Distributed Predictive Control and Estimation

–File 1–

João Miranda Lemos
jlml@inesc-id.pt

Instituto Superior Técnico

2022

Course presentation

Welcome to the course on Model Predictive Control (MPC).

MPC is one of the most popular control strategies in industry.

This is due to MPC being able to tackle:

- Constraints;
- Multivariable processes;
- Being easily tunable.

MPC uses:

- Discrete-time state models; hence we'll study its basics.
- access to state; hence we'll study state estimation.

For large scale plants, there can be separate controllers that need to be coordinated: hence we'll study some basics of distributed control.

Course objectives

After completely the corse with success, the student will be able to

- Formulate and solve simple deterministic predictive control problems using a sw package;
- Tune the parameters that configure an MPC controller;
- Understand the basics of the issues affecting stability and performance in a MPC controlled loop.

Bibliography

- J. Miranda Lemos. *Distributed Predictive Control and Estimation*. Slides of the course, available in Fénix.
- J. Miranda Lemos. *Distributed Predictive Control and Estimation – Problems for self study*, available in Fénix.
- J. Miranda Lemos. *Distributed Predictive Control and Estimation – Lab work guide*, available in Fénix.

- J. M. Maciejowski. *Predictive Control with Constraints*. Prentice Hall, 2002.
- J. B. Rawlings, D. Q. Mayne, and M. M. Diehl. *Model Predictive Control: Theory, Computation and Design*. Nob Hill Publishing, 2nd Ed., 2017.

Evaluation and grading

- (T) 1 exam in 2 dates (counts the best grade). Minimum grade for approval in the course: 9.0;
- (L) 1 lab project. No minimum grade;
- Final grade: $0,5 \times (T + L)$.

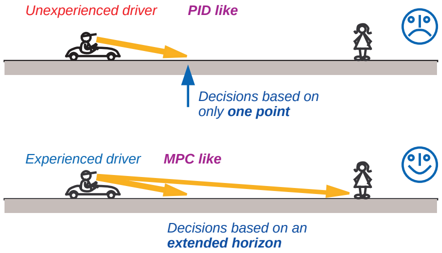
1

The MPC idea and the Receding Horizon Principle

Intuition behind MPC

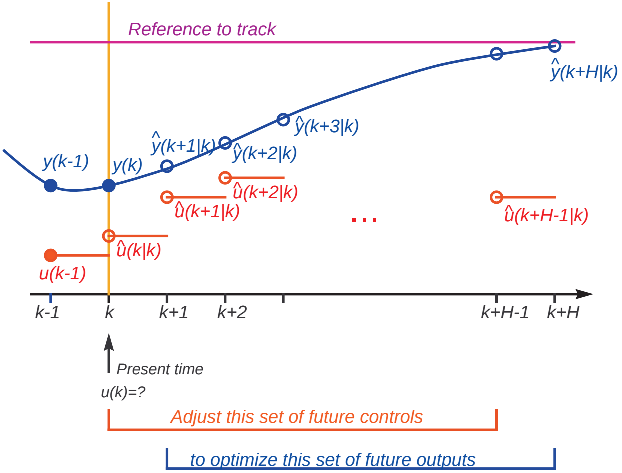
The inexperienced driver basis his decisions on just one point in the future.
Results in excessive actions, possibly instability.

The experienced driver basis his decisions on a horizon.
Anticipates future behaviour of the plant and disturbances.
Smoother actions. Stabilizes difficult systems.



Unexperienced driver
PID like
Decisions based on only **one point**
Experienced driver
MPC like
Decisions based on an **extended horizon**

Basic MPC algorithm



According to the receding horizon (RH) strategy, apply tyo the plant only the first sample of the optimized sequence of future controls and repeat the process at time $k+1$

Basic MPC algorithm (cont.)

1. At present time k, consider the samples of possible future control inputs

$$\hat{u}(k|k),\ \hat{u}(k+1|k),\ \ldots,\ \hat{u}(k+H-1|k)$$

2. This control sequence affect the sequence of predicted output samples

$$\hat{y}(k+1|k),\ \hat{y}(k+2|k),\ \ldots,\ \hat{y}(k+H|k)$$

3. Select the sequence of possible future controls that minimize the cost

$$J(k) = J(\hat{y}(k+1|k), \ldots, \hat{y}(k+H|k),\ \hat{u}(k|k), \ldots,\ \hat{u}(k+H-1|k)$$

   Since the input and output are related by a plant model, $J(k)$ is just a function of $\hat{u}(k|k)$, $\ldots$, $\hat{u}(k+H-1|k)$.

4. Apply to the plant only the first element of this sequence (RH strategy): $u(k) = \hat{u}(k|k)$

5. Increment time $k \leftarrow k+1$ and repeat the procedure from step 1.

MPC components

- Plant model (possibly including disturbances);
- State estimator, based on the plant model;
- State predictor algorithm, based on the plant model;
- Cost function (criterion) to express the desired system behaviour;
- Optimization solver (optimization algorithm) to find future control actions

Key strategy: receding horizon strategy.

Examples of cost functions

Minimum energy (quadratic cost):

$$J_1(k) = \sum_{i=1}^{H} (y(k+i) - r(k+i))^2 + \rho u(k+i-1)^2$$

Minimum fuel (modulus cost):

$$J_2(k) = \sum_{i=1}^{H} |y(k+i) - r(k+i)| + \rho |u(k+i-1)|$$

Notation: $u_{k_1}^{k_2} := \begin{bmatrix} u_{k_1} & \dots & u_{k_2} \end{bmatrix}^\top$.

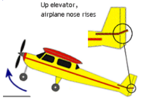
Choice of the cost function

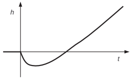
- The choice of the cost function is a key issue in MPC.
- The cost function must reflect the control objectives.
- Some costs reflect the objective of tracking a reference, while others may imply the optimization of economic quantities.
- Not only quadratic or modules costs can be used.
- Quadratic costs have the advantage of being differentiable.
- Modulus costs yield sparse control functions (zero when quadratic yield a small value).

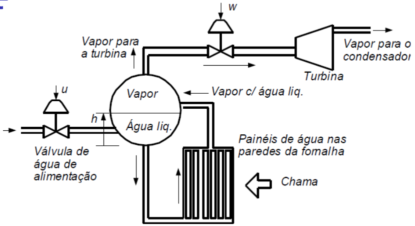
What is a difficult control problem?

- Inverse response;
- Open-loop unstable plants;
- Multivariable plants;
- Explicit incorporation of constraints in controller design
- Nonlinear plants.

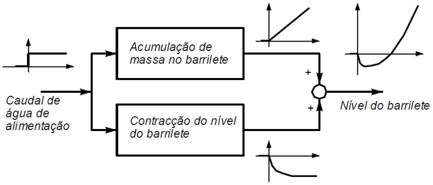
Difficulties: inverse response







Rising the elevator causes the aircraft to rotate, pointing up, but also causes an increase in drag, that reduces the lift force.
The aircraft decreases altitude initially, and only after a while does the altitude increase.


Acumulação de massa no barrilete
Caudal de água de alimentação
Contracção do nível do barrilete
+
+
Nível do barrilete

$$G(s) = \frac{1}{s} - \frac{1}{\tau s + 1} = \frac{(\tau - 1)s + 1}{s(\tau s + 1)}$$

Inverse response corresponds to a zero in the positive half plane.

This zero attracts root-locus branches and causes the closed-loop to be unstable with proportional control for high gains.

Difficulties: inverse response (example)

Discrete time plant with inverse response (zero outside the unit circle)

$$y(k) = \frac{z-1.1}{z(z-0.8)} u(k)$$

$$y(k+1) = 0.8y(k) + u(k) - 1.1u(k-1)$$

Control strategy:

- At present time k, select $u(k)$ such as to minimize the one-step ahead cost

$$J_1 = y^2(k+1)$$

Solution: use the model in J_1 to express $y(k+1)$ as a function of $u(k)$:

$$J_1 = (0.8y(k) + u(k) - 1.1u(k-1))^2$$

Difficulties: inverse response (example - Cont.)

$$J_1 = (0.8y(k) + u(k) - 1.1u(k-1))^2$$

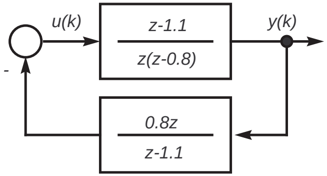
Compute the gradient of J_1 with respect to $u(k)$ and equate to zero, to get the optimal value of $u(k)$:

$$\frac{\partial J_1}{\partial u(k)} = 2(0.8y(k) + u(k) - 1.1u(k-1)) = 0$$

The "optimal" control is computed from

$$u(k) = -0.8y(k) + 1.1u(k-1)$$

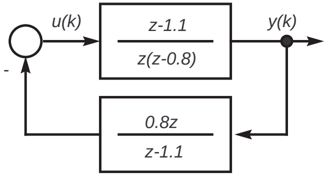
$$u(k) = -\frac{0.8z}{z-1.1}y(k)$$



Difficulties: inverse response (example - Cont.)

$$u(k) = -\frac{0.8z}{z-1.1}y(k)$$

$$y(k) = \frac{z-1.1}{z(z-0.8)}u(k)$$



In matrix form

$$\begin{bmatrix} 1 & -\frac{z-1.1}{z(z-0.8)} \\ \frac{0.8z}{z-1.1}y(k) & 1 \end{bmatrix} \begin{bmatrix} y(k) \\ u(k) \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \end{bmatrix}$$

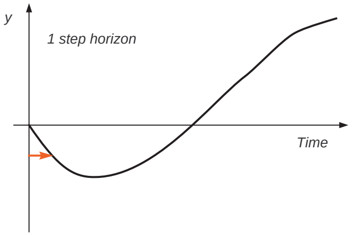
Closed-loop characteristic polynomial

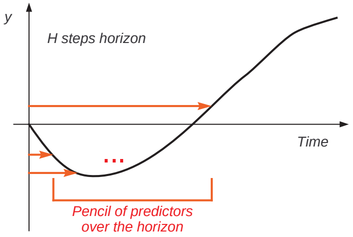
$$z(z-1.1)$$

Unstable!

MPC can tackle inverse responde provided that the horizon is large enough.

Difficulties: inverse response





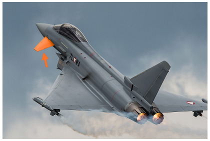
Control based on one-step prediction is mislead by the inverse response, since in the long run the sense of the response is inverted.

Warning: there are many cases in which the horizon most not be longer than the duration of the inverse response to achive stability of the closed-loop.

Difficulties: open-loop unstable plants

Classical examples of open-loop unstable plants:

The inclusion of carnard-wings (orange) to improve maneuverbility causes instability at low speed.



source: eurofighter

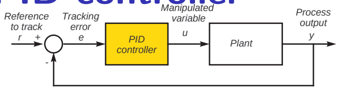
The space-schutle is designed to be unstable at the landing phase.

Source: NASA

Continuous stirred tank reactor with an exothermic reaction.

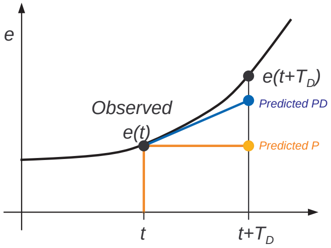
Source: IndiaMART

PID controller



Tracking error:

$$e(t) = r - y(t)$$



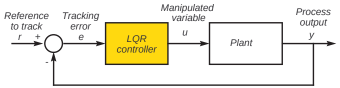
PID control law:

$$u(t) = K_p e(t) + K_D \frac{de}{dt} + K_I \int_0^t e(\tau) d\tau$$

The PD part forms a predictor T_d-secons ahead (Taylor approximation).

- Poor prediction capacity.
- Can't deal with constraints
- Deals poorly with multivariable processes
- Deals poorly with nonlinearities and inverse response

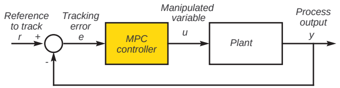
Optimal LQ controller



Optimizes a quadratic cost function defined over an infinite horizon.

- Very good prediction capacity.
- Can't deal with constraints
- Deals very well with multivariable processes
- Deals poorly with nonlinearities and inverse response
- Requires access to state

MPC controller



Optimizes a quadratic cost function defined over an infinite horizon.

- Very good prediction capacity.
- Deals very well with constraints
- Deals very well with multivariable processes
- Deals well with nonlinearities and inverse response
- Requires access to state

Controller comparison

	PID	LQ	MPC
Inverse response	No	Yes	Yes
Unstable plants	Some	Yes	Yes
Multivariable plants	No	Yes	Yes
Constraints	No	No	Yes
Nonlinear plants	No	No	Yes
Computational load	Very low	Very low	High/very high

The definite advantage of MPC is on the possibility to explicitly incorporate constraints.

How did it get this way?

Idea from process industry engineers.
(1970ies) Early motivation from the oil refinery industry

- Jacques Richalet, Adersa (France), *PFC - Predictive Functional Control* 1965.
- B. L. Ramaker and C. R. Cutler, Shell Co. (USA), *DMC - Dynamic Matrix Control*, 1970'ies.

Motivation: Need for a control algorithm easily retunable (respond to raw material - crude - coming from different sources, depending on market conditions), applicable to multivariable plants and that incorporate constraints.

Advances in numerical algorithms, computer hardware, and software allowed applications to fast processes (automotive, aircraft vehicles; spacecraft subjet of research) in

MPC applications: From slow (1970ies) to very fast processes (2020'ies)

Program

1. The MPC idea and the receding horizon principle
2. Review of basic topics
3. Unconstrained, linear, receding horizon
4. Optimization and Constraints
5. Nonlinear MPC
6. Stability constraints
7. State estimation
8. Distributed MPC

Program (detail - I)

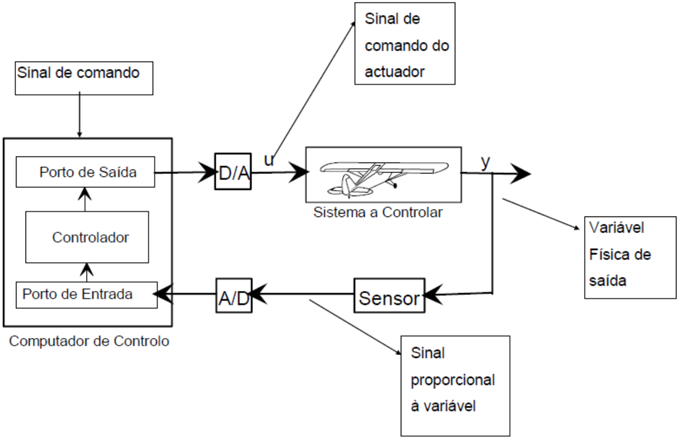
1. The MPC idea and the receding horizon principle
2. Review of basic topics
    1. Discrete time state model
    2. Obtaining discrete state models from continuous models
    3. Obtaining discrete state models from plant data
    4. Optimal control in discrete time
3. Unconstrained, linear, receding horizon
    1. The regulation problem
    2. Reference tracking and disturbance rejection

Program (detail - II)

4. Optimization and Constraints
    1. Convex functions
    2. Quadratic problems
    3. Dual optimization and Lagrange multipliers
5. Nonlinear MPC
6. Stability constraints
7. State estimation
8. Distributed MPC
    1. Decomposition coordination
    2. Game based distributed MPC
    3. ADMM based distributed MPC

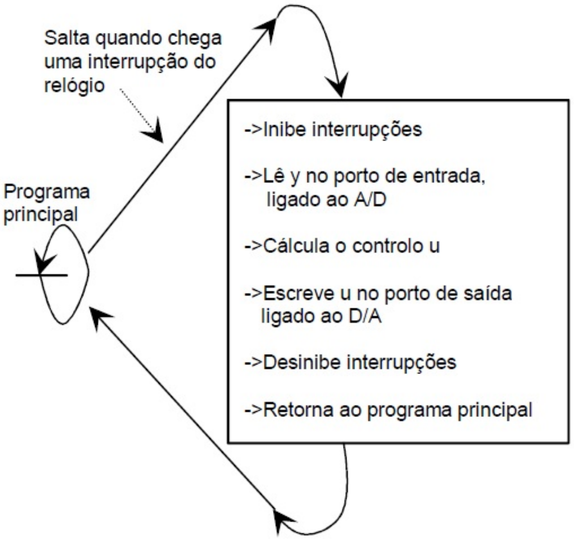
2
Review of basic topics

Computer Control structure

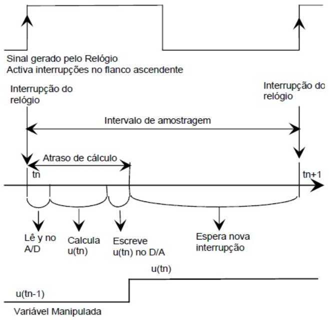


SoftwareStructure4CC

Software structure for Computer Control

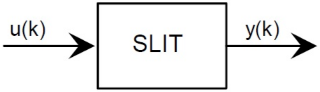

Salta quando chega uma interrupção do relógio
Programa principal
->Inibe interrupções
->Lê y no porto de entrada, ligado ao A/D
->Cálcula o controlo u
->Escreve u no porto de saída ligado ao D/A
->Desinibe interrupções
->Retorna ao programa principal

Discrete time



- The computer considers the variables only at sampling times;
- There is a delay due to computation and A/D and D/A conversion

Linear, Time Invariant Systems (SLITs)



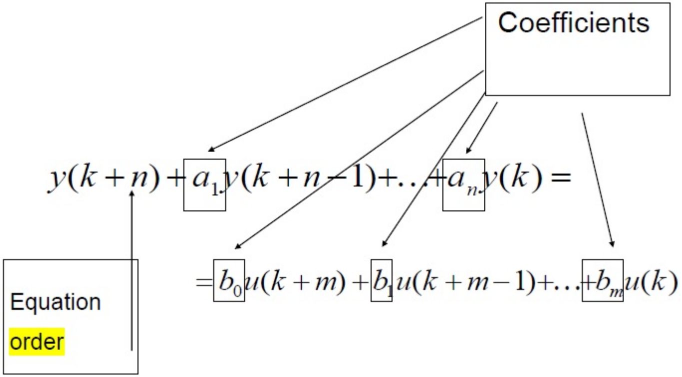
Linearity (Superposition Principle holds):

$$u_1(k) \rightarrow y_1(k)$$
$$u_2(k) \rightarrow y_2(k)$$
$$au_1(k) + bu_2(k) \rightarrow ay_1(k) + by_2(k)$$

Time Invariance

$$u(k) \rightarrow y(k)$$
$$u(k+k_0) \rightarrow y(k+k_0)$$

Difference equations



$$y(k+n)+\boxed{a_1}y(k+n-1)+\ldots+\boxed{a_n}y(k)=$$

$$=\boxed{b_0}u(k+m)+\boxed{b_1}u(k+m-1)+\ldots+\boxed{b_m}u(k)$$

Initial Conditions

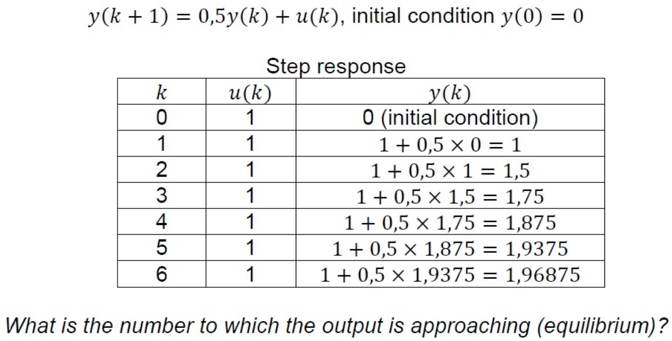
n initial conditions specified:

$$\begin{gathered} y(n-1) \\ \vdots \\ y(0) \end{gathered}$$

Show that:

- The linear difference equation with constant coefficients represents a linear time invariant system
- The solution of the linear difference equations with n initial conditions specified exists and is unique.

Example: 1st order difference equation



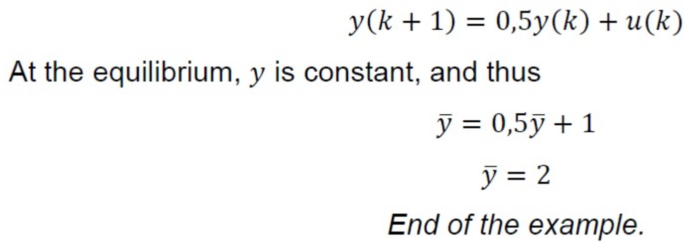
$y(k+1) = 0{,}5y(k) + u(k)$, initial condition $y(0) = 0$

Step response

k	$u(k)$	$y(k)$
0	1	0 (initial condition)
1	1	$1 + 0{,}5 \times 0 = 1$
2	1	$1 + 0{,}5 \times 1 = 1{,}5$
3	1	$1 + 0{,}5 \times 1{,}5 = 1{,}75$
4	1	$1 + 0{,}5 \times 1{,}75 = 1{,}875$
5	1	$1 + 0{,}5 \times 1{,}875 = 1{,}9375$
6	1	$1 + 0{,}5 \times 1{,}9375 = 1{,}96875$

What is the number to which the output is approaching (equilibrium)?

Example: equilibrium of the response



$$y(k+1) = 0{,}5y(k) + u(k)$$

At the equilibrium, y is constant, and thus

$$\bar{y} = 0{,}5\bar{y} + 1$$

$$\bar{y} = 2$$

End of the example.

Forward and backward representations

Difference equation with forward shifts:

$$y(k+n)+a_1y(k+n-1)+\ldots+a_ny(k)=$$
$$=b_0u(k+m)+b_1u(k+m-1)+\ldots+b_mu(k)$$

Difference equation with backwards shifts:

$$y(k)+a_1y(k-1)+\ldots+a_ny(k-n)=$$
$$=b_0u(k-(n-m))+b_1u(k-(n-m)-1)+\ldots+b_mu(k-n)$$

Transform one in the other by shifting time n steps.

Example

$$y(k+2)+a_1y(k+1)+a_2y(k)=u(k+1)$$

In this case, $n=2$, $m=1$.

Delaying the time 2 steps (the order of the system):

$$y(k)+a_1y(k-1)+a_2y(k-2)=u(k-1)$$

Causality

A system is **causal** if *y(k)* depends only on inputs and outputs up to time *k*.

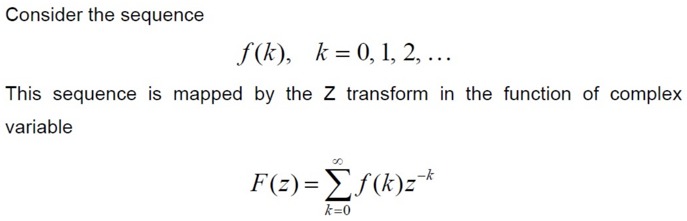
System described by a linear difference equation:

$$y(k+n)+a_1 y(k+n-1)+\ldots+a_n y(k)=$$
$$=b_0 u(k+m)+b_1 u(k+m-1)+\ldots+b_m u(k)$$

This system is causal if

$$\boxed{n>m}$$

The Z-transform


Consider the sequence

$$f(k), \quad k = 0, 1, 2, \ldots$$

This sequence is mapped by the Z transform in the function of complex variable

$$F(z) = \sum_{k=0}^{\infty} f(k) z^{-k}$$

Z-transform: example 1

Find the Z transform of the sequence

$$f(k) = 1 \quad k = 0, 1, 2, \dots$$

Help *(Z transform definition):*

$$F(z) = \sum_{k=0}^{\infty} f(k) z^{-k}$$

Help (Sum of the geometrical series):

$$\sum_{i=0}^{\infty} r^i = \frac{1}{1-r} \qquad \text{for } |r| < 1$$

Z-transform: example 1 (cont)

Solution

$$F(z)=\sum_{k=0}^{\infty} f(k)z^{-k}=\sum_{k=0}^{\infty} z^{-k}=1+\frac{1}{z}+\left(\frac{1}{z}\right)^2+\ldots=$$

$$=\frac{1}{1-\frac{1}{z}}=\frac{z}{z-1}$$

Z-transform: example 2

Find the Z transform of the sequence

$$f(k) = e^{ahk} \quad k = 0, 1, 2, ..$$

***Help** (Z transform definition):*

$$F(z) = \sum_{k=0}^{\infty} f(k) z^{-k}$$

Help (sum of the geometrical series):

$$\sum_{i=0}^{\infty} r^i = \frac{1}{1-r} \quad \text{for } |r| < 1$$

Z-transform: example 2 (cont)

Solution

$$F(z)=\sum_{k=0}^{\infty} e^{ahk} z^{-k} = \sum_{k=0}^{\infty} \left(e^{ah} z^{-1}\right)^{k} = \frac{1}{1-e^{ah}z^{-1}} = \frac{z}{z-e^{ah}}$$

Z-transform properties

1.Linearity

$$Z[\alpha f(k)+\beta g(k)]=\alpha F(z)+\beta G(z)$$

2.Time shift

$$Z[q^{-n} f(k)]=z^{-n} F(z)$$

$$Z[q^{n} f(k)]=z^{n}\left(F(z)-\sum_{j=0}^{n-1} f(j) z^{-j}\right)$$

Ex.: $n=1: \; Z[qf(k)]=z(F(z)-f(0))$

$n=2: \; Z[q^2 f(k)]=z^2(F(z)-f(0)-f(1)z^{-1})$

Z-transform properties (cont.)

3.Final value theorem

$$\lim_{k\to\infty} f(k) = \lim_{z\to 1}(1-z^{-1})F(z)$$

4.Convolution

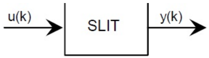
The convolution between two sequences $f(k)$ and $g(k)$ is defined by

$$f*g(k)=\sum_{j=0}^{k} f(j)g(k-j)$$

Then

$$Z\big(f*g\big)=F(z).G(z)$$

Discrete transfer function



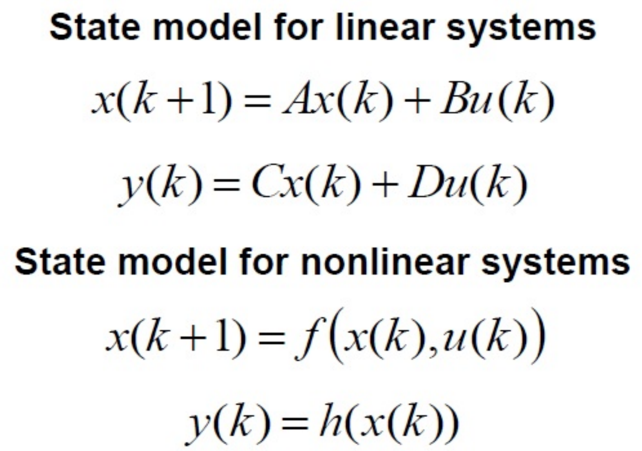
Assume the model represented by the forward difference equation

$$y(k+n)+a_1y(k+n-1)+\ldots+a_ny(k)=b_0u(k+m)+b_1u(k+m-1)+b_mu(k)$$

Take the Z transform with zero initial conditions to obtain the discrete transfer function:

$$G(z)=\frac{Y(z)}{U(z)}=\frac{b_0z^m+b_1z^{m-1}+\ldots+b_m}{z^n+a_1z^{n-1}+a_2z^{n-2}+\ldots+a_n}$$

state models



State model for linear systems

$$x(k+1) = Ax(k) + Bu(k)$$

$$y(k) = Cx(k) + Du(k)$$

State model for nonlinear systems

$$x(k+1) = f\big(x(k), u(k)\big)$$

$$y(k) = h(x(k))$$

obtaining the transfer function from the state model

Apply the Z transfer with zero initial conditions

$$x(k+1) = Ax(k) + Bu(k)$$

$$zX(z) = AX(z) + BU(z)$$

$$(zI - A)X(z) = BU(z)$$

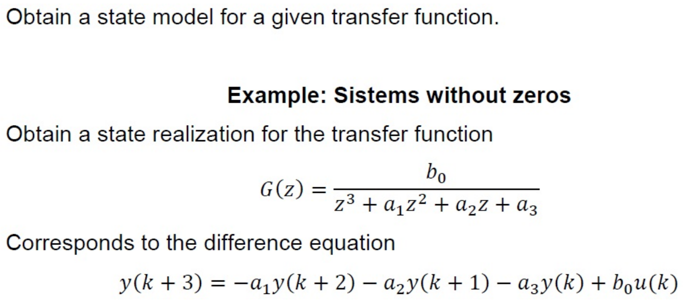
$$X(z) = (zI - A)^{-1}BU(z)$$

$$Y(z) = CX(z) = C(zI - A)^{-1}BU(z)$$

Conclusion: The discrete transfer function

$$G(z) = \frac{Y(z)}{U(z)} = C(zI - A)^{-1}B$$

Obtaining the state model from the transfer function - models without zeros



Obtain a state model for a given transfer function.

Example: Sistems without zeros

Obtain a state realization for the transfer function

$$G(z) = \frac{b_0}{z^3 + a_1 z^2 + a_2 z + a_3}$$

Corresponds to the difference equation

$$y(k+3) = -a_1 y(k+2) - a_2 y(k+1) - a_3 y(k) + b_0 u(k)$$

Obtaining the state model from the transfer function - models without zeros (cont.)

$$y(k+3) = -a_1 y(k+2) - a_2 y(k+1) - a_3 y(k) + b_0 u(k)$$

Take as state variables the output and its first $n-1$ forward shifts. (n is the system order). In this case, $n = 3$, and the state is defined as

$x_1(k) = y(k)$
$x_2(k) = y(k+1)$
$x_3(k) = y(k+2)$

From the definition of the state:

$x_1(k+1) = y(k+1) = x_2(k)$
$x_2(k+1) = y(k+2) = x_3(k)$
$x_3(k+1) = y(k+3)$

From the difference equation

$x_3(k+1) = -a_3 x_1(k) - a_2 x_2(k) - a_1 x_3(k) + b_0 u(k)$

Obtaining the state model from the transfer function - models without zeros (cont.)

$$x_1(k+1) = x_2(k)$$
$$x_2(k+1) = x_3(k)$$
$$x_3(k+1) = -a_3 x_1(k) - a_2 x_2(k) - a_1 x_3(k) + b_0 u(k)$$

State model in matrix form:

$$\begin{bmatrix} x_1(k+1) \\ x_2(k+1) \\ x_3(k+1) \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ -a_3 & -a_2 & -a_1 \end{bmatrix} \begin{bmatrix} x_1(k) \\ x_2(k) \\ x_3(k) \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ b_0 \end{bmatrix} u(k)$$

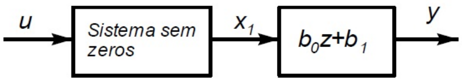
$$y(k) = \begin{bmatrix} 1 & 0 & 0 \end{bmatrix} \begin{bmatrix} x_1(k) \\ x_2(k) \\ x_3(k) \end{bmatrix}$$

Obtaining the state model from the transfer function - models with zeros

Systems with zeros

$$G(z) = \frac{b_0 z + b_1}{z^3 + a_1 z^2 + a_2 z + a_3}$$

Break the system in a part without zeros and a part without poles.
The state is defined as before, but taking x_1 instead of y.



Only the output equation is affected:

$$y(k) = b_1 x_1(k) + b_0 x_2(k)$$