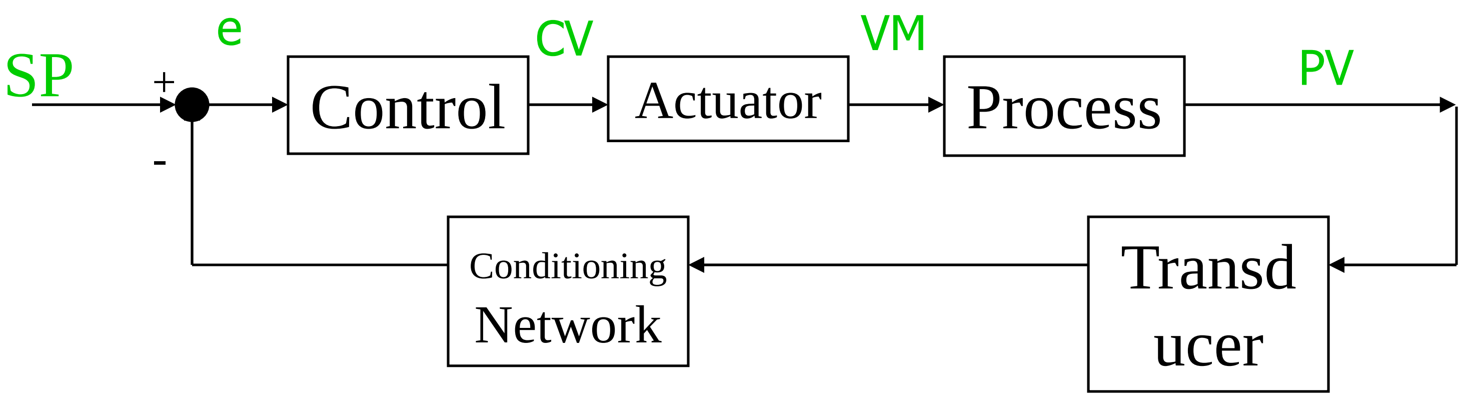
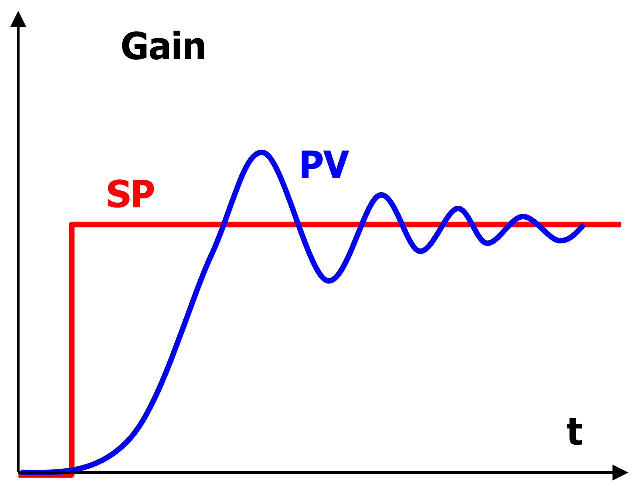
CONTROL ALGORITHMS

ALGORITMI DI CONTROLLO - INTRODUZIONE



- **Control algorithm** = the controller operating principle
- **Process** = the problem. Know the transfer function
- **Minimize** the gap between SP and PV (and dependency on environmental variables)
- **Oscillations** due to the velocity of the answer to SP variations ⇒ minimize the **transient**
- **Overshootings** ⇒ limit the amplitude
- **Stability** (no infinite or diverging oscillations)
- **Delays**



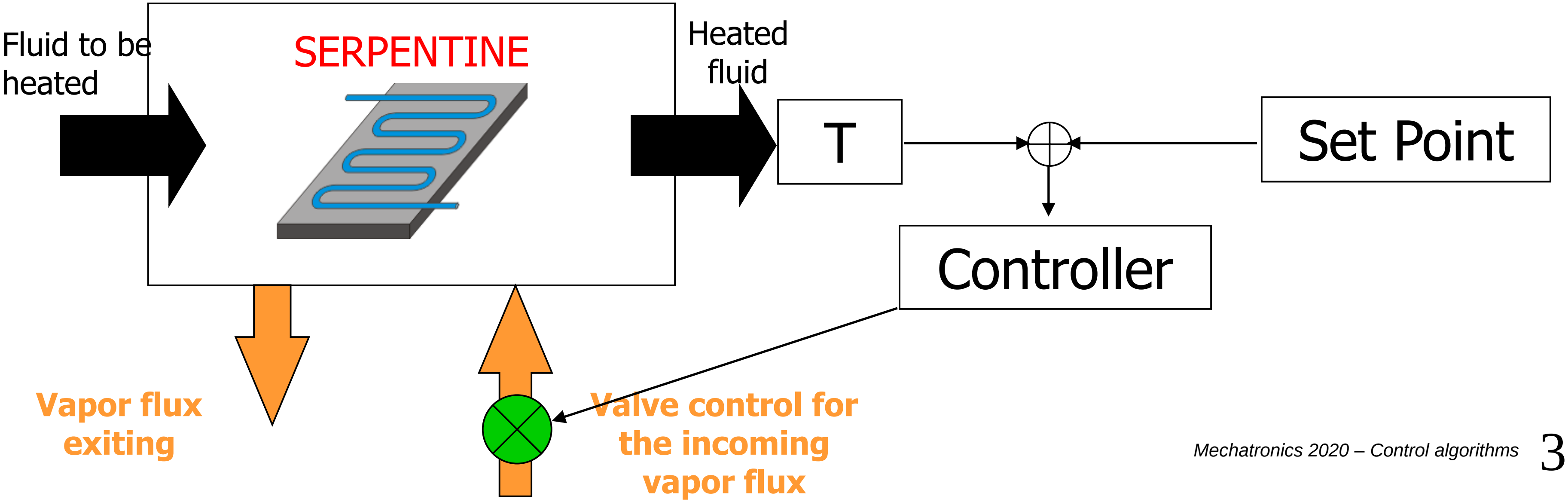
ON OFF CONTROL

REFERENCE CASE (heat exchanger)

- **Vapor sent on a serpentine to heat a fluid crossing it**
- **Temperature acquired through a transducer**
- **Comparison with a Set Point**
- **Controller for the vapor input valve opening/closing**

METHODOLOGY

- **Provide to the system a certain amount of vapor or nothing**



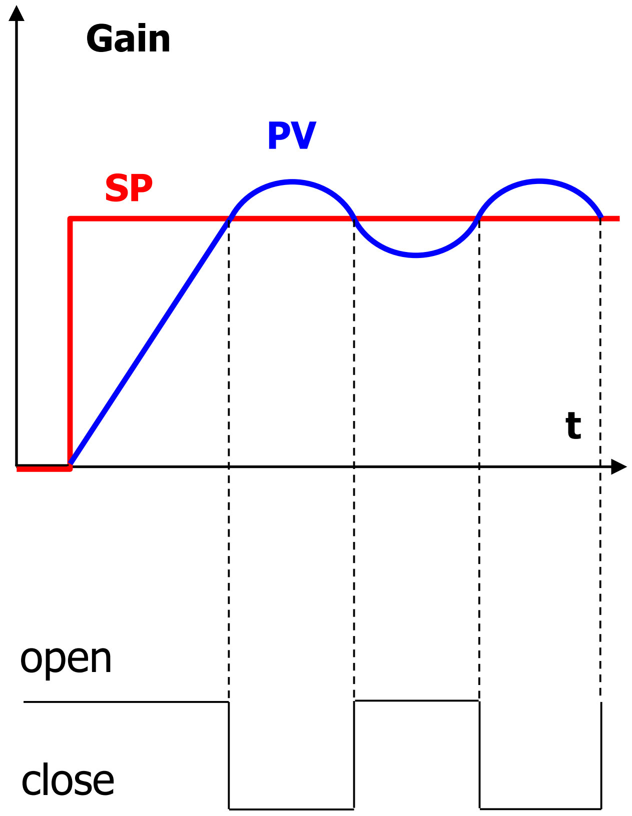
ON OFF CONTROL

OBSERVATIONS

- **SP and PV are temperatures**
- **Until SP>PV the valve is open to provide warm vapor**
- **When the valve is closed, the T° does not diminish immediately since inside the environment a certain amount of heat is still in propagation**
- **successively the temperature goes down to the SP (not immediately however), the valve is open again and the sequence is repeated**

PROBLEMS

- **the temperature reached in most of the cases is not equal to the SP (negligible error?)**
- **continuous solicitations to the valve**
- **connection with the process variation speed**

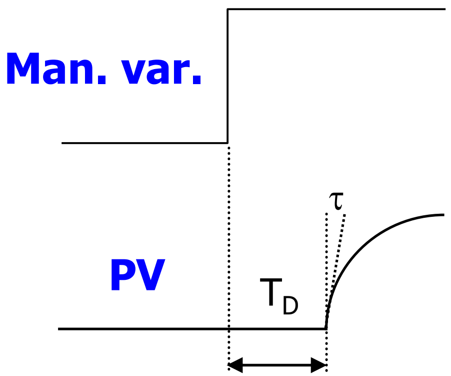


ON OFF CONTROL

REFINING THE MODEL

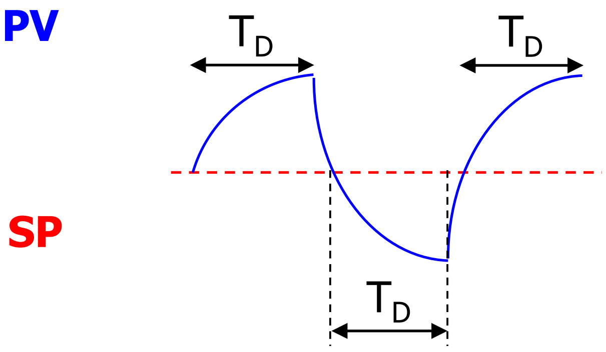
- **The process is slow to propagate the heat in the environment: low pass function can represent it (τ)**
- **A further 'natural' delay (T_D) to respond to the thermal solicitation**

$$\frac{e^{-sT_D}}{1+s\tau}$$



PROCESS ANSWER - 1

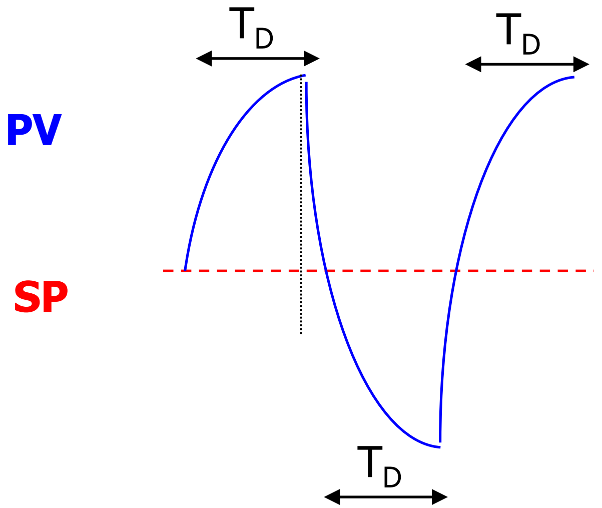
- **It is made of increasing or decreasing exponential parts with delay T_D from the last reached SP**
- **if T_D is small the response assumes a typical *sawtooth* shape (still frequent commutations)**



ON OFF CONTROL

PROCESS ANSWER - 2

- **If τ is small and T_D is constant, the amplitudes of descents and climbs are wider and the answer becomes more similar to a (wide) oscillation. This is not acceptable although depends on the oscillations amplitude.**



SUMMARISING …

- ✓ **τ represents a system ready to react to the input and then developing the answer during the time**
- ✓ **T_D represents a system that does not react immediately but after a while with a strong response: this is more critical in terms of stability**
- ✓ **If T_D >> τ wide oscillations so ON OFF is not good, ok instead if viceversa (a very small ripple take place).**
- ✓ **To diminish the commutations frequency, a hysterical control can be devised.**

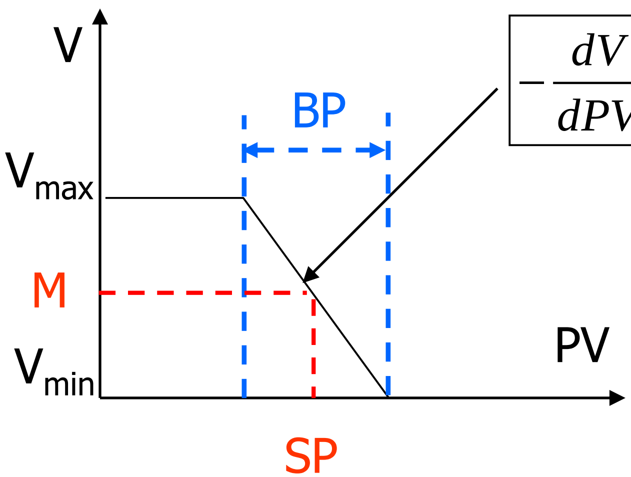
PROPORTIONAL CONTROL

This time a linear relationship is set between the set point and manipulated variable

$$V=k[SP-PV] + M$$

M reset variable, used to avoid the absence of control when the set point is reached.

M=power to provide to keep the process in the desired condition.



$$-\frac{dV}{dPV} = k$$

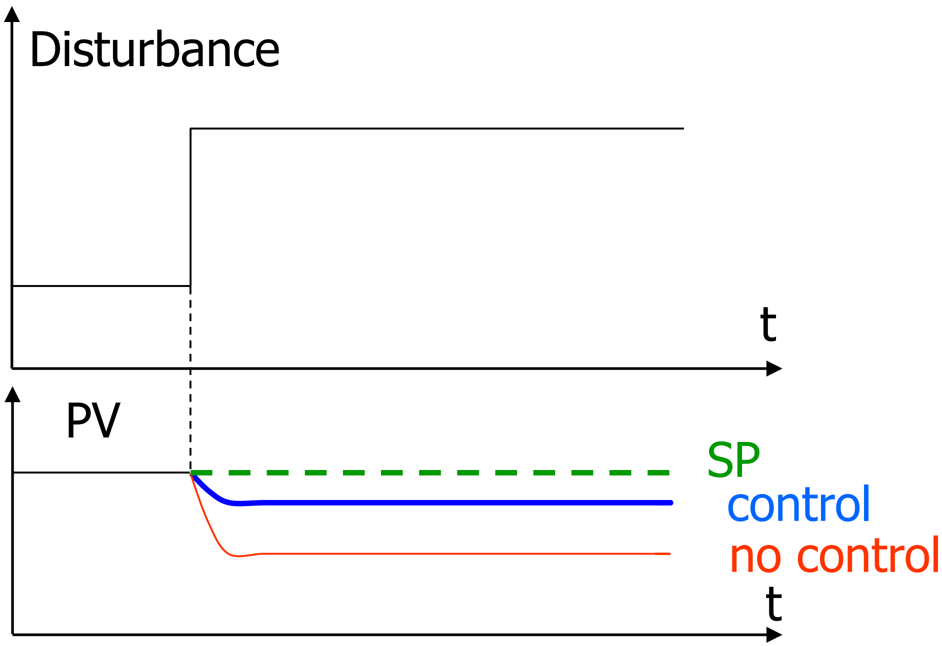
V limited since it is the 'action on the process'

The controller works limited to the range centered on the set point and defined 'band of proportionality' (BP) = V_{max}/k (V_{max}= max. power tolerated by the actuator)

if $k \rightarrow \infty$, ON-OFF controller

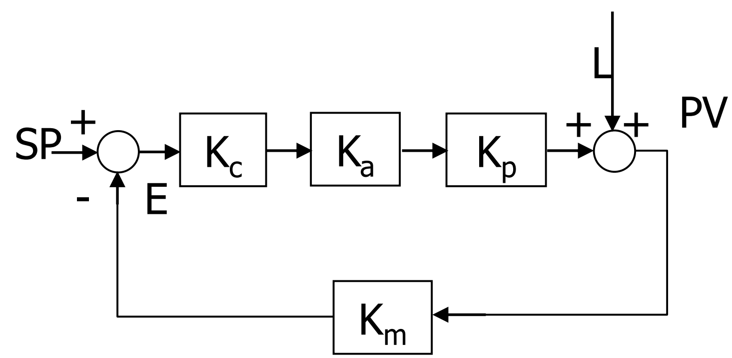
Generally k can not grow without restrictions and a small variation in the system is enough to have a error $\neq 0$ (a constant disturbance) $\Rightarrow$ difficult to assure a null asymptotic error.

PROPORTIONAL CONTROL



Example of a open ring proportional control in presence of a step-wise disturbance.

PV behavior in presence of control and without.



$$PV = \frac{L}{1 + K_p K_a K_m K_c}$$

If SP=0, PV becomes null only if L=0 (no disturbance) or if $1+K_pK_aK_mK_c \rightarrow \infty$ (but this can make the system unstable).

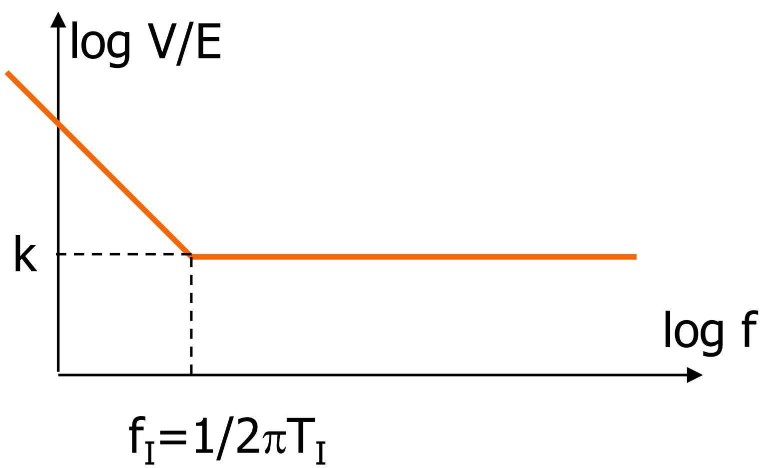
Better if the gain is < 1, but in this case PV will be never zero.

INTEGRAL-PROPORTIONAL CONTROL

In the proportional control, a continuous disturbance cannot be deleted unless we modify the value of M. A method to make this automatic is to introduce an integral term that is a value depending on the integral of the error.

The integration of the error E is a temporal «memory» that affects and influences the manipulated variable. This term replaces the M reset value.

$$\mathbf{V = k[SP - PV] + k' \int_{t0}^{t} [SP - PV]\,dt = k\left[E + \frac{1}{T_I}\int_{t0}^{t} E\,dt\right]} \qquad \mathbf{T_I = \frac{k}{k'}} \text{ (time to reset)}$$

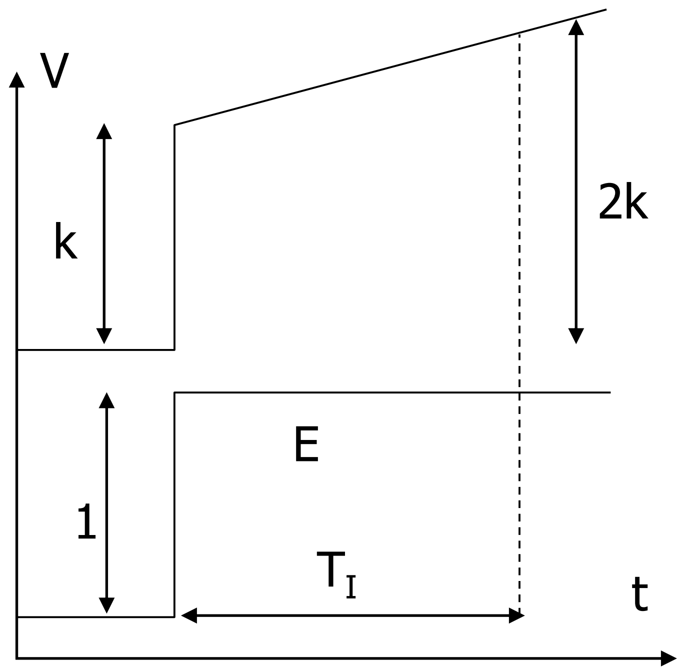


$$\mathbf{V(s) = kE(s)\frac{1}{sT_I}(1 + sT_I)}$$

At high frequencies the integral term does not exert any influence.

If the «nature» of the process should be derivative, to apply the PI control does not work.

INTEGRAL-PROPORTIONAL CONTROL

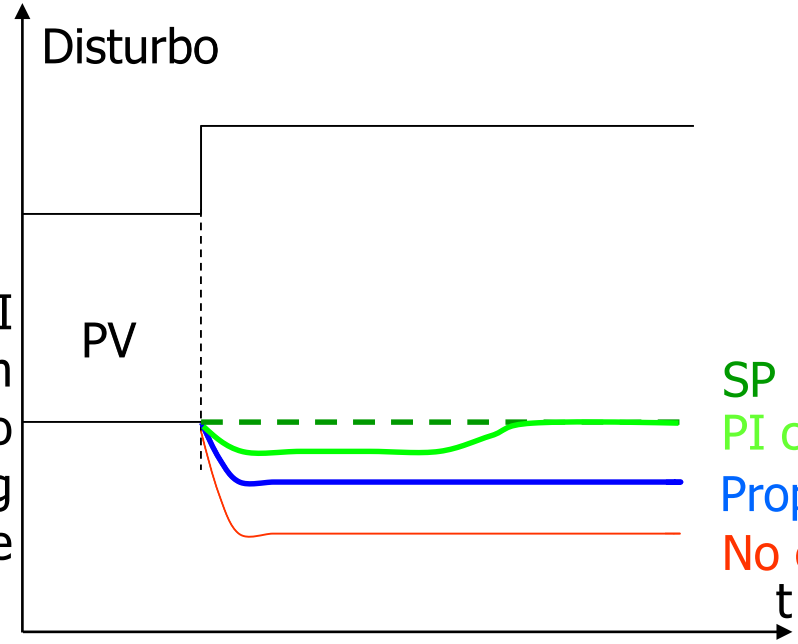


The figure shows the controller time response to a «stepwise» disturbance.

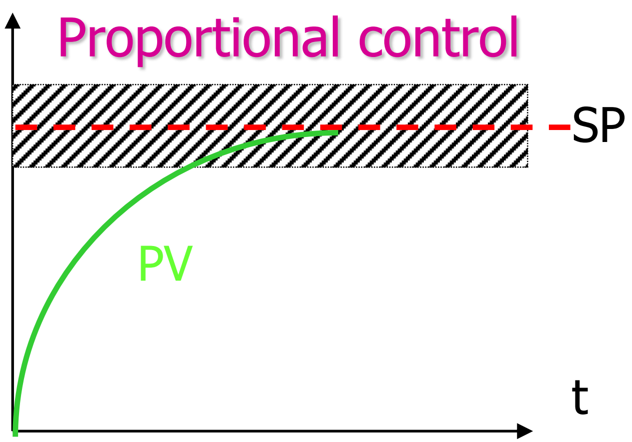
The integral term will reach the intensity of the proportional one after T_I time instants.

The action of the integral term will be as slower as greater is T_I .

When the transient is exhausted the PI controller gain tends to ∞ (different from the proportional) and this allows to delete the error without compromising the stability (typically process poles are at $f>f_I$ where the gain is constant)

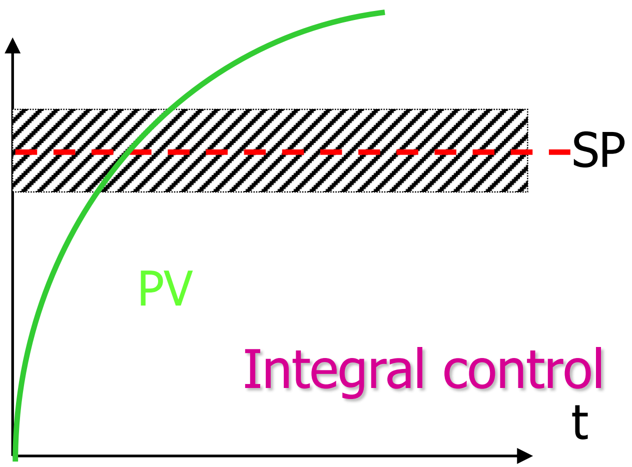


STRONG SET POINT VARIATIONS OR START UP



With a proportional control when booting the system, the PV grows up with the maximum power that is continuously applied until it is not within the BP area.

Within the BP the slope gradually diminishes while approaching the SP.



With a PI control, in t=0 the integral term is negligible and the maximum power can be considered as applied.

Approaching the BP the integral term starts to work and to influence the control, and the greater the time running from the beginning the heavier will be this PI contribution to the point that it can even cause the «exit» from the BP.

Better to use numerical techniques.

Moreover in case of quick transient variations of the SP, the effect of the initial deviation E affects long time the control, due to the «slow» component tied to the integral. This is a limit the suggest the introduction of the PID.

DERIVATIVE-INTEGRAL-PROPORTIONAL CONTROL

The derivate of the error (SP-PV) allows to «anticipate» the control strengthen it, that is adding a contribute that provides a very faster response than from the integral (that operates only after a suitable interval of time).

Of course a suitable coefficient is needed to avoid excessive o too low actions.

If, however, a similar expression would be used:

$$V = k\left[E + \frac{1}{T_I}\int_{t0}^{t} E dt + T_D \frac{dE}{dt}\right] \overset{\text{Laplace}}{\Rightarrow} kE\left[1 + \frac{1}{sT_I} + sT_D\right]$$

That diverges when s →∞

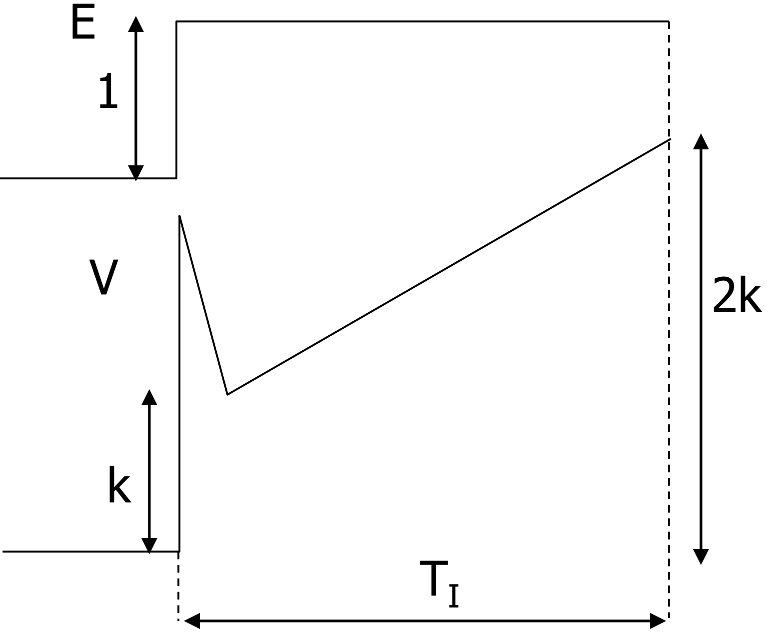
To keep the gain limited a low pass filter can be added.

$$V = kE\left[1 + \frac{1}{sT_I} + \frac{sT_D}{1+sT_F}\right] \quad \text{or} \quad V = \frac{kE\left[1 + \frac{1}{sT_I} + sT_D\right]}{1+sT_F}$$

This kind of control is defined PID. The three time constants T_D, T_I, T_F are conceived in a different way (medium, long, short) so do not interact reciprocally.

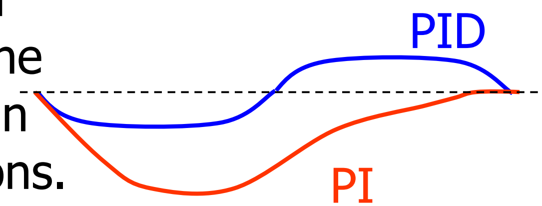
From here the «non interactive PID» takes origin.

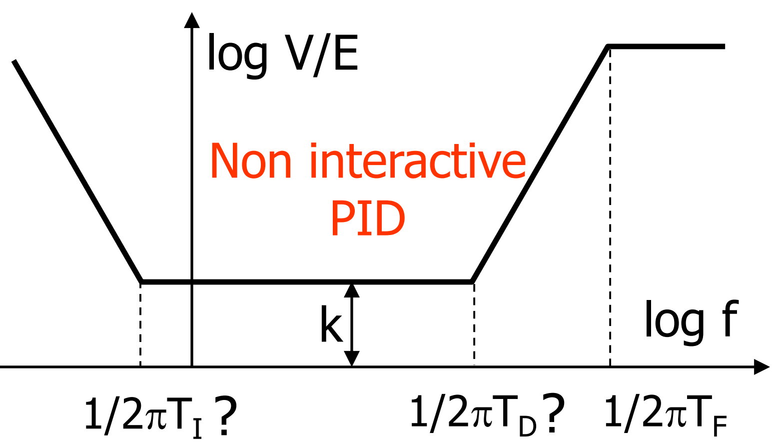
NON INTERACTIVE PID



If a stepwise error E is present (with gain = 1) the control output features a δ of Dirac at the beginning with area kT_D that is proportional to the transient. After this the control behaves as for the PI.

This allows to limit the error due to the quick action of the derivate. Sometimes this can introduce however oscillations.



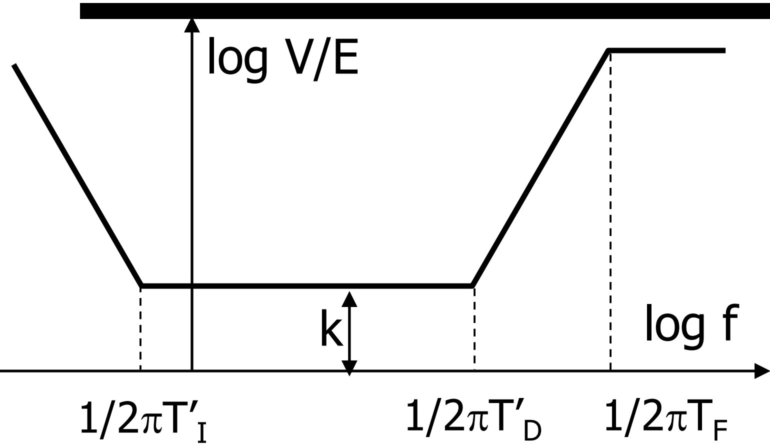


A possible instability (2 poles) can arise $\Rightarrow$ adaptation to the process characteristics in mandatory: not easy since these one change with the time.

$$V = \frac{kE\left[1 + \frac{1}{sT_I} + sT_D\right]}{1 + sT_F}$$

The parameters k, T_I, T_D, T_F must be adapted empirically to the process.

INTERACTIVE PID



The previous expression can be transformed in another one defined «interactive PID» (more comfortable to compensate the process poles).

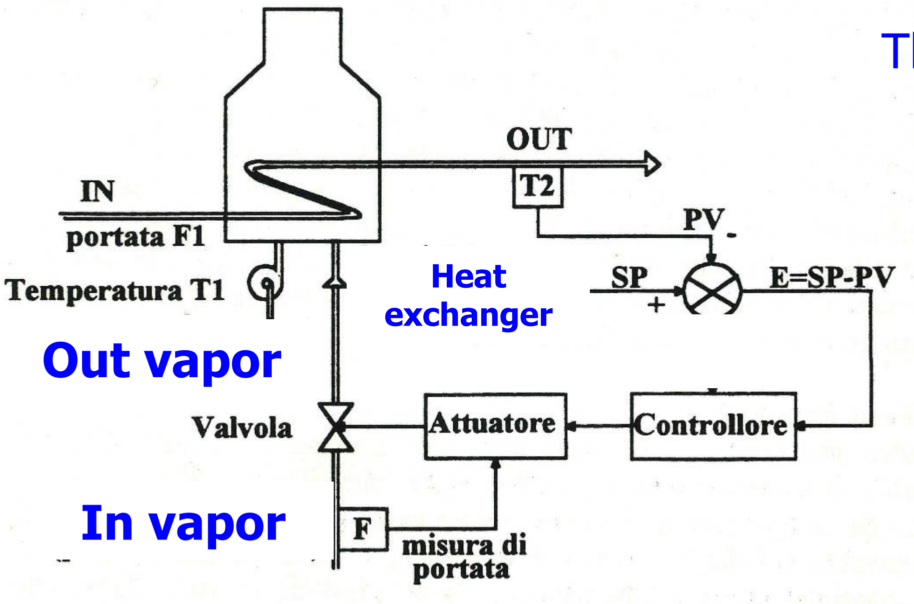
$$V = \frac{k'E\left(1+sT'_I\right)\left(1+sT'_D\right)}{sT'_I\left(1+sT_F\right)}$$

$$V = \frac{E}{1+sT_F}\left[\frac{k'}{sT'_I} + k' + k'\frac{T'_D}{T'_I} + k'sT'_D\right] = \frac{Ek'\left(1+{T'_D}/{T'_I}\right)}{1+sT_F}\left[\frac{1}{sT'_I\left(1+{T'_D}/{T'_I}\right)} + \frac{sT'_D}{\left(1+{T'_D}/{T'_I}\right)} + 1\right]$$

The two expressions are equivalent if we set: $q=1+T'_D/T'_I$, $k=k'q$, $T_I=T'_Iq$, $T_D=T'_D/q$ and when $q=1$ are perfectly equal. However q tends to 1 if $T'_D<<T'_I$ that is very reasonable because the two constants must be located one at high frequencies and the other at lower ones. In such a way we have written a new expression for the PID that highlights explicitly the «zeros» so simplifying the process compensation.

The PID is not convenient when the process features many poles or delays: in these situations a «cascade control» is more suitable.

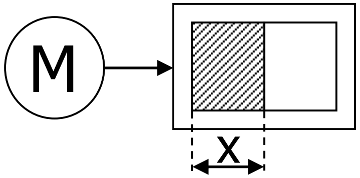
CASCADE CONTROL



The aim is to separate the control of the two time constants

The valve regulating the input vapor flow, is a gate ('slider') that when moving from left to right ('x') leaves a slit through which the vapor passes. The flow is proportional to the width of the slit.

The position control is made through a direct current motor.

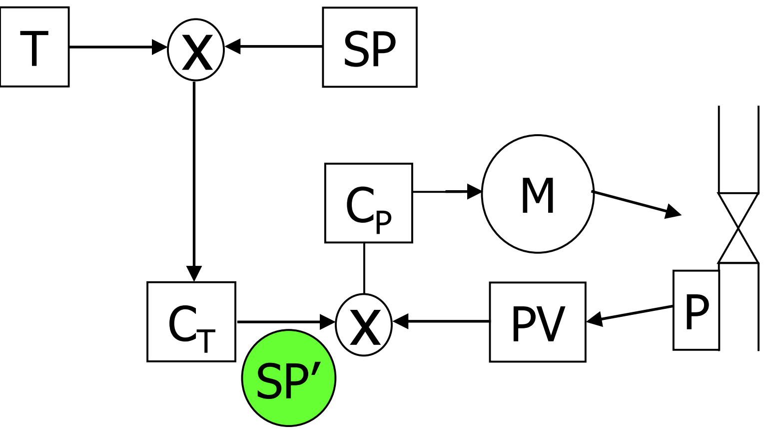


$$x = c\int V dt$$

The relationship between the position and the motor control voltage is integral (the linear valve velocity $\propto$ ω of the motor).

The process includes another pole and delays $\Rightarrow$ not suitable to use aPID.

CASCADE CONTROL



Let's break the control in two parts, by introducing a smaller, internal loop that regulates the valve position comparing it to a suitable set point. (P= position transducer, T=temperature transducer).

The benefit is in the introduction of a closed loop regulator that eliminates or reduces in part the effect of the poles due to the integration and to the motor.

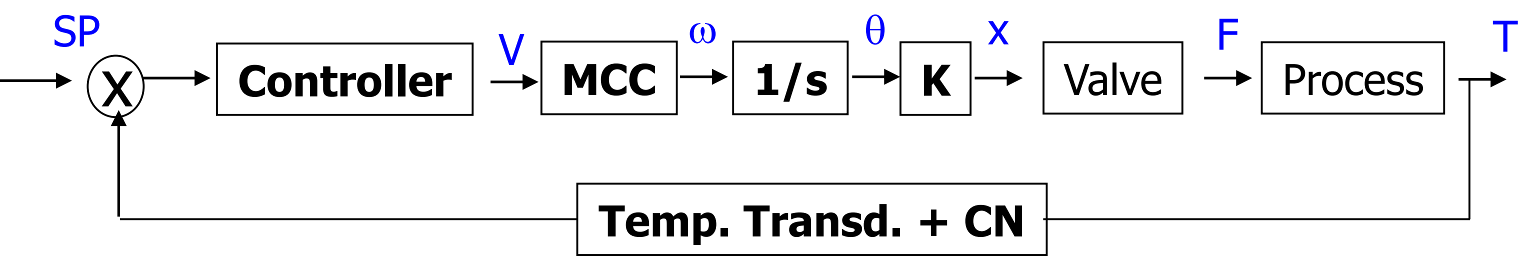
The time constant of the regulator can be as small as we wish so to reach SP immediately.

The velocity control is reduced to a new control without the integration term (one pole less).

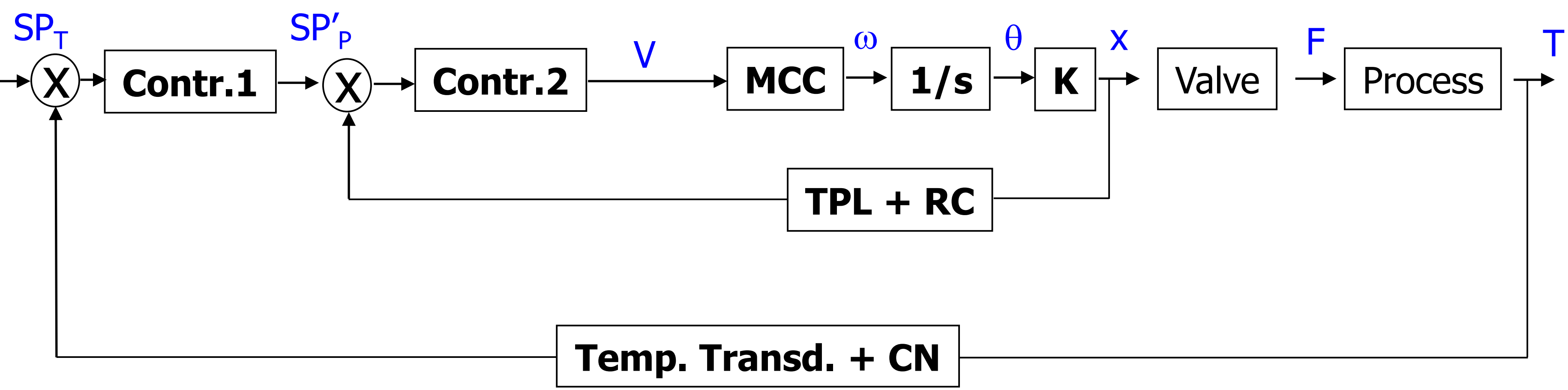
This technique is easy and it is independent on environmental conditions. However it can be applied only to the actuator and to the devices attached to it (in this case motor + valve) not to the process.

CASCADE CONTROL

From a situation like this, with many poles and a potentially unstable loop:



To a situation like this into which the poles due to the motor and to the integration ω-x do not influence the external ring since regulated through the internal control (Contr.2)



FEED-FORWARD

Even when the PV follows well the set-point, few disturbances can arise, not due to the process.

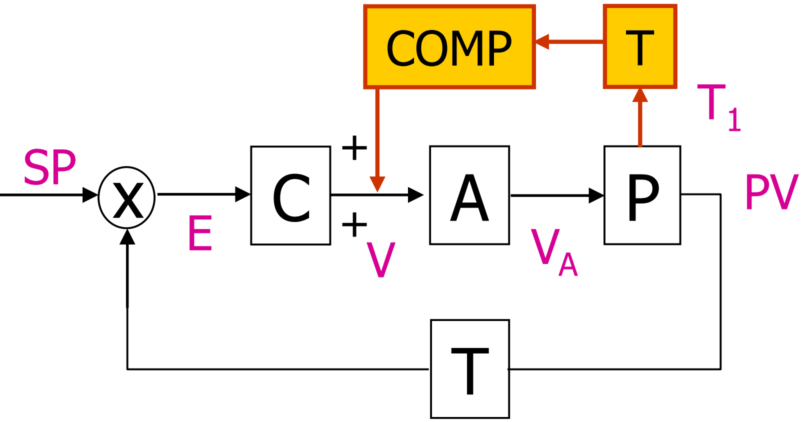
Surely the measured temperature depends on the released vapor amount.

Moreover it depends both on the temperature in the exchanger and on that (T_1) of the vapor (or of the liquid) when entered.

Even the flux control allows a more or less intensive heat exchange in the serpentine.

If these 2 factors change (or even only one), the process will react (slowly) by modifying the temperature, but the controller will act on these variations with a further DELAY (that is only when the *measured* PV deviates from the SP). Not always acceptable.

The effect of these disturbances can be additive or multiplicative or both.



By making suitable hypothesis on the process (how the T_{output} depends on T_1 and the flux) a parallel and complementary path can be can be carried out so as to ‘compensate’ eventual variations due to disturbances.

This path is said *feed-forward chain*.

FEED-FORWARD: variation of the liquid temperature

Generally $PV=\alpha T_1+\beta V_A \quad V_A=f_A[V+FF*T_1]$

Where $f_{A,}$ actuator transfer function, and FF transducer-compensator transfer function

By replacing we have

feed-forward contribution

$$PV= \beta f_A V+[\alpha+FF*\beta f_A]\ T_1$$

To remove the dependency of PV from a disturbance we need to clear [...] → $FF=-\alpha/\beta f_A$

Is FF constant? To answer it is necessary to be sure that T_1 does not feature any phase delay with respect the main controller. This is true in the majority of the cases so it is possible to say that FF is constant.

Conversely FF can depend on the frequency and anticipates or delays its action depending if the two paths are in phase or not. In other words FF can be expressed by

$$FF=\frac{1+s\tau_1}{1+s\tau_2}$$

Where phase shift is in advance or in delay according to $\tau_1>\tau_2$ or not.

In this case the *feed-forward* is called *dynamic*, as opposed with the previous one that is said *static* (FF is constant).

FEED-FORWARD: variation of the vapor flux

Thermal power in the exchanger:

$$W_E + FT_1C_t = FTC_t$$

From which:

$$T = T_1 + W_E / C_t F$$

W_E = thermal power entering (vapor)

F = input liquid flux

C_t = thermal capacity

FT_1C_t = thermal power of the liquid at the input

FTC_t = thermal power of the liquid at the output

We could eliminate the dependency on the flux by adding a term - $W_E / C_t F$ to the controller output: however, since the disturbance depends on the variation of the flux and the input power (multiplicative), this kind of compensation would eliminate also the manipulated variable. It does not make sense.

In case of small variations, however, we can transform the multiplicative disturbance in an additive one so as to apply the same approach as before

$$T = \frac{W_E}{C_T F_{nom}}\left[1 - \frac{F - F_{nom}}{F}\right] + T_1 \approx \frac{W_E}{C_T F_{nom}} + T_1$$

constant

when $F-F_{nom}$ is small

FEED-FORWARD – frequency response

The previous model describes the behavior when the transient expires. The transfer function must, however, take into account those components that imply a different answer when the input solicitations feature a certain frequency. As an example not all the entering heat will be immediately returned: the exchanger will 'store' a certain amount and this means that the temperature variation is caused by W_E - W_n

$$W_E + FT_1C_t = FT_2C_t + W_n$$

T_P=T_2 since the process temperature is relative to the output liquid = Q/MC_T
(Q=heat, M=liquid mass, C_T= thermal capacity

$$T_2 = T_P = \frac{Q}{MC_T} = \frac{\int W_n dt}{MC_T} = \frac{W_E + FC_T(T_1 - T_2)}{sMC_T}$$

$$sMC_T T_2 = W_E + FC_T(T_1 - T_2)$$

$$\boxed{\text{N.B.} \quad \mathbf{Q = \int W_n dt}}$$

$$T_2 = \frac{W_E + FC_T T_1}{sMC_T + FC_T} = \frac{T_1 + \frac{W_E}{FC_T}}{1 + \frac{sM}{F}}$$

When s→0 the already seen result is obtained

FEED-FORWARD – frequency response

The previous expression is like a low pass filter $\frac{\mathbf{k}}{\mathbf{1}+\mathbf{s}\,\tau}$ with $\mathbf{k}=\frac{\mathbf{W_E}}{\mathbf{FC_T}}+\mathbf{T_1}$ e $\tau=\frac{\mathbf{M}}{\mathbf{F}}$

The time constant shows that the greater the input flux the faster will be the answer of the system, and the greater is the liquid mass the longer will be the time required to heat it.

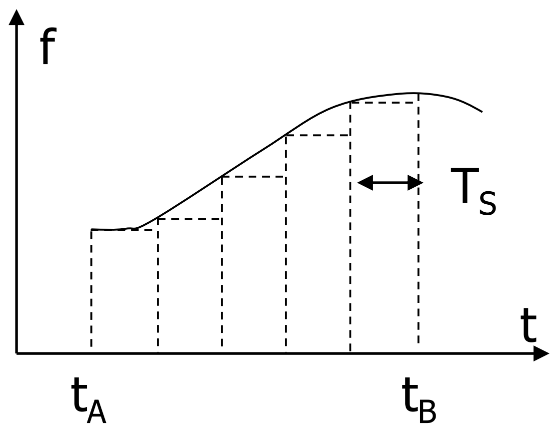
A further model will consist in the introduction of the delays due to the 2 inflow and outflow tubes with a temperature measurement at the input and at the output of the process. Finally a further delay must be considered between the variation of the temperature internally to the heater and the real measurement at the output.

$$T_2=\frac{T_1 e^{-s(\tau_1+\tau_2)}+\frac{W_E}{FC_T}e^{-s\tau_2}}{1+\frac{sM}{F}}$$

NUMERICAL CONTROLLERS

Numerical controllers allow a very accurate signal elaboration (sampled at sufficiently high frequency). They also allow to modify the control parameters on the basis of the monitored process conditions more quickly than with an analog controller.

This requirement is more critical when a complex elaboration must be executed on a variable (like integration or derivation) and in case this affects the process control.



$$I_{ab} = \int_{t_a}^{t_b} f(t)dt \cong \sum_{a}^{b-1} f_i T_S$$

Better approximation if T_S or $T_S * f_{max}$ are small

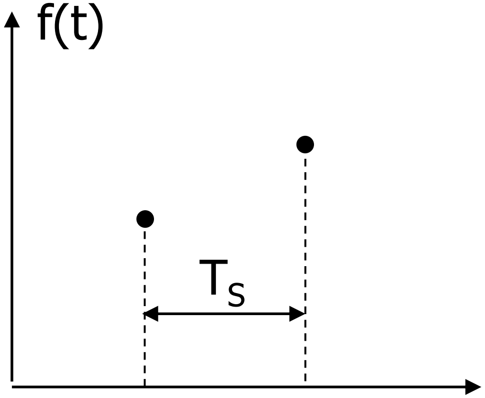
A potentially better approach consists in the interpolation of the curve with straight lines fitting the sampling points:

$$I_{ab}' = \sum_{a}^{b-1} \frac{f_{i+1} + f_i}{2} T_s = \sum_{a}^{b-1} \frac{f_{i+1}}{2} T_s + \sum_{a}^{b-1} \frac{f_i}{2} T_s =$$

$$\sum_{a}^{b-1} \frac{f_i}{2} T_s + \frac{1}{2} f_B T_s - \frac{1}{2} f_A T_s + \sum_{a}^{b-1} \frac{f_i}{2} T_s = I_{ab} + \frac{T_s}{2}\left(f_B - f_A\right)$$

≈ To the previous one if $f_B = f_A$ or T_S small

NUMERICAL DERIVATION



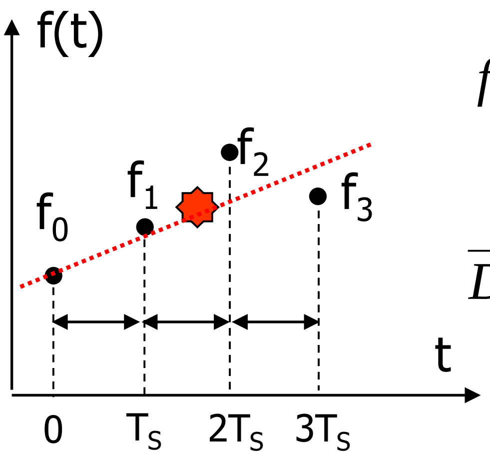
$$f'(t) = \frac{f_{i+1} - f_i}{T_s}$$

$$f'(t) = \frac{f_{i+2} - f_i}{2\,T_s}$$

If T_s is small possible errors due to spikes or outliers values present on the samples can be enhanced.

Sometimes it is better to sample at a reduced frequency to consider samples at higher distances (noise filtering).

More sophisticated techniques can be used, by identifying a *center* of the samples set



$$f_m = \frac{f_0 + f_1 + f_2 + f_3}{4} \qquad t_m = \frac{6T_s}{4} = \frac{3T_s}{2}$$

$$\overline{D} = \frac{1}{4}\left[\frac{f_m - f_0}{\frac{3}{2}T_S} + \frac{f_m - f_1}{\frac{1}{2}T_S} + \frac{f_2 - f_m}{\frac{1}{2}T_S} + \frac{f_3 - f_m}{\frac{3}{2}T_S}\right] = \frac{f_3 + 3f_2 - 3f_1 - f_0}{6T_S}$$

This approach needs more samples and the calculation implies a delay equal to $3/2T_s$ (the derivate is not available every T_s) that limits the response velocity.

DIFFERENTIAL EQUATIONS

1) Every function can be represented as a ratio among polynomials in s without pure delays.

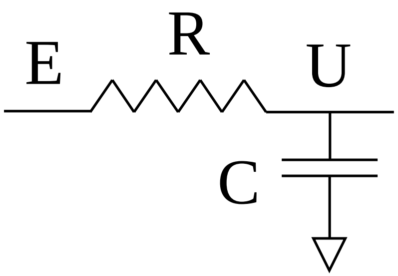
$$\frac{U(s)}{E(s)} = \frac{A(s)}{B(s)}$$

2) Terms like sE_n and like s^2E_n can be replaced with the correspondent incremental ratio.

$$\frac{\dfrac{E_n - E_{n-1}}{T_S} - \dfrac{E_{n-1} - E_{n-2}}{T_S}}{T_S} = \frac{E_n - 2E_{n-1} + E_{n-2}}{T_S^2}$$

3) Derivation turns out in recursive operations on E in the different moments. The equation U/E behaves in a linear relationship among several samples of the input and of the output. At the beginning the values E_n can be set to zero, so the initial values of U will be not the right ones but at the end they will converge.

RC FILTER – NUMERICAL IMPLEMENTATION



$$\frac{U(s)}{E(s)} = \frac{1}{1+s\tau}$$ from which $U + s\tau U = E$

that can be expressed in numerical form as

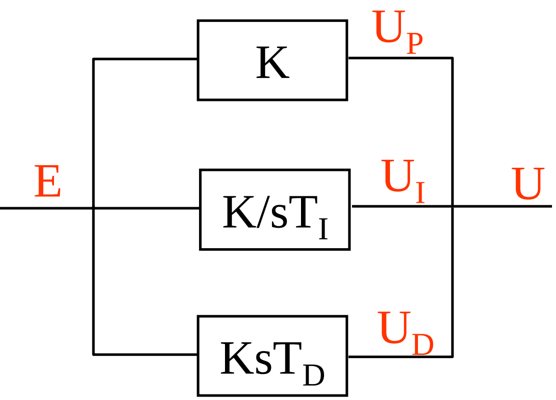
$$U_n + \tau \frac{U_n - U_{n-1}}{T_S} = E_n \qquad U_n = \frac{E_n + U_{n-1}\,\tau/T_S}{1+\tau/T_S} = U_{n-1} + \frac{E_n - U_{n-1}}{1+\tau/T_S}$$

- $K=(1+\tau/T_S)^{-1}<1$ does not cause overflow and can be calculated once and forever
- sometimes instead the sampling frequency can be modified
- a memory buffer is needed to keep K and the previous U_n values

NON INTERACTIVE PID CONTROLLER

$$\frac{U}{E} = k\left[1 + \frac{1}{sT_I} + sT_D\right]$$

Let's neglect the filtering term (T_F). A set of blocks can be carried out, that avoids to calculate second order derivative since every block contains or a pole ora zero.



$$U_{Pn} = KE_n$$

$$sT_I U_{In} = kE_n$$

$$\frac{T_I}{T_S}\left(U_{In} - U_{In-1}\right) = kE_n \quad da\ cui\ U_{In} = U_{In-1} + \frac{kT_S}{T_I}E_n$$

$$U_{Dn} = \frac{kT_D}{T_S}\left(E_n - E_{n-1}\right)$$

The TF filtering block can be introduced successively after the derivator or directly after the U exit.

INTERACTIVE PID CONTROLLER

Let's consider the expression of the interactive PID: $\frac{U}{E} = \frac{k(1+sT_I)(1+sT_D)}{sT_I(1+\gamma sT_D)}$ where $\gamma T_D = T_F$

Goal: to limit the high frequency gain ($1/\gamma$)
Let's implement the PID as a cascade of blocks each with only one zero/pole function:

$$E \rightarrow \boxed{\frac{1+sT_I}{sT_I}} \xrightarrow{F} \boxed{\frac{1+sT_D}{1+s\gamma T_D}} \rightarrow U$$

$$\frac{U}{F} = \frac{1+sT_D}{1+\gamma sT_D} \qquad F + sT_D F = U + sT_D U\gamma$$

In discrete form: $U_n + \gamma T_D \frac{U_n - U_{n-1}}{T_S} = F_n + T_D \frac{F_n - F_{n-1}}{T_S}$

O anche: $U_n = \dfrac{\gamma \frac{T_D}{T_S} U_{n-1} + F_n\left(1+\frac{T_D}{T_S}\right) - F_{n-1}\frac{T_D}{T_S}}{1+\gamma\frac{T_D}{T_S}}$

Can these terms cause a overflow when working with 8/16 bit integer values) ?

INTERACTIVE PID CONTROLLER

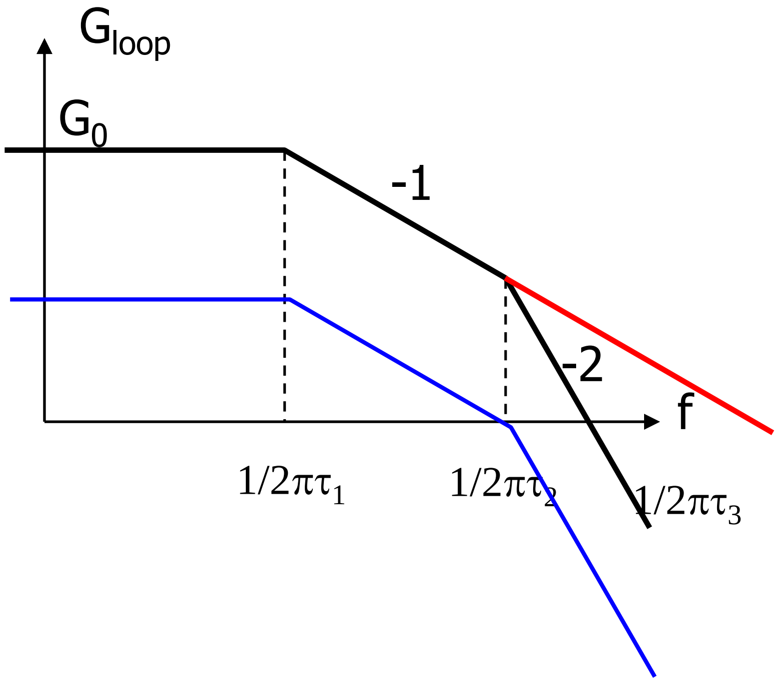
- ($\gamma T_D/T_S$ / ($1+\gamma T_D/T_S$)) = the first coefficient is less than 1. No possible overflow.
- $T_F<T_D \Rightarrow \gamma <1$, then $T_D>\gamma T_D>>T_S$, e ($1+T_D/T_S$)>>1. The II and III coefficients can cause overflow. ($\gamma T_D>>T_S$ due to Nyqvist theorem)
- Same consideration can be made for the coefficient of F_{n-1}

A possible modification that can avoid the overflow consists in writing the equation in a different way:

$$U_n = \frac{\gamma \frac{T_D}{T_S} U_{n-1} + F_n + \left(F_n - F_{n-1}\right)\frac{T_D}{T_S}}{1+\gamma \frac{T_D}{T_S}}$$

In this case the coefficient of (F_n-F_{n-1}) is ~ $1/\gamma >1$. However if F does not change too much (slow dynamics) the difference can be considered small and this term cannot be responsible of an overflow

DYNAMIC COMPENSATION



$$\frac{G_0}{(1+s\tau_1)(1+s\tau_2)}$$

If the system features 2 poles it is potentially unstable. Solutions:

- Reduce the gain to cross 0 db with lower slope
- A numerical block with anticipatory phase. $(1+s\tau_2)/(1+s\tau_3)$ with $\tau_2>\tau_3$.

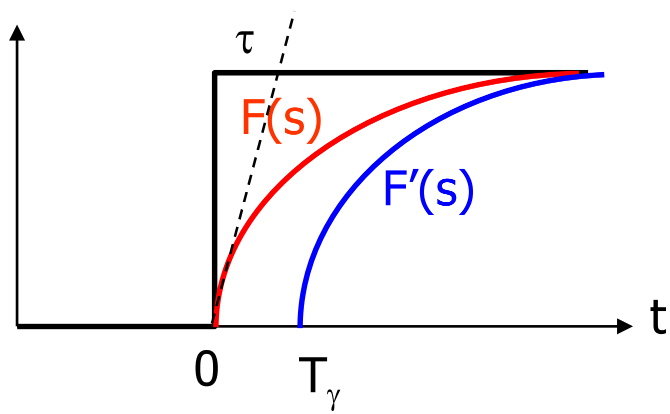
The introduction of a block with a pole-zero transfer function corresponding to the poles of the process is defined as *dynamic compensation* and its realisation in numerical form is similar to what already discussed for the PID controller.

PURE DELAY COMPENSATION

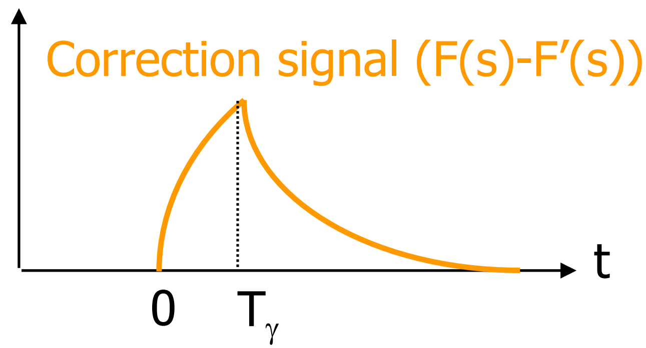
A delay corresponds to a phase shift that can compromise the so called *phase margin* in fact it can cause the crossing of the axis at 0 db with phase shift greater than 180°, compromising the system stability.

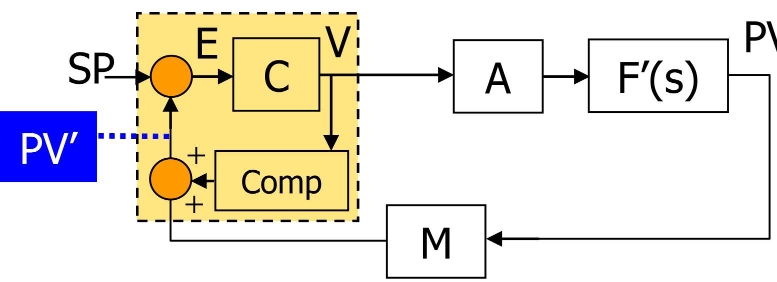
Moreover it introduces an attenuation in the amplitude.

By applying a step-case input for example, the system response can arrive too late.



$$F'(s) = \frac{e^{-sT_\gamma}}{(1+s\tau)}$$

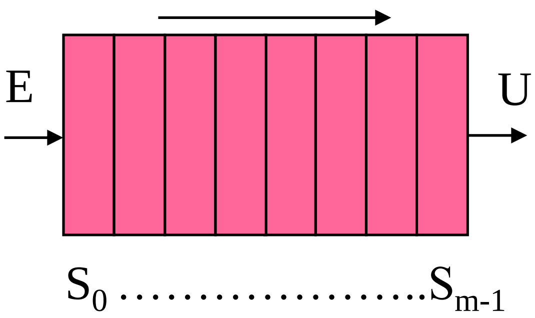




$$PV'(s) = \frac{V}{(1+s\tau)} \Rightarrow Comp. = \frac{1-e^{-sT_\gamma}}{(1+s\tau)}$$

SMITH PREDICTOR

The compensation block can be carried out through a RAM memory buffer like a FIFO, that works as a shift register.



$S_{m-1} \rightarrow U$

$S_{m-2} \rightarrow S_{m-1}$

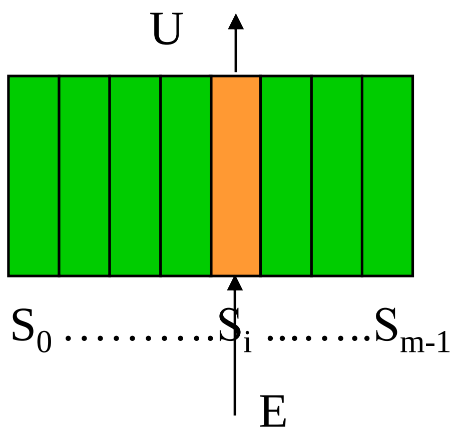
.

.

$S_0 \rightarrow S_1$

$E \rightarrow S_0$

The output at the n_{th} instant is equal to the input at the instant t_{n-m}. The delay is $T=mT_s$. Complex solution if the data number is big.



$S_i \rightarrow U$

$E \rightarrow S_i$

$i+1 \rightarrow i$ per $i+1<m$

$0 \rightarrow i$ per $i+1=m$

Another solution envisions the usage of a mobile pointer that address the i_{th} cell: this implies a very lower number of operations tha before independently on the delay that we want to generate.

In any case if the needed delay is long, a high n° of cells is required.

SMITH PREDICTOR

A cascade of m RC circuits can be used to achieve a total delay equal to the sum of every stage delay.

$$\frac{1}{(1+s\tau_1)}\frac{1}{(1+s\tau_2)}\cdots\frac{1}{(1+s\tau_m)} \rightarrow \mathbf{T} = \sum_{1}^{m}\tau_i$$

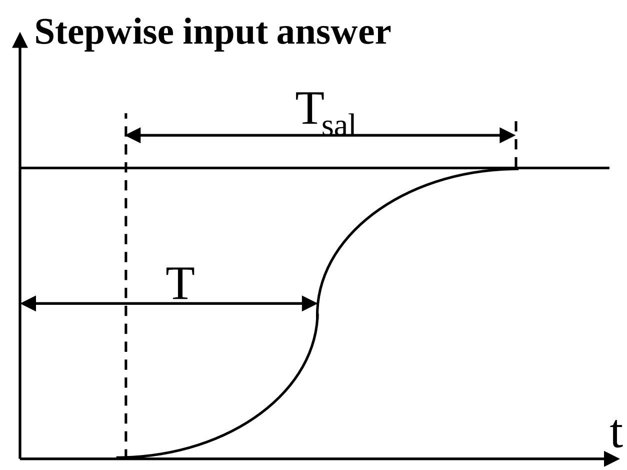
It can be noticed that:

$$e^{-sT} = \frac{1}{e^{sT}} \cong \frac{1}{1+sT+\frac{(sT)^2}{2}} \cong \frac{1}{1+sT+\frac{(sT)^2}{4}} = \frac{1}{\left(1+\frac{sT}{2}\right)^2}$$

This is easy to implement although in a numerical way (with a lower memory usage)

And in general:

$$e^{-sT} = \lim_{m\to\infty}\frac{1}{\left(1+\frac{sT}{m}\right)^m}$$



Assuming that T is the time when the answer overcomes the 50% of the asymptotic value, supposing that all the blocks feature the same delay τ_0, it can be demonstrated

$$T_{sal} = \tau_0\sqrt{m} = \frac{T}{\sqrt{m}}$$

when m $\rightarrow\infty$ it tends to a pure delay