}(t) &= A\mathbf{x}(t) + B\mathbf{u}(t) \in \mathbb{R}^n \\ \mathbf{y}(t) &= C\mathbf{x}(t) + D\mathbf{u}(t) \in \mathbb{R}^{n_y} \end{cases}, \mathbf{y}(s) = \mathbf{H}(s)\mathbf{u}(s) \in \mathbb{C}^{n_y}$$

Mixed interconnection

Continuous-time plant

Let us consider $\mathbf{H}$, a n_u inputs, n_y outputs linear dynamical system described by the **complex-valued function of order** n (n large or ∞) equipped with realisation $\mathcal{S}$

$$\mathcal{S}:\left\{\begin{array}{ccc} E\dot{\mathbf{x}}(t) & = & A\mathbf{x}(t)+B\mathbf{u}(t) \in \mathbb{R}^n \\ \mathbf{y}(t) & = & C\mathbf{x}(t)+D\mathbf{u}(t) \in \mathbb{R}^{n_y} \end{array}\right. , \mathbf{y}(s)=\mathbf{H}(s)\mathbf{u}(s) \in \mathbb{C}^{n_y}$$

Continuous-time controller

and $\mathbf{C}$, a n_y inputs, n_u outputs linear dynamical system described by the **complex-valued function of order** n_c equipped with realisation $\mathcal{C}$

$$\mathcal{C}:\left\{\begin{array}{ccc} \dot{\mathbf{x}}_c(t) & = & A_c\mathbf{x}_c(t)+B_c\mathbf{y}_c(t) \in \mathbb{R}^{n_c} \\ \mathbf{u}(t) & = & C_c\mathbf{x}_c(t)+D_c\mathbf{y}(t) \in \mathbb{R}^{n_u} \end{array}\right. , \mathbf{u}(s)=\mathbf{C}(s)\mathbf{y}(s) \in \mathbb{C}^{n_u}$$

Mixed interconnection

Continuous-time plant

Let us consider $\mathbf{H}$, a n_u inputs, n_y outputs linear dynamical system described by the **complex-valued function of order** n (n large or ∞) equipped with realisation $\mathcal{S}$

$$\mathcal{S} : \begin{cases} E\dot{\mathbf{x}}(t) &= A\mathbf{x}(t) + B\mathbf{u}(t) \in \mathbb{R}^n \\ \mathbf{y}(t) &= C\mathbf{x}(t) + D\mathbf{u}(t) \in \mathbb{R}^{n_y} \end{cases}, \mathbf{y}(s) = \mathbf{H}(s)\mathbf{u}(s) \in \mathbb{C}^{n_y}$$

Discrete-time controller (using your favorite discretisation method)

and $\mathbf{C}_s$, a n_y inputs, n_u outputs linear dynamical system described by the **complex-valued function of order** n_c equipped with realisation $\mathcal{C}_s$

$$\mathcal{C}_s : \begin{cases} \mathbf{x_s}(t_k + h) &= A_s\mathbf{x_s}(t_k) + B_s\mathbf{y}(t_k) \in \mathbb{R}^{n_c} \\ \mathbf{u}(t_k) &= C_s\mathbf{x_s}(t_k) + D_s\mathbf{y}(t_k) \in \mathbb{R}^{n_u} \end{cases}, \mathbf{u}(z) = \mathbf{C}_s(z)\mathbf{y}(z) \in \mathbb{C}^{n_u}$$

Mixed interconnection

The interconnection with block holder

A block holder transfer function $\mathbf{H}_{BOZ} : u_k \to u(t)$ can be seen as a memory a step shifted by another step at time h:

$$\mathbf{H}_{ZOH}(s) = \frac{1}{s} - \frac{e^{-sh}}{s} = \frac{1-e^{-sh}}{s} = \frac{\mathbf{u}_h(s)}{\mathbf{u}(s)}$$

It can be characterised by the **ODE** $\dot{\mathbf{u}}_h(t) = \mathbf{u}(t) - \mathbf{u}(t-h)$. As $\mathbf{u}(t) = C_s \mathbf{x}_s(t)$ we have

$$\dot{\mathbf{x}}_h(t) = C_s \mathbf{x}_s(t) - C_s \mathbf{x}_s(t-h)$$

Mixed interconnection

$$\left\{\begin{array}{rcll} \dot{\mathbf{x}}(t) & = & A\mathbf{x}(t) + B(r(t) - \mathbf{u}_h(t)) & \text{plant dyn. w. ref.} \\ \mathbf{y}(t) & = & C\mathbf{x}(t) & \text{plant output} \\ 0 & = & A_s\mathbf{x}_s(t) - I\mathbf{x}_s(t+h) + B_s\mathbf{y}(t) & \text{ctl alg. constraint w. sampling delay} \\ \dot{\mathbf{u}}_h(t) & = & C_s\mathbf{x}_s(t) - C_s\mathbf{x}_s(t-h) & \text{ctl dyn. w. holder delay} \\ \mathbf{u}(t) & = & C_s\mathbf{x}_s(t) & \text{ctl output} \end{array}\right.$$

$$\begin{pmatrix} I & 0 & 0 \\ 0 & I & 0 \\ 0 & 0 & 0 \end{pmatrix}\begin{pmatrix} \dot{\mathbf{x}}(t) \\ \dot{\mathbf{u}}_h(t) \\ \dot{\mathbf{x}}_s(t) \end{pmatrix} = \begin{pmatrix} A & -B & 0 \\ 0 & 0 & C_s \\ B_sC & 0 & A_s \end{pmatrix}\begin{pmatrix} \mathbf{x}(t) \\ \mathbf{u}_h(t) \\ \mathbf{x}_s(t) \end{pmatrix} + \begin{pmatrix} B \\ 0 \\ 0 \end{pmatrix} r(t) + \begin{pmatrix} 0 \\ 0 \\ -I \end{pmatrix}\mathbf{x}_s(t+h) + \begin{pmatrix} 0 \\ -C_s \\ 0 \end{pmatrix}\mathbf{x}_s(t-h)$$

A linear system with delays and algebraic constraint... what to do?

Model interpolation enters the game

Rational interpolation framework

(Tangential) interpolation is the path to optimal $\mathcal{H}_2$ approximation

SISO model: given $\mathbf{H}$, seek a reduced-order system $\hat{\mathbf{H}}$, such that

$$\begin{array}{rcll} \hat{\mathbf{H}}(\mu_i) & = & \mathbf{H}(\mu_i) & i=1,\dots,q \\ \hat{\mathbf{H}}(\lambda_j) & = & \mathbf{H}(\lambda_j) & j=1,\dots,k \end{array}$$

S. Gugercin and A C. Antoulas and C A. Beattie, *"$\mathcal{H}_2$ Model Reduction for Large Scale Linear Dynamical Systems"*, SIAM Journal on Matrix Analysis and Applications, vol. 30(2), June 2008, pp. 609-638.

A.J. Mayo and A.C. Antoulas, *"A framework for the solution of the generalized realization problem"*, Linear Algebra and its Applications, 425(2-3), 2007, pp 634-662.

Model interpolation enters the game

Rational interpolation framework

(Tangential) interpolation is the path to optimal $\mathcal{H}_2$ approximation

SISO model: given $\mathbf{H}$, seek a reduced-order system $\hat{\mathbf{H}}$, such that

$$\begin{array}{rcll} \hat{\mathbf{H}}(\mu_i) & = & \mathbf{H}(\mu_i) & i=1,\dots,q \\ \hat{\mathbf{H}}(\lambda_j) & = & \mathbf{H}(\lambda_j) & j=1,\dots,k \end{array}$$

MIMO model (tangential): in a similar way, given $\mathbf{H}$, seek $\hat{\mathbf{H}}$, such that

$$\begin{array}{rcll} \mathbf{l}_i^H\hat{\mathbf{H}}(\mu_i) & = & \mathbf{l}_i^H\mathbf{H}(\mu_i) & i=1,\dots,q \\ \hat{\mathbf{H}}(\lambda_j)\mathbf{r}_j & = & \mathbf{H}(\lambda_j)\mathbf{r}_j & j=1,\dots,k \end{array}$$

S. Gugercin and A C. Antoulas and C A. Beattie, *"$\mathcal{H}_2$ Model Reduction for Large Scale Linear Dynamical Systems"*, SIAM Journal on Matrix Analysis and Applications, vol. 30(2), June 2008, pp. 609-638.

A.J. Mayo and A.C. Antoulas, *"A framework for the solution of the generalized realization problem"*, Linear Algebra and its Applications, 425(2-3), 2007, pp 634-662.

Model interpolation enters the game

Rational interpolation in the Loewner framework

Given $\mathbf{H}(s)$ and $\{\mu_1, \ldots, \mu_q\} \in \mathbb{C}$, $\{\lambda_1, \ldots, \lambda_k\} \in \mathbb{C}$, we seek $\hat{\mathbf{H}}$, s.t.

$$
\begin{array}{rcll}
\hat{\mathbf{H}}(\mu_i) & = & \mathbf{H}(\mu_i) & i = 1, \ldots, q \\
\hat{\mathbf{H}}(\lambda_j) & = & \mathbf{H}(\lambda_j) & j = 1, \ldots, k
\end{array}
$$

Model interpolation enters the game

Rational interpolation in the Loewner framework

Given $\mathbf{H}(s)$ and $\{\mu_1, \ldots, \mu_q\} \in \mathbb{C}$, $\{\lambda_1, \ldots, \lambda_k\} \in \mathbb{C}$, we seek $\hat{\mathbf{H}}$, s.t.

$$\begin{array}{rcll} \hat{\mathbf{H}}(\mu_i) & = & \mathbf{H}(\mu_i) & i = 1, \ldots, q \\ \hat{\mathbf{H}}(\lambda_j) & = & \mathbf{H}(\lambda_j) & j = 1, \ldots, k \end{array}$$

$$\mathbb{L} = \begin{bmatrix} \frac{\mathbf{H}(\mu_1)-\mathbf{H}(\lambda_1)}{\mu_1-\lambda_1} & \ldots & \frac{\mathbf{H}(\mu_1)-\mathbf{H}(\lambda_k)}{\mu_1-\lambda_k} \\ \vdots & \ddots & \vdots \\ \frac{\mathbf{H}(\mu_q)-\mathbf{H}(\lambda_1)}{\mu_q-\lambda_1} & \ldots & \frac{\mathbf{H}(\mu_q)-\mathbf{H}(\lambda_k)}{\mu_q-\lambda_k} \end{bmatrix} \in \mathbb{C}^{q\times k}$$

$$\mathbb{L}_\sigma = \begin{bmatrix} \frac{\mu_1\mathbf{H}(\mu_1)-\mathbf{H}(\lambda_1)\lambda_1}{\mu_1-\lambda_1} & \ldots & \frac{\mu_1\mathbf{H}(\mu_1)-\mathbf{H}(\lambda_k)\lambda_k}{\mu_1-\lambda_k} \\ \vdots & \ddots & \vdots \\ \frac{\mu_q\mathbf{H}(\mu_q)-\mathbf{H}(\lambda_1)\lambda_1}{\mu_q-\lambda_1} & \ldots & \frac{\mu_q\mathbf{H}(\mu_q)-\mathbf{H}(\lambda_k)\lambda_k}{\mu_q-\lambda_k} \end{bmatrix} \in \mathbb{C}^{q\times k}$$

$\mathbf{W} = \begin{bmatrix} \mathbf{H}(\sigma_1) & \ldots & \mathbf{H}(\sigma_r) \end{bmatrix}$ and $\mathbf{V}^T = \begin{bmatrix} \mathbf{H}(\sigma_1) & \ldots & \mathbf{H}(\sigma_r) \end{bmatrix}$

$\hat{\mathbf{H}}(s) = \mathbf{W}(\mathbb{L}_\sigma - s\mathbb{L})^{-1}\mathbf{V}$ $\Rightarrow$ Rational interpolation

Model interpolation enters the game

Rational interpolation in the Loewner framework

Given $\mathbf{H}(s)$ and $\{\sigma_1,\dots,\sigma_r\}=\{\mu_1,\dots,\mu_q\}=\{\lambda_1,\dots,\lambda_k\}\in\mathbb{C}$, we seek $\hat{\mathbf{H}}$, s.t.

$$\begin{array}{rcll}\hat{\mathbf{H}}(\sigma_i) & = & \mathbf{H}(\sigma_i) & i=1,\dots,r\\ \hat{\mathbf{H}}'(\sigma_i) & = & \mathbf{H}'(\sigma_i) & \end{array}$$

Model interpolation enters the game

Rational interpolation in the Loewner framework

Given $\mathbf{H}(s)$ and $\{\sigma_1,\dots,\sigma_r\}=\{\mu_1,\dots,\mu_q\}=\{\lambda_1,\dots,\lambda_k\}\in\mathbb{C}$, we seek $\hat{\mathbf{H}}$, s.t.

$$\begin{array}{rcll} \hat{\mathbf{H}}(\sigma_i) & = & \mathbf{H}(\sigma_i) & i=1,\dots,r \\ \hat{\mathbf{H}}'(\sigma_i) & = & \mathbf{H}'(\sigma_i) & \end{array}$$

$$\mathbb{L} = \begin{bmatrix} \mathbf{H}'(\sigma_1) & \dots & \frac{\mathbf{H}(\sigma_1)-\mathbf{H}(\sigma_r)}{\sigma_1-\sigma_r} \\ \vdots & \ddots & \vdots \\ \frac{\mathbf{H}(\sigma_r)-\mathbf{H}(\sigma_1)}{\sigma_r-\sigma_1} & \dots & \mathbf{H}'(\sigma_r) \end{bmatrix} \in \mathbb{C}^{r\times r}$$

$$\mathbb{L}_\sigma = \begin{bmatrix} (s\mathbf{H}(s))'_{s=\sigma_1} & \dots & \frac{\sigma_1\mathbf{H}(\sigma_1)-\sigma_r\mathbf{H}(\sigma_r)}{\sigma_1-\sigma_r} \\ \vdots & \ddots & \vdots \\ \frac{\sigma_r\mathbf{H}(\sigma_r)-\sigma_1\mathbf{H}(\sigma_1)}{\sigma_r-\sigma_1} & \dots & (s\mathbf{H}(s))'_{s=\sigma_r} \end{bmatrix} \in \mathbb{C}^{r\times r}$$

$$\mathbf{W} = \begin{bmatrix} \mathbf{H}(\sigma_1) & \dots & \mathbf{H}(\sigma_r) \end{bmatrix} \text{ and } \mathbf{V}^T = \begin{bmatrix} \mathbf{H}(\sigma_1) & \dots & \mathbf{H}(\sigma_r) \end{bmatrix}$$

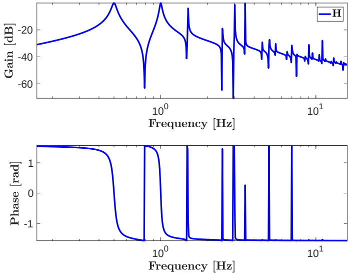
$\hat{\mathbf{H}}(s) = \mathbf{W}(\mathbb{L}_\sigma - \mathbb{L}s)^{-1}\mathbf{V}$ $\Rightarrow$ Hermite interpolation

Model interpolation enters the game

Example #3

Vibrating string (ends fixed & control and observation distributed along the string)

$$\mathbf{H}(s) = \frac{\frac{s}{2}\sinh(s) + 2\cosh(\frac{s}{2}) - 3\cosh^2(\frac{s}{2}) + 1}{s(s+\frac{1}{2})\sinh(s) + 2\cosh(\frac{s}{2}) - 3\cosh^2(\frac{s}{2}) + 1}$$

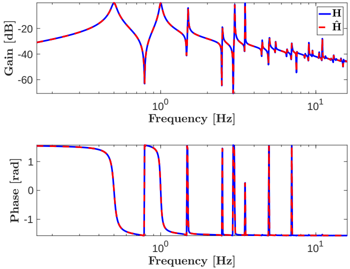


R. Curtain and K. Morris, *"Transfer functions of distributed parameter systems: A tutorial"*, Automatica, 45(5), 2009, pp. 1101-1116.

Model interpolation enters the game

Example #3

```
H    = @(s) (s/2*sinh(s)+2*cosh(s/2)-3*cosh(s/2)^2+1 ) / ...
FR   = mor.bode(H,W);
Hr   = mor.lti({W,FR},[]); % Exact rational approximation
Hred = mor.lti(Hr,r);      % H2 approximation r=40,30,20,10
```

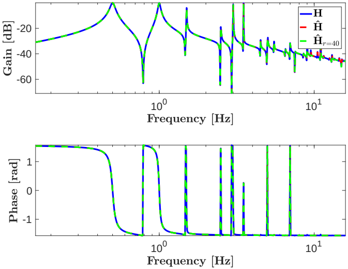


R. Curtain and K. Morris, *"Transfer functions of distributed parameter systems: A tutorial"*, Automatica, 45(5), 2009, pp. 1101-1116.

Model interpolation enters the game

Example #3

```
H    = @(s) (s/2*sinh(s)+2*cosh(s/2)-3*cosh(s/2)^2+1 ) / ...
FR   = mor.bode(H,W);
Hr   = mor.lti({W,FR},[]); % Exact rational approximation
Hred = mor.lti(Hr,r);      % H2 approximation r=40,30,20,10
```

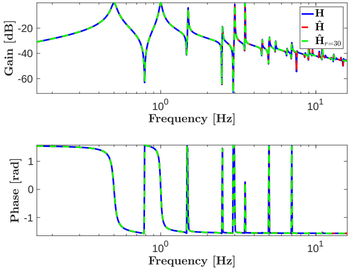


R. Curtain and K. Morris, *"Transfer functions of distributed parameter systems: A tutorial"*, Automatica, 45(5), 2009, pp. 1101-1116.

Model interpolation enters the game

Example #3

```
H     = @(s) (s/2*sinh(s)+2*cosh(s/2)-3*cosh(s/2)^2+1 ) / ...
FR    = mor.bode(H,W);
Hr    = mor.lti({W,FR},[]); % Exact rational approximation
Hred  = mor.lti(Hr,r);      % H2 approximation r=40,30,20,10
```

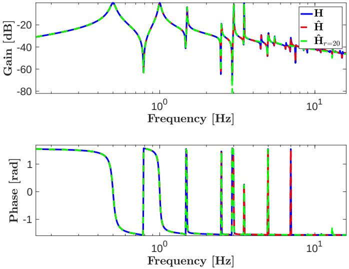


R. Curtain and K. Morris, *"Transfer functions of distributed parameter systems: A tutorial"*, Automatica, 45(5), 2009, pp. 1101-1116.

Model interpolation enters the game

Example #3

```
H     = @(s) (s/2*sinh(s)+2*cosh(s/2)-3*cosh(s/2)^2+1 ) / ...
FR    = mor.bode(H,W);
Hr    = mor.lti({W,FR},[]); % Exact rational approximation
Hred  = mor.lti(Hr,r);      % H2 approximation r=40,30,20,10
```

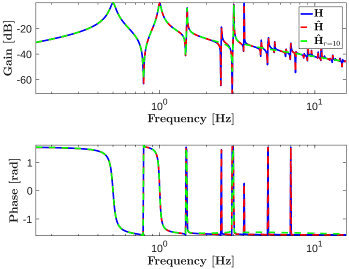


R. Curtain and K. Morris, *"Transfer functions of distributed parameter systems: A tutorial"*, Automatica, 45(5), 2009, pp. 1101-1116.

Model interpolation enters the game

Example #3

```
H    = @(s) (s/2*sinh(s)+2*cosh(s/2)-3*cosh(s/2)^2+1 ) / ...
FR   = mor.bode(H,W);
Hr   = mor.lti({W,FR},[]); % Exact rational approximation
Hred = mor.lti(Hr,r);      % H2 approximation r=40,30,20,10
```



R. Curtain and K. Morris, *"Transfer functions of distributed parameter systems: A tutorial"*, Automatica, 45(5), 2009, pp. 1101-1116.

Mixed continuous-sampled-time delay margin estimation

Hybrid Systems Delay Margin Algorithm (HSDMA)

Require: $\mathbf{H}(s)$ and $\mathbf{H}_s(z)$

1: Set $N \in \mathbb{N}$ s.t. $N \gg \dim(A_s)$

2: Set frequency grid $\{\omega_i\}_{i=1}^N \in \mathbb{R}_+$ satisfying

$$0 < \omega_i \leq \pi/h$$

3: Compute the frequency response of $\mathbf{H}_s(z)$over $\{\omega_i\}_{i=1}^N$ as

$$\{\Phi_i\}_{i=1}^N = \mathbf{H}_s(e^{\imath\omega_i h})$$

4: Given $\{e^{\imath\omega_i h}, \Phi_i\}_{i=1}^N$, compute the approximate model $\hat{\mathbf{H}}_s \in \mathcal{H}_\infty$ or $\in \mathcal{L}_\infty$ satisfying, for $i = 1, \ldots, N$

$$\hat{\mathbf{H}}_s(\imath\omega_i) = \Phi_i$$

5: [opt.] If $\mathbf{H}_s(z)$ is stable (*i.e.* $\in \mathbb{D}$), enforce $\hat{\mathbf{H}}_s(s)$ to be stable too (*i.e.* $\in \mathcal{H}_\infty$)

6: Compute $\mathbf{L}(s) = \hat{\mathbf{H}}_s(s)\mathbf{H}(s)$

7: Compute the delay margin **DM**, based on $\mathbf{L}(s)$

Mixed continuous-sampled-time delay margin estimation

Hybrid Systems Delay Margin Algorithm (HSDMA)

Require: $\mathbf{H}(s)$ and $\mathbf{H}_s(z)$

1: Set $N \in \mathbb{N}$ s.t. $N \gg \dim(A_s)$

2: Set frequency grid $\{\omega_i\}_{i=1}^N \in \mathbb{R}_+$ satisfying

$$0 < \omega_i \leq \pi/h$$

3: Compute the frequency response of $\mathbf{H}_s(z)$over $\{\omega_i\}_{i=1}^N$ as

$$\{\Phi_i\}_{i=1}^N = \mathbf{H}_s(e^{\imath\omega_i h})$$

4: Given $\{e^{\imath\omega_i h}, \Phi_i\}_{i=1}^N$, compute the approximate model $\hat{\mathbf{H}}_s \in \mathcal{H}_\infty$ or $\in \mathcal{L}_\infty$ satisfying, for $i = 1, \dots, N$

$$\hat{\mathbf{H}}_s(\imath\omega_i) = \Phi_i$$

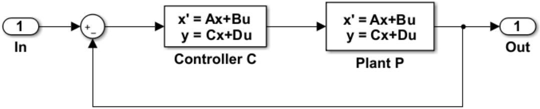
5: [opt.] If $\mathbf{H}_s(z)$ is stable (*i.e.* $\in \mathbb{D}$), enforce $\hat{\mathbf{H}}_s(s)$ to be stable too (*i.e.* $\in \mathcal{H}_\infty$)

6: Compute $\mathbf{L}(s) = \hat{\mathbf{H}}_s(s)\mathbf{H}(s)$

7: Compute the delay margin **DM**, based on $\mathbf{L}(s)$

Mixed continuous-sampled-time delay margin estimation

Example #4

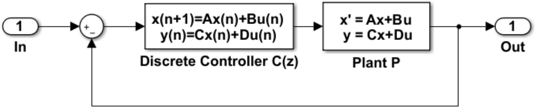


$$\texttt{P}: \begin{cases} \dot{\mathbf{x}}(t) = \begin{bmatrix} -10 & -5 \\ 4 & 0 \end{bmatrix} \mathbf{x}(t) + \begin{bmatrix} 0.5 \\ 0 \end{bmatrix} \mathbf{u}(t) \\ \mathbf{y}(t) = \begin{bmatrix} 0 & 0.5 \end{bmatrix} \mathbf{x}(t) \end{cases} \quad (1)$$

$$\texttt{C}: \begin{cases} \dot{\mathbf{x}}_s(t) = \begin{bmatrix} -0.001 & 7,854 \\ 0 & -62.83 \end{bmatrix} \mathbf{x}(t) + \begin{bmatrix} 0 \\ 8 \end{bmatrix} \mathbf{y}(t) \\ \mathbf{u}(t) = \begin{bmatrix} 70 & 235.6 \end{bmatrix} \mathbf{x}(t) \end{cases} \quad (2)$$

Mixed continuous-sampled-time delay margin estimation

Example #4



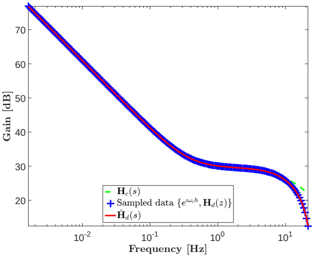
$$\mathtt{P}: \begin{cases} \dot{\mathbf{x}}(t) &= \begin{bmatrix} -10 & -5 \\ 4 & 0 \end{bmatrix} \mathbf{x}(t) + \begin{bmatrix} 0.5 \\ 0 \end{bmatrix} \mathbf{u}(t) \\ \mathbf{y}(t) &= \begin{bmatrix} 0 & 0.5 \end{bmatrix} \mathbf{x}(t) \end{cases} \tag{3}$$

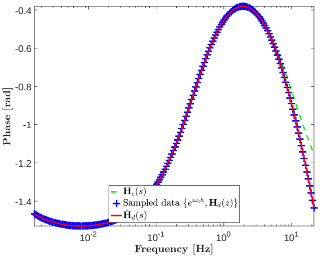
$$\mathtt{C}: \begin{cases} \mathbf{x_s}(t_k+h) &= A_s\mathbf{x_s}(t_k) + B_s\mathbf{y}(t_k) \\ \mathbf{u}(t_k) &= C_s\mathbf{x_s}(t_k) + D_s\mathbf{y}(t_k) \end{cases} \tag{4}$$

Mixed continuous-sampled-time delay margin estimation

Example #4

Approximate the discretised controller

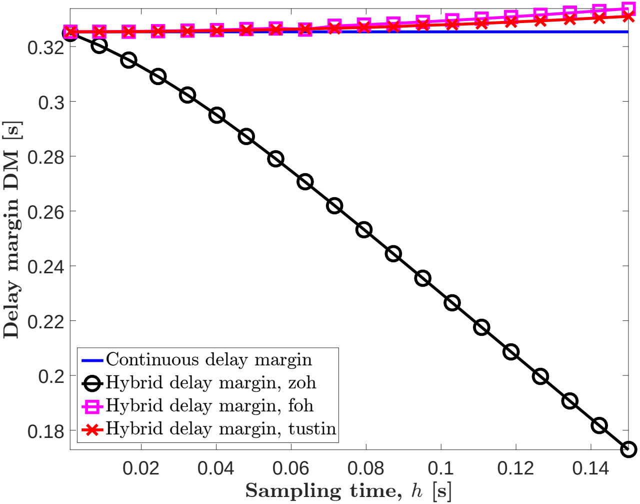




Note the mismatch in high frequency

Mixed continuous-sampled-time delay margin estimation

Example #4



Outlines

What has been said

Mixed continuous-sampled-time dynamical models

- turned as a delayed **DAE**
- or an infinite irrational transfer function...
- ... which has been approximated using the interpolatory framework (with some "guarantee")
- ... and embedded into (delay) margin estimation process

What to do?

- investigate the index of the **DAE**
- ... to treat it in the interpolation context
- ... extend to multi-sampling h
- ... link with previous talk

What has been said

Mixed continuous-sampled-time dynamical models

- turned as a delayed **DAE**
- or an infinite irrational transfer function...
- ... which has been approximated using the interpolatory framework (with some "guarantee")
- ... and embedded into (delay) margin estimation process

What to do?

- investigate the index of the **DAE**
- ... to treat it in the interpolation context
- ... extend to multi-sampling h
- ... link with previous talk

Interpolatory framework for mixed continuous-sampled-data models interconnection analysis

C. Poussot-Vassal

ONERA
THE FRENCH AEROSPACE LAB

2019, Toulouse, France
(coll. w. P-L. Garoche, T. Loquen, C. Pagetti and P. Vuillemin)

FEANICSES