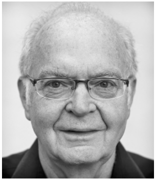
A glimpse at Sampled-Data systems

Nothing new, just some elements from literature

Pierre Vuillemin (ONERA / DTIS / COVNI)

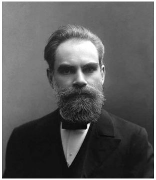
FEANICSES 2019 Workshop

Idea behind the research team COVNI:



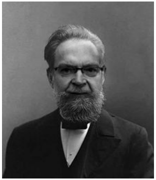
computer theory

+



automatic control

=

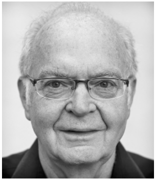
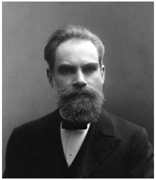
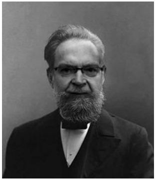


something interesting for computer-controlled systems

[1] Boyd and Barratt. *Linear controller design: limits of performance*. 1991

[2] Chen and Francis. *Optimal sampled-data control systems*. 1995

Idea behind the research team COVNI:

				
computer theory	+	automatic control	=	something interesting for computer-controlled systems

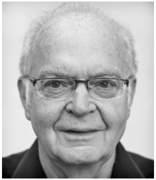
How do we assess dynamical performances of an implemented control-law?

- give some insights concerning what "performance" may refer to for control people
- introduce some concepts from sampled-data systems theory
- talk restricted to Linear Time Invariant (LTI) models

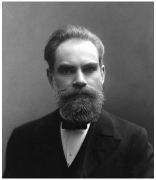
[1] Boyd and Barratt. *Linear controller design: limits of performance*. 1991

[2] Chen and Francis. *Optimal sampled-data control systems*. 1995

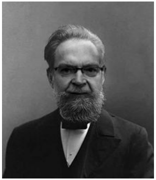
Idea behind the research team COVNI:


computer theory

+


automatic control

=


something interesting for computer-controlled systems

How do we assess dynamical performances of an implemented control-law?

- give some insights concerning what "performance" may refer to for control people
- introduce some concepts from sampled-data systems theory
- talk restricted to Linear Time Invariant (LTI) models

References:

- a comprehensive introduction to control engineering can be found in the first chapters ($1 \rightarrow 3$) of [1]
- what will be presented on sampled-data systems can be found in [2]

---

[1] Boyd and Barratt. *Linear controller design: limits of performance*. 1991

[2] Chen and Francis. *Optimal sampled-data control systems*. 1995

Outline

What is control engineering all about? I

In a nutshell

$$\textit{dynamical system} + \textit{actuators}/\textit{sensors} + \underbrace{\textit{control objectives}}_{\text{tracking, disturbance rejection, etc.}} = \textit{control problem} \quad (1)$$

What is control engineering all about? I

In a nutshell

$$\textit{dynamical system} + \textit{actuators}/\textit{sensors} + \underbrace{\textit{control objectives}}_{\text{tracking, disturbance rejection, etc.}} = \textit{control problem} \quad (1)$$

And in practice...?

Different philosophies: pole placement, loop-shaping, convex synthesis...

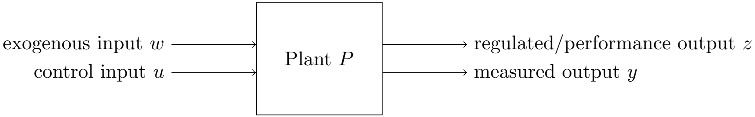
$\hookrightarrow$ *Here: focus on the approach relying on the specification of a performance function to be optimized* $\sim$ *optimal feedback control*

What is control engineering all about? I

In a nutshell

$$\textit{dynamical system} + \textit{actuators}/\textit{sensors} + \underbrace{\textit{control objectives}}_{\text{tracking, disturbance rejection, etc.}} = \textit{control problem} \quad (1)$$

And in practice...?
Different philosophies: pole placement, loop-shaping, convex synthesis...
$\hookrightarrow$ *Here: focus on the approach relying on the specification of a performance function to be optimized* $\sim$ *optimal feedback control*



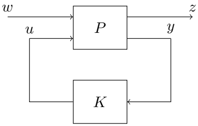
Modelling of the control problem = derive the generalized plant P:

- models for the dynamical system + actuators/sensors $\rightarrow$ this sets u and y
- identify reference signals, possible disturbances or noise $\rightarrow$ w
- select signals of interest for the control objective $\rightarrow$ z

What is control engineering all about? II

Standard form

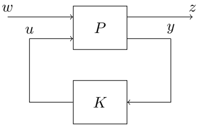
The loop is closed with some controller K,



- most of the control architectures can be represented as this standard form,
- K can have different structures: a gain, PI, PID, etc.
- the closed-loop is the transfer from w to z, denoted $T_{w \to z}(P, K)$,
- in general, one wants K to stabilize (internally) the plant P.

What is control engineering all about? II

Standard form
The loop is closed with some controller K,



- most of the control architectures can be represented as this standard form,
- K can have different structures: a gain, PI, PID, etc.
- the closed-loop is the transfer from w to z, denoted $T_{w \to z}(P, K)$,
- in general, one wants K to stabilize (internally) the plant P.

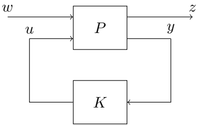
Optimal control
Find K that stabilizes P internally and minimizes $\|T_{w \to z}\|$

For fixed plant P, performance of the closed-loop = value of $\|T_{w \to z}\|$

What is control engineering all about? II

Standard form

The loop is closed with some controller K,



- most of the control architectures can be represented as this standard form,
- K can have different structures: a gain, PI, PID, etc.
- the closed-loop is the transfer from w to z, denoted $T_{w\to z}(P, K)$,
- in general, one wants K to stabilize (internally) the plant P.

Optimal control

Find K that stabilizes P internally and minimizes $\|T_{w\to z}\|$

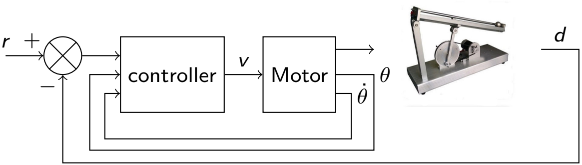
For fixed plant P, performance of the closed-loop $=$ value of $\|T_{w\to z}\|$

Linear-Time Invariant case

- practical interpretation of $T_{w\to z}$ convenient in the frequency-domain (transfer function)
- useful systems norms: $\mathcal{H}_\infty$, $\mathcal{H}_2$

What is control engineering all about? III

A tracking example: ball on the beam

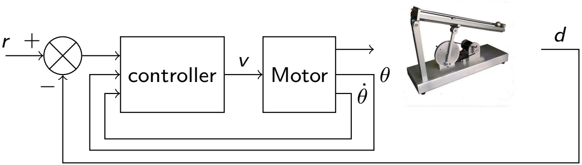


Physical modelling

- dynamical sytem: beam + ball,
- actuator: motor e.g. controlled by a voltage $v \rightarrow u$
- sensed signals: the distance d of the ball to the center of the beam, the angular position θ and velocity $\dot{\theta}$ of the shaft of the motor $\rightarrow y = [r - d, \theta, \dot{\theta}]^T$

What is control engineering all about? III

A tracking example: ball on the beam



Physical modelling

- dynamical sytem: beam + ball,
- actuator: motor e.g. controlled by a voltage $v \rightarrow u$
- sensed signals: the distance d of the ball to the center of the beam, the angular position θ and velocity $\dot{\theta}$ of the shaft of the motor $\rightarrow y = [r - d, \theta, \dot{\theta}]^T$

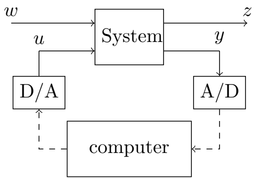
Signals for the control problem:

- tracking objective $\rightarrow z = r - d$
- only exogenous input $w = r$

The *performance channel* $T_{w \rightarrow z}$ describes how fast the system will follow the reference. It can be filtered, e.g. with a low-pass filter.

Fifty shades of models

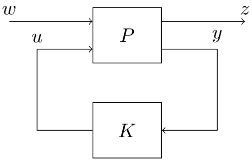
"Real world"



- need a model

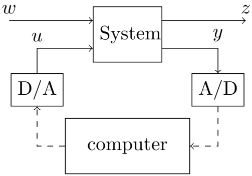
Fifty shades of models

Continuous-time/analog modelling



- idealized (LTI)
- lots of techniques for design, analysis, etc.

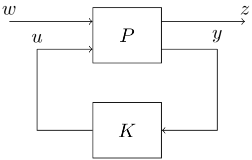
"Real world"



- need a model

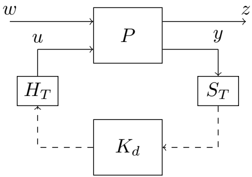
Fifty shades of models

Continuous-time/analog modelling



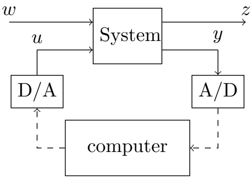
- idealized (LTI)
- lots of techniques for design, analysis, etc.

Hybrid modelling, sampled-data systems



- less idealized (still...)
- more involved to manipulate (not LTI!)

"Real world"



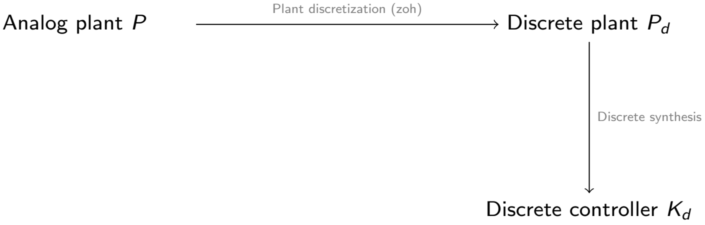
- need a model

Optimize then discretize...?

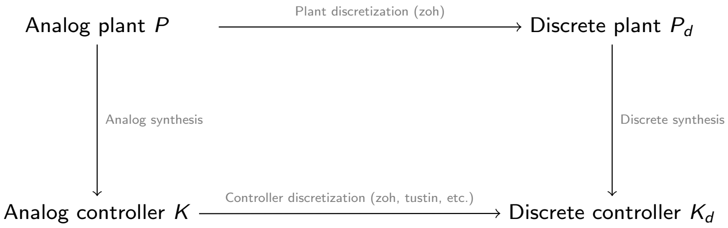
Analog plant P

Discrete controller K_d

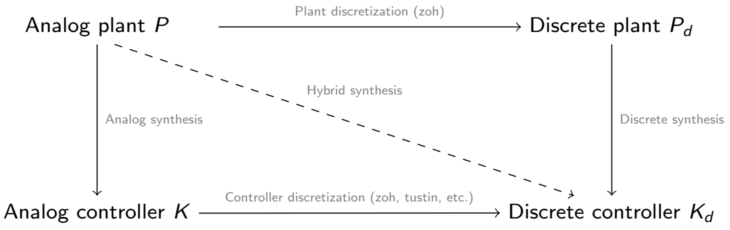
Optimize then discretize...?



Optimize then discretize...?



Optimize then discretize...?



- indirect methods loose optimality
- intersample behaviour is discarded
- decreasing T requires higher computational power and accuracy
  ↪with shift operator, $\lim_{T\to 0} A_d = I$ and $\lim_{T\to 0} B_d = 0$

Outline

Ideal sampler & holder

Reminder: LTI model G, i/o relationship in frequency domain

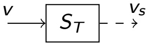
$$\text{analog:}\, Y(j\omega) = G_a(j\omega)U(j\omega) \qquad \text{discrete: } Y(e^{j\omega}) = G_d(e^{j\omega})U(e^{j\omega})$$

Ideal sampler & holder

Reminder: LTI model G, i/o relationship in frequency domain

$$\text{analog:}\, Y(j\omega) = G_a(j\omega)U(j\omega) \qquad \text{discrete:}\;\; Y(e^{j\omega}) = G_d(e^{j\omega})U(e^{j\omega})$$

Sampler S_T



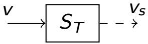
- time domain: $v_s[k] = v(kT)$
- modelled by an impulse-train $\sum_k \delta(t - kT)$ = T-periodic function
- frequency domain: $V_s(e^{-j\omega T}) = \frac{1}{T} \sum_{k\in\mathbb{Z}} V(j\omega + jk\omega_s)$ frequency aliasing
- /!\not bounded for all signals in $\mathcal{L}_2$ (bandlimited signals or anti-aliasing filter)

Ideal sampler & holder

Reminder: LTI model G, i/o relationship in frequency domain

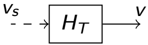
$$\text{analog:}\, Y(j\omega) = G_a(j\omega)U(j\omega) \qquad \text{discrete: } Y(e^{j\omega}) = G_d(e^{j\omega})U(e^{j\omega})$$

Sampler S_T



- time domain: $v_s[k] = v(kT)$
- modelled by an impulse-train $\sum_k \delta(t - kT)$ = T-periodic function
- frequency domain: $V_s(e^{-j\omega T}) = \frac{1}{T}\sum_{k\in\mathbb{Z}} V(j\omega + jk\omega_s)$ frequency aliasing
- /!\not bounded for all signals in $\mathcal{L}_2$ (bandlimited signals or anti-aliasing filter)

Holder H_T



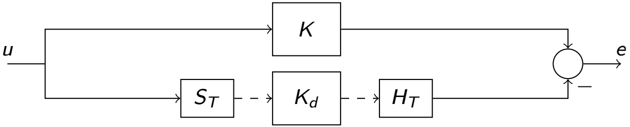
- time domain: $v(t) = v_s[k]$, $kT \leq t < (k+1)T$
- modelled as a difference of delayed steps: $h_T(t) = \frac{1}{T}1(t) - \frac{1}{T}1(t-T)$
- frequency domain: $H_T(s) = \frac{1-e^{-sT}}{sT}$ and $V(j\omega) = TH_T(j\omega)V_s(e^{-j\omega T})$

Discretization error

Let K be an analog LTI model and K_d its discretization: **Meaningful notion of error?**

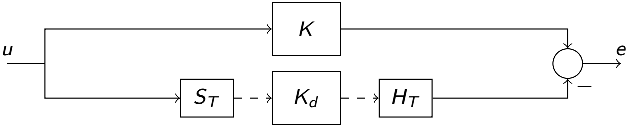
Discretization error

Let K be an analog LTI model and K_d its discretization: **Meaningful notion of error?**



Discretization error

Let K be an analog LTI model and K_d its discretization: **Meaningful notion of error?**

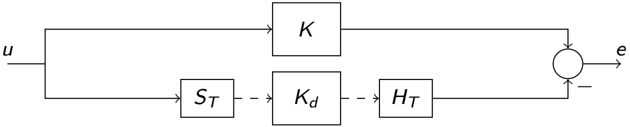


- the error system is time-varying (T-periodic) $\rightarrow$ no transfer function
- $H_T S_T$ is not bounded for all signals in $\mathcal{L}_2$ $\rightarrow$ need bandlimited signal u or anti-aliasing filter
- frequency-domain error:

$$E(j\omega) = K(j\omega)U(j\omega) - H_T(j\omega)K_d(e^{-j\omega T})\sum_k U(j\omega + jk\omega_s) \qquad (2)$$

Discretization error

Let K be an analog LTI model and K_d its discretization: **Meaningful notion of error?**



- the error system is time-varying (T-periodic) $\rightarrow$ no transfer function
- $H_T S_T$ is not bounded for all signals in $\mathcal{L}_2$ $\rightarrow$ need bandlimited signal u or anti-aliasing filter
- frequency-domain error:

$$E(j\omega) = K(j\omega)U(j\omega) - H_T(j\omega)K_d(e^{-j\omega T})\sum_k U(j\omega + jk\omega_s) \qquad (2)$$

suppose $U(j\omega) = 0$ for $\omega > \omega_N$ (bandlimited),

$$E(j\omega) = \Big(K(j\omega) - H_T(j\omega)K_d(e^{-j\omega T})\Big)\, U(j\omega) \qquad (3)$$

$\rightarrow$ $error(\omega) = \left|K(j\omega) - H_T(j\omega)K_d(e^{-j\omega T})\right|$

Discretization error: example

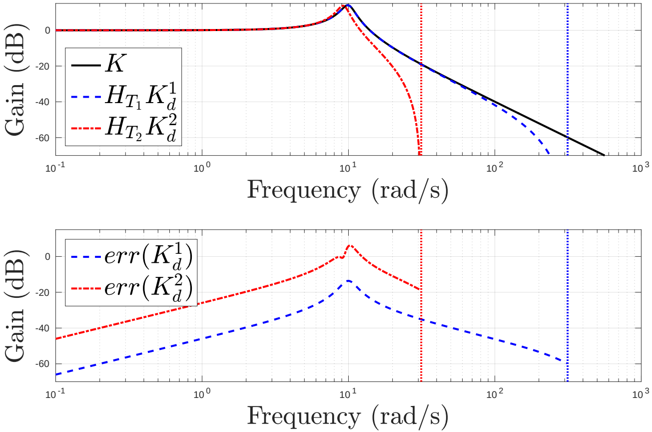
$$K(s) = \frac{1}{1 + 0.7s/10 + s^2/100} \tag{4}$$

- $T_1 = 0.01 \rightarrow K_d^1$
- $T_2 = 0.1 \rightarrow K_d^2$

Discretization error: example

$$K(s) = \frac{1}{1 + 0.7s/10 + s^2/100} \quad (4)$$

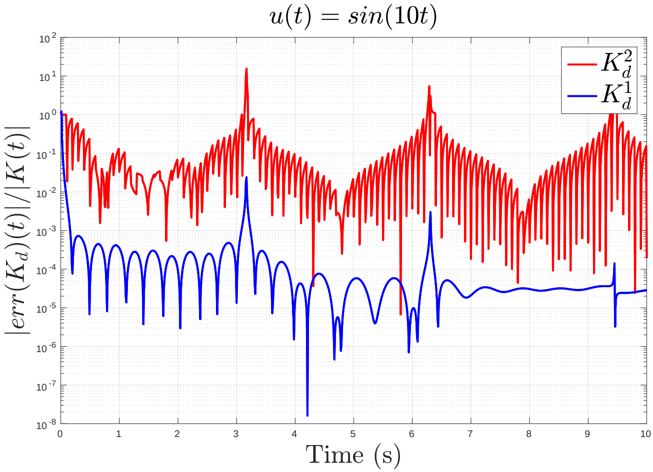
- $T_1 = 0.01 \rightarrow K_d^1$
- $T_2 = 0.1 \rightarrow K_d^2$



Discretization error: example

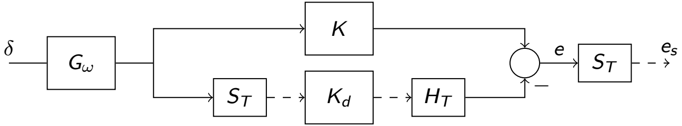
$$K(s) = \frac{1}{1 + 0.7s/10 + s^2/100} \quad (4)$$

- $T_1 = 0.01 \rightarrow K_d^1$
- $T_2 = 0.1 \rightarrow K_d^2$



Optimal discretization?

- frequency-domain error not usual system norm
- possible to look for the best discretization scheme w.r.t. the sampling instants and w.r.t. a predefined class of signals
  ↪section 4.6 in [2]
  ↪usual discrete norm can be used to measure e_s
  ↪e.g. step invariant discretization (ZOH) is the solution for steps



[2] Chen and Francis. *Optimal sampled-data control systems*. 1995

Outline

Lifting continuous signals

Frequency-domain considerations are central in some design methods...but SD have no transfer function

Lifting continuous signals

Frequency-domain considerations are central in some design methods...but SD have no transfer function

Continuous lifting
Consider a signal $u(t)$ such that $\int_{-\infty}^{\infty} u(t)^2 dt < \infty$ (i.e. $\in \mathcal{L}_2(\mathbb{R})$)

1. divide u in chunks of length T (sampling time): $\ldots$, $[-T, 0)$, $[0, T)$, $[T, 2T)$, $\ldots$

Lifting continuous signals

Frequency-domain considerations are central in some design methods...but SD have no transfer function

Continuous lifting
Consider a signal $u(t)$ such that $\int_{-\infty}^{\infty} u(t)^2 dt < \infty$ (i.e. $\in \mathcal{L}_2(\mathbb{R})$)

1. divide u in chunks of length T (sampling time): $\ldots$, $[-T, 0)$, $[0, T)$, $[T, 2T)$, $\ldots$
2. call the k-th chunk u_k such that $u_k(t) = u(kT + t)$ for $0 \leq t < T$

Lifting continuous signals

Frequency-domain considerations are central in some design methods...but SD have no transfer function

Continuous lifting
Consider a signal $u(t)$ such that $\int_{-\infty}^{\infty} u(t)^2 dt < \infty$ (i.e. $\in \mathcal{L}_2(\mathbb{R})$)

1. divide u in chunks of length T (sampling time): $\ldots$, $[-T, 0)$, $[0, T)$, $[T, 2T)$, $\ldots$
2. call the k-th chunk u_k such that $u_k(t) = u(kT + t)$ for $0 \leq t < T$
3. introduce the discrete-time signal $\underline{u} = \{u_k\}_k$

Lifting continuous signals

Frequency-domain considerations are central in some design methods...but SD have no transfer function

Continuous lifting
Consider a signal $u(t)$ such that $\int_{-\infty}^{\infty} u(t)^2 dt < \infty$ (i.e. $\in \mathcal{L}_2(\mathbb{R})$)

1. divide u in chunks of length T (sampling time): ..., $[-T, 0)$, $[0, T)$, $[T, 2T)$, ...
2. call the k-th chunk u_k such that $u_k(t) = u(kT + t)$ for $0 \leq t < T$
3. introduce the discrete-time signal $\underline{u} = \{u_k\}_k$
4. $\underline{u}$ lies in $l_2(\mathbb{Z}, \mathcal{K})$ where $\mathcal{K} = \mathcal{L}_2([0, T))$

Lifting continuous signals

Frequency-domain considerations are central in some design methods...but SD have no transfer function

Continuous lifting

Consider a signal $u(t)$ such that $\int_{-\infty}^{\infty} u(t)^2 dt < \infty$ (i.e. $\in \mathcal{L}_2(\mathbb{R})$)

1. divide u in chunks of length T (sampling time): $\ldots$, $[-T, 0)$, $[0, T)$, $[T, 2T)$, $\ldots$
2. call the k-th chunk u_k such that $u_k(t) = u(kT + t)$ for $0 \leq t < T$
3. introduce the discrete-time signal $\underline{u} = \{u_k\}_k$
4. $\underline{u}$ lies in $l_2(\mathbb{Z}, \mathcal{K})$ where $\mathcal{K} = \mathcal{L}_2([0, T))$

Lifting operator:

$$\begin{array}{rccc} L: & \mathcal{L}_2(\mathbb{R}) & \rightarrow & l_2(\mathbb{Z}, \mathcal{K}) \\ & u(t) & \rightarrow & \underline{u} \end{array} \tag{5}$$

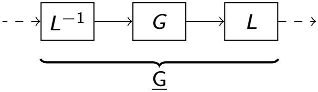
- it turns out L is an isomorphism
  $\hookrightarrow$working with u or $\underline{u}$ is 'equivalent'

Lifting systems I

$$G : \begin{cases} \dot{x}(t) &= Ax(t) + Bu(t) \\ y(t) &= Cx(t) + Du(t) \end{cases} \quad \text{such that} \quad y = Gu \tag{6}$$

Lifted system $\underline{G}$ obtained by lifting input/output

- $\underline{u} = Lu$
- $\underline{y} = Ly$
- thus $\underline{y} = \underline{G}\underline{u}$ with $\underline{G} = LGL^{-1}$

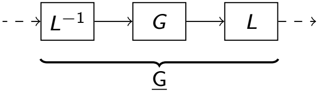


Lifting systems I

$$G : \begin{cases} \dot{x}(t) &= Ax(t) + Bu(t) \\ y(t) &= Cx(t) + Du(t) \end{cases} \quad \text{such that} \quad y = Gu \qquad (6)$$

Lifted system $\underline{G}$ obtained by lifting input/output

- $\underline{u} = Lu$
- $\underline{y} = Ly$
- thus $\underline{y} = \underline{G}\underline{u}$ with $\underline{G} = LGL^{-1}$



$\underline{G}$ represented by discrete-time equations

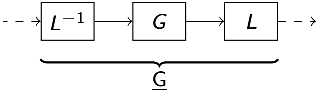
$$\underline{G} : \begin{cases} \xi_{k+1} &= \underline{A}\xi_k + \underline{B}u_k \\ y_k &= \underline{C}\xi_k + \underline{D}u_k \end{cases} \quad \text{with} \quad \xi_k = x(kT) \qquad (7)$$

Lifting systems I

$$G:\begin{cases}\dot{x}(t) &= Ax(t)+Bu(t)\\ y(t) &= Cx(t)+Du(t)\end{cases}\qquad \text{such that}\quad y=Gu \tag{6}$$

Lifted system $\underline{G}$ obtained by lifting input/output

- $\underline{u}=Lu$
- $\underline{y}=Ly$
- thus $\underline{y}=\underline{G}\underline{u}$ with $\underline{G}=LGL^{-1}$



$\underline{G}$ represented by discrete-time equations

$$\underline{G}:\begin{cases}\xi_{k+1} &= \underline{A}\xi_k+\underline{B}u_k\\ y_k &= \underline{C}\xi_k+\underline{D}u_k\end{cases}\qquad \text{with}\quad \xi_k=x(kT) \tag{7}$$

where the transformations are

$$\begin{array}{lll}\underline{A}: & \mathcal{E}\to\mathcal{E}, & \underline{A}\xi=e^{TA}\xi\\ \underline{B}: & \mathcal{K}\to\mathcal{E}, & \underline{B}u=\int_0^T e^{(T-\tau)A}Bu(\tau)d\tau\\ \underline{C}: & \mathcal{E}\to\mathcal{K}, & (\underline{C}\xi)(t)=Ce^{tA}\xi\\ \underline{D}: & \mathcal{K}\to\mathcal{K}, & (\underline{D}u)(t)=Du(t)+\int_0^t Ce^{(t-\tau)A}Bu(\tau)d\tau\end{array} \tag{8}$$

Lifting systems II

Interest?

- D_T and D_T^*: time delay and time advance of T
- U and U^*: unit time delay and advance on $l_2(\mathbb{Z}, \mathcal{K})$

Lifting systems II

Interest?

- D_T and D_T^*: time delay and time advance of T
- U and U^*: unit time delay and advance on $l_2(\mathbb{Z}, \mathcal{K})$

Note that time-delays translate to shifts in the lifted domain:

$$LD_T^* = U^* L \quad \text{and} \quad L^{-1} U = D_T L^{-1} \tag{9}$$

Lifting systems II

Interest?

- D_T and D_T^*: time delay and time advance of T
- U and U^*: unit time delay and advance on $l_2(\mathbb{Z}, \mathcal{K})$

Note that time-delays translate to shifts in the lifted domain:

$$LD_T^* = U^* L \quad \text{and} \quad L^{-1} U = D_T L^{-1} \tag{9}$$

G is T-periodic $\Leftrightarrow$ $\underline{G}$ is time invariant

Proof ($\Rightarrow$)

$$\begin{aligned} U^* \underline{G} U &= U^* L G L^{-1} U \\ &= L D_T^* G D_T L^{-1} \\ &= L G L^{-1} \\ &\triangleq \underline{G} \end{aligned} \tag{10}$$

Lifting systems II

Interest?

- D_T and D_T^*: time delay and time advance of T
- U and U^*: unit time delay and advance on $l_2(\mathbb{Z}, \mathcal{K})$

Note that time-delays translate to shifts in the lifted domain:

$$LD_T^* = U^* L \quad \text{and} \quad L^{-1} U = D_T L^{-1} \tag{9}$$

G is T-periodic $\Leftrightarrow$ $\underline{G}$ is time invariant

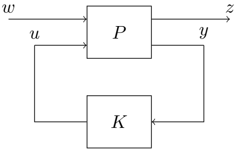
Proof ($\Rightarrow$)

$$\begin{aligned} U^* \underline{G} U &= U^* L G L^{-1} U \\ &= L D_T^* G D_T L^{-1} \\ &= L G L^{-1} \\ &\triangleq \underline{G} \end{aligned} \tag{10}$$

A SD system can be lifted into a time-invariant discrete-time system with infinite-dimensional input and output spaces

Measure of performance of closed-loop SD systems I

Reminder: $\|T_{w\to z}(K)\|$ sums up the performances of the closed-loop

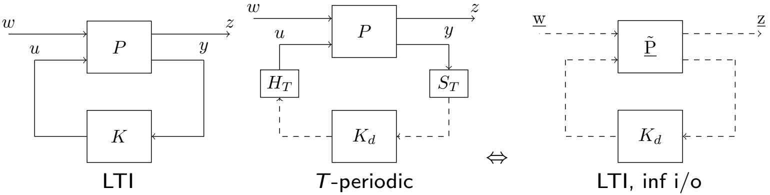


LTI

[2] Chen and Francis. *Optimal sampled-data control systems*. 1995

Measure of performance of closed-loop SD systems I

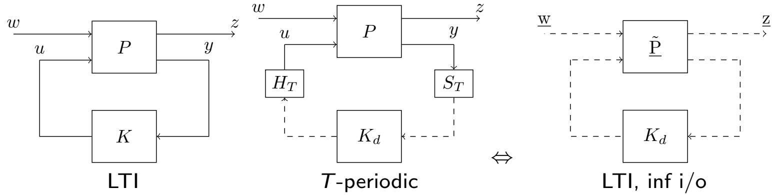
Reminder: $\|T_{w\to z}(K)\|$ sums up the performances of the closed-loop



[2] Chen and Francis. *Optimal sampled-data control systems*. 1995

Measure of performance of closed-loop SD systems I

Reminder: $\|T_{w\to z}(K)\|$ sums up the performances of the closed-loop



In chap. 12-13 of [2]:

- similar norms (2 and ∞) can be defined for $T_{w\to z}(K_d)$ through $T_{\underline{w}\to\underline{z}}(K_d)$
- ...leading to some optimal synthesis framework
- ...and analysis
- but more difficult due to infinite dimensionality of input/output spaces

---

[2] Chen and Francis. *Optimal sampled-data control systems*. 1995

Measure of performance of closed-loop SD systems II

Some interesting applications in [2]
Consider two discretizations $K_d^1(T)$ and $K_d^2(T)$ of some controller K

- compare their performances against sampling rate T
  $\hookrightarrow$ example 13.7.3
- determine T_{max} such that some prescribed performance bound is fulfilled
  $\hookrightarrow$ example 13.8.1
- perform optimal synthesis in SD framework
  $\hookrightarrow$ example 13.8.1

[2] Chen and Francis. *Optimal sampled-data control systems*. 1995

Measure of performance of closed-loop SD systems II

Some interesting applications in [2]
Consider two discretizations $K_d^1(T)$ and $K_d^2(T)$ of some controller K

- compare their performances against sampling rate T
  $\hookrightarrow$ example 13.7.3
- determine T_{max} such that some prescribed performance bound is fulfilled
  $\hookrightarrow$ example 13.8.1
- perform optimal synthesis in SD framework
  $\hookrightarrow$ example 13.8.1

Other?

- given P, K, for fixed T, find K_d that minimizes the loss of performance w.r.t. analog design

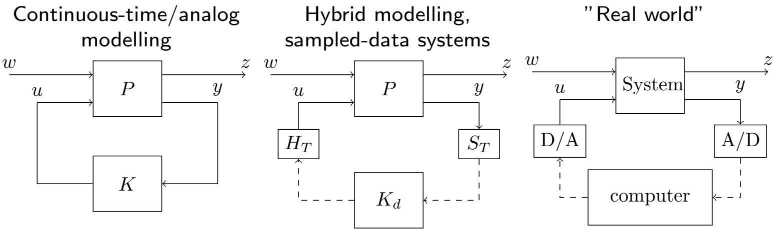
$$\| T_{\underline{w}\to\underline{z}}(K) - T_{\underline{w}\to\underline{z}}(K_d) \| \tag{11}$$

$\hookrightarrow$everything must be considered in the lifted domain

[2] Chen and Francis. *Optimal sampled-data control systems*. 1995

Outline

Conclusion



- Sampled-Data formalism suited for computer-controlled systems
- Lifting technique to extend LTI tools
- Lifting can also be used to handle multiple sampling rates, chap. 8 [2]
- infinite dimensionality...
- still not implemented controller (code)...

Tools: [3], [4]

[2] Chen and Francis. *Optimal sampled-data control systems*. 1995

[3] Fujioka, Hara, and Yamamoto. "Sampled-data control toolbox: object-oriented software for sampled-data feedback control systems". 2004

[4] Polyakov, Rosenwasser, and Lampe. "DIRECTSD 3.0 toolbox for MATLAB: Further progress in polynomial design of sampled-data systems". 2006

References I

- S. Boyd and C. Barratt. *Linear controller design: limits of performance*. Prentice Hall Englewood Cliffs, NJ, 1991.
- T. Chen and B.A. Francis. *Optimal sampled-data control systems*. Springer Science & Business Media, 1995.
- H. Fujioka, S. Hara, and Y. Yamamoto. "Sampled-data control toolbox: object-oriented software for sampled-data feedback control systems". In: *International Conference on Robotics and Automation*. 2004, pp. 19–24.
- K. Polyakov, E. Rosenwasser, and B. Lampe. "DIRECTSD 3.0 toolbox for MATLAB: Further progress in polynomial design of sampled-data systems". In: *Conference on Computer Aided Control System Design*. 2006, pp. 1946–1951.