

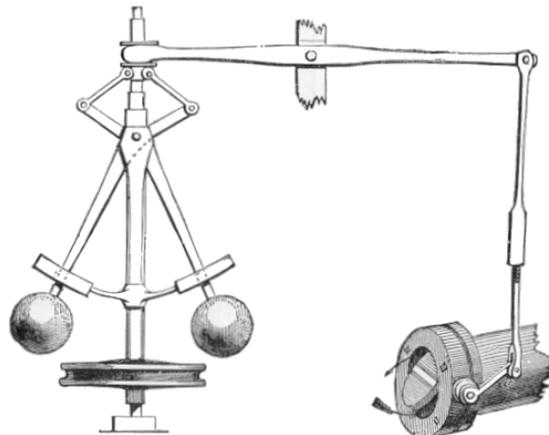


ELECTRICAL ENGINEERING AND INDUSTRIAL IT

Semester 3

AUTOMATIC CONTROL ENGINEERING

Practical Assignments



1st Series

2020 – 2021

First series and planning 2020 - 2021

A1- Modeling and identification for the control of an industrial climatic chamber.

A2- Modeling and identification of a hydraulic process : Matlab version

A3.1- Feedback-based closed-loop control of position, testing and analysis: Part 1.

A3.2- Feedback-based closed-loop control of position, testing and analysis: Part 2.

A5.1- CAO MATLAB, simulation of dynamical systems: Part 1.

A5.2- CAO MATLAB, simulation of dynamical systems: Part 2.

A1

MODELING AND IDENTIFICATION FOR THE CONTROL OF AN INDUSTRIAL CLIMATIC CHAMBER.

I. AIM OF THE OPERATION

The aim is to study an industrial system in order to determine the adjustment parameters for the controller during a subsequent Practical Assignment (PA). After a theoretical analysis based on general physics equations, an analysis of the behavior of the system in response to the stresses imposed can be used to determine the parameters of a simple but adequate model.

Important: when you arrive, open the chamber door, without turning on the power, in order to make sure that the internal temperature is equal to the ambient temperature when the measurement process begins.

II. DESCRIPTION OF THE EQUIPMENT

A DV 30 CLIMATIC CHAMBER manufactured by Sécasi Technologies (figure 1) is used. It consists of an externally insulated chamber in which heating or refrigeration is produced by two systems.



Figure 1 – Photograph of the climatic chamber

This type of chamber is, in particular, used for testing the operation of electronic components and systems under extreme temperatures. Therefore, it is important to monitor the temperatures inside the chamber and their variations precisely in order to avoid exceeding the envisaged temperatures.

It has a capacity of 30 liters and the temperature may vary between $-40\text{ }^{\circ}\text{C}$ and $+150\text{ }^{\circ}\text{C}$; it is controlled by an industrial controller and the following components are mounted above the chamber door:

- on/off switch for the chamber
- on/off switch for the refrigeration unit
- EUROTHERM industrial controller
- manual override button for heating
- manual override button for refrigeration
- DIN socket (25-pin) for digital communication
- BNC socket delivering a signal (0 / 10 volts) reflecting the internal temperature
- mains power plug
- thermostat for temperature limitation

The data are transmitted to a computer via an RS 232 digital link and formatted by a program written in LABVIEW.

III – MODELING OF THE THERMAL CHAMBER

The following notations are used:

$u(t)$: power amplifier input control voltage, which supplies power to an electrical element for heating or to a refrigeration unit for cooling.

$P_{Act}(t)$: active thermal power (heating or refrigeration).

$\theta_a(t)$: ambient temperature.

$\theta(t)$: temperature inside the chamber.

$\theta_m(t)$: measurement of internal temperature given by a PT 100 probe placed inside the chamber.

Changes in the internal temperature $\theta(t)$ depend on the ambient temperature $\theta_a(t)$ and the thermal power $P_{Act}(t)$: positive for heating and negative for cooling. The power leakage ($P_{leakage}(t)$) represents the heat exchanges between the inside of the chamber and the PA room. Finally, the measured temperature $\theta_m(t)$ is a delayed reflection of the internal temperature.

Exercise 2.1 in the first series of Tutoed Assignments (TA) allowed us to establish that the dynamic behavior can be represented by the block diagram in figure 3:

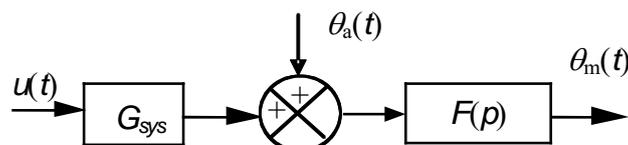


Figure 2 - Block diagram of the climatic chamber

and the transfer functions:

$$F(p) = \frac{\theta_m(p)}{\theta_a(p)} = \frac{e^{-Tp}}{1 + \tau p} \quad G(p) = \frac{\theta_m(p)}{U(p)} = G_{\text{sys}} F(p)$$

IV. EUROTHERM CONTROLLER

A 902-type board controller manufactured by Eurotherm regulates the temperature inside the climatic chamber.

1. Definition

The function of the PID-type controller is to calculate a control signal $u(t)$ based on the error $\varepsilon(t)$ between the setpoint $\theta_c(t)$ and the measurement $\theta_m(t)$.

$u(t)$ is calculated on the basis of the following relationship:

$$u(t) = K \left[\varepsilon(t) + \frac{1}{t_i} \int \varepsilon(t) dt + t_d \frac{d\varepsilon(t)}{dt} \right] \quad \varepsilon(t) = \theta_c(t) - \theta_m(t)$$

For the remainder of the PA, it shall be stipulated that the terms Integral and Derivative are null and void, therefore giving:

$$u(t) = K\varepsilon(t)$$

2. Description of the controller

As illustrated by figure 3, the front face of the controller comprises a keypad and a display.



Figure 3 – Photograph of the controller

a) Keypad

There are six keys, which perform the following functions, from left to right:

- Select (or fast scroll) \diamond
- Run
- Automatic / Manual
- Decrement ∇
- Increment \triangle
- Scroll \curvearrowright

b) Display

When powered on, the controller displays its model number – 902S – for a brief moment on the central part of the display (large-format 7-bar characters), followed by the temperature measured $\theta_m(t)$ inside the chamber in °C.

The letter (W) appears on the bottom left-hand side of the display, followed by the temperature setting $\theta_c(t)$ in °C (small-format 7-bar characters), which is adjusted via the increment, decrement and fast scrolling keys.

The Scroll key displays the letter (Y) which indicates the output power between + and - 100%.

Press the Scan key to return to the setpoint.

3. Programming the EUROTHERM industrial controller

a) Accessing the lists

Scroll down the main menu by pressing and holding the Scroll key for 1 s. \curvearrowright
Choose the rEGL (OPERATOR) menu by repeatedly pressing the Scroll key

Never access the other menus

List of settings available to the operator:

- P_b Proportional band expressed as a %
- t_i Integral action time expressed in seconds
- rES Manual reset (**do not change**)
- t_i Derivative action time expressed in seconds

None of the other settings should be modified, under any circumstances

cbL	Holdback (upper threshold)	[0.0]
cbL	Holdback (lower threshold)	[NO]
HL	Output limitation as a %	[100%]
HC	Output 1 cycle time in s	[10 s]
Cr	Relative gain of output 2, P_b (output 2) = P_b / Cr	[1]
CL	Output limitation as a %	[-100%]
Cc	Output 2 cycle time in s	[10 s]
db	Deadband output 1 - output 2 (% P_b)	[0.0]

b) Definition

The proportional band P_b is the variation, as a percentage, of the controller input required to obtain an output variation of 100%.

The relationship between the gain K and the proportional output P_b expressed as a % is:

$$K = 100 / P_b$$

The deadband db is the distance separating the two proportional bands (inverse output hot and direct output cold); it may vary between -5 and +5% of the value of P_b .

c) Programming the parameters

Choose the parameter to be modified from the rEGL (OPERATOR) menu
Press the key (decrement or increment) to modify a parameter; press both keys simultaneously to change the parameters quickly (fast scroll and decrement or increment).

Press the Scroll key to return to the permanent display.
Press the Select key to return to the main menu.

NB. The display duration is 15 s; the main menu will then reappear if no keys are pressed.

V. OPERATION

Before powering on the unit, make sure that the compartment has been left open for sufficient time to allow the internal and external temperatures to equalize.

- Turn on the computer, run the LABVIEW software and open the enceinte_climatique.vi file (a description of which is given in the annex), which you will find on the desktop.
- Activate the POWER and COOLING switches.
- Record the temperature measurement shown on the display: this value will act as the ambient temperature measurement throughout the entire operation.
- Close the climatic chamber door.
- Set the P_b parameter of the controller to 4 (which corresponds to $K = 25$), and display NO for the parameters t_i and t_d (by setting these values at 0).
- Check directly on the controller that the parameters have been programmed correctly.
- Apply a step function of 90°C to the controller setting.

The controlled system (climatic chamber and controller) is represented by the block diagram in figure 4, which shows:

- the ambient temperature $\theta_a(t)$ as the disturbance input
- the measured temperature $\theta_m(t)$ as the output
- the temperature setting $\theta_c(t)$.

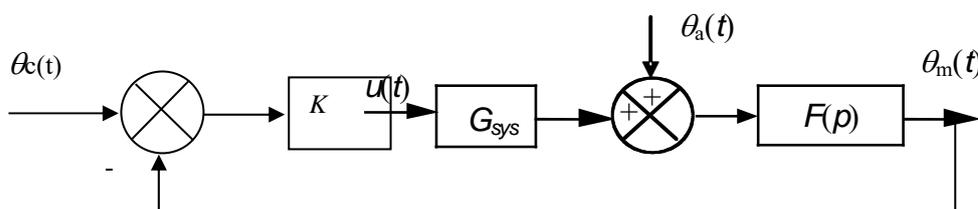


Figure 4 – Block diagram of proportional closed-loop control.

Taking account of the presence of the controller, describe the equation that links the Laplace transforms of the temperatures $\theta_c(p)$, $\theta_m(p)$ and $\theta_a(p)$.

The signals $\theta_c(t)$ and $\theta_a(t)$ are considered to be step functions that appear when the appliance is powered on. From this, use the Final Value Theorem to determine the equation that describes the steady temperature state measured inside the chamber.

1. Identification of static gain G_{sys}

(i) Wait for the steady state to be established (approximately 30 minutes), i.e. when there is no further temperature variation, and record the temperature value measured in a steady state.

(ii) Using the steady state equation above, calculate the value of the parameter G_{sys} .

(iii) Give the units of G_{sys} .

2. Identification of the time constant τ and time lag T

The method used to identify the time constant τ and the delay T consists of creating continuous oscillations in the regulation loop.

The oscillation period T_{osc} and the critical gain of the controller K_{osc} which causes these oscillations allows for the evaluation of the parameters τ and T .

(i) Determination of the gain K_{osc} and the angular frequency ω_{osc}

1- Make sure that the derivative and integral actions of the controller (parameters t_d and t_i) are still inactive (NO on the display)

2- With the temperature setting θ_c remaining unchanged (90 °C), adjust the P_b value to 2% .

3- Display a setpoint variation of ± 5 °C, and observe the changes in the measured temperature on the computer screen.

4- If these changes are in the form of dampened oscillations, reduce P_b by 0.1% and go back to stage 3 above.

When continuous oscillations of the output temperature are observed:

- note the value of P_{bosc} and measure the period of oscillation T_{osc} using the curve displayed on the screen, or a stopwatch

(i) Determination of the gain K_{osc} and the angular frequency ω_{osc}

(ii) Determination of the parameters T and τ of the transfer functions for the climatic chamber based on the aforementioned measurements.

Remember that at the closed-loop stability limit:

$$K_{osc} G_{sys} \frac{e^{-jT\omega_{osc}}}{1 + j\tau\omega_{osc}} = -1$$

where:

$$K_{osc} G_{sys} \frac{1}{|1 + j\tau\omega_{osc}|} = 1$$

and

$$-T\omega_{osc} - \text{Arctg}(\tau\omega_{osc}) = -\pi$$

3. Frequency response curve of $G(p)$

This work can be carried out at the same time as the efforts to establish continuous oscillations that allow for the determination of T and τ .

Using Matlab software, the aim is to analyze the **frequency response** of the system described by its transmittance:

$$G(p) = \frac{G_{sys} e^{-Tp}}{1 + \tau p} \quad \text{with } G_{sys} = 2.34, \tau = 3600\text{s and } T = 50\text{s}$$

This transmittance corresponds to the transmittance that characterizes the behavior of the climatic chamber that has just been analyzed (there may, however, be significant variations in the numerical values of the coefficients).

In order to facilitate the correction of the program, it will be inputted in a text file with the extension.m (executable by matlab), which will be opened via the **File** menu and then by selecting **New**, followed by **M File**.

a) *The transfer functions are inputted* by entering the polynomials of the numerator and denominator separately. The square brackets show that it is a polynomial, with its coefficients always written **in decreasing order of the powers of the Laplace variable p** (referred to as **s** in Matlab):

```
num1=2.34;
den1=[3600,1];    (the comma may also be replaced by a space)
```

Always adopt the habit of putting a semicolon at the end of each line, in order to prevent Matlab from displaying the results of intermediate calculations and thus give a clearer printout of the lines of code selected at the end of the entry.

The time lag can only be accounted during the inputting of the numerator and denominator of the transfer function, and it must be carried out artificially during the calculation of the process gain and phase

b) *The gain and phase* can be calculated using the following series of instructions:

```
w=logspace(-5,-1,100);    % ligne 1
[g,p]=bode(num1,den1,w);  % ligne 2
g_db=20*log10(g);        % ligne 3
```

Line (ligne) 1 allows for the definition of the angular frequency range over which the gain and phase calculation is carried out. In fact, in the variable w, the instruction

$w=\text{logspace}(n,m,q)$ reverses q points of the angular frequency 10^n rad/s at the angular frequency 10^m rad/s, with a logarithmic distribution over this angular frequency range.

Line 2 reverses the modulus and phase of the system in the variables g and p respectively.

Line 3 allows for the conversion of the modulus into gain expressed in dB.

At this stage, the effect of the time lag is not yet taken into account. The following instruction is used to account for the time lag:

```
p=p-180*50*w'/pi; % ' transpose le vecteur
```

 Justify such an instruction by literally expressing the gain and phase added by a time lag.

b) The gain and phase curve can be produced using the following series of instructions:

```
figure; % ligne 1
subplot(211); % ligne 2
semilogx(w,g_db); % ligne 3
xlabel('Pulsations (rad/s)'); % ligne 4
ylabel('Gain (dB)'); % ligne 5
title('Diagrammes de Bode de F1(p)'); % ligne 6
grid; % ligne 7
subplot(212); % ligne 8
semilogx(w,p); % ligne 9
xlabel('Pulsations (rad/s)'); % ligne 10
ylabel('Phase (deg)'); % ligne 11
grid; % ligne 12
```

Line (ligne) 1 allows for the opening of a new figure.

Line 2 allows for the division of the window into several parts. The subplot(m,n,q) function divides the screen into a table of m rows, n columns and selects box q of this table as the designated location for the insertion of any new graph.

Line 3 allows the gain to be plotted on a semi-logarithmic scale.

Line 4 adds a key to the x-axis of the graph.

Line 5 adds a key to the y-axis of the graph.

Line 6 adds a title to the graph.

Line 7 adds a grid to the graph.

These instructions are repeated on lines 8 to 12 for the phase diagram graph.

 Print out the curves obtained, accompanied by the corresponding lines of code.

d) Exploitation of the results

For this system:

-  use a ruler to mark out the asymptotes at the gain curves obtained;
-  measure the static gain (for $\omega \rightarrow 0$);
-  determine the cut-off angular frequency value ω_c (at -3 dB).

Annex: Description and use of the LABVIEW program

To launch the software tool used to supervise the climatic chamber, click the "Climatic_chamber" icon situated on the desktop. To run the program, select "Run" from the menu. To stop the program, select "Stop" from the menu.

A Windows configuration is required (but should already exist):

Start menu: → Parameters → Configuration panel → Regional parameters
 → Number: the "decimal" must imperatively be present, otherwise the acquisition will not operate correctly.

Figure A.1 shows the supervision screen that appears when the "enceinte_climatique.vi" program is run.

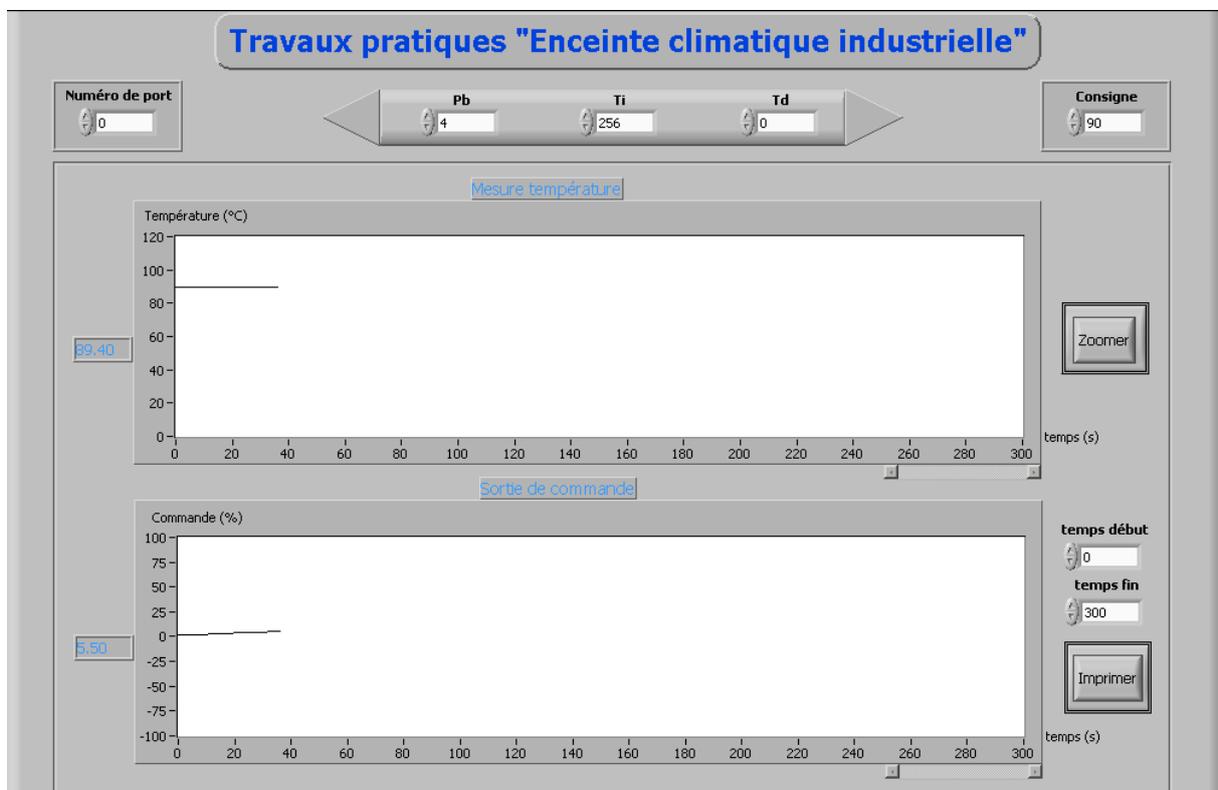


Figure A.1 – Supervision screen

The supervision screen can be used to:

- view the temperature measured inside the oven and the control applied to the climatic chamber
- modify the P_b , T_i and T_d parameters of the controller
- "zoom" in on the window displaying the temperature; the area to which the zoom applies is defined by the "start time" and "finish time" parameters
- print out the windows displaying the temperature and the zone; the area covered by the printout is defined by the "start time" and "finish time" parameters

When the zoom function is used, the window in figure A.2 appears. The scales of the x-axis and y-axis can be directly modified by clicking directly on the maximum value and changing its value.

To end the "zoom function", simply close the window.



Figure A.2 – Zoom screen

A2

MODELING AND IDENTIFICATION OF A
HYDRAULIC PROCESS
MATLAB VERSION

1 - Description of the process to be modeled

The process studied in the framework of this PA consists of one operational part and one control / supervision part. As shown by the diagram in figure 1, the following components are used in the operational part:

- graduated vessel (reference B102)
- tank (reference B101)
- centrifugal pump (reference P101)
- electropneumatic valve (reference V102))
- ultrasound depth sensor (reference LIC102)
- flow rate sensor (reference FIC 101).

This entire process can be seen as a miniaturized representation of a pressurized water distribution system supplying a network of consumers. In this situation, the different components listed above have the following equivalents:

- the graduated vessel is the tank situated at the top of the water tower
- the tank is the well or a storage facility situated below the water tower
- the centrifugal pump is used to pump water up to the water tower
- the electropneumatic valve is used to create a leakage from the vessel towards the tank, simulating the demand for water from the network of consumers
- the ultrasound depth sensor is used to measure the water level in the vessel
- the flow rate sensor measures the rate at which the tank is filled by the pump.

Obtaining consistent pressure throughout the entire distribution system is highly dependent on maintaining the water level in the vessel at a constant level, irrespective of how much water is consumed by the network.

The operational part is controlled by *Labview* software via an S7-type programmable logic controller (PLC) manufactured by Siemens. Communication between the PLC and the PC is carried out using the Profibus DP protocol and via an *Applicom* board. In this PA, the PLC acts exclusively as the communication module between the PC and the power and measurement interfaces situated on the operational part. Indeed, it is solely responsible for transferring the commands originating from the PC to the actuators, and for transferring the information originating from the sensors to the PC. Here, the power interfaces are used to power the direct-current motor that drives the pump and the electropneumatic valve. The measurement interface is used to put the information that originates from the ultrasound sensor and the flow rate sensor into a format that can be exploited by the PLC.

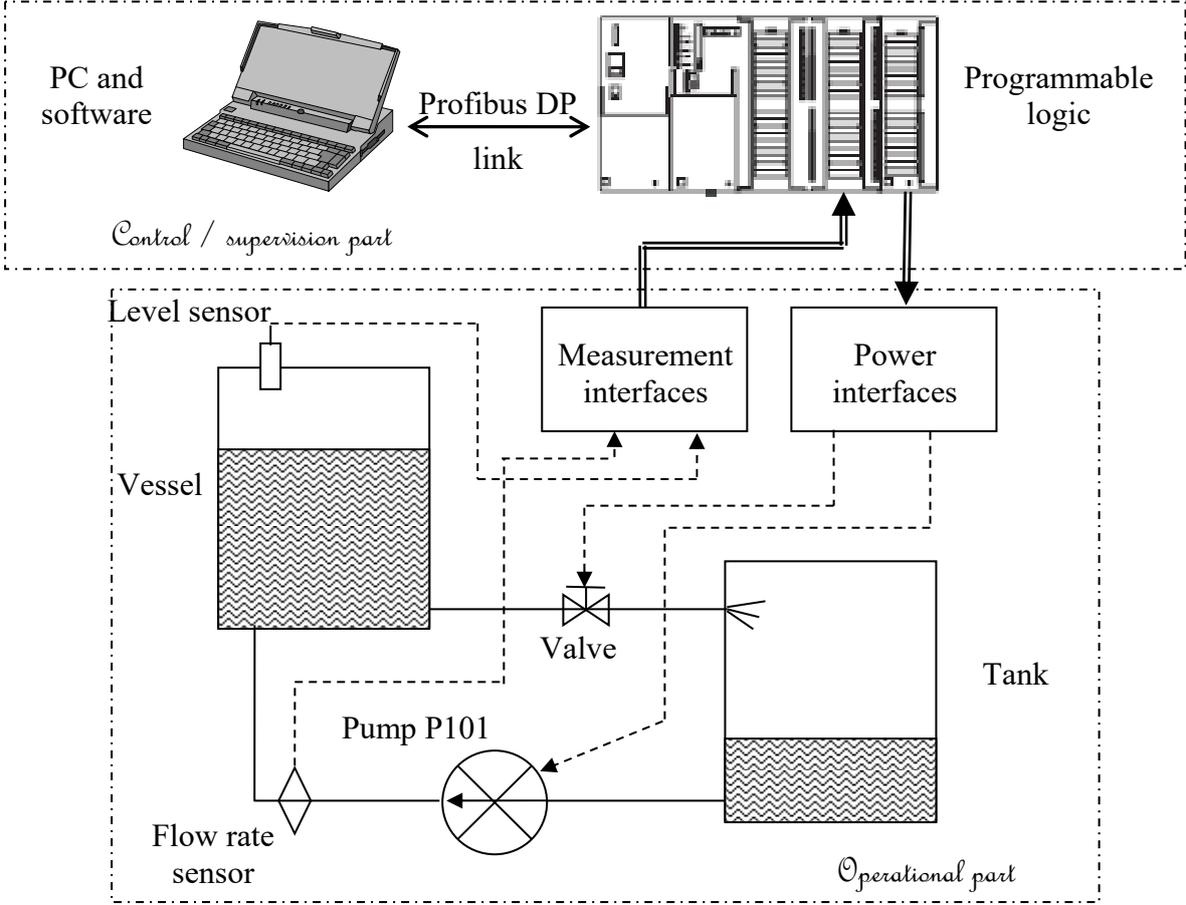


Figure 1 – Process diagram



2 – Modeling and identification

It is recommended for all of the measurement values to be expressed in the MKSA system

The height of the water level in the vessel is inferred by the difference between the flow rate of the water going into the vessel and the flow rate of the water coming out of it. Mathematically, such a balance is expressed by the following formula:

$$S \frac{dH(t)}{dt} = Q_e(t) - Q_s(t), \quad (1)$$

where:

S designates the cross-section of the vessel

$Q_e(t)$ designates the flow rate going into the vessel

$Q_s(t)$ designates the flow rate going into the vessel

2.1 – Reminders about fluid mechanics

In many situations, the behavior of a hydraulic system can be expressed by the Bernoulli equation. This equation is an energy balance applied to two points of a conduit and reflects the energy conservation of the fluid as circulates in the conduit. Between points 1 and 2 of the conduit described by figure 2, it can therefore be stated that:

$$\frac{P_1}{\rho} + gz_1 + \frac{v_1^2}{2} = \frac{P_2}{\rho} + gz_2 + \frac{v_2^2}{2} + S_o - P_e \quad (2)$$

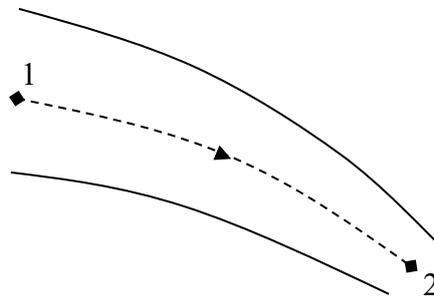


Figure 2 – The conduit in question

In the relationship (2):

- P is the pressure of the fluid at the point in question (in Pa)
- ρ is the density of the fluid (in kg/m^3)
- g is the acceleration due to gravity (9.81 m/s^2)
- z is the elevation of the point in question (in m)
- v is the speed of the fluid at the point in question (m/s)
- S_o is a source term if the conduit includes components such as pumps
- P_e is a loss term if the fluid flows down a slope.

Before starting the experiments, make sure that:

- the Controller/PLC switch is in the PLC position
- the manual valves V103, V107, V108, V109 are in the closed position

- the manual valves V101, V104, V112 are in the open position

Refer to annex 1 for information on powering the operational part and running the supervision software.

★ The PLC produces a control signal with an amplitude of 0-10V and allows for the acquisition of a measuring signal of the same amplitude. Create a block diagram of the procedure used, showing the following components:

- programmable logic controller (PLC)
- pump
- tank
- level sensor

Write down the unit of the signal situated at the output of each block and its range of variation.

2.2 – Model of the vacuum pump

Open the manual valve V104 and the electropneumatic valve. Close the manual valve V101. Drain the tank completely and then leave the electropneumatic valve open. Activate the pump with a command U of 100%, and raise the output flow rate of water from the pump until the maximum flow rate is obtained. Now show that the dynamic behavior of the pump can be described by the transmittance:

$$\frac{Q(p)|_{H=0}}{U(p)} = \frac{K}{1 + \tau p}$$

$Q(t)|_{H=0}$ designates the pump output (flow rate), for a given command $U(t)$ and for a depth of water close to zero.

Using the curve of figure 3, measure the value of the pump output at a steady state and use it to determine the value of the static gain K .

Determine the value of τ .

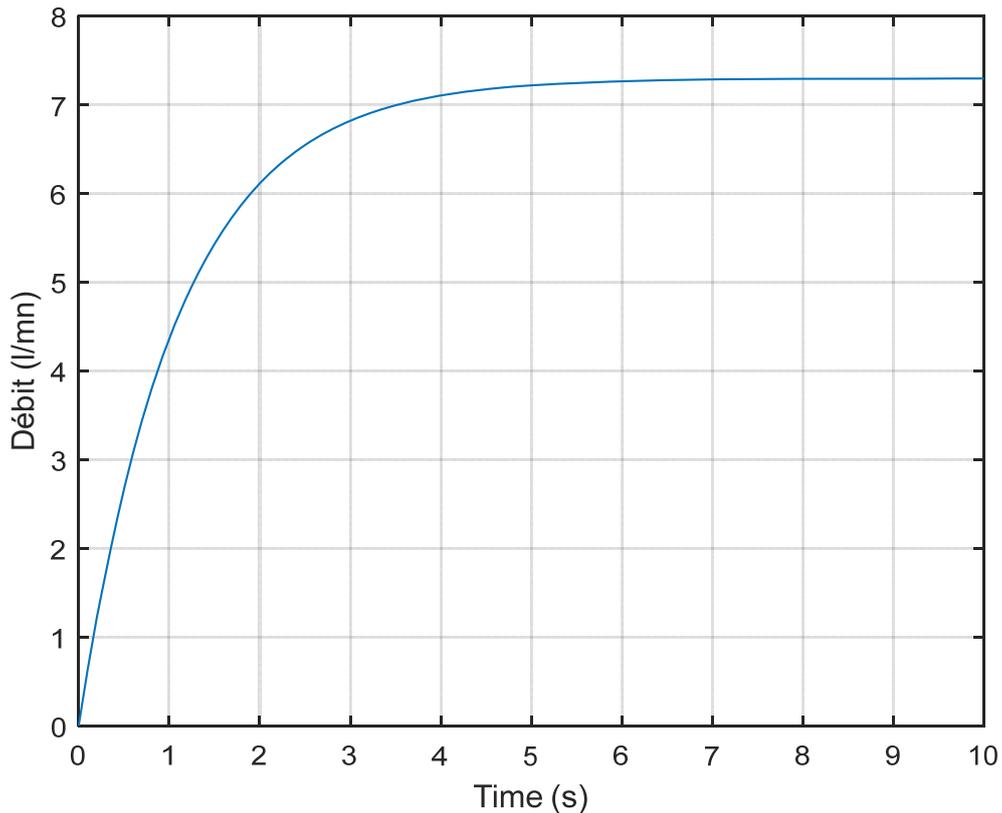


Figure 3 – Measure of the flow rate over time

2.3 – Filling model

When the pump is operating, the more the water level rises in the vessel, the greater the increase in pressure on the pump. The pump must therefore provide more power in order to continue to raise the water level in this vessel at the same speed.

Using the Bernoulli equation applied between points 1 and 2 shown in figure 4, it is assumed that the flow rate entering the vessel is of the following form:

$$Q_e(t) = Q(t)_{H=0} - aH(t). \quad (3)$$

with

$H(t)$: depth of water in the vessel

a : coefficient to be determined (and according to the pump power supply)

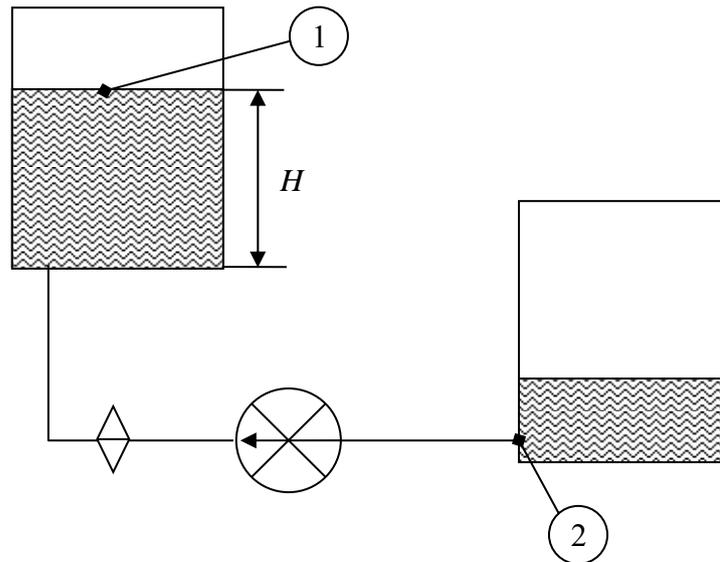


Figure 4 – Application of the Bernoulli equation for leakage-free operation without a pump

We would therefore like to determine coefficient a .

To this end, open valve V104 and the electropneumatic valve, and close the manual valve V101.

With the pump command U percentage at 100%, record the value of $Q(t)$ when the steady output state has been established.

Then close valve V104 and the electropneumatic valve, and open the manual valve V101.

Using the measure of $Q_e(t)$ and $H(t)$ shown in figure 5, and use them to determine the value of a .

Comment: the pump shuts down automatically when the water level in the tank drops too low.

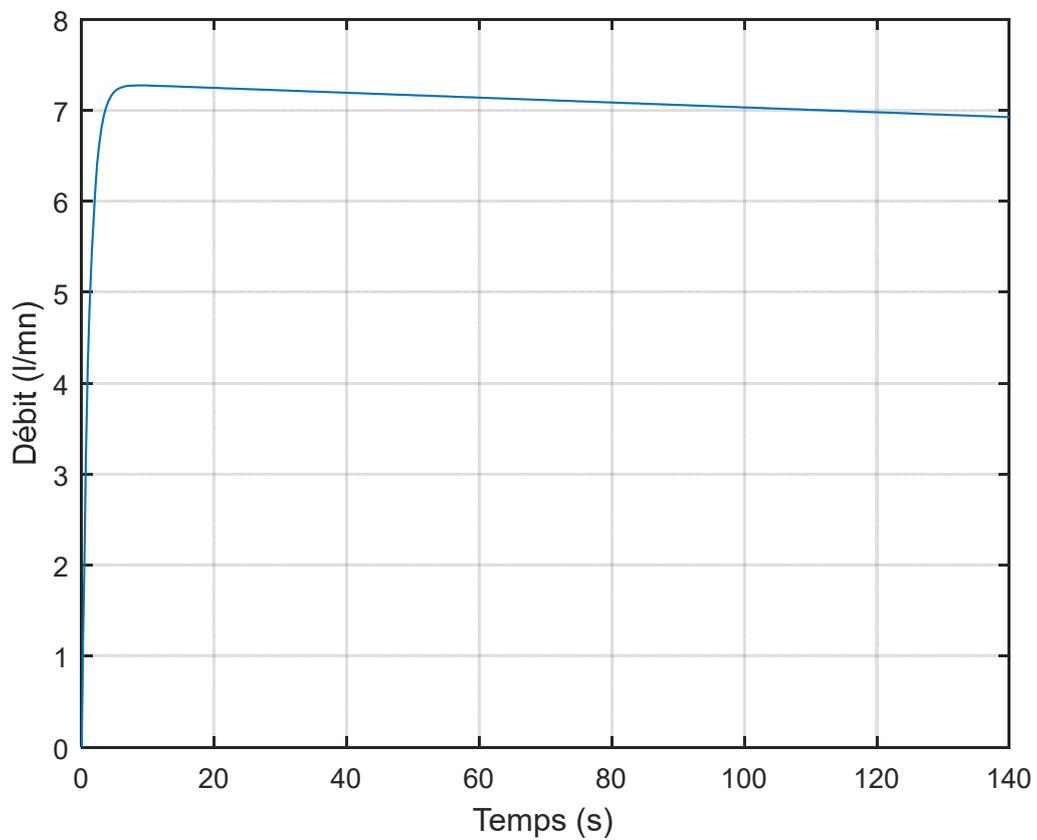
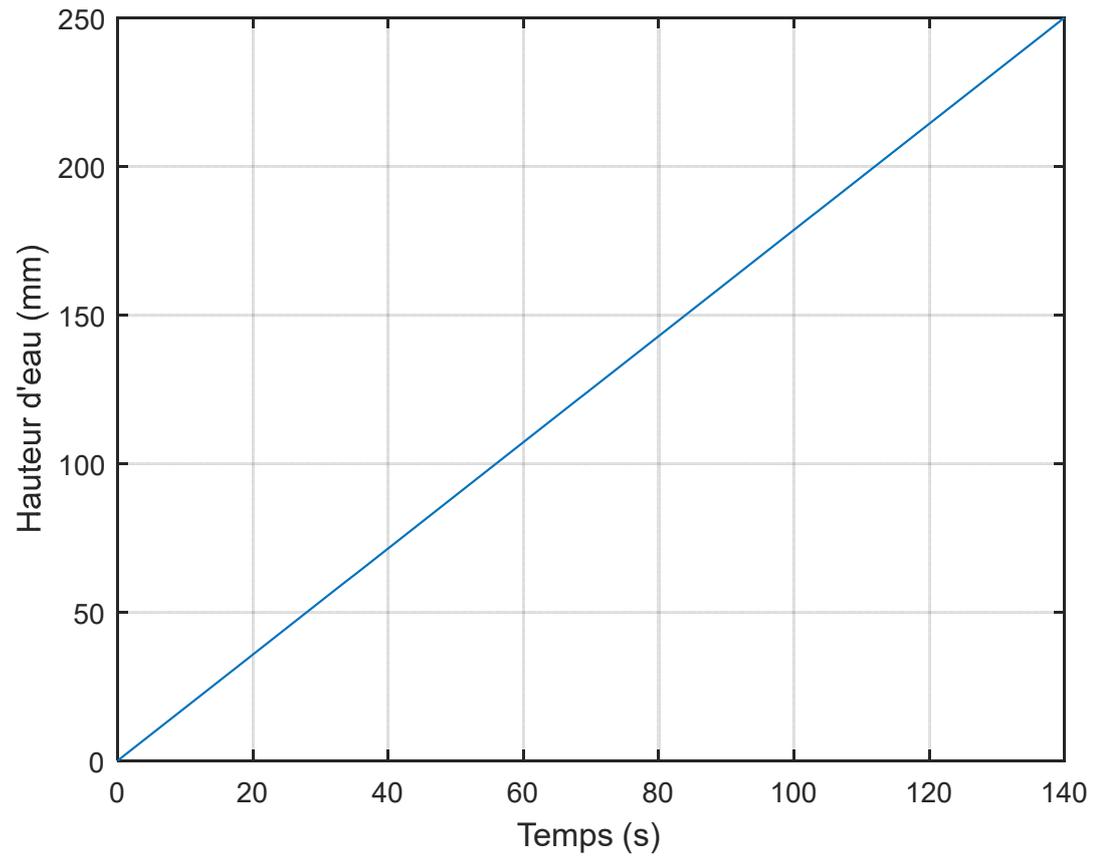


Figure 5 – Records of the variations of the water level in tank 1 and the flow rate as a function of time

2.4 – Leakage model

In the event of the system operating with a leakage and without a pump, the Bernoulli equation applied between points 1 and 2, as shown on the diagram in figure 4, can be used to determine that the flow rate emerging from the tank is given by:

$$Q_{s(t)} = \alpha \sqrt{H(t)} \quad (4)$$

α is a coefficient that can be applied in order to take account of the valve opening rate, for example.

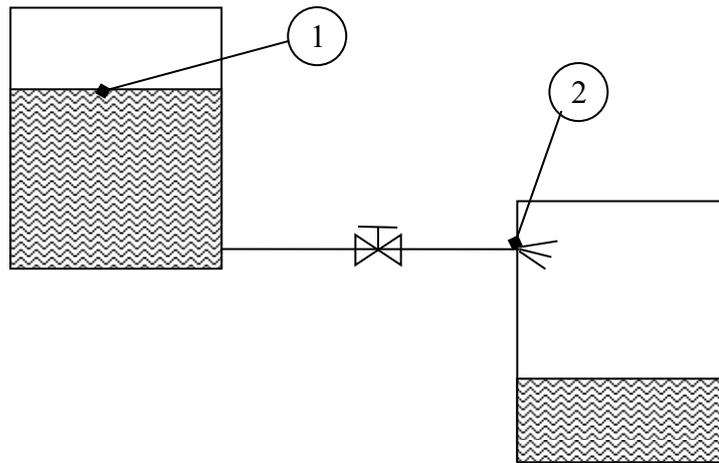


Figure 4 – Application of the Bernoulli equation for operation with a leakage and without a pump

We would like to determine the coefficient α . To this end, fill the vessel to its maximum level by activating the pump.

When the maximum level has been reached (the pump shuts down automatically), close the electropneumatic valve and the manual valve V101 to prevent any leakage through the pump.

Open the solenoid valve and record the changes in the water level in the vessel.

With the knowledge that:

$$- S \frac{dH(t)}{dt} = -Q_s(t) \text{ and that } \frac{H(t_1) - H(t_2)}{t_1 - t_2} \approx \frac{dH(t)}{dt} \text{ for } t_1 \text{ close to } t_2,$$

- the cross-section of the vessel is 0.19 m by 0.175 m:

evaluate the coefficient α from several measurements using the curves given in figure 7.

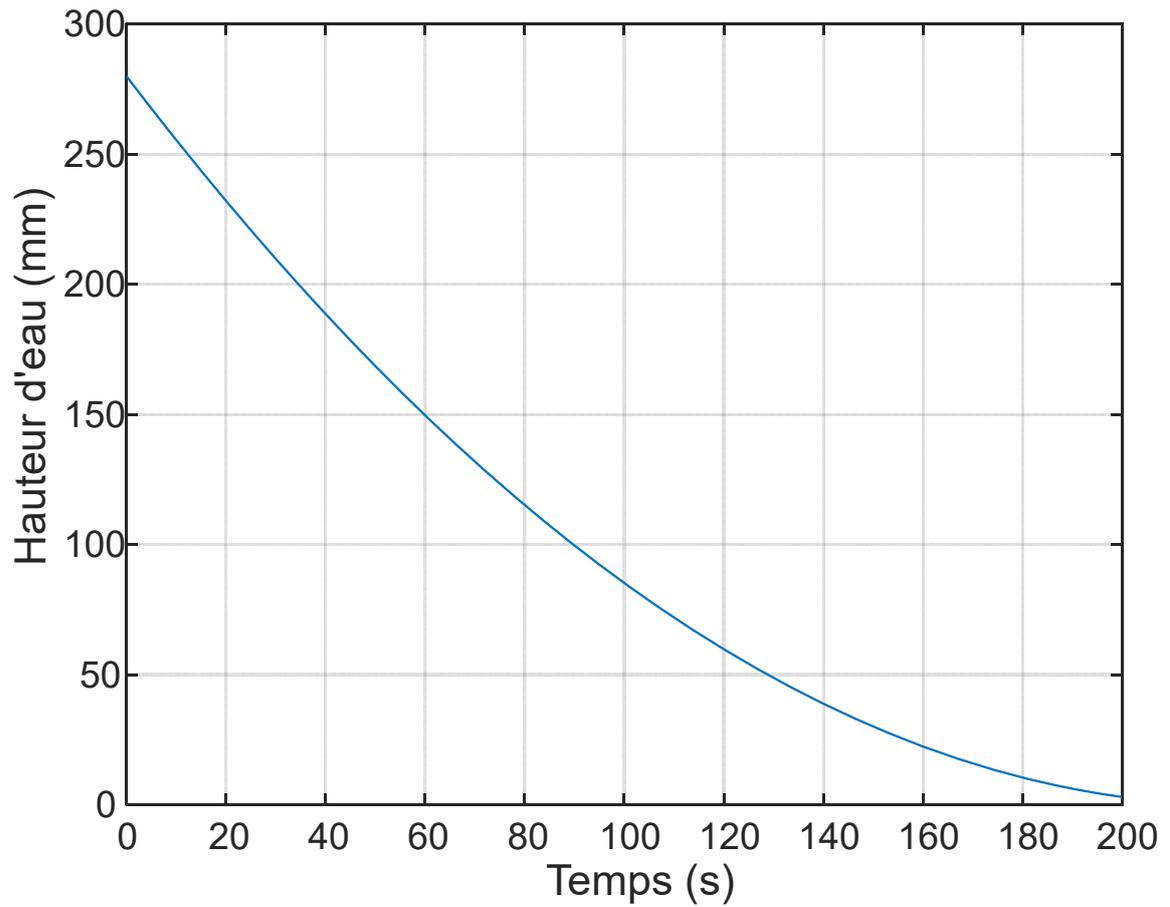


Figure 7 – Record of the water height in tank 1 as a function of time

2.5 – Overall model

The block diagram of the whole system is given by figure 8.

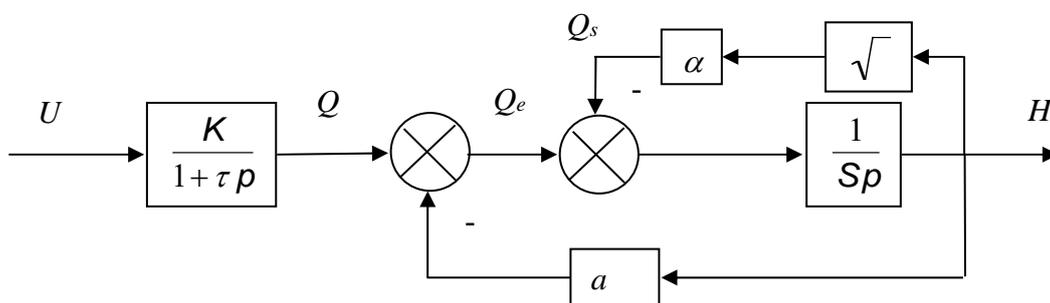


Figure 8 – Block diagram

3 – Validation of the model

3.1 – Validation of the model for leakage-free operation with a pump

Drain the tank completely and then close the electropneumatic valve. Activate the pump with a maximal command and record the changes in the depth of water in the vessel.

At the same time, run the Matlab software to simulate the model obtained in questions 2.2 and 2.3.

To this end, in Matlab, *open Simulink* by typing:

```
>>simulink
```

The Simulink window opens and reveals a series of libraries of different functions (see figure 5). If necessary, you may minimize this window and keep it in the top right-hand corner of the screen.

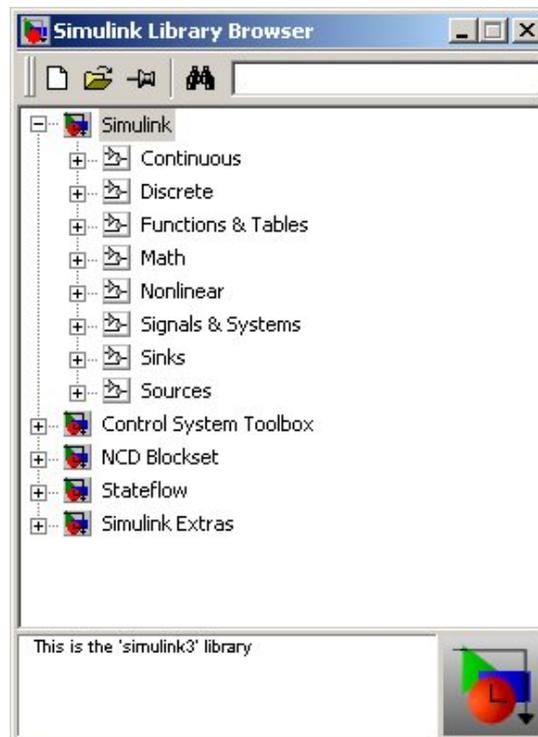


Figure 9 – Simulink control window and directory in which the blocks to be used are stored

Now open your work area, by clicking the blank page icon in the Simulink window.

The aim is to simulate the system's response to a step signal with an amplitude of 1 (max. control of 100%),

$$G_1(p) = \frac{K}{1 + \tau p} \frac{1}{a + Sp}$$

which characterizes the dynamic behavior of the operational part by taking account of the pump dynamics.

Create the simulation diagram in the working window ("Untitled").

To this end, open the "Sources" directory in the Simulink window. Select and drag the **Step** block into the working window (left-click on the mouse and hold while moving it to the desired location, then release).

In the same way, move the following elements consecutively into the working window:

- two transfer functions (Transfer Fcn), dragged from the "Continuous" directory
- and one oscilloscope (Scope), from the "Sinks" directory.

Line up these elements in the correct order, and *make all of the connections* using the mouse (press the left-hand button inside the small output triangle for a block, drag it by holding the button down until you reach the small input triangle for the next block, and then release it).

Then program each element by double-clicking its icon:

- with the Step time of 0 s for the moment at which this step is applied, an Initial value of 0 and a Final value of 1. Validate by clicking "OK".
- with the transfer functions, by entering the numerator and denominator of $G_1(p)$.

Finally, program the parameters chosen for the simulation, by clicking the 'parameters' command in the 'Simulation' menu in the "Untitled" window: select Ode45 (this is the digital integration method), and enter the simulation finish time that is compatible with the experiment being carried out (Stop time). Do not modify the other values and validate by clicking "OK".

Then run the simulation ('Start' command in the 'Simulation' menu), and double-click the oscilloscope icon to open the viewing window.

Compare the response obtained with the measurement recorded. State your conclusions, especially concerning the validity of the parameters τ and a .

3.2 – Validation of the model for operation with leakage and without a pump

Using Matlab software, the differential equation

$$S \frac{dH(t)}{dt} = -\alpha \sqrt{H(t)}$$

allows for the following representation:

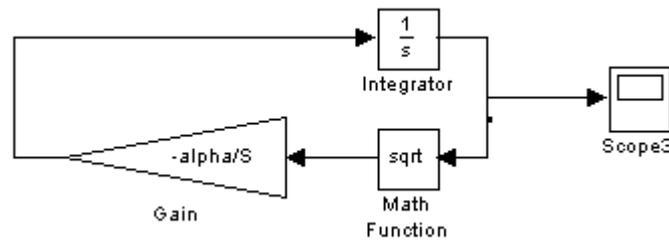


Figure 10 – Leakage simulation diagram

Justify such a diagram by entering the name of the variables at the output of each block. Correctly initialize the integrator and run the simulation.

Compare the time response obtained with the time response recorded in paragraph 2.4.

State your conclusions, especially concerning the validity of the parameter α .

3.3 – Validation of the model with a leakage and with a pump

Due to the presence of the term $\sqrt{H(t)}$, the equation obtained is non-linear in $H(t)$. To linearize this equation, we shall assume that the depth of water remains approximately H_0 so we can state that:

$$H(t) = H_0 + h(t) \quad \text{with} \quad h(t) \ll H_0$$

With the understanding that $\sqrt{1+x} \approx 1 + \frac{x}{2}$ for $x \ll 1$ we can therefore write:

$$q_s(t) = \alpha \sqrt{H_0} \left(1 + \frac{h(t)}{2H_0} \right)$$

where $q_s(t)$ represents the variation in the leakage flow rate around the nominal flow rate Q_{s0} which corresponds to the nominal depth H_0 :

$$Q_s(t) = Q_{s0} + q_s(t)$$

It should be remembered that the equation for the variations obtained is only valid if the hypotheses that allow it to be obtained are also valid, i.e. $h(t) \ll H_0$. In practice, the equation obtained will only describe the system accurately if the depth of water $H(t)$ remains close to H_0 .

Close the manual valve V104 and open the electropneumatic valve. Apply a command of 85%. Wait until the water level in the vessel reaches its steady state.

Record the H_0 level at a steady state.

Once the steady state has been obtained, apply a command of 95% and record the changes in the depth and flow rate of the water over time until a new steady state has been established.

Then record the variation in flow rate $q(t)$ between the two steady states. This record is represented by figure 11.

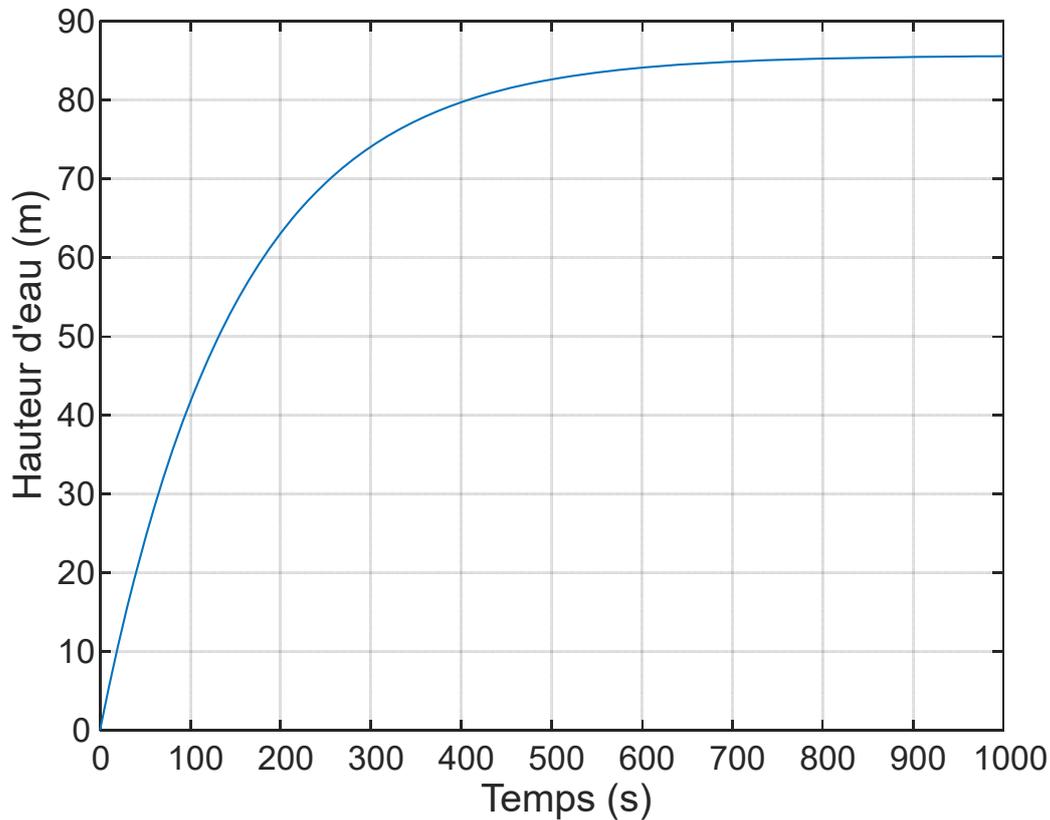


Figure 11 – Record of the change in water depth in tank 1 as a function of time for a flow step of 10%

As shown by the theoretical analysis, the changes in the level around the chosen point of operation can be likened to a first order system of transmittance:

$$G_2(p) = \frac{1}{sp + a + \frac{\alpha}{2\sqrt{h_0}}} = \frac{h(p)}{q(p)}$$

From the measurements, record the time constant and the static gain of $G_2(p)$. Compare them with the theoretical values.

At the same time, using the Matlab/Simulink software, plot the response of the transfer function $G_2(p)$ at an amplitude step function of Δq . Compare the response obtained with the measurement recorded. State your conclusions.

Annex 1 – Starting up the operational part of the supervision system

To run the supervision program for the operational part, double-click the "Manip_festo.vi" icon situated on the desktop, which runs the Labview software and the supervision program.

The supervision screen in figure A.1 then appears.

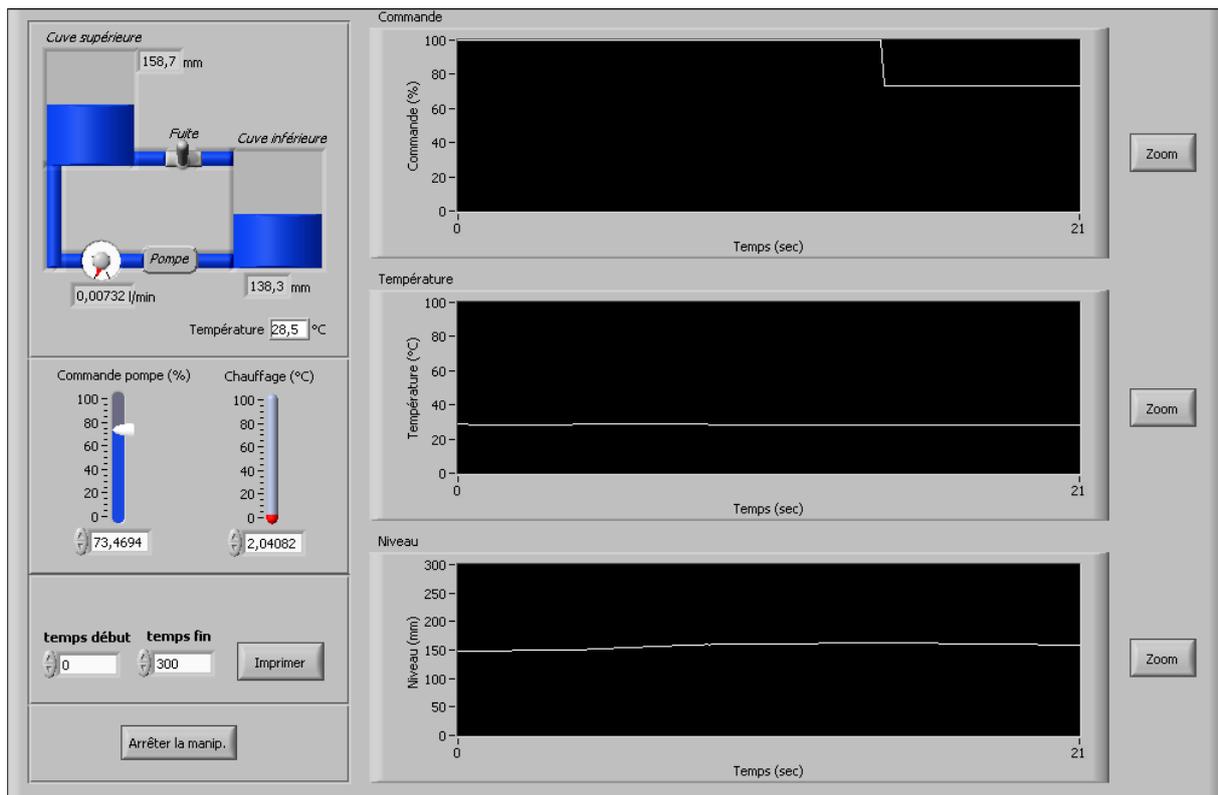


Figure A.1 – Supervision screen

The installation diagram is shown in the top left-hand corner of the screen. The graphic windows show (from top to bottom), the valve status, the pump control level (as a %), the pump output flow rate (l/min) and the level in the vessel (in mm)

The pump control level can be modified by moving the cursor in the pump control window or by entering this value in the "pump control" window in the "Control:" field.

The status of the valve that controls the leakage can be controlled directly in the vessel window by moving the control lever.

The numerical value of the level in the vessel and the flow rate are shown in the vessel window.

To facilitate the reading of the curves, the zoom function allows them to be displayed in full-screen mode. The window must be closed in order to return to the normal viewing mode.

To print out the curves, press the "Print" key; the graphs corresponding to the area defined by the "start time" and "finish time" parameters will then be printed out.

A3.1 / A3.2

FEEDBACK-BASED CLOSED-LOOP POSITION CONTROL, TESTING AND ANALYSIS

I DESCRIPTION OF EQUIPMENT

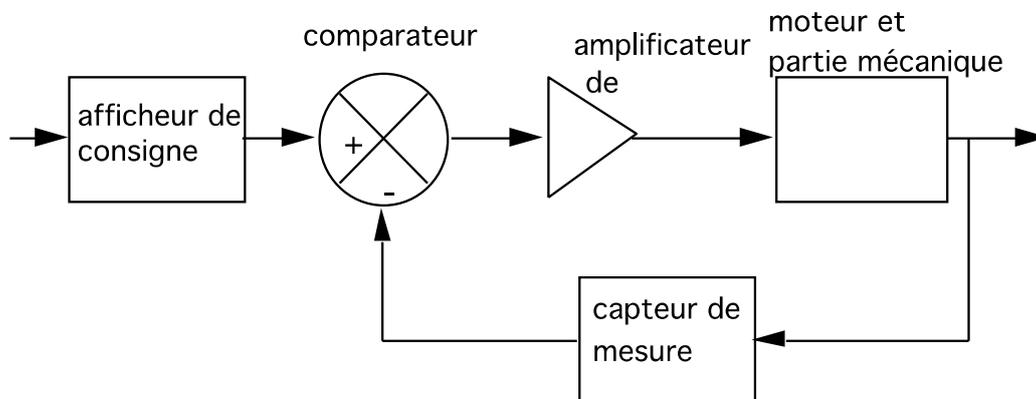
It is important to read this description attentively, in preparation for and then during the practical assignment session, in order to familiarize yourself with the equipment that will be used on several occasions, and avoid any problems in carrying out the different wiring operations.

I-1 Introduction

This equipment allows for the creation of and analysis of closed-loop control systems:

- closed-loop control of speed (analog and digital)
- closed-loop position control (analog and digital)

We shall initially restrict ourselves to studying analog feedback systems as shown in figure 1.



afficheur de consigne=> set point display; comparateur=> comparator; amplificateur de puissance=> power amplifier; moteur et partie mécanique=> motor and mechanical part; capteur de mesure=> measurement sensor

Figure 1 - Analog system

The practical creation of these different feedback systems will be carried out by interconnecting two main blocks – the analog block and the mechanical unit block, which will now be described.

I-2 Mechanical unit

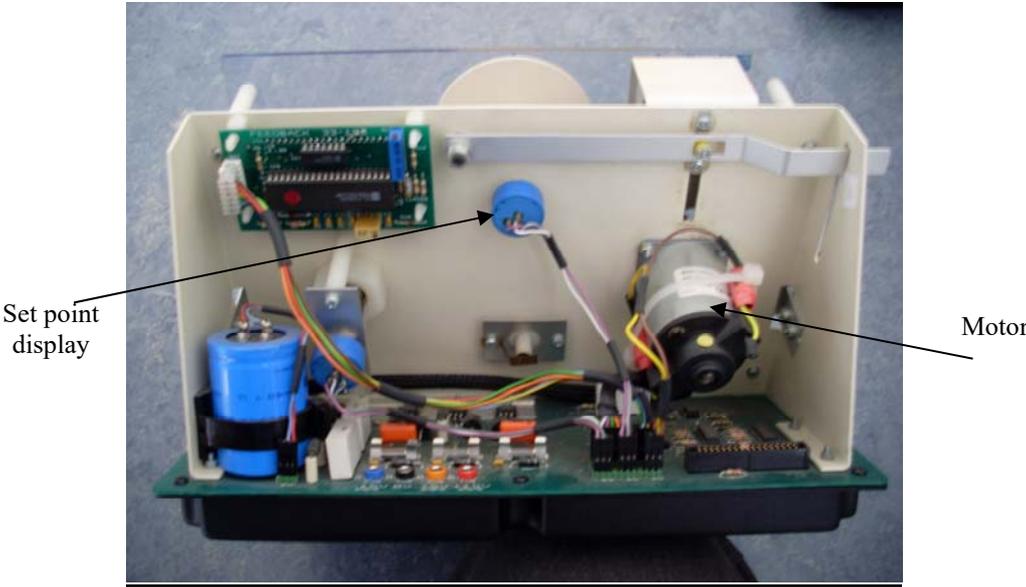
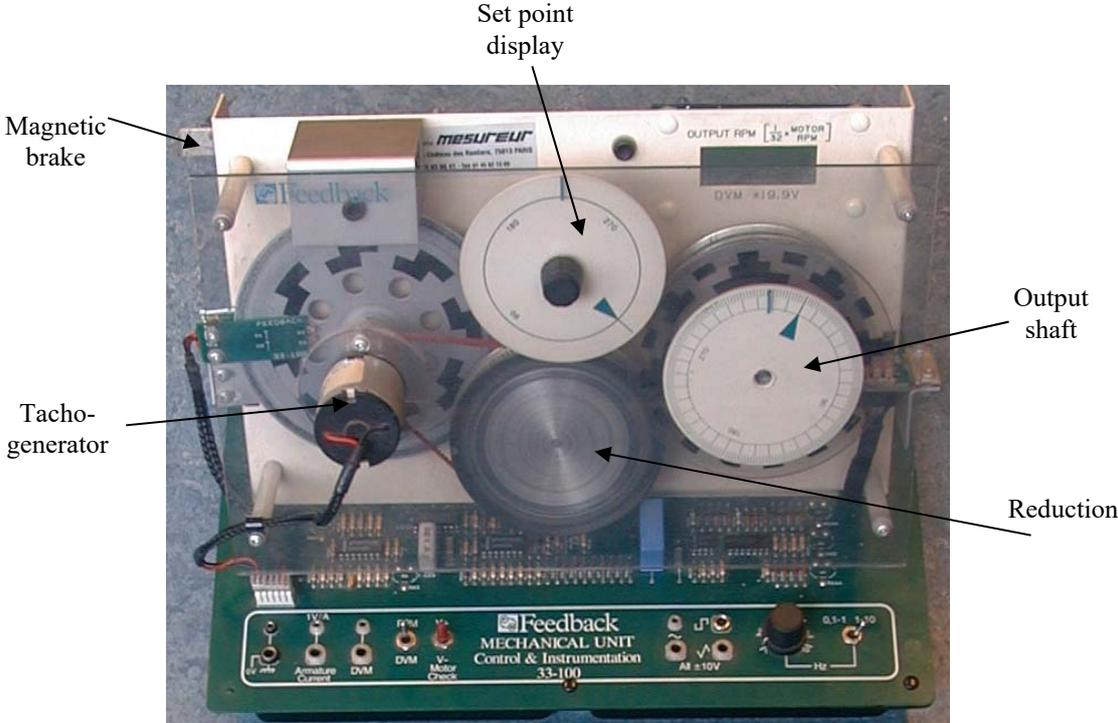


Figure - Photograph of the mechanical unit on the breadboard

This part consists of all components required for the study and creation of closed-loop position control (analog and digital), and speed control (analog and digital) systems. It consists of:

- A direct current motor

The following components are mounted on the motor shaft:

- A **brake magnet**, which is used to vary the motor load and therefore for performing load tests. This brake is controlled by a lever (the brake lever).
- A **tachogenerator**, mounted directly on the motor shaft. The outlet of this generator delivers voltage U_T that is proportional to the rotation speed of the motor ($U_T = K_T * \Omega$). This generator acts as an analog speed sensor.
- A **brake disk & speed track**, consisting of two tracks, which, via a track reader, provides digital information about the motor rotation speed. The external part of the disk is used by the brake magnet.
- A **1/32 reduction gear** on which the following are mounted: an **output potentiometer** giving analog information about position, and a **disk** with several tracks, which, via track readers, provides digital information about position.

- An **input potentiometer** (input shaft) mounted on a separate shaft from the motor, on which the position set point is displayed.

- A **function generator** (in the bottom right-hand corner) that provides square-wave, triangle waveform and sine-wave signals with amplitude of +/- 10 Volts, and adjustable frequency.

- A **digital display** (3 digits, output shaft: rpm or DVM), which displays either the rotation speed at the reduction gear output, or the voltage value transmitted to the pin marked DVM Input (in the bottom left-hand corner). The choice of information to be displayed is made via the "RPM/DVM switch" situated in the bottom left-hand corner.

- A pin (armature current signal) which delivers a voltage that is proportional to the current in the armature of the motor (1V/ A).

- Power supply inputs.

- A ribbon cable (34 wires) which provides a link between the mechanical unit and the analog unit.

I-3 Analog unit (Figure 3)

This part consists of the electronic components required for the creation of the different closed-loop control systems and receives information transmitted by the different sensors through the 34-wire "ribbon" cable.

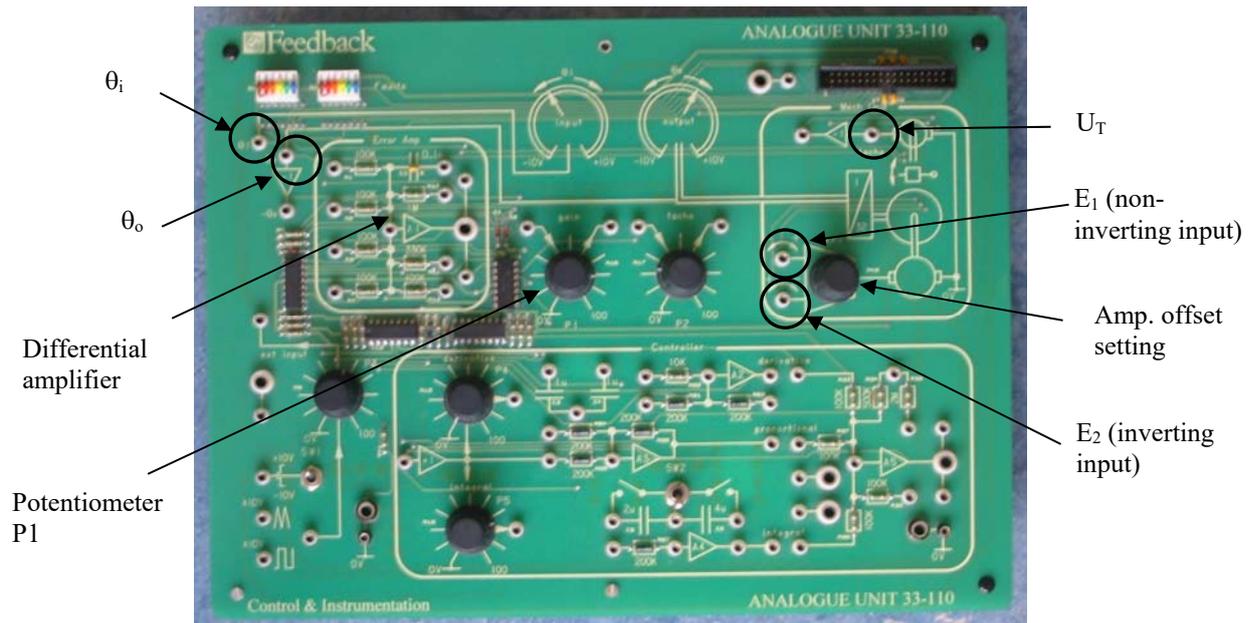


Figure - Photograph of the analog unit on the breadboard

1-3-1 Information originating from sensors

* **U_T (tacho)** tachogenerator output. This signal can be used directly or inversely.

* **θ_i (input shaft angle signal)** output of the set point display potentiometer. This signal can be used directly or inversely.

* **θ_o (output shaft angle signal)** output of the position measurement potentiometer. This signal can be used directly or inversely.

1-3-2 Electronic components

* **P3 (external input potentiometer)**. This potentiometer is used to adjust the amplitude of the signals that need to be sent to the input of the "error amp" comparator. Indeed, only fixed amplitude signals are available via the ribbon cable or directly on the mechanical unit. In the bottom left-hand corner of this unit, there is a triangle waveform, a square-wave signal and direct voltage of ± 10 V.

* **Error amp and P1**. Both of these components can be used to create a comparator followed by an adjustable gain amplifier. This will be studied in a subsequent stage.

* **P2 (tacho)**. This potentiometer is used to set the amplitude of the signal delivered by the tachogenerator.

* **Power amplifier and zero adjustment**. The power stage supplies the power required for controlling the motor. The output is directly connected to the armature of the motor.

This stage has two inputs that allow for the inversion of the motor control voltage. If a positive voltage is sent to the top input (- +, E1), the motor rotates in a given direction; if the same voltage is sent to the bottom input (+ -, E2), it rotates in the opposite direction.

* **Controller** This unit allows for the integration of correctors into feedback systems, which help to improve the performance of these systems. This controller unit will be studied during subsequent operations.

II PRELIMINARY STUDY

II-1 Study of the amplifying comparator stage

Connections are shown in the diagram in figure 4.

II-1-1 If R_1 is the input resistor, R_2 is the degenerating resistor, p_1 is the attenuation caused by P1 and if the + input is assumed to be directly connected to ground, express V_1 according to R_1 , R_2 , p_1 , V_r and V_e .

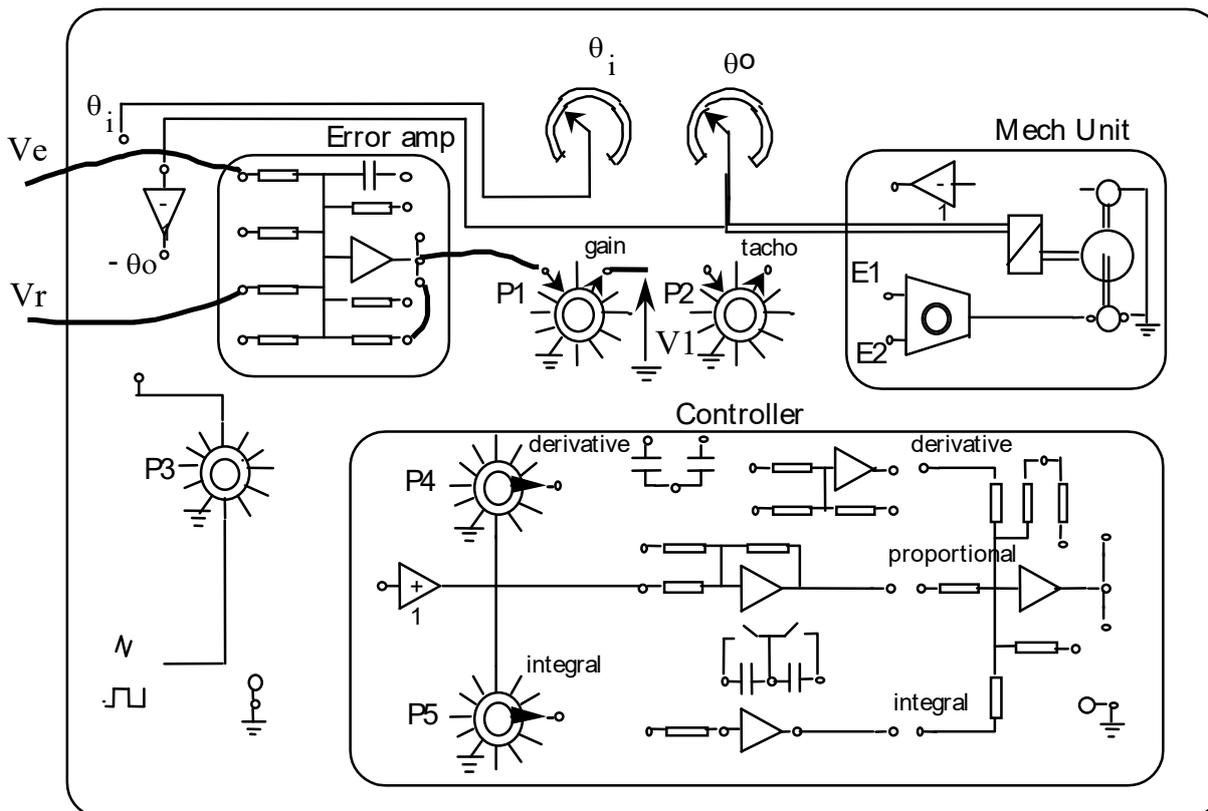


Figure 4: Wiring of the comparator

II-1-2 Express V_1 in the system in figure 5

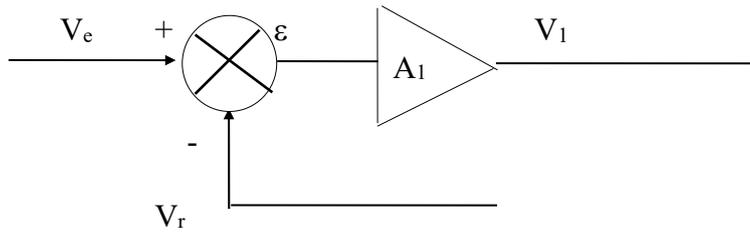


Figure 5: Equivalent wiring diagram

What is required to make the two expressions of V_1 in figures 4 and 5 identical?

II-2 **Closed-loop position control**

- With reference to figure 1 and the description of the existing system, draw a block diagram corresponding to closed-loop position control. This diagram must show the comparator, the gain adjustment A_1 , the power amplifier, the motor and, if necessary, the reduction gear and type of sensor used.

- In figure 6, complete the links between the different components, in bold type, in order to create a closed-loop control system for position. For this purpose, the .pdf file in figure 6, which can be found at "Groupe B/Autom/Abaques et papier milli et semilog/schema_feedback.pdf" can be printed out.

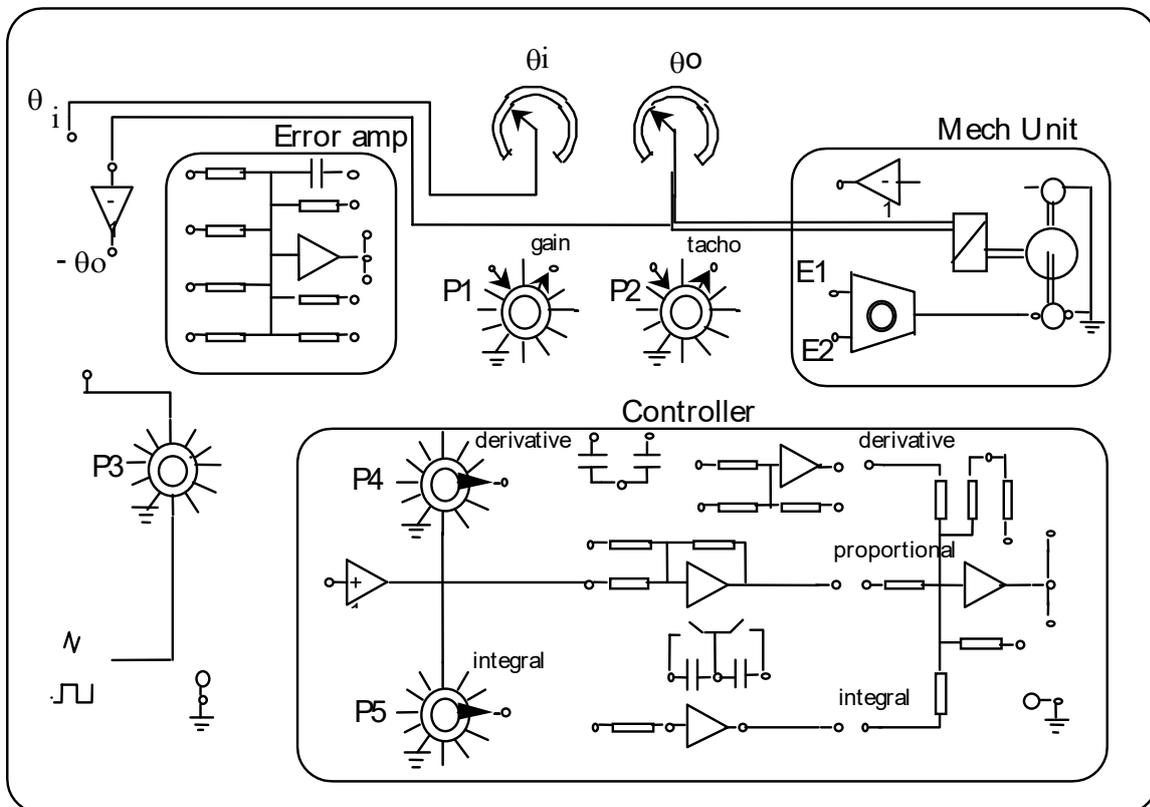


Figure 6 - Diagram to be completed

III OPERATION

III-1 Closed-loop control diagram

Create a block diagram for closed-loop position control, showing:

- the set point display
- the differential amplifier
- the correction gain
- the power amplifier
- the motor
- the reduction gear
- the tachometer generator

Write down the unit of the signal situated at the output of each block and its range of variation. What is the technological nature of the set point display?

III-2 Analysis of closed-loop position control

Several connections of our equipment can be carried out:

V_r may be equal to θ_0 or $-\theta_0$

The direct chain may be reversed at the power amplifier by using the input E_1 or the input E_2

III-2-a With the power off, establish the connections that allow for closed-loop position control (figure 6), and make sure that the connections are checked prior to powering on.

With a degenerating resistor $R_2 = 100 \text{ K}\Omega$, P_1 in the 20% position, in the four following cases, observe the behavior of the system while moving the set point display potentiometer. Comment on the results obtained.

$V_r = \theta_0$ and input E_1

$V_r = -\theta_0$ and input E_1

$V_r = \theta_0$ and input E_2

$V_r = -\theta_0$ and input E_2

III-2-b When the system is operating correctly, for $\theta_i = 0$, apply the setting $\theta_0 = 0$ using the zero adjustment potentiometer (power amplifier), then rotate the set point display potentiometer through 90° and comment on the results (direction of movement of disk and accuracy).

III-2-c For $V_r = -\theta_0$ and E_2 , adjust the offset for the 0 position of θ_0 and study the behavior of the system, while rotating the set point display potentiometer from 0° to 90° , for the following cases:

$R_2 = 100 \text{ K}\Omega$ P_1 20%

$R_2 = 100 \text{ K}\Omega$ P_1 50%

$$R2 = 100K\Omega \quad P1 \ 100\%$$

$$R2 = 1M\Omega \quad P1 \ 100\%$$

What do you observe?

III-2-d Still for $V_r = -\theta_o$ and E2, and in order to improve the interpretation of the preceding results, replace θ_i with a square-wave signal whose amplitude can be adjusted by P3. With P3 positioned at 40%, adjust the frequency of the square-wave signal so that θ_o can be viewed correctly on the storage oscilloscope. (approx. 0.1 Hz).

For each of the preceding cases (R2 and P1) plot the input (output P3) and the output θ_o of the system (depending on the available equipment, the digital oscilloscope or the Labview virtual oscilloscope will be used).

For each of the readings, show the response time at 5% of the system (see figure 7: Definition of response time).

Using the preceding curves and results, answer the following questions:

- Is closed-loop position control equal to a first or second-order system?
- How does the response time vary according to the gain of the direct chain?
- Is the output directly equivalent to the input?

III-2-e Analysis of the disturbance effect

On the block diagram of closed-loop position control, show the offset voltage as a disturbance at the power amplifier input.

Still for $V_r = -\theta_o$ and E2, and for $\theta_i = 0$, turn the offset voltage adjustment potentiometer to its maximum setting.

Measure the position error for each gain value corresponding to the following settings:

$$R2 = 100K\Omega \quad P1 \ 20\%$$

$$R2 = 100K\Omega \quad P1 \ 50\%$$

$$R2 = 100K\Omega \quad P1 \ 100\%$$

$$R2 = 1M\Omega \quad P1 \ 100\%$$

State your conclusions.

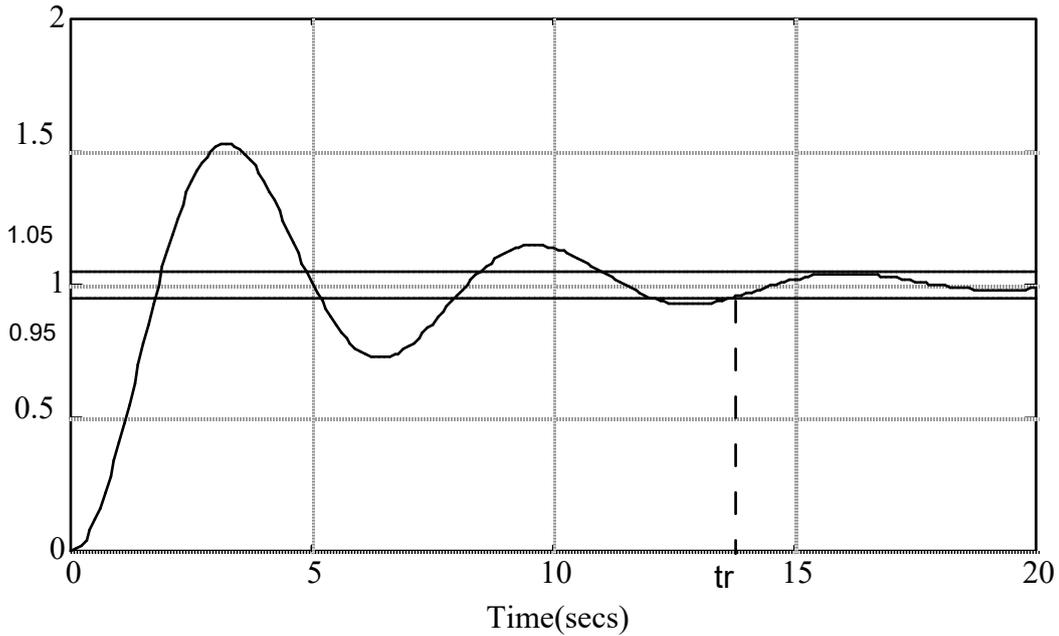


Figure 7 – Definition of the response time at 5%.

III-3 Identification of the motor unit

III-3-a Time identification

Disconnect the preceding assembly and connect the outlet P3 (40%) directly to the input E1 or E2 of the power amplifier.

Send a +/- 10V steady-rate signal to input P3 and check for the reversal of the motor's direction of rotation.

Replace the steady-rate signal with a +/-10V square-wave signal and adjust its frequency in in order to allow for the proper observation of the motor speed response (approx. 0.2 Hz).

Once these adjustments have been made, record the motor speed response (depending on the available equipment, the digital oscilloscope or the Labview virtual oscilloscope will be used).

From this reading, subtract the electromechanical time constant τ for the motor and the static gain K . Is the power amplifier and motor assembly a first-order or second-order system?

III-3-b Frequency identification

It is often useful to know a system's response to variable-frequency sinus-wave signals, and to show this response on Bode plans or Nichols plots.

To this end, an analyzer software program created by students in the GEII department at Bordeaux IUT, in Labview (see instructions at the end of the 1st series of the Automatic Control Engineering PA workbook), will be used.

The following settings will be applied on the front of the analyzer:

- frequencies of 0.1 Hz to 10 Hz, 20 points per decade;
- amplitude 2 V, offset 3 V.

1. GEII analyzer in Labview.

- When the power supply of the Feedback system is turned on, adjust the zero adjustment potentiometer so that, in the absence of input voltage for the power amplifier, the motor does not rotate. Also make sure that the brake magnet is not on (the brake lever is raised).

- Connect the output of the Labview analyzer generator to input E1 of the Feedback power analyzer and to channel 0 of the Labview analyzer.

- Connect the tachometer generator output (U_T) for the Feedback to channel 1 of the Labview analyzer.

- Using Bode coordinates, plot the frequency response of the system, from 0.1 Hz to 10 Hz. The generator will be programmed so that the amplitude of the sinus-wave signal is 2 V, and the offset is 3 V.

III-4 Diagram of frequency and time responses in Matlab

III-4-a Time simulation

Run the Matlab software to simulate the model obtained in question III.3.a.

To this end, in Matlab, *open Simulink* by typing:

```
>>simulink
```

The Simulink window opens and reveals a series of libraries of different functions (see figure 8). If necessary, you may minimize this window and keep it in the top right-hand corner of the screen.

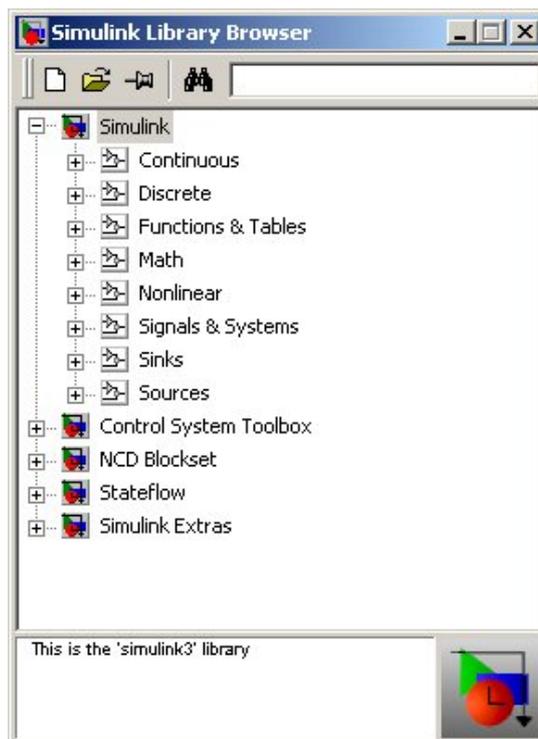


Figure 8 – Simulink control window and directory in which the blocks to be used are stored

Now open your work area by clicking the blank page icon in the Simulink window.

The aim is to simulate the system's response to a step signal with an amplitude of 1 (max. control)

$$\Omega_s(p)/U(p) = K/(1 + \tau p)$$

which characterizes the dynamic behavior of the operational part; the values of the parameters K and τ are those determined in question III.3.a.

Create the simulation diagram in the working window ("Untitled").

To this end, open the "Sources" directory in the Simulink window. Select and drag the **Step** block into the working window (left-click on the mouse and hold while moving it to the desired location, then release).

In the same way, move the following elements into the working window:

- one transfer function (Transfer Fcn), dragged from the "Continuous" directory
- and one oscilloscope (Scope), from the "Sinks" directory.

Line up these elements in the correct order, and *make all of the connections* using the mouse (press the left-hand button inside the small output triangle for a block, drag it by holding the button down until you reach the small input triangle for the next block, and then release it).

Then program each element by double-clicking its icon:

- with the Step time of 0 s for the moment at which this step is applied, its Initial value of 0 and its Final value of 1. Validate by clicking "OK".

- with the transfer functions, by entering the numerators and denominator.

Finally, program the parameters chosen for the simulation, by clicking the 'parameters' command in the 'Simulation' menu in the "Untitled" window: select Ode45 (this is the digital integration method), and enter the simulation finish time that is compatible with the experiment being carried out (Stop time). Do not modify the other values and validate by clicking "OK".

Then run the simulation ('Start' command in the 'Simulation' menu), and double-click the oscilloscope icon to open the viewing window.

Compare the response obtained with the measurement recorded for question III.2.a.

III-4-b Frequency identification

Using Matlab software, the aim is to analyze the **frequency response** of the system described by its transmittance:

$$\Omega_s(p)/U(p) = K/(1 + \tau p)$$

the values of the parameters K and τ are those determined in question III.3.a.

This transmittance corresponds to the transmittance which characterizes the behavior of the motor that has just been studied (there may, however, be significant variations in the numerical values of the coefficients).

In order to facilitate the correction of the program, it will be inputted in a text file with the extension.m (executable by matlab), which will be opened via the **File** menu and then by selecting **New Script**.

The transfer functions are inputted by entering the polynomials of the numerator and denominator separately. The square brackets show that it is a polynomial, with its coefficients always written **in decreasing order of the powers of the Laplace variable p** (referred to as **s** in Matlab):

```
num1= K;
den1=[ \tau,1];    (the comma may also be replaced by a space)
```

Always adopt the habit of putting a semicolon at the end of each line, in order to prevent Matlab from displaying the results of intermediate calculations and thus give a clearer printout of the lines of code selected at the end of the entry.

The gain and phase can be calculated using the following series of instructions:

```
w=logspace(-1.1,100);    % ligne 1
[g,p]=bode(num1,den1,w); % ligne 2
g_db=20*log10(g);        % ligne 3
```

Line (ligne) 1 allows for the definition of the angular frequency range over which the gain and phase calculation is carried out. In fact, in the variable w , the instruction $w=\text{logspace}(n,m,q)$ reverses q points of the angular frequency 10^n rad/s at the angular frequency 10^m rad/s, with a logarithmic distribution over this angular frequency range.

Line 2 reverses the modulus and phase of the system in the variables g and p respectively.

Line 3 allows for the conversion of the modulus into gain expressed in dB.

The gain and phase graph can be created using the following series of instructions:

```

figure;                                % ligne 1
subplot(211);                            % ligne 2
semilogx(w,g_db);                        % ligne 3
xlabel('Pulsations (rad/s)');          % ligne 4
ylabel('Gain (dB)');                   % ligne 5
title('Diagrames de Bode');            % ligne 6
grid;                                   % ligne 7
subplot(212);                            % ligne 8
semilogx(w,p);                          % ligne 9
xlabel('Pulsations (rad/s)');          % ligne 10
ylabel('Phase (deg)');                 % ligne 11
grid;                                   % ligne 12

```

Line (ligne) 1 allows for the opening of a new figure.

Line 2 allows for the division of the window into several parts. The `subplot(mnq)` function divides the screen into a table of m rows, n columns and selects box q of this table as the designated location for the insertion of any new graph.

Line 3 allows the gain to be plotted on a semi-logarithmic scale.

Line 4 adds a key to the x-axis of the graph.

Line 5 adds a key to the y-axis of the graph.

Line 6 adds a title to the graph.

Line 7 adds a grid to the graph.

These instructions are repeated on lines 8 to 12 for the phase diagram graph.

Print out the curves obtained, accompanied by the corresponding program lines and compare them with the curves obtained in paragraph III-3-b.

A5.1 / A5.2

MATLAB CAD, SIMULATION OF DYNAMIC SYSTEMS

The aim of this first Practical Assignment (PA) is to familiarize students with Matlab software and, in particular, with certain functionalities that are useful for resolving automatic control engineering problems.

Before starting the session, it is essential to have read through and worked on the course handouts and your notes made during the lectures.

1. Using Matlab: preliminary questions

Before going any further, read the introductory annex on Matlab below.

- Enter the following transfer function in Matlab:

$$G(p) = \frac{p^2 - 2p + 2}{p^3 + 4p^2 + 9p + 10}$$

The poles of a transfer function $G(p)$ entered beforehand are calculated using the `pole` function:

```
>> P=pole(G)
```

where `P` is a vector containing the list of poles for $G(p)$.

2. Frequency responses in Matlab

Bode diagrams and Nicols plots can be plotted using the `bode` and `nichols` functions in the following manner:

```
>> bode(G);  
>> grid on;  
>> nichols(G);  
>> grid on;
```

The range of angular frequencies used is chosen automatically according to the system, but it can be set manually by means of a second argument.

- Plot the Bode diagram and Nichols plot of a:
 - stable first-order system, e.g. $\{K = 10; \tau = 5\}$;
 - stable second-order system, of damping $0 < \zeta < 1$, for example $\{K = 10; \zeta = 0.1, \omega_n = 5\}$.
- Check that these diagrams show the expected trends.

3. Representation of a feedback system

Considering the closed-loop system shown in figure 1.

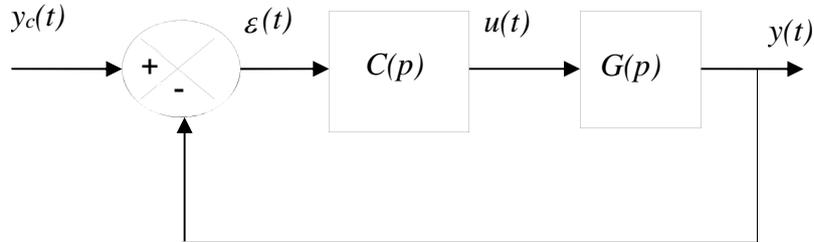


Figure 1 - Feedback system

Using the feedback function, determine the expression of the transfer function $\frac{Y(p)}{Y_c(p)}$ for:

- $G(p) = \frac{p^2 - 2p + 2}{p^3 + 4p^2 + 9p + 10}$ and $C(p) = 10$;
- $G(p) = \frac{0.333p^2 + 4.333p + 10}{2.5p^3 + 5.25p^2 + 10.5p + 1}$ et $C(p) = 2 \left(1 + \frac{1}{10p} \right)$.

By calculating the poles of the closed loop system, assess the stability of the loop in both cases.

4. Control of a 3rd order system

Considering a system described by the transfer function $G(p)$ given by:

$$G(p) = \frac{150(1 + p/400)}{(1 + p/10)(1 + p/40)(1 + p/60)}$$

This is a closed-loop control system according to figure 2.

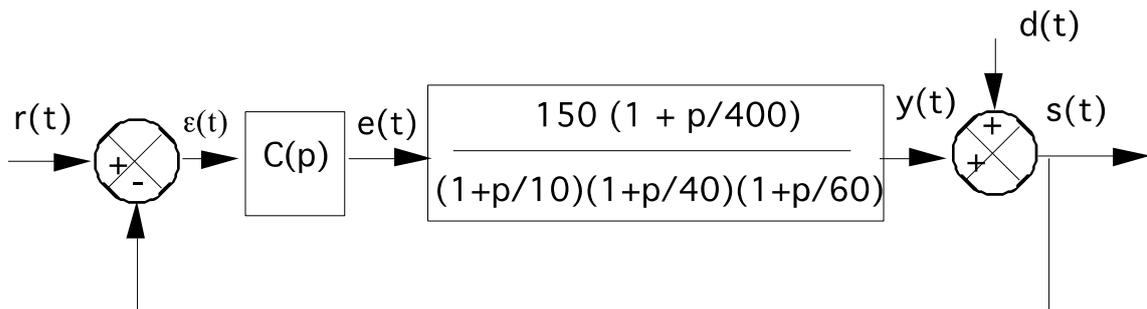


Figure 2 – Closed-loop system

Using Matlab, plot the Bode diagrams and Nichols plot of the function $G(p)$.

The proportional controller consists of a comparator and an amplifier with adjustable gain. Therefore, the system represented in figure 2 by the transmittance block $C(p)$ is reduced to a real number C representing the amplifier gain. In the sequel, the following value of C_1 will be used:

$$C_1 = 0.02 \quad \text{and} \quad C_1 = 0.04$$

- a. Using the feedback function and for the two previous values of C_1 , calculate the transfer function of the closed-loop system (transfer function $S(p)/R(p)$). Plot the two corresponding gain diagrams. On these two curves, record the static gain, the resonance value and the bandwidth of the closed-loop system. State your conclusions regarding the changes in these values according to C_1 .
- b. In Simulink, perform a temporal simulation of the feedback system for the values of C_1 , using:
 - $r(t) = 2 * h(t)$;
 - $d(t) = -h(t-1)$. $h(t)$ refers to the Heaviside step function. For the simulation, $0.01s$ will be chosen as the maximum value of the calculation increment, with a simulation period of $2s$.

On the simulation curves obtained:

- record the static error before and after the appearance of the disturbance;
- measure the response time at $\pm 5\%$ (before the disturbance);
- measure the rise time (before the disturbance);
- measure the first overshoot (before the disturbance);
- theoretically check the static error values (with and without disturbance).

5. Annex: Introduction to Matlab

When Matlab is run, a control window is displayed. This window contains the command prompt in which the instructions can be entered. The description of each instruction can be obtained by entering the `help nom_instruction` command. In addition, if an instruction is followed by a semicolon, its result is not displayed in the control window.

Entry of variables and basic operations

In contrast to compilable programming languages such as C and C++, the type of variables is not requested in Matlab. A variable is declared and assigned by means of a unique instruction:

```
>> a=5;
```

This variable can then be used in an expression:

```
>> 2*a+3
```

```
ans =
```

```
13
```

A complex number is entered by means of the predefined variable `i` or `j`:

```
>> b=4+3*i
```

```
b =
```

```
4.0000 + 3.0000i
```

The modulus and argument of the complex number `b` are calculated using the functions `abs` and `angle`

```
>> c=abs(b)
```

```
c =
```

```
5.0000
```

A row vector can be entered in the following manner:

```
>> x=[1 2 3]
```

```
A =
```

```
1      2      3
```

and a column vector by:

```
>> x=[10;11;12]
```

```
x =
```

```
10  
11  
12
```

It is important to note that Matlab makes a distinction between upper case and lower case letters with regard to the naming of variables. Variables must not contain punctuation characters or begin with a number.

Creation of scripts

A script is a text file containing a sequence of Matlab commands. This file must imperatively have the extension `.m`. A script can be executed by entering its name in the Matlab command prompt (without the extension `.m`). For example, the `monscript.m` file is executed by typing:

```
>> monscript
```

in the command prompt. So that it can be executed, the script must be situated in the Matlab current directory (checkable with the `pwd` command). If this is not the case, the directory can be changed using the command `cd` or with `File->Set Path`. By default, the current directory corresponds to `work`

Instructions used for the manipulation of files are listed in the following table:

<code>cd, pwd</code>	display the reader and the current directory
<code>cd c :</code>	go to the logical unit <code>c</code>
<code>cd ..</code>	go up a level in the tree structure (parent directory)
<code>dir</code> or <code>ls</code>	display the list of files and directories contained in the current directory
<code>pwd</code>	display the current directory path
<code>save</code>	save the data in a file in the memory (<code>.mat</code> extension)
<code>load</code>	loads the data contained in a file in the memory

Graphic functionalities

Matlab possesses a large number of functions that can be used to produce 2D or 3D curves. These functions accept data in the form of vectors or matrices as inputs. The axes are automatically scaled according to the data to be plotted.

The most frequently used command is `plot`. It generates a display of the components of a vector or of the columns of a matrix. In this way, if `y` is a vector, `plot(y)` plots a diagram with the vector indices (from 1 at the dimension of the vector) on the x-axis, and the values contained in the vector on the y-axis.

If two vectors of the same dimension are specified as inputs, `plot(x,y)` displays the components of the vector `y` on the y-axis according to the values of `x` on the x-axis.

The style of the curves (color, line, etc.) is specified by means of a third parameter in the form of a character string.

Finally, the `xlabel`, `ylabel` and `title` functions are used to state the sizes on the axes and add a title to the figure. The `legend` function is used to add a key if multiple curves are plotted.

The basic graphical functions are shown in the following table:

<code>plot(x,y)</code>	plot y according to x
<code>plot(x,y,'s')</code>	the variable s is used to choose the type of plot and its color
<code>plot(t,x,t,y)</code>	plot x and y according to the index t on the same graph
<code>grid</code>	add a grid in the background of the graph
<code>xlabel('titre')</code>	add a title to the x-axis
<code>ylabel('titre')</code>	add a title to the y-axis
<code>title('titre')</code>	add a title to the figure
<code>legend('courbel','courbe2')</code>	add a key to each curve
<code>figure</code>	open a new graphical window
<code>hold on</code>	superimpose several figures
<code>subplot(m,n,p)</code>	divide the current graphical window into m rows and n columns and display the graphic according to zone p
<code>axis</code>	establish the viewing intervals for the axes of the figure
<code>clf</code>	delete the contents of the graphical window

Declaring a transfer function in Matlab

A transfer function

$$G(p) = \frac{b_m p^m + b_{m-1} p^{m-1} + \dots + b_0}{a_n p^n + a_{n-1} p^{n-1} + \dots + a_0}$$

can be stored in the form of a single object in Matlab using the function:

```
>> G=tf([b_m b_{m-1} ... b_0],[a_n a_{n-1} ... a_0])
```

where the parameters a_i and b_j are the coefficients of the numerator and the denominator.

Simulation of dynamic systems in Simulink

Simulink is a simulation software program that is especially used for calculating and viewing the temporal changes of different magnitudes which are representative of a dynamic system. The model of the system is entered in graphical form by means of elementary blocks found in libraries. The layout of these blocks in relation to one another constitutes a block diagram of the system.

The calculation of the temporal changes of the different signals requires the resolution of differential equations. These differential equations are resolved using numerical integration methods based on the developments of Taylor, or implicit methods such as Runge-Kutta.

The signals obtained can be viewed directly in Simulink or be stored for exploitation in Matlab.

Simulink can be launched by entering the following command:

```
>> simulink;
```

A window showing the different libraries is then displayed. Each library can be explored by double-clicking it.

A simulator can now be created by clicking the `File/New/Model` menu. The block diagram is then obtained by dragging and dropping items from the elementary blocks contained in the libraries into the simulator window. The parameters of the blocks can then be modified by double clicking inside the simulator window. The diagram can be saved with an `.mdl` extension using the `File/Save` menu.

The essential elements are:

- the sources, contained in the `Sources` library;
- the viewing elements, contained in the `Sinks` library;
- items for the representation of continuous-time dynamic systems in the `Continuous` library.
- mathematical operations.