Capítulo 1

Introduction to Process Control

1. Systems Control

The goal is to keep one or more variables of interest in the original system close to reference values. To this end, an additional subsystem, will be connected to the original system.

An automated process is a dynamical system with an industrial purpose. The goal may be:

- To obtain a product with a given temperature
- To obtain a certain flow of a product
- To store a product with a given height
- To generate electrical energy, etc.

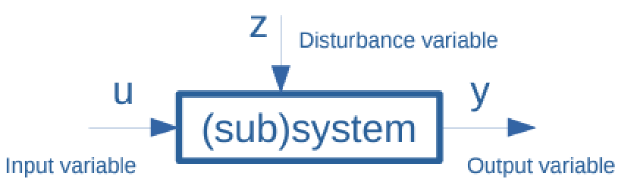
2. Process Control

Question: What is the adequate input value to control the output? I.e. What is the control law?

Answer: It depend on the system behavior

Conclusion: We need a system model to control and analyze the system behavior.

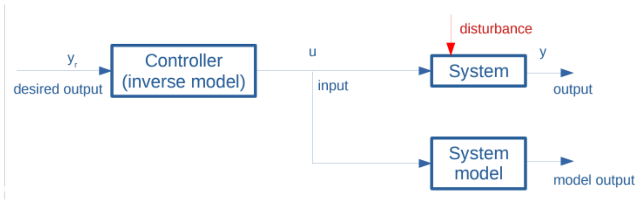
Automated control of systems and processes is usually represented by means of Block Diagrams. Each block represents a process component or subsystem: valves, tanks, reactors, etc. Transfer Functions are used as mathematical models. Each block has its own input and output variables, depicted as arrows.



2.1. Closed Loop Control

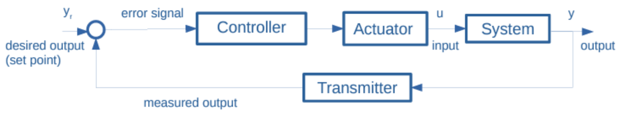
Open Loop Control

- Simple and cheap
- Not useful in the presence of important model errors or distrubances



Closed Loop Control

- Robust in the presence of model errors
- Robust in the presence of distrubances



Main features of Process Control:

- System models are usually non-linear
- Processes are often stable and over-damped
- Delays play a vital role
- Improving the behavior in the presence of distrubances is more important than dealing with set point changes

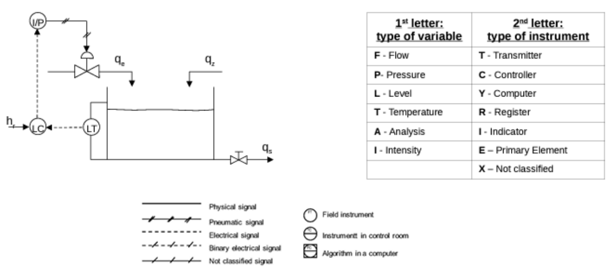
2.2. Process Control Methods

Control systems may present one single loop (Basic Regulatory control or BRC Control) or several closed loops (Cascade-based Advanced Control). Other advanced control schemes that we will see are: Feed Forward Control and the Smith Predictor.

3. P&ID

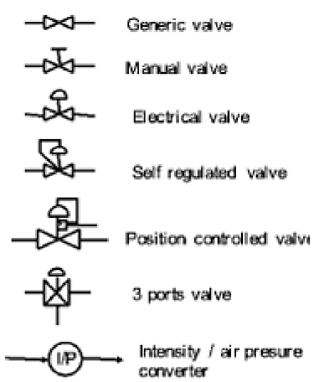
P&ID Representation of Processes:

- P&ID: Piping and Instrimentation Diagram
- International notation, essential to specify and deploy the industrial control systems layout



1st letter: type of variable	2nd letter: type of instrument
F - Flow	**T** - Transmitter
P- Pressure	**C** - Controller
L - Level	**Y** - Computer
T - Temperature	**R** - Register
A - Analysis	**I** - Indicator
I - Intensity	**E** – Primary Element
	X – Not classified

Different types of valves have different P&ID representations


Generic valve
Manual valve
Electrical valve
Self regulated valve
Position controlled valve
3 ports valve
Intensity / air presure converter

Capítulo 2

System Models and Control Requirements

1. Models

A model of a system is a mathematical approximation of its behavior.

An input-output model of a systme is a set of equations that provide a nice estimation of the actual output values of the system within a certain range of input values.

1.1. Operation point

A continuous automated process operates at a specific woring point in order to comply wiht the production goal.

The working point is usually an equilibrium point of the system, which meand that all involved variables are constant over time (i.e. their derivatives are zero).

1.2. Equilibrium point

The variable values at the Equilibrium Point (equilibrium values) are usually denoted with a 0 subscript. SInce they are the final values when all the variables are kept constant, they are also referred to as the values when $t = \infty$.

Some equilibrium values must be known to determine the rest.

Broadly speaking, industrial processes are systems that present a non linear behavior. Even though there are serveral techniques to implement their control systems, this course will ofcus on the methods for linear incremental modeling around the so-called Equilibrium Point (E.P.)

In order to design and implement the control system of an industrial process we will need, first of all, a linear model around a certain equilibrium point.

The model may be obtained by meand of:

- Linearizarion of differential equations

- Identification of actual systme behavior

Linearization of a multi-input dynamical system

Original system equations:

$$f(y, y', y'', ..., u, u', u'', ...) = 0$$

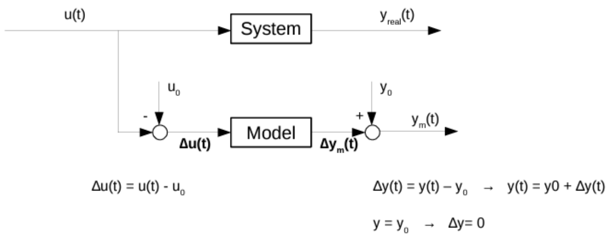
Nominal values:

$$y_0, y_0', y_0'', ..., u_0, u_0', u_0'', ...$$

Linearized model

$$\left.\frac{\partial f}{\partial y}\right|_0 \Delta y + \left.\frac{\partial f}{\partial y'}\right|_0 \Delta y' + \left.\frac{\partial f}{\partial y''}\right|_0 \Delta y'' + ... + \left.\frac{\partial f}{\partial u}\right|_0 \Delta u + \left.\frac{\partial f}{\partial u'}\right|_0 \Delta u' + \left.\frac{\partial f}{\partial u''}\right|_0 \Delta u'' + ... = 0$$

The linear model is an incremental model



The controller, the actuator, the transmitter, etc. are dynamical systems that we well model by means of differential equations.

In order to tune the controllers properly, transfer functions defined in the Laplace domain will be used instead of differential equations.

A transfer function is a linear model around a specific equilibrium point.

The linear approximation quality degrades as the variables get away from the selected equilibrium point.

1.3. Linearization of a tank system

In the following system:

a) Calculate the value for h_0 at the equilibrium point defined by $F_{10} = 1m^3/s$. Using Simulink, verify that this is an equilibrium point ($dh/dt = 0$)

$$dh/dt = 0 \rightarrow F_{i0} - 0{,}3\sqrt{2gh_0} = 0$$

$$h_0 = \frac{F_{i0}^2}{2ga^2}$$

$$F_{i0} = 1 \rightarrow h_0 = 0{,}5669$$

b) Obtain the liner model around this equilibrium point and calculate $H(s)/F_i(s)$

$$A\Delta\dot{h} = \Delta F_i(t) - \left.\frac{a\sqrt{2g}}{2\sqrt{h}}\right|_0 \Delta h(t)$$

$$\Delta\dot{h}(t) + 0{,}882\Delta h(t) = \Delta F_i(t)$$

$$\frac{H(s)}{F_1(s)} = \frac{1}{s + 0{,}882} = \frac{1{,}134}{1 + 1{,}134s}$$

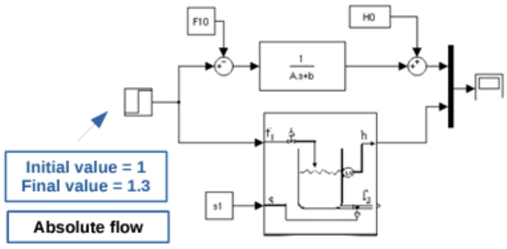
c) Obtain the new steady value h_{0_2}, both using the linear model and the non-linear model, when the flow grows 30 % form the previous equilibrium value ($F_{i0_2} = 1{,}3m^3/s$)

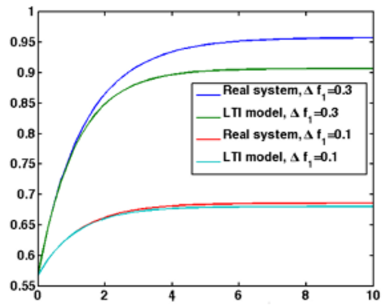
$$\Delta h(\infty) = \frac{\Delta F_i(\infty)}{0{,}0882} = \frac{F_{i02} - F_{i0}}{0{,}0882} = \frac{0{,}3}{0{,}882} = 0{,}34$$

$$h_{linear}(\infty) = h_0 + \Delta h(\infty) = 0{,}5669 + \Delta h(\infty) = 0{,}907$$

$$h_{real}(\infty) = \frac{F_i^2(\infty)}{2ga^2} = \frac{1{,}3^2}{2 \cdot 9{,}8 \cdot 0{,}3^2} = 0{,}958$$

d) Using simulinkg, compare the evolution of $h(t)$ based on the linear model and the non-linear model, when the systems goes drom the equilibrium defined by $F_{i0} = 1$ to the new equilibrium point defined by $F_{i02} = 1{,}3$ (30 % variation) What about 10 % variation?

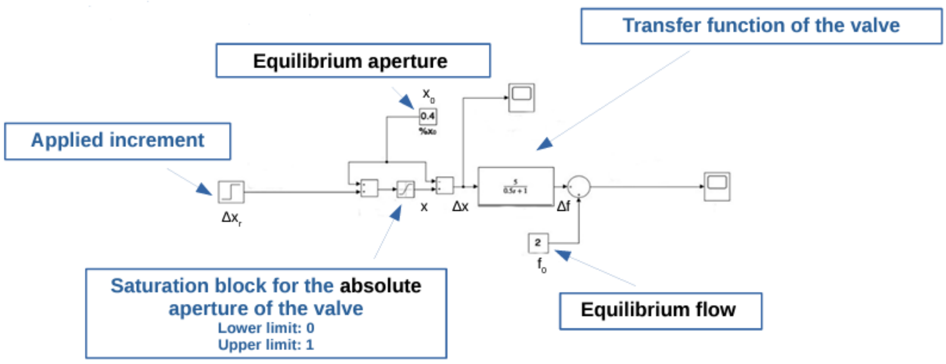




The transfer funciton model is an incremental model. Hence the equilibrium values must always be added, as shown in the previous figures.

The linear model is an approximation of the real system. This is the reason why there is a difference between the exact new equilibrium value and the value porvided by the linear model.

1.4. Modeling a valve



How does a valve work?

- The valve has an aperture range of $[0, 1]$
- The valve is initially opened a given % of the maximum degree of aperture. This is the valve operation point (equilibrium point). E.g. the valve is initially 40 % opened.
- The valve aperture is modified a given % from the operation point (opened or closed). If the valve is opened an additional 10 % then the valve is now 44 % opened $(0{,}4 + 0{,}04 = 0{,}44)$
- The new flow value from the valve is: the flow due to the increment (or reduction) of aperture from the operation point + the equilibrium flow.
- This is only true as long as the valve is not in a saturation state.
- The dynamical model of the valve, the transfer funciton Gv(s), is only used for the incremental opening or closing. The operation point flow f_0 (E.P) is always considered as a constant that must be added to the incremental flow obtained from the model of the valve. This is vary important.

Capítulo 3

Basic Control

1. Introduction

Basic control is a tupe of closed-loop control, with a single control loop. In this section, the transfer functions of the systems to be controlled are of first order or first order + deadtime. We will learn how to design PID controllers for this kind of systems.

The transfer functions that constitute the dynamics of the system to be controlled typically include two types:

Actuator transfer function. This usually corresponds to a flow control valve, steam valve, gas valve, etc.

Transfer functions of the plant (process). This is usually a first order transfer function of a tank, a boiler with heat exchanger, a reactor with heat exchanger, etc.

Controller. The objective of the basic control is to design a PID controller by cancellation of poles and zeros (analythical method) or by empirical methods (tables).

Process: a dynamic system with a particular purpose.

The goal of the control system is to correct deviations from the desired operation point.

2. Controllers Design Methodology

Controllers design is the central aspect of these sections and perhaps the most relevant aspect of the course. For this goal, two methodologies will be studied:

1. Analytical design based on pole-zero cancellation
2. Empirical design based on tables

Depending on the system model we will select the corresponding option.

2.1. Processes without delay

In this case, analytical methods are applied. The main options are root locus analysis, the pole-zero cancellation method or the Ziegler Nichols (Z-N) tables for closed loop processes.

It is not possible to use other tables such as Open-loop Z-N or the AMIGO method.

Transfer function for the design of process controllers without delay:

$$\frac{K_p}{1 + t_p s}$$

Desging methods:

- Closed-loop Z-N methods (applicable with or without delay)
- Analytical pole-zero cancellation design method (also known as direct synthesis).

2.2. Processes with a delay

These are controllers based on Ziegler Nichols (Z-N) tables and the AMIGO mehtod.

The design using Z-N tables or the AMIGO method, is fundamentally based on obtaining a first order function with a delay (First Order Plus Dead Time) of the process.

FOPDT model of the process:

$$\frac{K_p}{1+t_p s}e^{-t_m s}$$

Where:

- K_p: static gain
- t_p: time constant (or period)
- t_m: delay

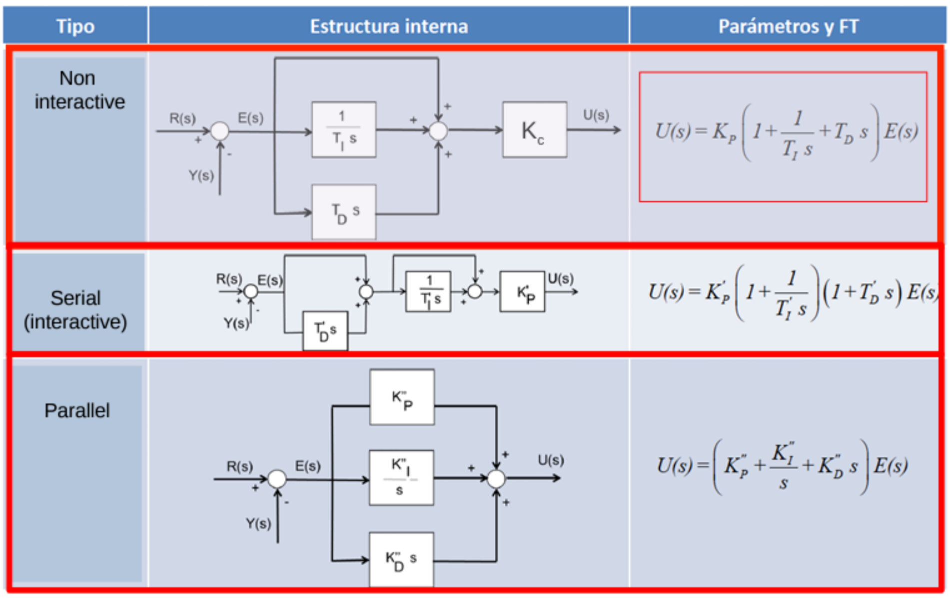
A FOPDT model is representative of many types of processes.

Design methods:

- Closed-loop Z-N method (applicable with or without process delay)
- Open-loop Z-N method (applicable with process delay)
- AMIGO method (applicable with process delay)

3. Basic concepts of PID controller design

PID (Proportional Integral Derivative) controllers are undoubtedly the most widely used in the industry. A PID controller is the result of combining three control actions: proportional action + derivative action + integral action.



3.1. Proportional control: P

Proportional control K_p makes the response speed faster, which is desirable, but it increases the setling time.

A strong proportional control action increases the oscilation and inestability of the system. For safety reasons, overshooting should not exceed 30 %.

The proportional part does not consider time, therefore, the best way to solve the permanent error and consider the variation with respect to time, is to include and set the integral and derivative actions.

3.2. PD Controller

PD control with K_p and T_d gains reduces the response speed and the setling time.

The T_d parameter increases the stability of the system, reducing the oscillations caused by the proportional component K_p.

The derivative action manifests itself when there is a change in the absolute value of the error.

The function of the derivative action is to correct the error proportionally with the same speed as it is produced, this way it prevents the error from increasing.

3.3. PI Controller

PI control with K_p and T_i gains increases the response speed and setling time.

The integral control mode is intended to decrease and eliminate the steady state error, caused by external disturbances and which cannot be corrected by proportional control.

The integral control acts when ther is a deviation between the variable and the set point, integrating this deviation in time and adding it ot the proportional action.

The integral control is used to avoid the inconvenience of an offset (permanent deviation of the variable with respect to the set point).

3.4. PID Controller

In general, when there is no knowledge of the process, the PID controller is the most controller. By properly is the most appropriate controller. By properly adjusting the three variables (gains) in the PID control algorithm, the controller can provide control actions tailored to the requirements of the specific process.

The response of the controller can be described in terms of control response to an error, the degree to which the controller overshoots the set point, and the degree of system oscillation. Note that the use of PID control does not guarantee optimal system control or system stability.

PI controllers are particularly common, since the derivative actions is very sensitive to noise.

While PID controllers are applicable to most control problems, they can perform poorly in some particular applications. This is why specific adaptations and other advanced methods were proporsed.

Tuning a control loop meand adjusting the control system parameters to the optimum values for the desired control system response.

The optimal behavior to a process change or a setpoint change varies depending on the application.

The most effective method generally requires developing some form of the process model, then choosing P, I and D based on the dynamic model parameters. Manual tuning methods can be very inefficient. The choice of a method will depend on whether the loop can be “switched off” to adjust it, and on the response time of the system.

4. Basic Control

4.1. 1st PID design method: Pole-zero cancellation controller

Transfer function of the processes without delay:

$$\frac{K_p}{1+t_p s}$$

For clarity purposes, we will denote the transfer function of the process (tank, boiler, heater, etc.) as $G_p(s)$.

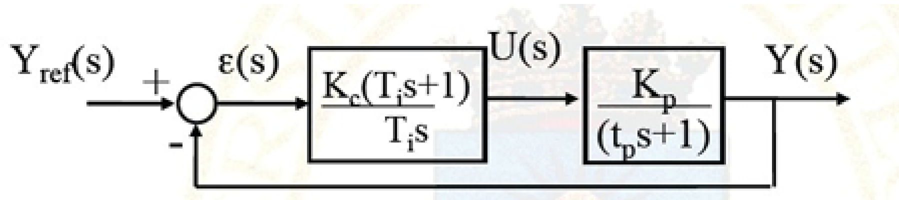
The design of a controller for a transfer function of a process without delay follows the analytical design method of pole-zero cancellation.

The controller function for a first order system with no delay is of type PI (Proportional and Integral) and hence it has the following structure:

$$G_c(s) = K_c\left(1+\frac{1}{T_i s}\right) = \frac{K_c(T_i s+1)}{T_i s}$$

Where:

- K_c is the proportional gain
- T_i is the integral constant



The goal is to determine the controller parameters, T_i and K_c.

Design criterion.

The design criterion is based on the closed-loop transfer function of the system, $M(s)$.

1. If we set T_i equal to the time constant of the process, $T_i = t_p$, then the closed loop controlled system behaves as a 1st order system:

$$M(s) = \frac{Y(s)}{Y_{ref}(s)} = \frac{G_c G_p}{1+G_c G_p} = \frac{K_c K_p}{t_p + s + K_c K_p} = \frac{1}{1+\frac{t_p}{K_c K_p}s} = \frac{1}{1+t_c s}$$

2. The closed-loop time constant, t_c is chosen by the designer.
3. The controller gain is obtained from the closed-loop equation $M(s)$

$$t_c = \frac{t_p}{K_c K_p} \rightarrow K_c = \frac{t_p}{K_p t_c}$$

4.2. 2nd Controller Design Method: Processes with a delay

Transfer function of the processes with delay:

$$\frac{K_p}{1+t_p s}e^{-t_m s}$$

For clarity purposes, we will denote the transfer function of the process (tank, boiler, heater, etc.) as $G_p(s)$.

The design of controllers for processes with a deadtime is empirical and its based on tables, for example the Ziegler-Nichols (Z-N) tables or the AMIGO tables.

Open-loop Z-N table (rules) based method

The open-loop Z-N rules, published in 1942, were obtained by experimenting with a large number of simulated system on an analog simulator from the Taylor Instrument company, and on the Differential Analyzer, an MIT's analog computer.

The authors studied P, PI and PID controllers. To understand the process the followed, we will focus on the PI controller. For each of the simulated systems, the authors obtained two sets of data:

- Descriptive data on the dynamic behavior of the simulated process ($G_p(s)$ parameters)
- PI regulator parameters (K_c and T_i), obtained by trial and error, which gave optimum loop performance.

1/4 error signal decay rate in response to a step in the distrubance variable.

They then correlated the data from the first group with those from the second group for the different systems simulated, to obtain laws or rules that would make it possible to determine, from the data describing the dynamic behavior of a process, the parameters of the PI controller that gave optimum operation of the loop. The same was done for P and PID type controllers.

This was done for two different ways of describing the dynamic behavior of a simulated process: the step response of the open chain process and the response of the closed loop with a P regulator and a controller gain that makes the loop critically unstable.

1. Firstly, an FOPDT Model must be obtained by applying the previously presented methods.

$$G_p(s) = \frac{K_p}{1 + t_p s} e^{-T_t s}$$

2. The followin parameters are extracted from the FOPDT model of the system to be controlled:
   - K_p: Static gain
   - T_t: Delay Time
   - T_p: Time constant
3. With the K_p, T_p and T_t parameters, go to the Z-N table and calculate the parameters for the controller type of your choice: P, PI or PID.

Controller	K_c Gain	T_i Integral time	T_d Derivative time
P	$\frac{1}{K_p}\frac{t_p}{T_t}$		
PI	$\frac{0{,}9}{K_p}\frac{t_p}{T_t}$	$3{,}33\ T_t$	
PID	$\frac{1{,}2}{K_p}\frac{t_p}{T_t}$	$2\ T_t$	$0{,}5\ T_t$

The operation range for Z-N Open Loop tables:

$$0{,}1 \leq r = \frac{T_t}{t_p} \leq 0{,}9$$

Two methods to obtain the FOPDT function

1. FOPDT obtained by linearization of the system equations (Analytical function method) + multiplication by the existing delays. This only applies if the system + actuator system if a 1st order system.
2. FOPDT obtained by the graphical method of the tangent and 63 %, based on an experimental curve resulting from a 10 % increment in the considered input signal.

The MIGO method

MIGO stands for "M-constrained Integral Gain Optimization". This method comes from the Åström group (Lundt University, Sweden) and was published in the last years of the 20th century.

In this method, the PID controller parameters are calculated so that the integral gain of the PID controller is maximized, while keeping the robustness of the controller within given levels.

Maximize K_c/T_i while keeping the relative stability above a threshold, in response to a step in the distrubance variable.

The AMIGO method

Optimization of the MIGO method is performed by individually for each process, based on its tf, which is laborious in a plant where there may be hundreds of control loops. To facilitate the tuning of conttollers, the AMIGO rules were developed from the MIGO method.

AMIGO is the acronym for Approximate MIGO. It is a set of rules for PI and PID controllers, dating from 2002 and 2004; they are dur to Åström and Hägglund.

These rules follow the philosophy of the Ziegler-Nichols rules. First a large set of tfs, representative of those appearing in process control (9 different types of tfs, which when parametrerized give a total of 134 tfs), was developed.

For each of the aforementioned tfs, the authors obtained:

- descriptive data on the dynamic behavior of the tfs.
- parameters of the optimum regulator according to the MIGO method

They the correlated the data of the first group with those of the second group for the different systems simulated, in order to try to obtain laws or rules that would make it possible to determine from the descriptive data of the dynamic behavior of a process, the parameters of the regulator that gave an optimal operation of the loop.

controller	K_c	T_i	T_d
PI	$\frac{1}{K_p}\left(0.15 + \frac{0.35}{r} - \frac{1}{(1+r)^2}\right)$	$T_t\left(0.35 + \frac{13}{1+12r+7r^2}\right)$	

$$r = \frac{T_t}{t_p}$$

Where K_p, T_t, and t_p are the parameters of the FOPDT model of the process, generally obtained by the tangent and the 63 % method.

The approximation between the results obtained with the MIGO method and those obtained with these AMIGO rules is very good.

controller	K_c	T_i	T_d
PID	$\frac{1}{K_p}\left(0.2 + \frac{0.45}{r}\right)$	$T_t\frac{8+4r}{1+10r}$	$T_t\frac{5}{10+3r}$

The approximation between the results obtained with the MIGO method and those obtained with the AMIGO rules for K_c and T_i is very good for $r \geq 0{,}25$ and conservative for $r < 0{,}25$. "Conservative" means that the error separates us from the optimal solution given by MIGO but still keeping the loop within the recommended values of robustness.

The approximation for T_d is very good for $r \geq 1$ and conservative for $r < 1$.

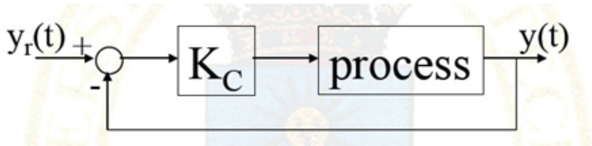
Operation range for AMIGO tables:

$$0{,}25 \leq r = \frac{T_t}{t_p} < 3$$

The closed-loop ZN method

The fourth method of controller design is based on the closed-loop ZN rules.

In this case the controller design strategy is to close the loop of a process using a proportional controller with gain K_c, as illustrated in the figure below.



The design method consists of the following steps:

1. Vary the gain K_c of the closed-loop system until the output oscilates in a sustained manner, as shown in the attached figure. We will denote this gain value the ultimate gain $K_u = K_c$.
2. Based on the oscillatory output curve we measure on the curve the perior T_u, which we will call, the ultimate period.
3. Determine the parameters of the PID controller from the table.

5. Controller design improvements

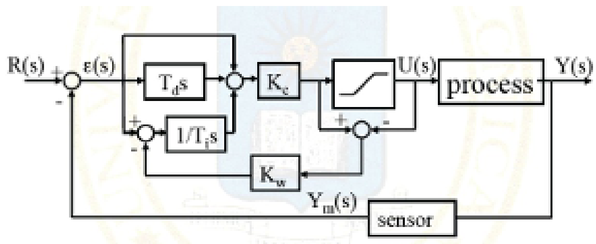
5.1. Anti-windup filter

As a result of the actuators saturation, the steady state error remains and the integral control output may accumulate very high values. In our tank systems, the elements that may present a saturation are the valves.

The adverse effect of the integral component generally occurs when the error is very high and is maintained for a long time. In this case the system is saturated and the integral control cannot perform its function.

In these cases it is recommended to disable or attenuate the intergal control by means of a filter so that excessive overshoot does not occur.

There are serveral way to implement the anti wind-up control, here we will study the one illustrated in the figures below.



Capítulo 4

Advanced Control

1. Introduction

Cascade control is a type of closed-loop control with two or more control loops.

The actuator + the process can usually be modeled as different sub-systems connected to each other, so that there is one or more intermediate variables available for measurement.

Concept: controlling intermediate variables before their alterations by disturbances affect the output.

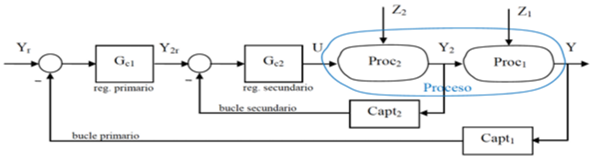
Result: cascade control minimizes the effect of disturbances in the inner loop (not so good for distrubances in the outer loop or for set point variations)

The objective of cascade control is to improve the dynamic response of a process by designing internal control loops according to identificable dynamics of sub-systems.

The tuning methods for the controllers are the ones studied in the basic control section.

2. Cascade Control Design Methodology

As mentioned in the intoruction, for cascade control, in addition to the controlled variable, one or more intermediate variables are fed back, giving rise to the appearance of nested loops in which the reference of an internal controller comes from the output of the controller immediately outside. The following figure shows the block diagram of a two-level cascade control, in which variable y_2 is fed back, in addition to the output.



Cascade control should only be designed if BRC does not provide satisfactory results. The key step is selecting the secondary variable to feed back.

- It should be a variable that can be measured.
- It must indicate the occurence of an important distrubance
- There must be a causal relationship between the manipulated and secondary variables.
- The secondary variable dynamics must be faster than the primary variable dynamics.

1. Identify the FOPDT functions of each subsystem by:

- Applying the studied methods, e.g. identification by the 63 % - tangent
- If applicable, applying the Fast-Slow method reduction.

2. After determination of the FOPDT functions of the processes, design the PID/PI controllers.
   a) Tune the inner controller, based on the model of this part of the process. A PI controller is usually enough.
   b) Tune the outer controller, based on the model of the inner loop and the part of the process outside it. The inner loop model has no steady error and is usually very fast compared to the other part of the model.
3. Compare the results with a basic control and decide the most suitable control strategy.

3. Feed forward control

The purpose of feed-forward control is to cancel or at least reduce the effect of external disturbances on the system. In general it is not possible to completely cancel a disturbance, but when this perturbation occurs, at least its static component can usually be eliminated.

The objective of $G_{CA}(s)$ controller design is to eliminate the effect of distrubance $G_D(s)$.

$$\frac{Y(s)}{D(s)} = 0 \rightarrow G_{CA}(s) = -\frac{G_D(s)}{G_P(s)G_S(s)}$$

The theoretical t.f. that fully eliminates the disturbance effect is not always feasible.

1. t_m in $G_{CA}(s)$ is not feasible when $t_{md} < (t_{mp} + t_{ms})$
2. When the number of zeros($G_{CA}(s)$) ¿number of poles($G_{CA}(s)$)

4. Large delay systems: Smith's Predictor

In previous sections we have studied the $rm = \frac{t_m}{t_p}$ coefficient, which defines the application ranges for ZN open-loop controllers $0{,}1 < r < 0{,}9$ and for the AMIGO controllers $0{,}25 < r < 3{,}0$.

Regarding the upper limit for the AMIGO controller, the upper limits can be defined approximately as a function of the relative delay τ:

$$\tau = \frac{T_t}{T_t + t_p}$$

For PID controllers: For values higher thatn 0.5 of the relative delay τ, the PID control is comparable to PI. Hence, for slightly higher values, $\tau > 0{,}5$, a PI controller may be sufficient, although the designer should verify this properly, through simulations and select the best solution, PID or PI.

In any case, the upper bound is not easy to define and there are numerous research studying variations on PID controllers and the use of advanced techniques such as generic algorithms, etc.

Based on the above, if the controller design is in the upper limits (in this course we accept up to $r < 3{,}0$), we must simulate the behavior of the AMIGO controller vs. a controller for larga delays such as the Smith's Predictos and select the best option.

4.1. Preliminaries

Currently, there is a growing interest in time-delay plant control strategies, because time delay constitutes a very common phenomenon in the dynamic behavior of practically all industrial plants, process units, ecological, agricultural, biotechnological systems, etc.

Time delay is a phenomenon that originates due to the temporal displacement between two or more variables of a plant which can be generated, for example, by the time needed to transpor mass, energy or information. The time delay represents a significant limitation in control systems and must be considered, both in the analysis stages, as in the design controllers.

The design of appropriate controllers for plants with dominant time delay requires a grat effort, because if the disturbances are not detected in time, the control action, which depends on the timely measurement, does not occur at the right time, and takes time to take effect on the dynamic behavior of the plant. A delayed response of the plant to the control signal can cause a reaction of the controller that does not correspond to the required one, which can lead to loss of stability of the control system.

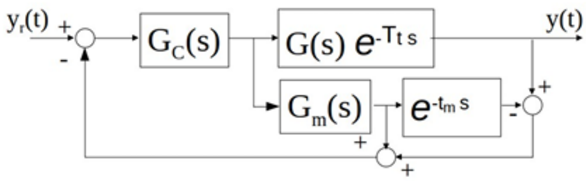
4.2. Smith's Predictos (SP)

The Smith's Predictor (SP) was proposed in 1957 by the American Otto Smith. It is undoubtedly the most widely used dead time compensator in the control of the time-delayed plants due to its high effectiveness and simple implementation.

This SP structure cannot be used in plants with integrators or in unstable plants, and it may also present a bad behavior for disturbance rejection.

Original idea: "Take" the delay out of the plant and design the controller for an internal $G_m(s)$ model with "perfect" prediction.

Smith Predictor: add the actual error compared to the predicted output.



Original idea: "Take" the delay out of the plant and design the controller for an internal model $G_m(s)$ with "perfect" prediction. Error feedback corrects the imperfections of the fast model $G_m(s)e^{\tau s}$