6 - Minimum Variance Control (MVC)

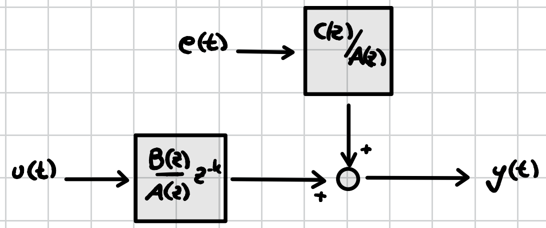
Why a chapter on "control design"?
- we complete the list of tasks in a control-system design problem:
  1) Modeling of the system (see chapter 1)
  2) Estimation of non-measurable states (software sensing) (see chapter 3)
  3) Design of the control algorithm
- the mathematical tools of MVC are very similar to the ones used for system identification
- we can consider MVC as a general tool for stochastic optimization of dynamical feedback systems

The problem of MVC is set around a standard ARMAX model of an I/O system

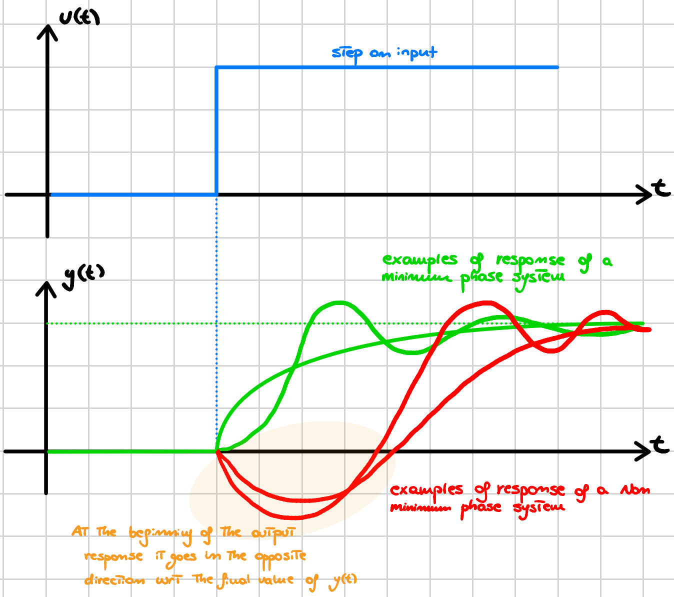
$$y(t) = \frac{B(z)}{A(z)} u(t-k) + \frac{C(z)}{A(z)} e(t) \qquad e(t) \sim WN(0, \lambda^2)$$

pure delay of the system

Noise model (Assumption: written in canonical form)



where
- $B(z) = b_0 + b_1 z^{-1} + \dots + b_p z^{-p}$ with $b_0 \neq 0$, k is the true delay of the system
- $A(z) = 1 + a_1 z^{-1} + \dots + a_m z^{-m}$
- $C(z) = 1 + c_1 z^{-1} + \dots + c_m z^{-m}$



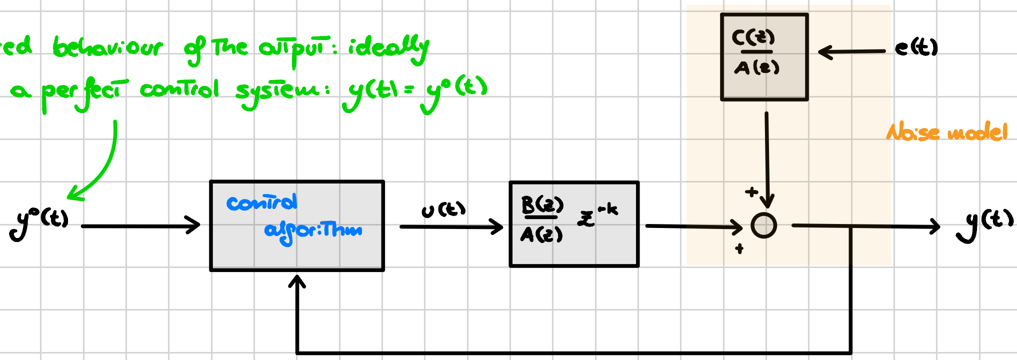
Assumption: $B(z)/A(z)$ is minimum phase (all the zeros of B(z) are strictly inside the unit cycle).

Controlling a non-minimum phase system is much more complicated: you can take easily the wrong decision after the first part of the response

MVC cannot be applied to non-minimum phase systems.
We need its generalization (generalized MVC)

The classical goal of a control system is to let y(t) follow a target. The easiest way to see it is through the block diagram

desired behaviour of the output: ideally in a perfect control system: $y(t) = y^{\circ}(t)$



In a formal way, we can define the problem as an optimal control problem: find the input signal $u(t)$ that minimizes this performance index J:

$$J = E\left[\left(y(t) - y^{\circ}(t) \right)^2 \right]$$

variation of the tracking error $y(t) - y^{\circ}(t)$

This is why is called Minimum variance control.

To solve this problem we need some additional technical assumptions of $y^{\circ}(t)$:

- $y^{\circ}(t)$ and $e(t)$ are uncorrelated ($y^{\circ}(t) \perp e(t)$)
- we assume (worst case assumption) that $y^{\circ}(t)$ is un-predictable: we have no preview on the future desired output $\hat{y}^{\circ}(t+k|t) = y^{\circ}(t)$ (at present time t the best prediction is simply $y^{\circ}(t)$)

Summary of the general setup:

- $\mathcal{S}$: $y(t) = \dfrac{B(z)}{A(z)} u(t-k) + \dfrac{C(z)}{A(z)} e(t) \qquad e(t) \sim WN(0, \lambda^2)$
- $B(z)$ has all the roots inside the unit cycle (minimum phase system)
- $C(z)/A(z)$ is written in canonical form
- $y^{\circ}(t) \perp e(t)$
- $y^{\circ}(t)$ is unpredictable $\Rightarrow \hat{y}^{\circ}(t+k|t) = y^{\circ}(t)$
- Goal: minimize $J = E\left[\left(y(t) - y^{\circ}(t) \right)^2 \right]$

Proof:

Main trick is to split $y(t)$ into predictor and prediction error:

$$\varepsilon(t) = y(t) - \hat{y}(t|t-k) \qquad \overset{\text{real value at time } t \text{ (future)}}{y(t)} = \underbrace{\hat{y}(t|t-k)}_{\text{prediction when present time is } t-k} + \underbrace{\varepsilon(t)}_{\text{prediction error}}$$

So, plugging $y(t) = \hat{y}(t|t-k) + \varepsilon(t)$ into J we obtain

$$\begin{aligned} J &= E\left[\left(\hat{y}(t|t-k) + \varepsilon(t) - y^{\circ}(t) \right)^2 \right] \\ &= E\left[\left(\left(\hat{y}(t|t-k) - y^{\circ}(t) \right) + \varepsilon(t) \right)^2 \right] \\ &= E\left[\left(\hat{y}(t|t-k) - y^{\circ}(t) \right)^2 \right] + \underbrace{E\left[\varepsilon^2(t) \right]}_{\text{Does not depend on } u(t) \text{ (just a function of noise) and not subject of minimization}} + \cancel{2E\left[\varepsilon(t) \left(\hat{y}(t|t-k) - y^{\circ}(t) \right) \right]} \end{aligned}$$

$\varepsilon(t)$ is a combination of $e(t), e(t-1) \dots$

by assumption $y^{\circ}(t) \perp e(t)$

predictor on prediction error must be uncorrelated by construction

Minimizing J with respect to $u(t)$ is equivalent to minimize $E\left[\left(\hat{y}(t|t-k) - y^{\circ}(t) \right)^2 \right]$ with respect to $u(t)$.
We have the minimum of J when $\hat{y}(t|t-k) = y^{\circ}(t)$.

Next step is to find the optimal prediction of $y(t)$ (ARMAX model)

- make the k-step polynomial division $C(z)/A(z)$
- Solution is $E(z)$, residual is $R(z) = \tilde{R}(z) z^{-k}$ so we can write that $\dfrac{C(z)}{A(z)} = E(z) + \dfrac{\tilde{R}(z)}{A(z)} z^{-k}$

Formula of k-step ahead prediction of ARMAX is

$$\hat{y}(t|t-k) = \frac{B(z)E(z)}{C(z)} u(t-k) + \frac{\tilde{R}(z)}{C(z)} y(t-k)$$

Now we can force this identity: $\hat{y}(t+k|t) = y^o(t+k)$

at present time t, $y^o(t+k)$ is not available. The best we can do is replace $y^o(t+k)$ with $y^o(t)$. So we use the identity $\hat{y}(t+k|t) = y^o(t)$. We plug this into the prediction formula

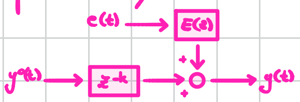
$$\hat{y}(t|t-k) = \frac{B(z)E(z)}{C(z)}u(t) + \frac{\tilde{R}(z)}{C(z)}y(t-k) = y^o(t)$$

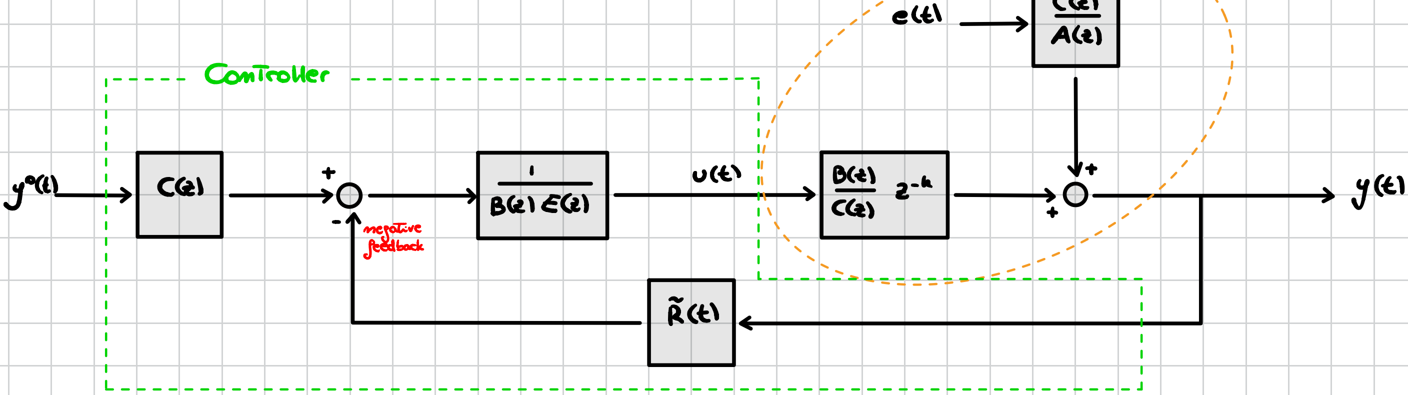
If we explicit $u(t)$:

$$u(t) = \frac{1}{B(z)E(z)}\left(C(z)\,y^o(t) - \tilde{R}(t)\,y(t)\right)$$

general formula for MVC

In closed loop this is equivalent to





Now we can move to analysis of this control system by checking the stability and the performance

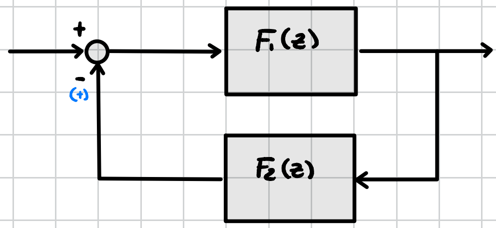
Stability: recall of a basic result of system theory for feedback system

To check the stability of a feedback system, we must compute the so called loop function $L(z)$:

$$L(z) = F_1(z)\,F_2(z)$$

and build the so called characteristic polynomial $X(z)$

$$X(z) = \underbrace{L_N(z)}_{\text{numerator}} \overset{(-)}{+} \underbrace{L_D(z)}_{\text{denominator}}$$



The feedback system is asymptotically stable if and only if all the roots of $X(z)$ are strictly inside the unit cycle.

Apply this result to our case (considering the controller section of the block diagram):

$$L(z) = \frac{1}{B(z)E(z)} \cdot \frac{B(z)}{A(z)} z^{-k}\tilde{R}(z)$$

do not make this cancellation

$$X(z) = L_N(z) + L_D(z) = B(z)\tilde{R}(z)z^{-k} + B(z)E(z)A(z)$$
$$= B(z)\left[\tilde{R}(z)z^{-k} + E(z)A(z)\right]$$
$$= B(z)\left[R(z) + E(z)A(z)\right] = B(z)C(z)$$

- Roots of $B(z)$ are stricty inside the unite cycle thanks to the minimum phase assumption
- Roots of $C(z)$ are stricty inside the unite cycle thanks to the canonical form assumption on $C(z)/A(z)$

So, MVC system is always guaranteed to be asymptotically stable!

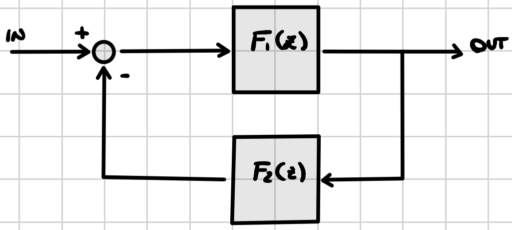
Performance Analysis: relationship $y^o(t) \to y(t)$ (complementary sensitivity) and $e(t) \to y(t)$ (sensitivity to noise) in closed loop. Notice that in closed loop, the output $y(t)$ can be written as

$$y(t) = F_{y^o y}(z)\, y^o(t) + F_{ey}(z)\, e(t)$$

Recall: in a feedback system, the transfer function from IN to OUT

all we find from IN to OUT (direct path)

$$\frac{F_1(z)}{\underbrace{1 + F_1(z)F_2(z)}_{\text{loop function}}}$$



Apply to our system:

- Transfer function from $y^o(t)$ to $y(t)$

$$F_{y^o y}(z) = \frac{C(z)\cdot \frac{1}{B(z)E(z)} \cdot \frac{B(z)}{A(z)} z^{-k}}{1 + \frac{1}{B(z)E(z)} \frac{B(z)}{A(z)} \tilde{R}(z) z^{-k}} = \dots = z^{-k}$$

- Transfer function from $e(t)$ to $y(t)$:

$$F_{ey}(z) = \frac{C(z)/A(z)}{1 + \frac{1}{B(z)E(z)} \frac{B(z)}{A(z)} \tilde{R}(z) z^{-k}} = \dots = E(z)$$

The closed loop optionally controlled system has a really simple behaviour

impossible to obtain 0-noise sensitivity. The best is to limit the output noise to the prediction error

$$y(t) = y^o(t-k) - \boxed{E(z)\, e(t)}$$

we have perfect tracking but with a k-step delay due to the unpredictability of $y^o(t)$

Ideal behaviour will be

$$y(t) = y^o(t) + 0$$

MVC is optimal, impossible to do more or better than this. Notice that after feedback, the resulting I/O system is extremely simple; a lot of dynamics are hidden inside the system as non observable / controllable parts.

Main limit of MVC:

1. Can be applied only to minimum phase system
2. we cannot moderate the effort / intensity of the control action $u(t)$
3. we cannot design a specific behaviour / relationship between $y^o(t)$ and $y(t)$

To overcome these limits, the is an extension of MVC (generalized MVC - GMVC)

Comparison between the performance index of MVC and GMVC

MVC: $J = E[(y(t) - y^o(t))^2]$

we can introduce a reference model $P(z)$ to indicate the desired $y^o(t) \to y(t)$ behaviour [3]

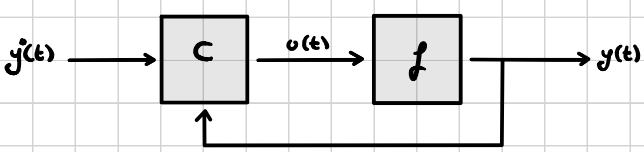
GMVC: $J = E[(y(t) - P(z)\, y^o(t) + \underline{Q(z)\, u(t)})^2]$

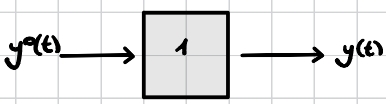
[1] GMVC can manage non-minimum phase system by putting in $P(z)$ the non-minimum phase zeros of $B(z)$

[2] New term: penalization of the control effort

If $P(z)$ and $Q(z)$ are 0, we go back to the simpler MVC. $P(z)$ and $Q(z)$ are designed transfer functions

Reference model $P(z)$: in general in a control feedback system, the expected result is perfect tracking





In this case the reference model is $P(z) = 1$

Is $P(z) = 1$ always the best closed-loop behaviour? In this application is better to give a target behaviour that is $P(z) \neq 1$ but $P(z) = LPF$

Example: consider the following ARMAX system

$$y(t) = \frac{2}{3}y(t-1) + u(t-1) + \frac{1}{3}u(t-2) + 3e(t-2) \qquad e(t) \sim WN(0,1)$$

Design if possible the MVC for this system

Check the assumptions:

1. Transfer function:

$$y(t) = \frac{1+\frac{1}{3}z^{-1}}{1-\frac{2}{3}z^{-1}}u(t-1) + \frac{3}{1-\frac{2}{3}z^{-1}}e(t-2)$$

zero inside the unit disk

→ Not a canonical form

- minimum phase? Yes ($1+\frac{1}{3}z^{-1} = z + \frac{1}{3} = 0 \Rightarrow z = -\frac{1}{3}$ inside the unit cycle)
- canonical form? No ($\frac{3}{1-\frac{2}{3}z^{-1}} e(t-2)$ not monic $\Rightarrow \eta(t) = 3e(t-2) \sim WN(0,9)$

$$y(t) = \frac{1+\frac{1}{3}z^{-1}}{1-\frac{2}{3}z^{-1}}u(t-1) + \frac{1}{1-\frac{2}{3}z^{-1}}\eta(t) \qquad \eta(t) \sim WN(0,9)$$

Standard system representation

Type of ARMAX: $ARMAX(\underbrace{1}_{A(z)},\underbrace{0}_{C(z)},\underbrace{1+1}_{B(z)+\text{delay } k=1})$ with $\lambda^2 = 9$

↑ This is the version of the system to be used

MVC application:

1) Division of k steps between $C(z)$ and $A(z)$

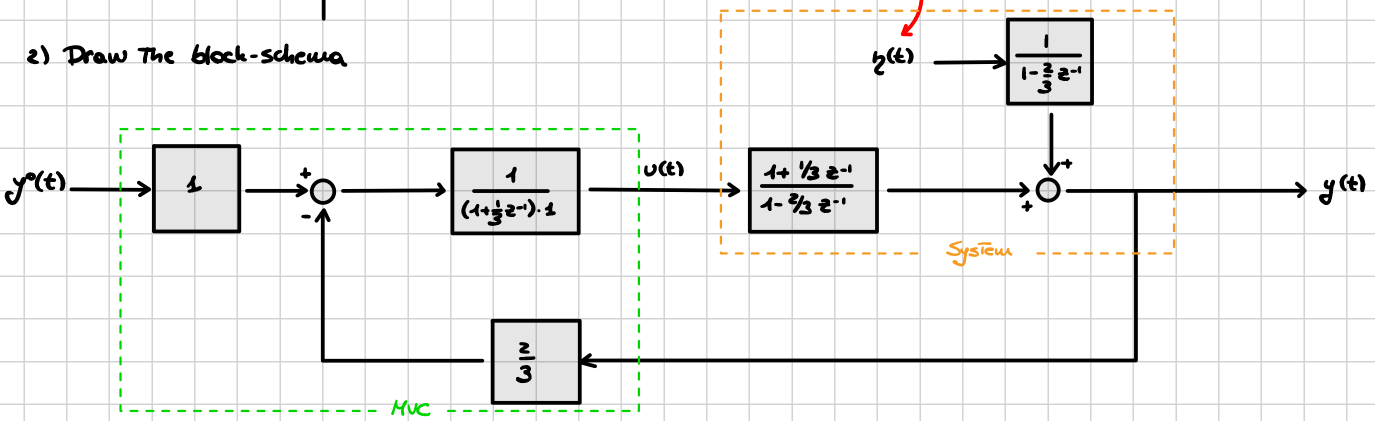
$$\begin{array}{l|l} C(z): \ 1 & A(z): \ 1-\frac{2}{3}z^{-1} \\ \hline -1+\frac{2}{3}z^{-1} & E(z): \ 1 \\ \hline R(z)=\tilde{R}(z)z^{-1}: \ \frac{2}{3}z^{-1} & \end{array}$$

From the long division we obtain

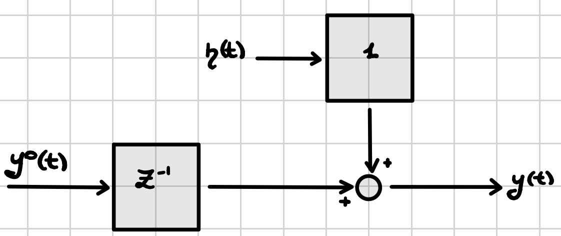
$E(z) = 1 \qquad \tilde{R}(z) = \frac{2}{3}$

we have to use the noise from the canonical version of the model ($\eta(t)$ and NOT $e(t)$)

2) Draw the block-schema



If we compute the transfer function $y^o \to y$ and from $\eta(t) \to y(t)$ we obtain

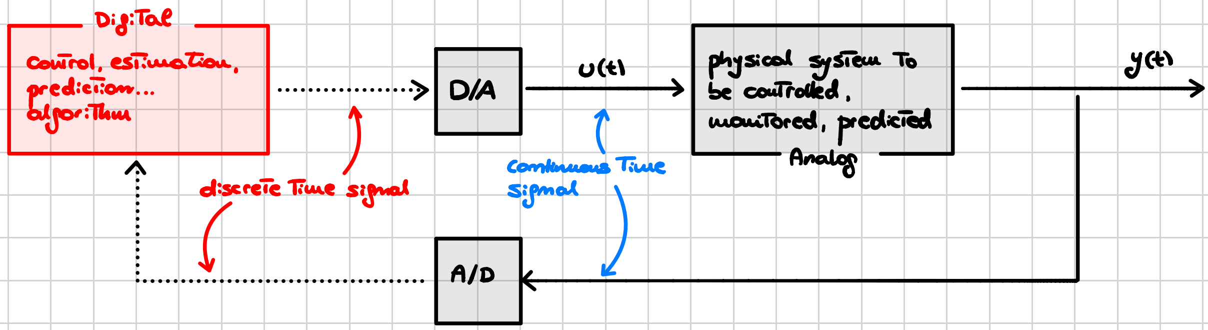


The closed loop behaviour is $y(t) = y^o(t-1) + \eta(t)$

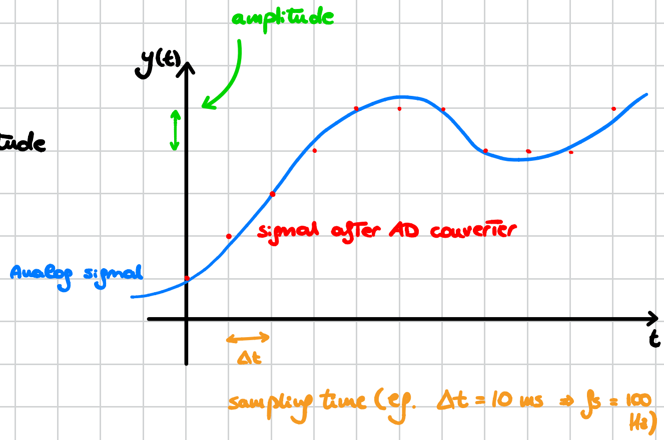
Stability is guaranteed since closed loop poles are the roots of

$$X(z) = B(z)C(z) = (1+\frac{1}{3}z^{-1})\cdot 1 \Rightarrow \text{poles: } z = -\frac{1}{3}$$

Appendix: discretization of analog dynamical systems (important for control, modeling, prediction, SW-sensing ...)



Analog and digital domains are connected with A/D and D/A converters

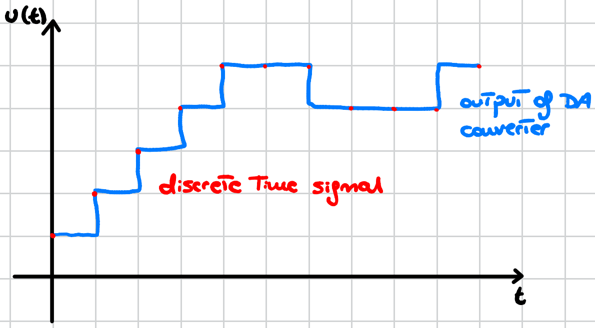


A/D converter makes a discretization of both time and amplitude

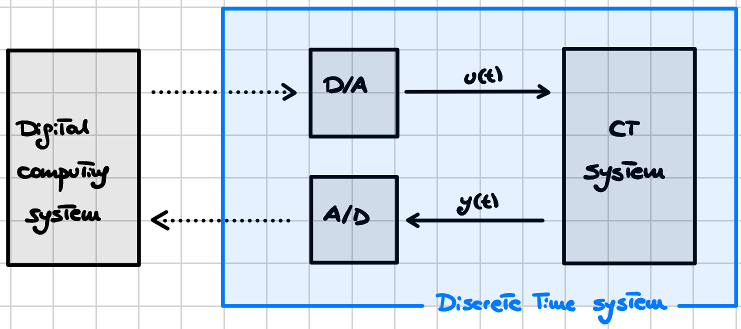
- Time discretization → Δt
- Amplitude discretization → # available discretization levels
  (eg. 10 bit converter → $2^{10} = 1024$ levels)

The cost of an AD converter is linked and mostly dependent on Δt (smaller is better) and # levels (more, better)

DA converter is typically called holder



This step-like converter is called "zero-order hold" (keeps the value until the next). Higher order holder can be used, in practice most of real DA converter are zero-order. If Δt is sufficiently small, the steplike behavior is negligible.
The quality of a DA converter is related to its sampling time



What is the discrete time model of the system? 2 very different solution

1. Make an experiment, collect $\{u(1), \dots, u(N)\}, \{y(1), \dots, y(N)\}$ and estimate (black box) a dynamical model directly from data (learning from data)
2. We have a state-space continuous time model, we want to convert this CT model into DT model

The most used discretization method is the State Space Transformation:

$$S: \begin{cases} \dot{x} = Ax + Bu \\ y = Cx + Du \end{cases} \xrightarrow{\text{Given the sampling time } \Delta t} \begin{cases} x(t+1) = Fx(t) + Gu(t) \\ y(t) = Hx(t) + Du(t) \end{cases}$$

$\{A, B, C, D\}$ continuous time $\qquad\qquad \{F, G, H, D\}$ discrete time

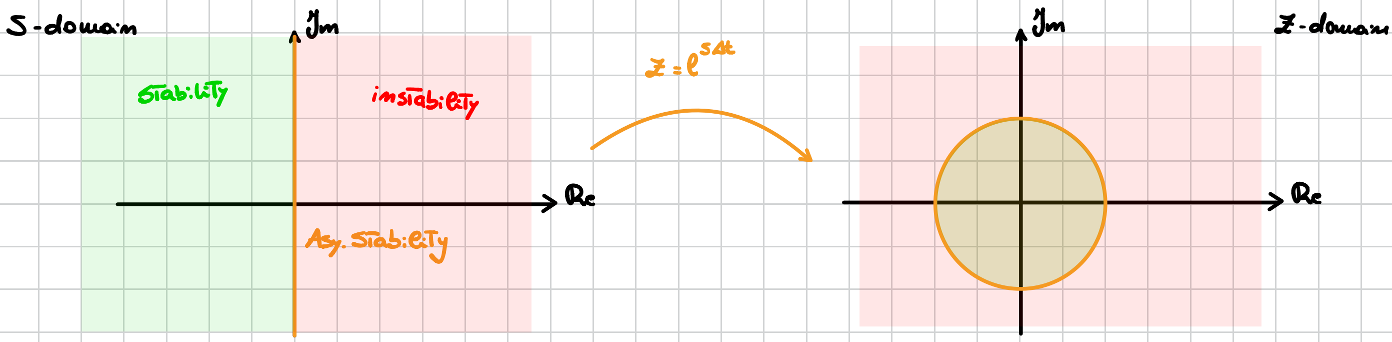
We need to find a way to move from $\{A,B,C,D\}$ in continuous time to $\{F,G,H,D\}$ in discrete time.
State space transformation formula

$$F = e^{A\Delta t} \qquad G = \int_0^{\Delta t} e^{A\delta} B \, d\delta \qquad H = C \qquad D = D$$

Now we have to find how the EIG(A) are transformed into EIG(F)? They simply follow the sampling transformation rule $z = e^{s\Delta t}$; so

$$\ell_F = e^{\lambda_A \Delta t}$$

with $\ell_F = EIG(F)$ and $\lambda_A = EIG(A)$



What about the zeros of the new transformation functions $W(s) \rightarrow W(z)$

$$W(s) = \frac{\text{poly in } s \text{ with } M \text{ zeros}}{\text{poly in } s \text{ in } N \text{ poles}} \qquad M<N \text{ if system is strictly proper}$$

$\{A,B,C,D\}$ ↑

$$W(z) = \frac{\text{poly in } z \text{ with } (N-1) \text{ zeros}}{\text{poly in } z \text{ with } N \text{ poles}}$$

we have n-h-1 zeros generated by the discretization step ; they are called "hidden zeros". Unfortunately, they do not follow a simple rule and usually some of them are outside the unit cycle ($W(z)$ becomes non-minimum phase). We need GMVC...

Another simple and frequently used approach of discretization is the discretization of the time derivative of $\dot{x}$

Eulero Backward: $\quad \dot{x} \approx \dfrac{x(t) - x(t-1)}{\Delta t} = \dfrac{x(t) - z^{-1}x(t)}{\Delta t} = \dfrac{z-1}{z\Delta t} x(t)$

Eulero Forward: $\quad \dot{x} \approx \dfrac{x(t+1) - x(t)}{\Delta t} = \dfrac{z\,x(t) - x(t)}{\Delta t} = \dfrac{z-1}{\Delta t} x(t)$

There is a formula which makes the convex combination of Eulero backward / forward

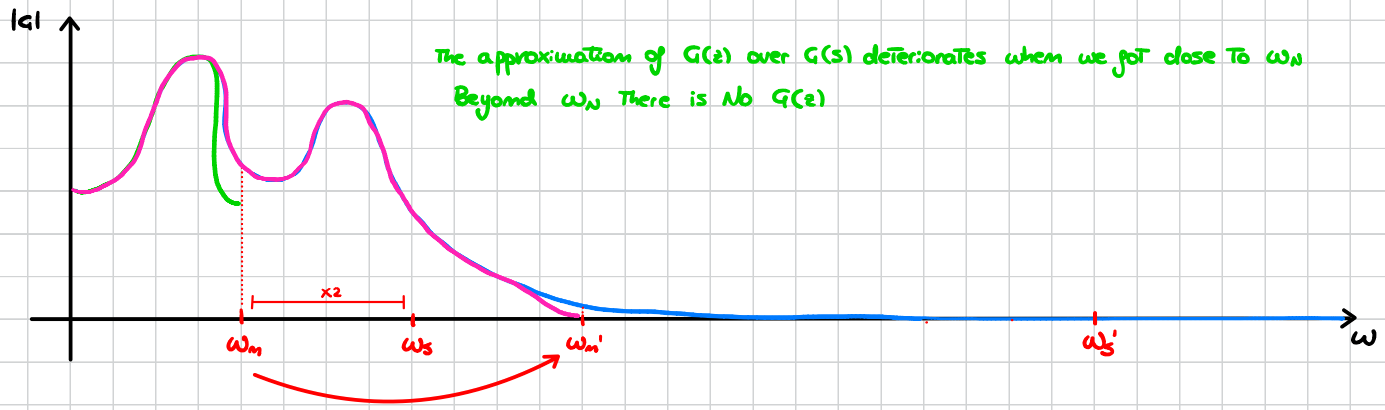
$$\dot{x} \approx \left[\frac{z-1}{\Delta t} \frac{1}{\alpha z + (1-\alpha)} \right] x(t) \qquad \forall \alpha, \quad 0 \le \alpha \le 1$$

where $\alpha \ge 0$ for the Eulero Forward and $\alpha \le 1$ for Eulero Backward. Very used is $\alpha = 1/2$ (Tustin Method)

The most critical choice in discretization is the choice of Δt

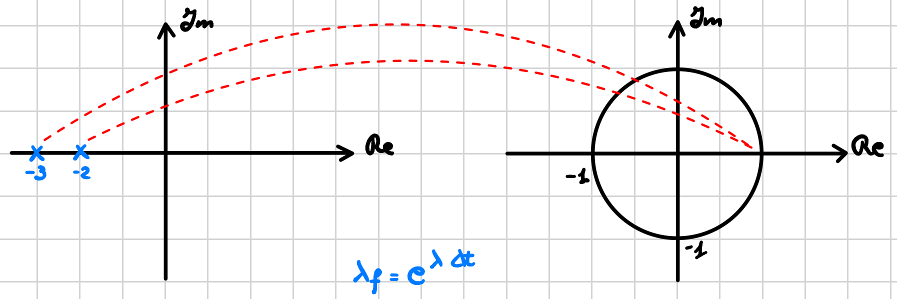
Recap of Notation
- Δt : sampling time [s]
- $f_s = 1/\Delta t$: sampling frequency [Hz]
- $f_N = \frac{1}{2} f_s$: Nyquist frequency [Hz]
- $\omega_s = 2\pi/\Delta t$: sampling angular frequency [Hz]
- $\omega_N = \frac{1}{2}\omega_s$: Nyquist angular frequency [Hz]

General intuitive rule to select f_s : "The bigger, the better". Why?



The bigger is f_s (ω_s) the wider is the bandwidth of good approximation of G(z) are G(s).

Hidden problems of a f_s too big (Δt too small):
1. Sampling devices (A/D and D/A) are more expensive
2. Computational cost: updating and algorithm at 1 μs is much more demanding than updating than 1 ms
3. Cost of memory: buffering a signal at 1 μs sampling time is more demanding than at 1 ms
4. Numerical precision cost:

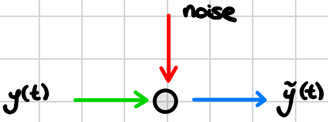


If Δt is very small, the poles in discrete time are very close to 1
0.999994 0,999993
All the poles are squeezed very close to 1, so we need high precision implementation of the algorithm

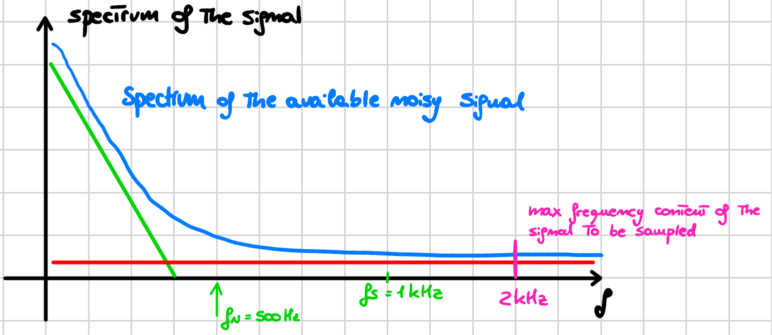
The rule of thumb for the selection of a balanced Δt for a dynamical system is $f_N \approx 10$ times the expected control system bandwidth (or of the bandwidth of interest)

Aliasing: be careful to the aliasing problem (big and critical hidden issue related to A/D converters)

Shannon theorem: the max frequency content of a continuous time signal to be sampled should be smaller than f_N

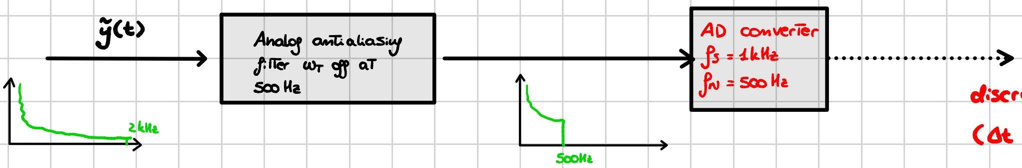


we must discretize $\tilde{y}(t)$



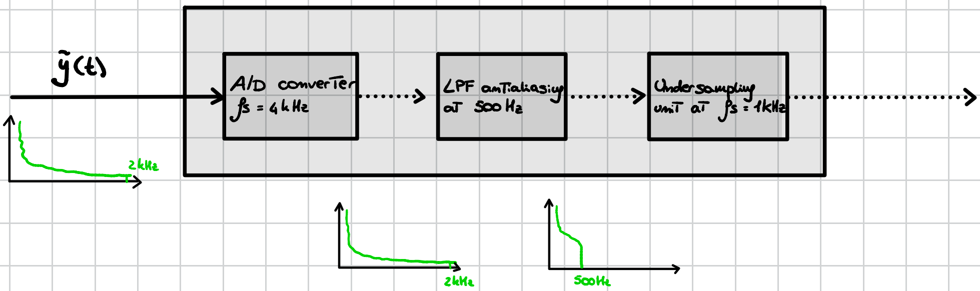
Ex: max frequency content of $\tilde{y}(t)$ is 2kHz, but we want to use a $f_N = 500$ Hz (so $f_S = 1$ kHz).
We cannot directly sample at 1kHz, otherwise we can generate aliasing

- classical way to manage the problem is to use N analog anti-aliasing filters



discrete time signal with $f_S = 1$ kHz ($\Delta t = 1$ms) and $f_N = 0.5$ kHz

- No analog filter is needed but an initial oversampling is required



discrete time signal $f_S = 1$ kHz with No aliasing