Polynomial Methods for Control Analysis and Design

4/ Discrete-time systems

Overview

[PolyX]

Overview

Ch. 4. Discrete-time systems

D-t systems and polynomials

- Two approaches
- Discrete-time signals
- Sequences
- Delay operator
- SS and IO in discrete time

Feedback design using *d*

- D-t plant
- Pole placement
- Stabilization
- Asymptotic regulation

Deadbeat regulation

- Deadbeat
- Weak deadbeat
- Strong deadbeat
- Deadbeat for c-t systems

Asymptotic tracking

- Reference generator
- Typical signals
- Two-degrees-of-freedom
- Classical structure
- Signals in the structure
- Asymptotic tracking
- Deadbeat tracking

Stochastic problems

Discrete-time systems and polynomials

Two approaches
Discrete-time signals
Sequences
Delay operator
SS and IO in discrete time

Two approaches

Discrete-time signals and systems

are usually described either by

- forward shift operator z or by
- backward shift (delay) operator z^{-1}
- both resulting from Z-transform.

Using

- z is very similar to s in continuous-time while using
- z^{-1} is somewhat simpler

We shall exercise both, slightly preferring z^{-1}

Discrete-time signals

- Discrete-time signals are typically infinite sequences
- An infinite sequence of real numbers looks like this

$$a = \{\alpha_n, \alpha_{n+1}, \alpha_{n+2}, \ldots\},\ \alpha_k \in \mathrm{R}$$

  with an integer n
- With element-wise addition and convolutory multiplication, such sequences form a field.
- We denote $d = \{0; 0, 1, 0, \ldots\}$

semicolon separates elements with negative and positive indices

- Then d^k is a sequence of zeros except for 1 at k-th position
- the sequence a above can be written as

$$a = \alpha_n d^n + \alpha_{n+1} d^{n+1} + \alpha_{n+2} d^{n+2} + \cdots$$

formal power (Laurent) series

$\boldsymbol{d}$ is a position-marker, called indeterminate or right/backwards shift

Sequences

- Sequence $a=\{\alpha_n,\alpha_{n+1},\alpha_{n+2},\ldots\},\ \alpha_k\in\mathrm{R}$ is recurrent if there exists integers r,s and reals $\lambda_1,\lambda_2,\ldots,\lambda_r$ such that

$$\alpha_{j+r}+\lambda_1\alpha_{j+r+1}+\cdots+\lambda_r\alpha_j=0,\ j=n+s,n+s+1,\ldots$$

  Recurrent sequences form a field denoted by $\mathrm{R}(d)$
- Recurrent sequence is causal, strictly causal and bicausal if its lowest index (power) is nonnegative, positive and 0, respect.
- Causal sequence is Schur stable if it converges to zero ($\forall\varepsilon>0\ |\alpha_k|<\varepsilon$ for almost all k)
- Causal sequence with only a finite number of nonzero elements is a polynomial .
- A polynomial fraction is defined as usually.
- Set of polynomial fractions is isomorphic with the set of recurrent sequences

$$a=a_2/a_1,\ a_1=d^n(1+\lambda_1 d+\cdots+\lambda_r d^r),a_2=a_1a$$

Sequences - 2

- A polynomial $p(d)$ is causal $1/p(d)$ is a causal sequence, that is if $p(0) \neq 0$

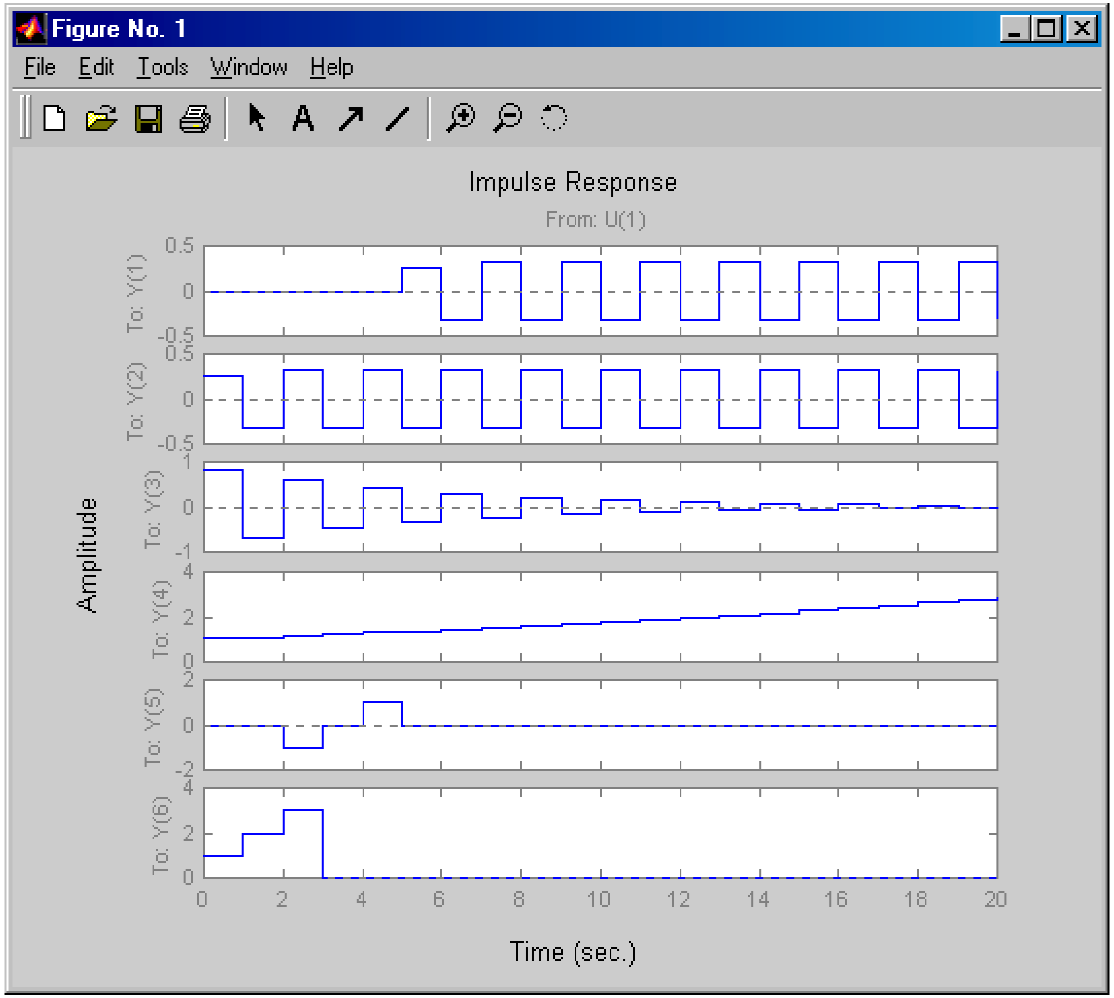
Examples

–1	0	1	2	3	4	5	6	7	8
●	●	●	●	●	●	●	●	●	●
		●	●	●	●	●	●	●	●
		●	●	●	●	●	●	●	●
	●	●	●	●	●	●	●	●	●
		●	●	●		●	●		
	●	●	●						

index (time)

non-causal sequence
causal sequence
strictly causal sequence
polynomial
causal polynomial

Sequences - 3



$\dfrac{d^5}{(1+d)(4+d)}$ s. causal

$\dfrac{1}{(1+d)(4+d)}$ bicausal

$\dfrac{1}{1.2+d}$ stable

$\dfrac{1}{0.95-d}$ unstable

$-d^2+d^4$ polynomial

$1+2d+3d^2$ causal pol.

Delay operator

[PolyX]

- Of course, d means the well-known delay or backward shift operator, often denoted $z^{-1} = d$
- Instead of usual *z*-transform, we have derived here almost everything necessary formally, without any mathematics.

SS and IO in discrete-time

[PolyX]

Computation of the IO representation in *d*

$$\begin{aligned} x_{k+1} &= Ax_k + Bu_k, \quad x_0 \\ y_k &= Cx_k + Du_k \end{aligned}$$

$$C(I - dA)^{-1}Bd + D = \frac{b(d)}{a(d)}$$

$$C(I - dA)^{-1}dx_0 = \frac{c_{x_0}(d)}{a(d)}$$

$$y(d) = \frac{b(d)}{a(d)}u(d) + \frac{c_{x_0}(d)}{a(d)}$$

$$a(d)y(d) = b(d)u(d) + c_{x_0}(d)$$

Assumption: constructibility → $Cd(I - dA)^{-1}$ coprime → $(a, b, c_{x_0}) = 1$

$(a, c_{x_0}) = 1$

(a, b) represents uncontrollable modes, i.e. common factors in $(I - dA)^{-1}Bd$

if no modes are hidden

SS and IO in discrete-time

Computation of the IO representation in *z*

$$x_{k+1} = Ax_k + Bu_k,\ x(0) = x_0$$
$$y_k = Cx_k + Du_k$$

$$x(z) = \sum_{k=0}^{\infty} x_k z^{-k}$$
$$x_{k+1}(z) = z\,x(z) - z\,x_0$$

$$C(zI - A)^{-1}B + D = \frac{b(z)}{a(z)}$$
$$C(zI - A)^{-1}x_0 = \frac{c_{x_0}(z)}{a(z)}$$

$$y(z) = \frac{b(z)}{a(z)}u(z) + \frac{c_{x_0}(z)}{a(z)}$$
$$a(z)y(z) = b(z)u(z) + c_{x_0}(z)$$

Assumption: observability → $C(zI - A)^{-1}$ coprime → $(a, b, c_{x_0}) = 1$

$(a, c_{x_0}) = 1$

(a, b) represents unreachable modes, i.e. common factors in $(zI - A)^{-1}B$

if no modes are hidden

Properties

- reachability = coprimness of $(zI - A)^{-1}B$
- controllability = coprimness of $(I - dA)^{-1}Bd$
- controllability = reachability of non-finite modes
- observability = coprimness of $C(zI - A)^{-1}$
- constructibility = coprimness of $Cd(I - dA)^{-1}$
- constructibility = observability of non-finite modes
- $\dfrac{b(z)}{a(z)}$ is causal (realizable) iff $\deg a(z) \geq \deg b(z)$
- $\dfrac{b(d)}{a(d)}$ is causal (realizable) iff $a(0) \neq 0$
- order of $\dfrac{b(z)}{a(z)}$ is $\deg a(z)$, order of $\dfrac{b(d)}{a(d)}$ is $\deg[a(z), b(z)]$

Feedback design using *d*

[PolyX]

D-t plant
Pole placement
Stabilization
Asymptotic regulation
Deadbeat regulation
Weak deadbeat
Strong deadbeat
Deadbeat for c-t systems

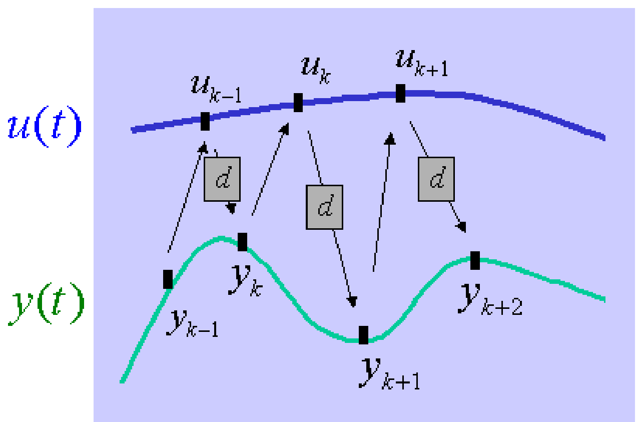
Discrete-time plant

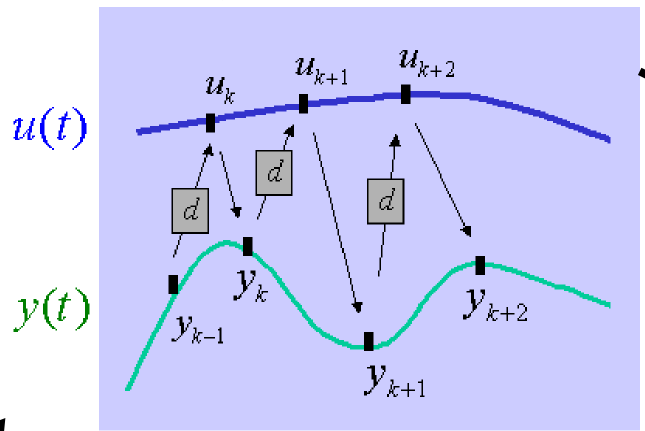
Discrete-time plant is usually assumed

- free of hidden modes,
- having initial state and hence c_{x_0} unknown
- strictly causal $a(0) \neq 0,\ b(0) = 0$

sampling and holding instants never coincide, we may index as

delay in plant or as delay in feedback





reformulate: what is known is plant

Pole placement

Pole placement in *d*

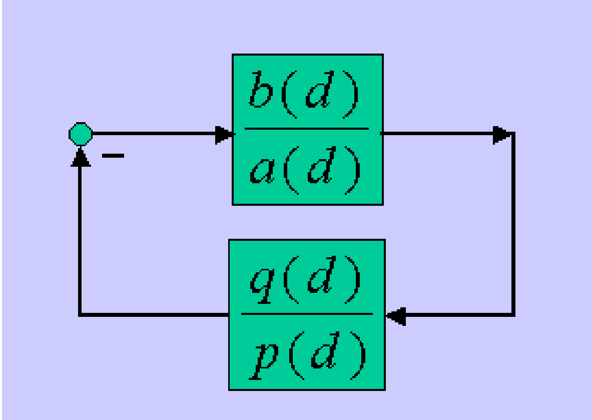
Given causal

$$m(d) = (d - d_1)\cdots(d - d_k)$$

solve

$$ap + bq = m$$

Any solution gives rise to a controller placing the poles accordingly.



Plant and controller free of hidden modes!

Solvability condition $(a, b) \mid m$

Eventual uncontrollable modes must be preserved.
Of course, they must be stable.

Pole placement in *z* and most of other design are similar to the continuous-time and hence are omitted.

Pole placement - 2

[PolyX]

Comments

- The number *k* of desired poles is arbitrary. If necessary, a proper number of finite modes is automatically added ← while in d
- *m* is pseudo-characteristic (finite modes are "out of control" but one need not care)
- Any solution is causal (just write the equation at *d=0:*

$$a(0)p(0)+b(0)q(0)=m(0) \Rightarrow a(0)p(0)=m(0) \Rightarrow p(0)\neq 0$$

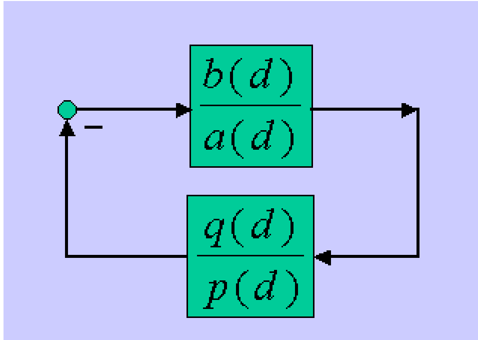
$$b(0)=0 \qquad a(0)\neq 0\neq m(0)$$

- Any solution does the job but usually a small (minimum) order controller is taken, which has small (minimum)

$$\max\{\deg p, \deg q\}$$

Stabilization

Stabilization in *d*



Formulation

Given plant, find a controller such the feedback system is stable.

Solution

All stabilizing controllers are parameterized by

$$\frac{q}{p} = \frac{y + at}{x - bt}$$

where t is an arbitrary polynomial fraction with stable denominator and x, y satisfy

$\bar{a}x + \bar{b}y = 1$ where $a = (a,b)\bar{a},\ \ b = (a,b)\bar{b}$

Solvability

Plant free of unstable hidden modes and (a,b) stable.

Youla-Kucera parameterization

Stabilization - 2

Proof

1) Any controller from Y-K is stabilizing:

Denote $t = n/m, \ m$ stable. Then we can write $\dfrac{q}{p} = \dfrac{my + an}{mx - bn}$

and $ap + bq = a(mx - bn) + b(my + an) = axm + bym = (a,b)m$

2) Any stabilizing controller is in Y-K:

Any stabilizing controller yields $ap + bq$ stable, that is

$$\bar{a}p + \bar{b}q = m$$

for some stable m .

As the general solution of ↑ reads

$$p = mx - \bar{b}\bar{n}$$

$$q = my + \bar{a}\bar{n}$$

with an arbitrary polynomial parameter $\bar{n}$,

Y-K results from the choice $\bar{n} = (a,b)n$.

3) The solvability condition is clearly n&s for the stability of $ap + bq$ •

Stabilization -3

[PolyX]

Comments

- The Y-K parameterization can also be written as $\dfrac{q}{p} = \dfrac{my + an}{mx - bn}$ as soon as a cancellation is done when possible.
- m equals c-l characteristic polynomial (up to *(a,b)*)
- degrees are not interesting (because of *d*)
- Nothing like Y-K exists in state space, confer also classical methods
- Y-K parameterization is the greatest success of polynomial methods
- Indeed all controllers are present, incl. non-generic orders
- Parameterization of all stabilizing controllers up to order *l* are easy if *l* is generic or higher, but almost intractable otherwise.
- Taking $t = 0$ results in the “most stable” system having constant char. pol.: $ap' + bq' = 1$ (for $(a,b) = 1$).

Asymptotic regulation

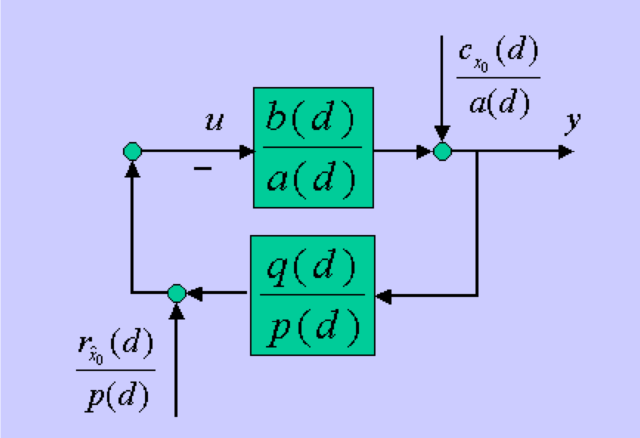
Asymptotic regulation

Formulation

Achieved iff both sequences

$y(d), u(d)$ are Schur stable

for any combination of $c_{x_0}, r_{\hat{x}_0}$



Solution

All asymptotic regulators result from the solution of

$$ap + bq = m$$

for a stable polynomial m .

Solvability condition

(a, b) stable

AR is equivalent to stabilization !

Asymptotic regulation - 2

Derivation

Simple inspection of the formulas

$$y = \frac{b}{ap+bq} r_{\hat{x}_0} + \frac{p}{ap+bq} c_{x_0}$$

$$u = \frac{a}{ap+bq} r_{\hat{x}_0} - \frac{q}{ap+bq} c_{x_0}$$

reveals that both y and u are stable sequences (fractions) for all particular values of $r_{\hat{x}_0}, c_{x_0}$ (that is, for all initial conditions) iff

$$ap + bq = m$$

is a stable polynomial.

Asymptotic regulation - 3

For a controller satisfying $ap + bq = m$, the external signals read

$$y = \frac{b}{m} r_{\hat{x}_0} + \frac{p}{m} c_{x_0}$$

$$u = \frac{a}{m} r_{\hat{x}_0} - \frac{q}{m} c_{x_0}$$

Exercise

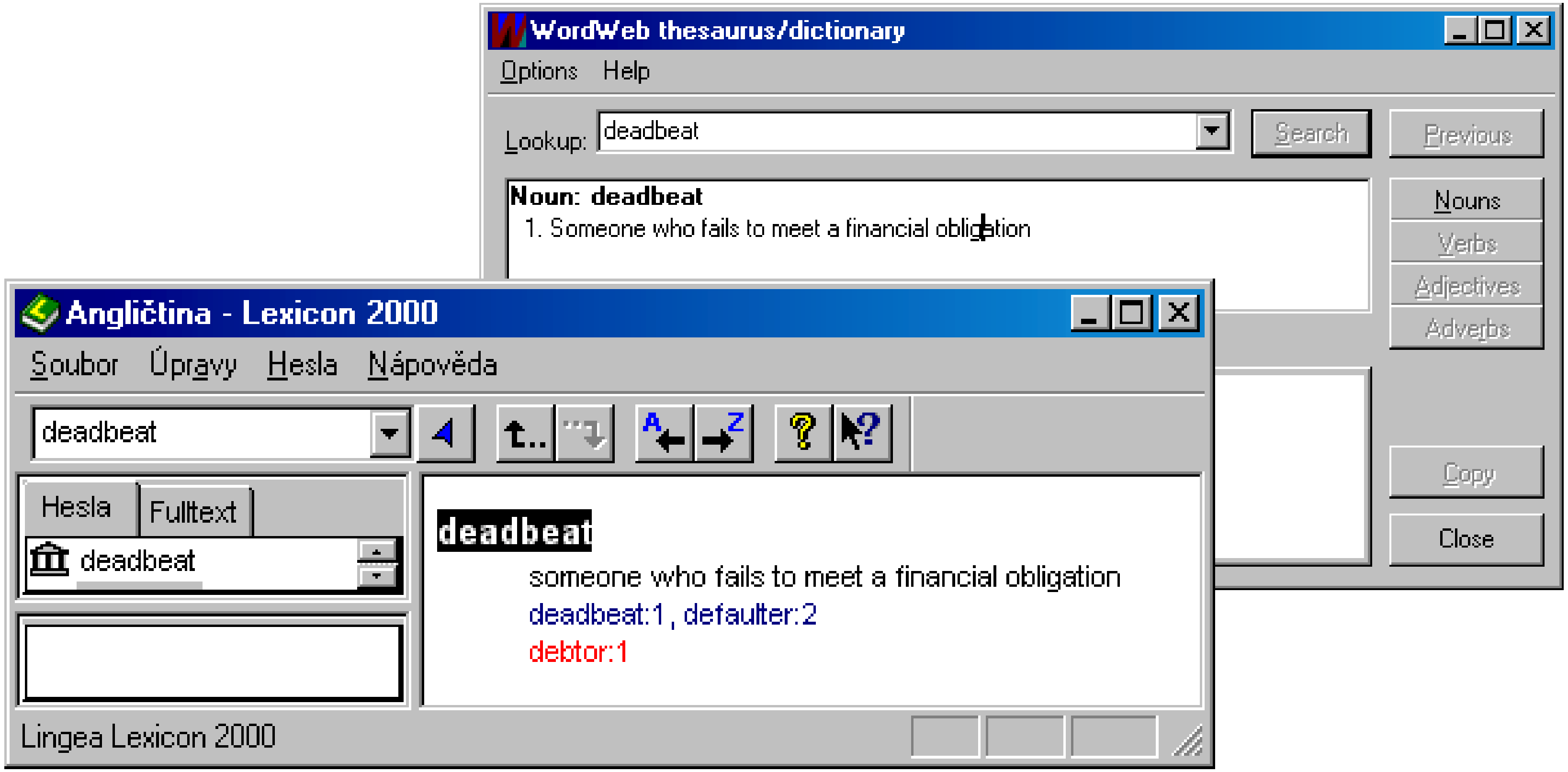
To familiarize the polynomial way of thinking:

- repeat the derivation under an (unrealistic) assumption of (partly) fixed initial conditions leading to some (partly) fixed $r_{\hat{x}_0}(d), c_{x_0}(d)$. Consider namely the case of unstable $(r_{\hat{x}_0}, c_{x_0})$
- repeat the derivation assuming that only *y* but not *u* is to be stable (or vice versa).
- argue what happens if $(a, b) \neq 1$ both stable and unstable.

Deadbeat regulation

Deadbeat regulators
Weak version
Strong version
Solution via z
Deadbeat poles
Deadbeat for c-t plant

Contradictio in adjecto?



Is “deadbeat controller” a contradiction in attribute ?

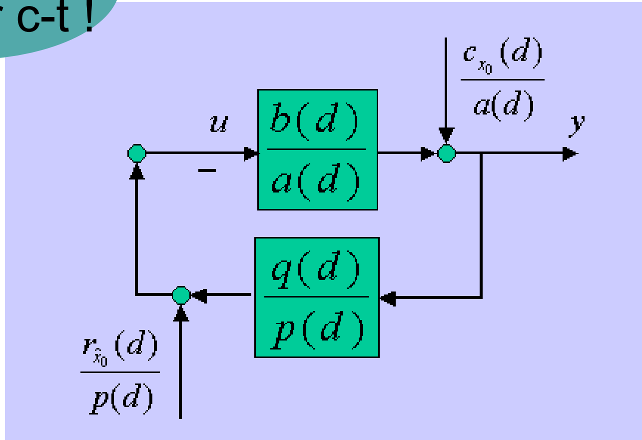
Deadbeat regulators

Deadbeat strategy

(finite number of steps)

Impossible in linear c-t !

Discrete-time control allows driving some signal to zero in finite time and holding it there for all discrete (sampling) times after



Deadbeat regulation

Formulation 1 / Strong version

Achieved iff both sequences y, u have finite length (i.e., $y(d), u(d)$ are polynomials!) for any combination of $c_{x_0}, r_{\hat{x}_0}$.

Formulation 2 / Weak version

Achieved iff y has finite length while u only converges to zero, (i.e., $y(d)$ is a polynomial and $u(d)$ is infinite but stable!) for any combination of $c_{x_0}, r_{\hat{x}_0}$.

Deadbeat regulators - 2

Different motivation

- In the strong deadbeat both external signals are to disappear after a finite number of steps. As a consequence, the whole system becomes at rest.
- In the weak version, only the output is to disappear. Other signals remain nonzero but reasonable (stable).
  The system does not get at rest in any finite time.

Weak deadbeat regulator

Derivation

The same formulas but different reasoning is used in the both cases

$$y = \frac{b}{ap+bq} r_{\hat{x}_0} + \frac{p}{ap+bq} c_{x_0}$$

$$u = \frac{a}{ap+bq} r_{\hat{x}_0} - \frac{q}{ap+bq} c_{x_0}$$

1) Weak version

The first term in *y* becomes polynomial if, e.g., we put $ap + bq = b$
As *b* need not be stable, however, this choice is not correct in general. All we can do is to take just a stable part of *b*. To this end, factor $b = b^+ b^-$ where b^+ is Schur stable while b^- is antistable.

Weak deadbeat regulator - 2

Now we can take $ap + bq = b^{+}$ which results in

$$y = b^{-} r_{\hat{x}_0} + \frac{p}{b^{+}} c_{x_0}$$

$$u = \frac{a}{b^{+}} r_{\hat{x}_0} - \frac{q}{b^{+}} c_{x_0}$$

To make the second term in *y* is polynomial, we set

$p = b^{+} x$ for some polynomial *x* . This leads to

$$y = b^{-} r_{\hat{x}_0} + x c_{x_0}$$

$$u = \frac{a}{b^{+}} r_{\hat{x}_0} - \frac{q}{b^{+}} c_{x_0}$$

Now *y* is indeed polynomial. Moreover, *u* is stable as was also required, so the design is complete.

Weak deadbeat regulator - 3

The controller we need must satisfy $ap + bq = b^+$ and $p = b^+ x$

It can be found by solving the polynomial equation

$$ax + b^- q = 1$$

which is solvable iff $(a, b^-) = 1$, i.e. (a, b) is stable. To shorten the transient period, we pick the solution with minimum degree of *x*.

Weak deadbeat regulator – Solution

The weak deadbeat regulator results from solving $ax + b^- q = 1$ and taking $p = b^+ x$ where b^+ is the stable part of b . For the shortest transient, take the solution with minimum degree of *x* .

Weak deadbeat regulator – Solvability condition: (a, b) stable

Strong deadbeat regulator

Derivation

The same formulas but different reasoning will be used

$$y = \frac{b}{ap+bq} r_{\hat{x}_0} + \frac{p}{ap+bq} c_{x_0}$$

$$u = \frac{a}{ap+bq} r_{\hat{x}_0} - \frac{q}{ap+bq} c_{x_0}$$

2) Strong version

Here we cannot use $ap + bq = b^+$ as b^+ cannot divide at the same time all *a,b,p,q* that appear in *y* and *u.* As no other polynomial can do it either. So what?

Strong deadbeat regulator - 2

We simply take

$$ap + bq = 1$$

for a comparison:
week version

$$y = b^- r_{\hat{x}_0} + x c_{x_0}$$

$$u = \frac{a}{b^+} r_{\hat{x}_0} - \frac{q}{b^+} c_{x_0}$$

which leads to

$$y = b r_{\hat{x}_0} + p c_{x_0}$$

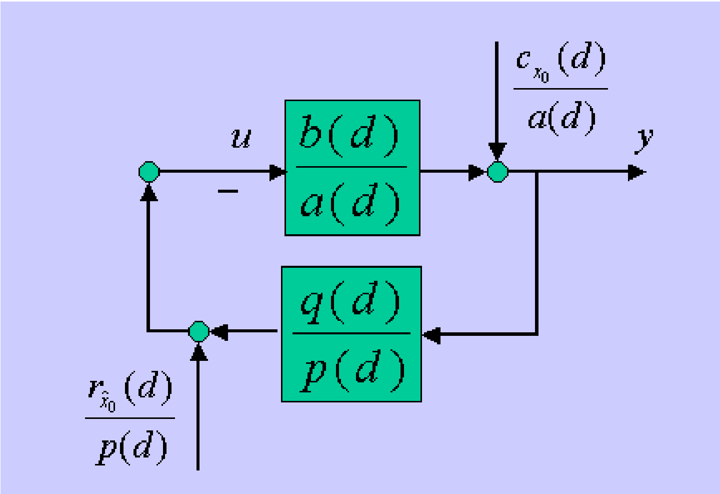
$$u = a r_{\hat{x}_0} - q c_{x_0}$$

As all four terms are polynomial, the job is done.

The equation is clearly solvable iff *a* and *b* are coprime.

For the shortest transient possible, we use its minimum degree solution.

Strong deadbeat regulator - 3



Strong deadbeat regulator – Solution

The strong deadbeat regulator results from solving $ap + bq = 1$.
The shortest transient period is obtained by taking the minimum degree solution.

Strong deadbeat regulator – Solvability condition: $(a,b) = 1$

Deadbeat designed using *z*

PolyX

Design using forward shift operator

The method using *d* will be modified into *z* solution. With $z^{-1} = d$ the original deadbeat equation reads

$$a(z^{-1})p(z^{-1}) + b(z^{-1})q(z^{-1}) = 1$$

Its minimum-degree solution is characterized by

$$\deg p(z^{-1}) \le \deg b(z^{-1}) - 1$$
$$\deg q(z^{-1}) \le \deg a(z^{-1}) - 1$$

where generically the equality holds.

Denoting the plant order by *n* $\max\{\deg a, \deg b\} = n$,

the controller order is generically $\max\{\deg p, \deg q\} = n - 1$

Hence the above equation is turned into *z* by multiplying

$$a(z^{-1})p(z^{-1}) + b(z^{-1})q(z^{-1}) = 1 \quad \otimes\ z^{2n-1}$$

$$a(z)p(z) + b(z)q(z) = z^{2n-1}$$

Solution in *z*

Strong deadbeat regulator – Solution in *z*

For a plant $b(z)/a(z)$ of order $n = \max\{a, b\}$, strong deadbeat regulator results from solving the equation

$$a(z)p(z) + b(z)q(z) = z^{2n-1}$$

The shortest transient period is obtained by taking the minimum degree solution w.r.t. q

Weak deadbeat regulator – Solution in *z* briefly

$$a(z)x(z) + b^{-}(z)q(z) = z^{m}$$ and $$p(z) = b^{+}(z)x(z)$$

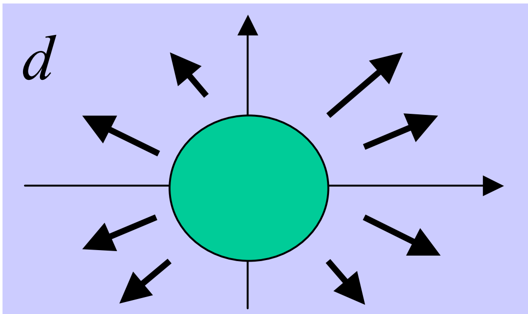
where $m = 2n - 1 - \deg b^{+}(z)$

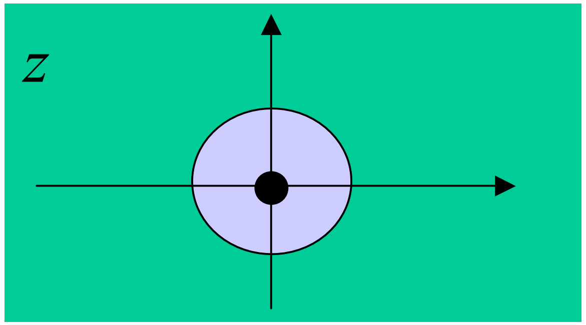
finite modes must be handled "manually" while in *z*

Deadbeat poles

Comments

- Strong deadbeat regulator assigns (pseudo) characteristic polynomial to (1) z^{2n-1}, hence it places all poles to (infinity) (zero). $a(d)p(d)+b(d)q(d)=1$ $\quad a(z)p(z)+b(z)q(z)=z^{2n-1}$





- It is the "most stabilizing" controller, reacts very fast, sometimes too fast (noise)
- Weak deadbeat regulator assigns (pseudo) characteristic polynomial to $b^+(d)$ ($z^m b^+(z)$).

$$a(d)p(d)+b(d)q(d)=b^+(d) \qquad a(z)p(z)+b(z)q(z)=z^m b^+(z)$$

Deadbeat for c-t plant

- can never be achieved with a linear c-t controller
- but a d-t regulator can do the job (why?)

Strong deadbeat

$$y = b^{-} r_{\hat{x}_0} + x c_{x_0}$$
$$\deg b^{+} \text{ shorter}$$

- may take more d-t steps but
- sets the whole d-t system at rest in a finite time, hence
- the c-t system gets also at rest so that
- its output remains zero even in between of sampling instants

Weak deadbeat (on the contrary)

$$y = b r_{\hat{x}_0} + p c_{x_0}$$

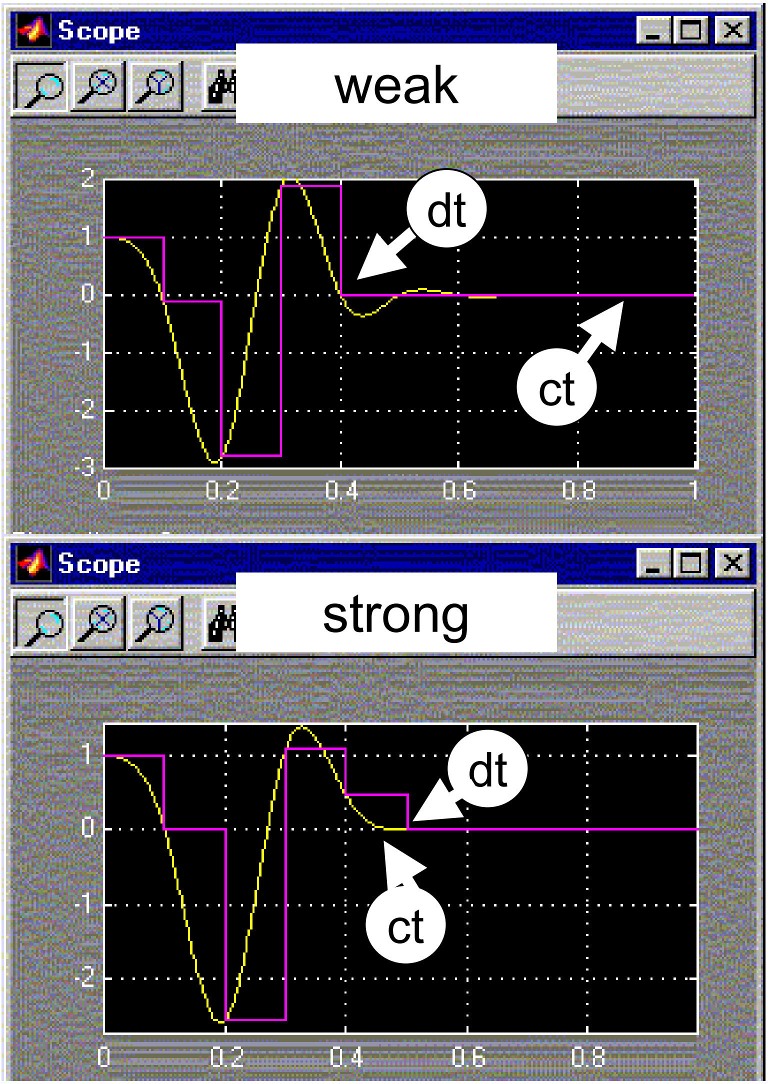
- takes less d-t steps but
- makes d-t output finite while the d-t plant "keeps moving"
- and so does the "true" c-t plant. As a result,
- the c-t output gets to zero in the sampling instants but not (necessarily) between them!

Deadbeat for c-t plant - 2

Example

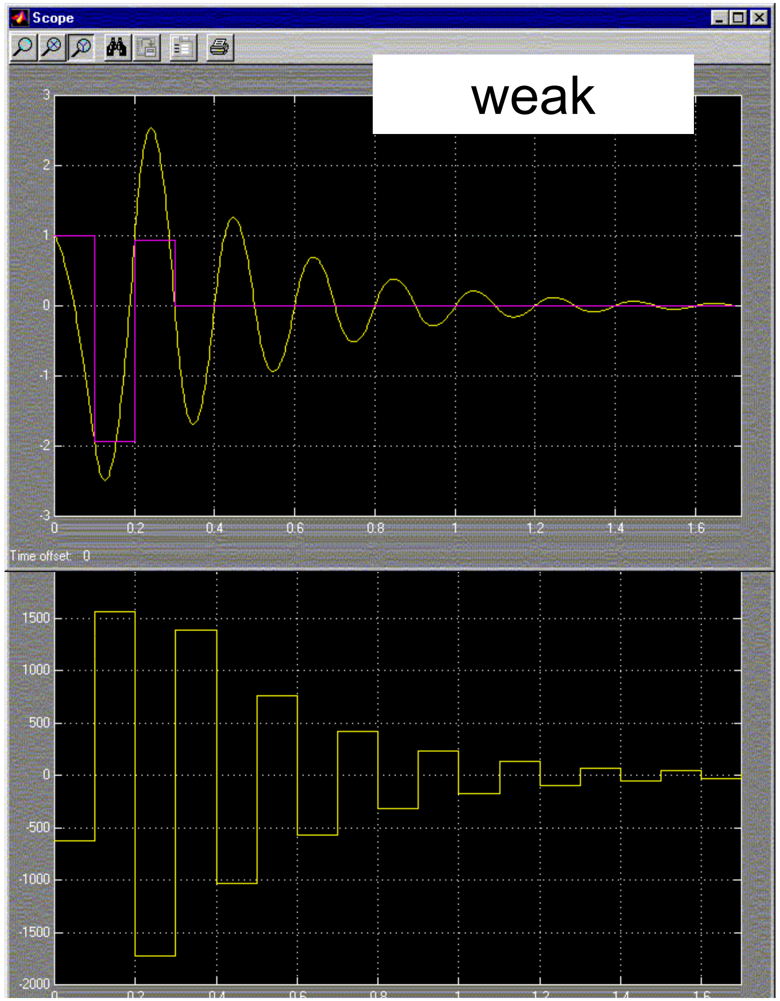
Modified Ball and beam

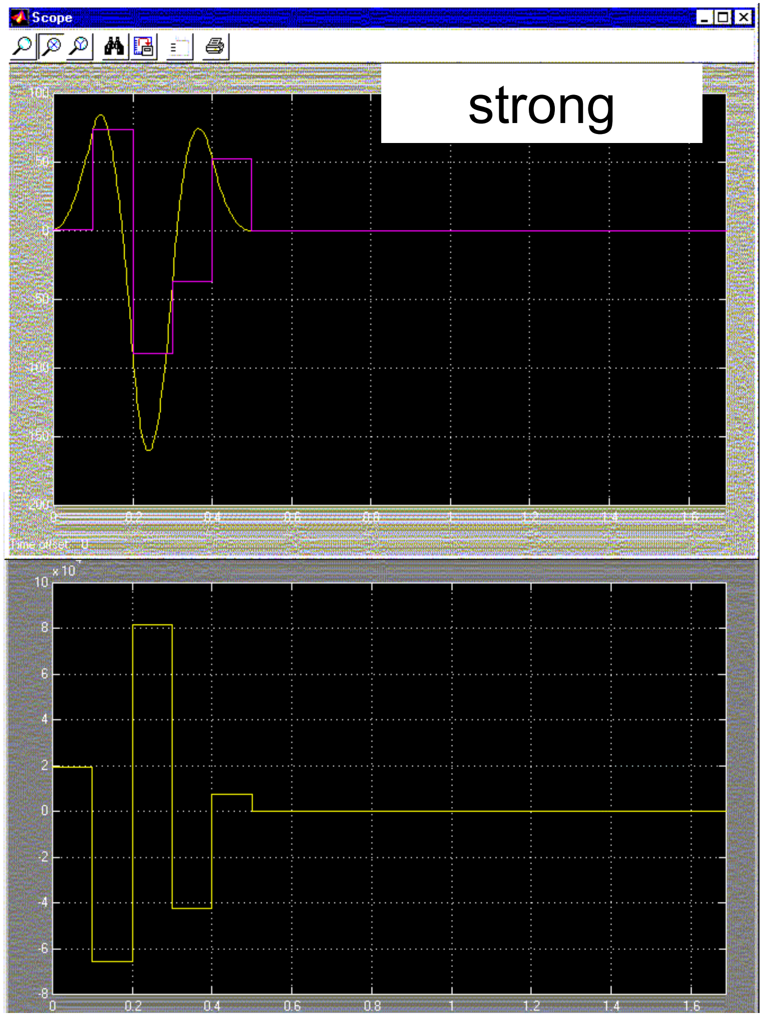
It is clear that weak version
is one step shorter in d-t but
(in fact infinitely) longer in c-t!



Deadbeat for c-t plant - 3

Another example (artificial)





Tracking problems

PolyX

Reference generator
Typical signals
Two-degrees-of-freedom
Classical structure
Signals in the structure
Asymptotic tracking
Deadbeat tracking

Tracking

Reference signal tracking

is an important control task.

- We may wish to design a controller that tracks a specific signal (e.g., a unit step starting at time zero). Then the resulting system is tuned to this particular signal and may not be able to track another one, even very similar.
- Or the controller is designed to track all signals from a given class, such as all ramps (with any starting time, any initial value and any slope).
- Such a class is conveniently described by a reference generator system with unspecified initial conditions.

Reference generator

Reference generator

An artificial autonomous system considered only for design:

State space model

$$\tilde{x}_{k+1} = \tilde{A}\tilde{x}_k, \quad \tilde{x}_0$$
$$y_{r,k} = \tilde{C}\tilde{x}_k$$

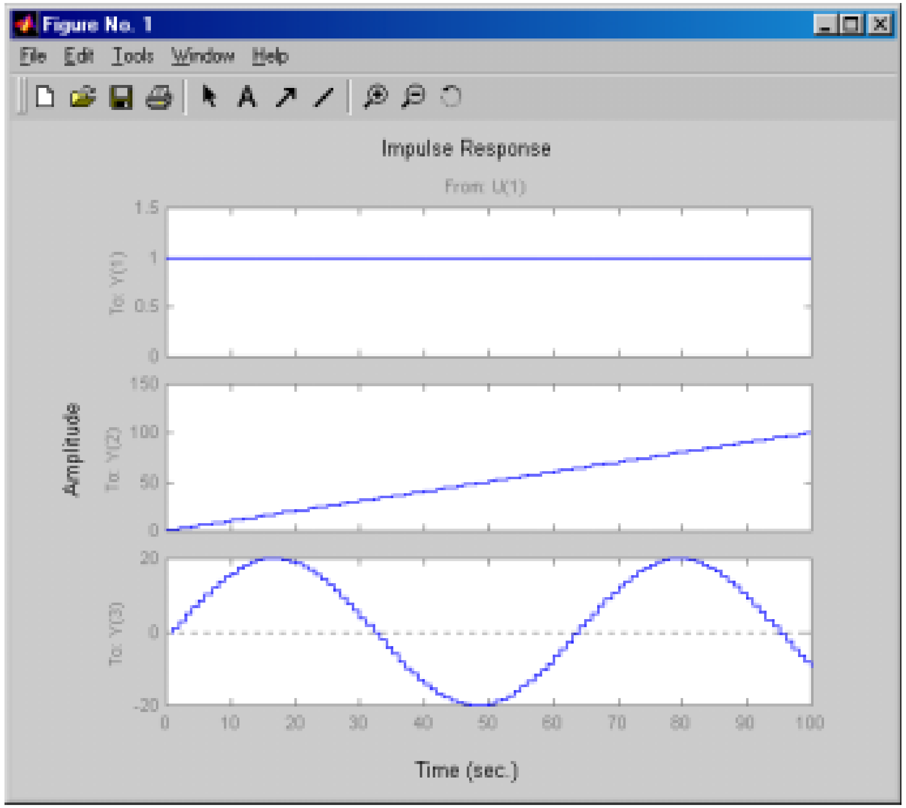
IO type model

$$y_r(d) = \frac{g_{x_0}(d)}{f(d)}$$ where $$\frac{g_{x_0}(d)}{f(d)} = \tilde{C}(I - d\tilde{A})^{-1} d\tilde{x}_0$$

- $f(d)$ chosen (typically unstable) but
- $g_{x_0}(d)$ unspecified - to describe the whole class of signals

Typical reference signals

Plots of $y_r(d) = \dfrac{1}{f(d)}$ for different $f(d)$:



$$f(d) = 1 - d$$

$$f(d) = (1-d)^2$$

$$f(d) = 1 - 1.99d + d^2$$

In real-world applications

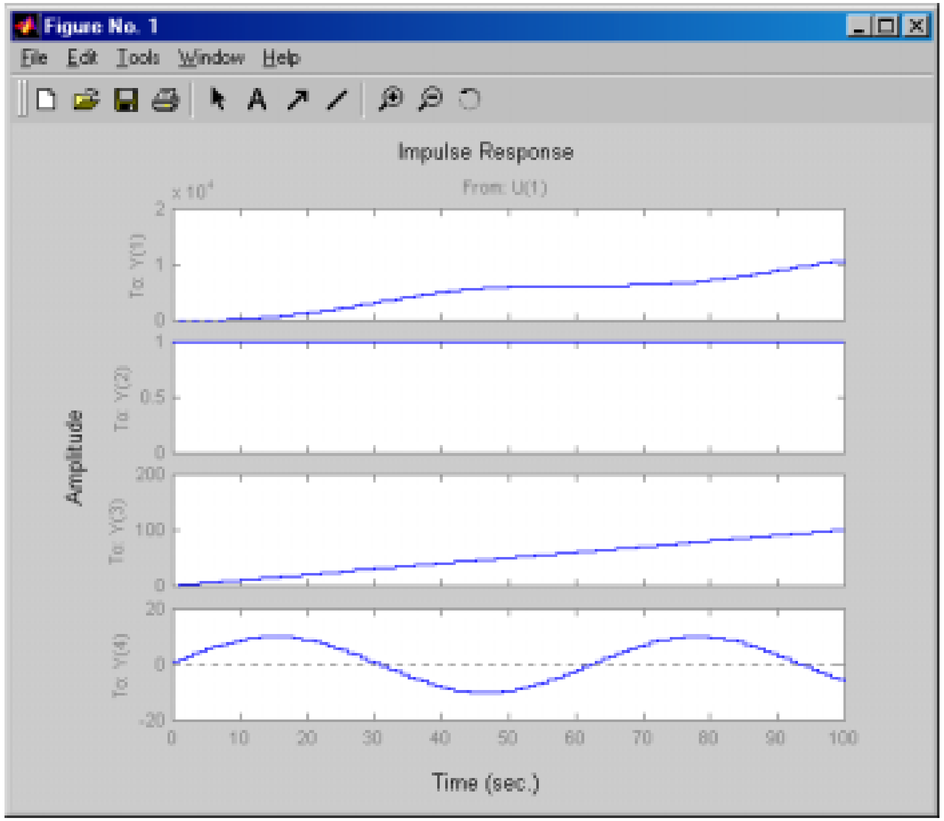
- y_r comes from outside
- it is not exactly of any shape produced by the generator
- yet can be considered as composed of them.

Members of a class

[PolyX]

Plots of $y_r(d) = \dfrac{g_{x_0}(d)}{f(d)}$ for $f = (1-d)^2(1-1.99d+d^2)$

but different $g_{x_0}(d)$:



$g_{x_0} = 1$

$g_{x_0} = 1 - 2.99d + 2.99d^2 - d^3$

$g_{x_0} = 1 - 1.99d + d^2$

$g_{x_0} = 1 - 2d + d^2$

Controller: two degrees of freedom

Controller for tracking

- operates on two signals - plant output y and reference signal y_r - to create an input for the plant
- so the most natural (and most general) linear controller is a two-input one-output system described by

$$p(d)u(d) = -q(d)y(d) + r(d)y_r(d) + s_{\hat{x}_0}(d)$$

or, equivalently, by

$$u(d) = -\frac{q(d)}{p(d)}y(d) + \frac{r(d)}{p(d)}y_r(d) + \frac{s_{\hat{x}_0}(d)}{p(d)}$$

+

feedback + feedforward ← two degrees of freedom

- It must be realized a a single dynamical system with dynamic described by. Its initial condition is unknown but in general nonzero and so is the polynomial $s_{\hat{x}_0}(d)$

Classical controller

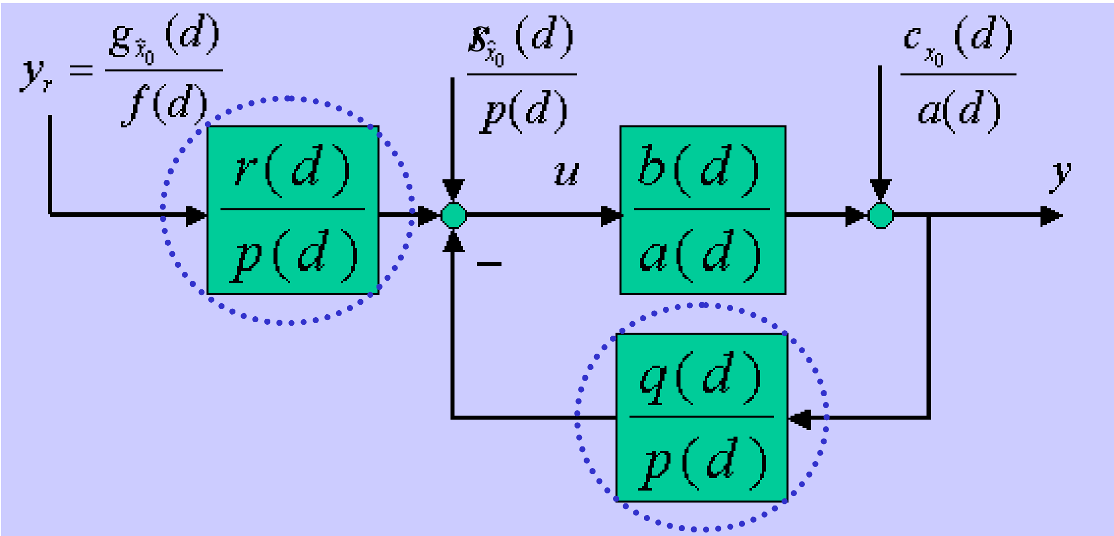
- Classical controller operating on the tracking error $e = y_r - y$ is just a particular case:

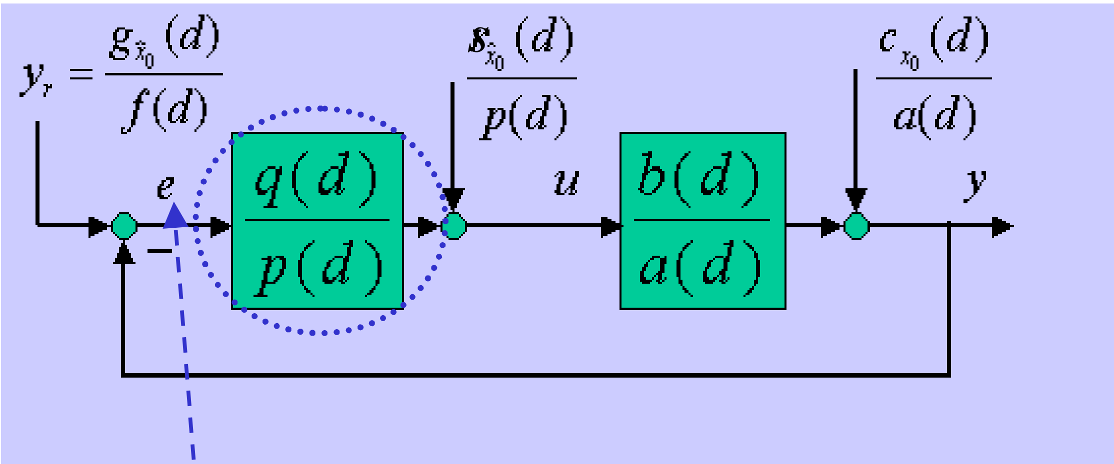
$$u = -\frac{q}{p}y + \frac{r}{p}y_r + \frac{s_{\hat{x}_0}}{p} \quad \xrightarrow{q=r} \quad u = \frac{q}{p}e + \frac{s_{\hat{x}_0}}{p}$$

two degrees of freedom — one degree of freedom

- It is a little more restrictive

Modern and classical structure





tracking error – "error actuated" control

tracking error signal does not exists

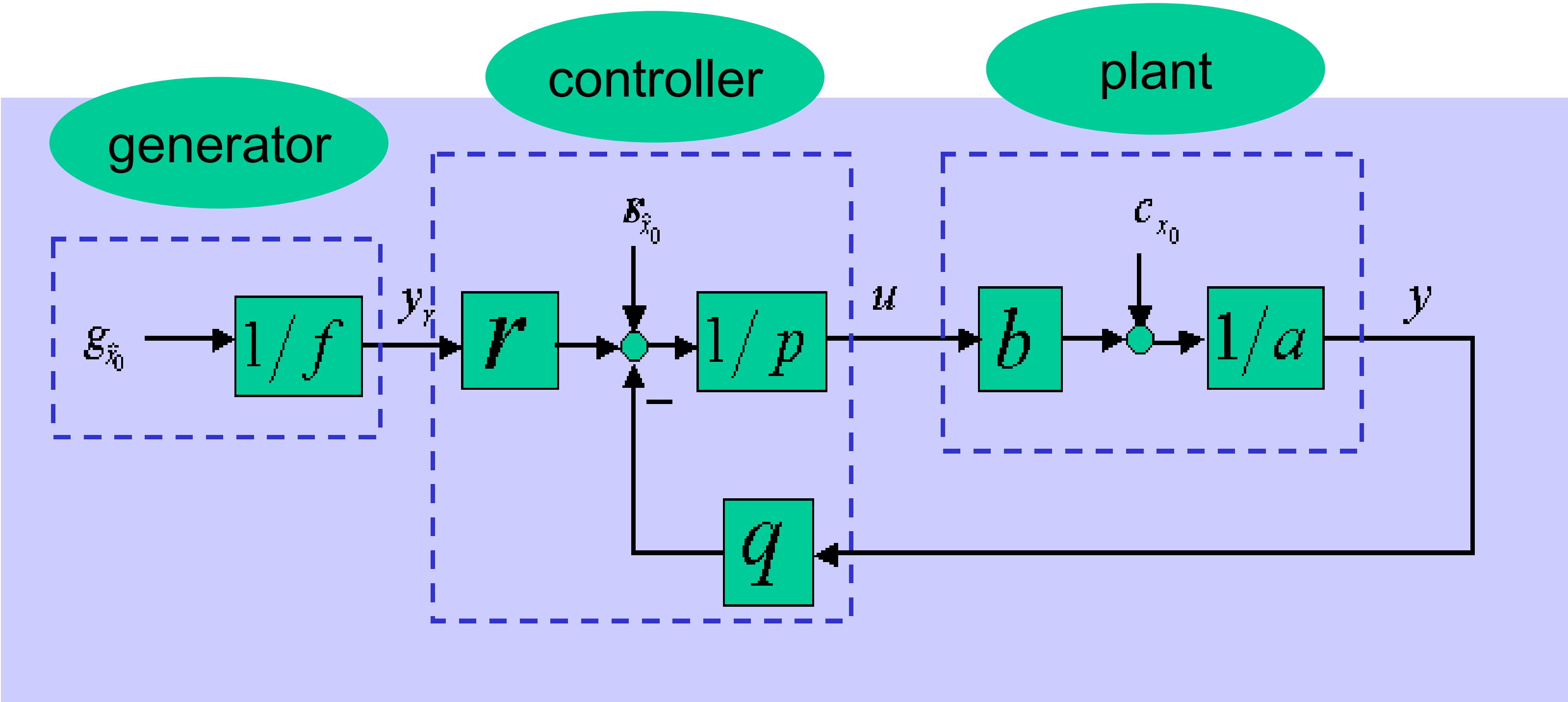
Two

degrees of freedom

One

Two-degrees-of-freedom structure

Two degrees of freedom



- tracking error does not exists – it only serves to measure quality of tracking
- no initial condition is fixed = $g_{\tilde{x}_0}, s_{\hat{x}_0}, c_{x_0}$ undetermined

Signals

Interesting signals

in the two-degrees-of-freedom structure are described by

$$u = \frac{a}{ap+bq} s_{\hat{x}_0} - \frac{q}{ap+bq} c_{x_0} - \frac{r}{ap+bq} \frac{a}{f} g_{\tilde{x}_0}$$

$$y = \frac{b}{ap+bq} r_{\hat{x}_0} + \frac{p}{ap+bq} c_{x_0} + \frac{r}{ap+bq} \frac{b}{f} g_{\tilde{x}_0}$$

$$e = y_r - y = -\frac{b}{ap+bq} r_{\hat{x}_0} - \frac{p}{ap+bq} c_{x_0} + \left(1 - \frac{br}{ap+bq}\right) \frac{1}{f} g_{\tilde{x}_0}$$

Asymptotic tracking

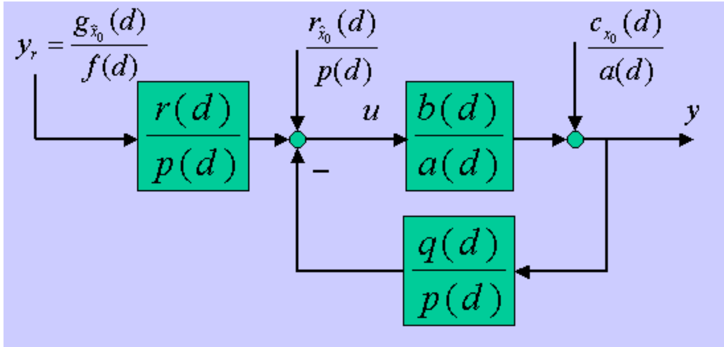
Asymptotic tracking

Formulation

Achieved iff both sequences

$u, e = y_r - y$ are Schur stable

for any combination of $c_{x_0}, s_{\hat{x}_0}, g_{\tilde{x}_0}$



Solution

All asymptotic regulators result from the solution of two equations

$$ap + bq = m \quad \text{and} \quad f^- t + br = m$$

for a stable polynomial m .

Solvability

1) (a, b) stable; 2) $(f^-, b) = 1$; 3) $f^- \mid a$.

Derivation

Simple inspection of the formulas

$$u = \frac{a}{ap+bq} s_{\hat{x}_0} - \frac{q}{ap+bq} c_{x_0} - \frac{r}{ap+bq} \frac{a}{f} g_{\tilde{x}_0}$$

$$e = -\frac{b}{ap+bq} r_{\hat{x}_0} - \frac{p}{ap+bq} c_{x_0} + \left(1 - \frac{br}{ap+bq}\right) \frac{1}{f} g_{\tilde{x}_0}$$

reveals that most of the terms in u and e are stable sequences for all particular values of $s_{\hat{x}_0}, c_{x_0}, g_{\tilde{x}_0}$ iff

$$\boxed{ap + bq = m}$$

is a stable polynomial.

Derivation - 2

This actually results in

$$u = \frac{a}{m} s_{\hat{x}_0} - \frac{q}{m} c_{x_0} - \frac{ra}{mf} g_{\tilde{x}_0}$$

$$e = -\frac{b}{m} r_{\hat{x}_0} - \frac{p}{m} c_{x_0} + \frac{m - br}{mf} g_{\tilde{x}_0}$$

The third term in e is stable only if

$$m - br = f^- t$$

for some polynomial t . This is accomplished by computing r from a another polynomial equation

$$\boxed{f^- t + br = m}$$

Derivation - 3

For already computed *p,q* and *r*, the investigated signals look like

$$u = \frac{a}{m} s_{\hat{x}_0} - \frac{q}{m} c_{x_0} - \frac{ra}{mf} g_{\tilde{x}_0}$$

$$e = -\frac{b}{m} r_{\hat{x}_0} - \frac{p}{m} c_{x_0} + \frac{t}{mf^+} g_{\tilde{x}_0}$$

with all the terms stable but one. To have also the third term in *u* stable, its unstable factor of *f* must cancel. But we can use neither *g* (undetermined) neither *r* (the second equation would not be solvable. Hence it can only cancel with *a,* so we have the condition

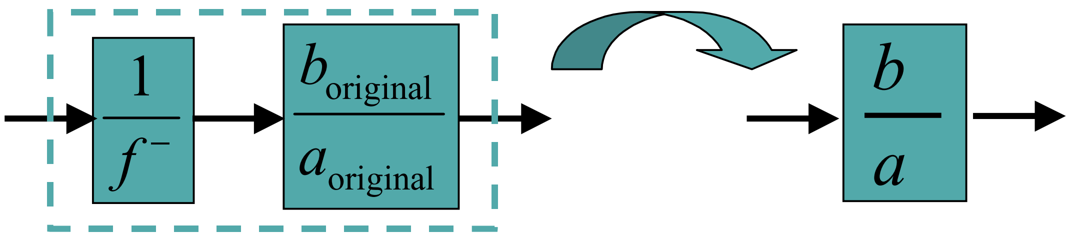
$$f^- \mid a$$

but this cannot be influenced by design. The other two conditions arise from solvability of the two equations.

Physical interpretation

The solvability conditions have a nice physical interpretation:

- Stability of (a,b) means stabilizability of the plant
- $(f^-,b)=1$ is a general condition for tracking arising from the definition of plant zeros: no unstable modes can be transmitted through the plant that equal its zeros.
- $f^- | a$ is also quite natural: the plant driven by a stable input can asymptotically track only such unstable signals that it is able to generate by itself.
- If $f^- | a$ does not hold, the tracking system can still do the job but we must give up the requirement of stable input. This is often done in practical applications. Typically, the system used as plant for design is actually augmented from an original plant and a prefilter



Comments

Two degrees of freedom vs. one degree of freedom

- the particular case of *q=r* is easily identified. The two design equations

$$ap + bq = m$$
$$f^{-}t + br = m$$

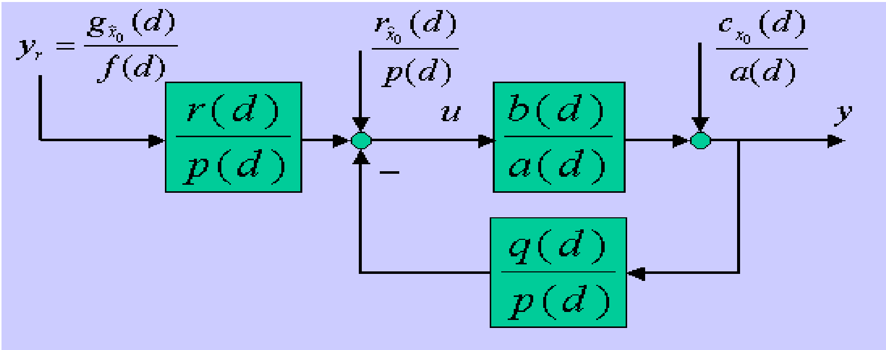
have a solution with *q=r* iff $ap = f^{-}t$
hence the classical structure requires unstable modes of reference to be in the plant or to be put in the controller even when unstable input is allowed. But this is just a minor restriction.

The choice of solution

- Any solutions of the equations do the job. The get a minimum order controller, we pick the one which minimizes

$$\max\{\deg p, \deg q, \deg r\}$$

Deadbeat tracking



Impossible in c-t !

Deadbeat tracking

Formulation 1 / Strong version

Achieved iff both sequences e, u have finite-length (i.e., $e(d), u(d)$ are polynomials!) for any combination of $c_{x_0}, s_{\hat{x}_0}, g_{\tilde{x}_0}$

Formulation 2 / Weak version

Achieved iff e has finite-length while u converges to zero, (i.e., $e(d)$ is s polynomial and $u(d)$ is infinite but stable!) for any combination of $c_{x_0}, s_{\hat{x}_0}, g_{\tilde{x}_0}$

Solution

Solution - Weak

All deadbeat tracking controllers result from the solution of two equations $ap + bq = b^+$ and $ft + br = b^+$

Solvability- Weak

1) (a,b) stable; 2) $f \mid a$.

Solution - Strong

All deadbeat tracking controllers result from the solution of two equations $ap + bq = 1$ and $ft + br = 1$

Solvability- Strong

1) $(a,b) = 1$; 2) $f^- \mid a$.

Stochastic problems

To be completed