Politecnico di Milano
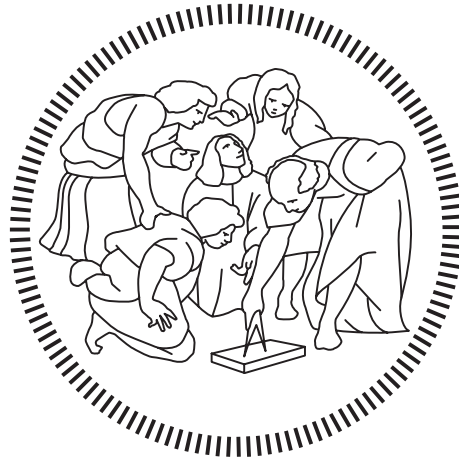
SCHOOL OF INDUSTRIAL AND INFORMATION ENGINEERING

Master of Science – Automation and Control Engineering

Control of a glue spreading system of a woodworking machinery

Supervisor

Prof. Luca Ferrarini

Co-Supervisor

Prof. Soroush Rastergarpour

Candidate

Alessandro Rinaldi – 927963

Acknowledgments

Sommario

Al momento la produzione di pannelli di compensato nell'azienda che ha commissionato il lavoro, la Fabinmec Europe S.R.L, viene fatta con una macchina incollatrice che non presenta un controllo automatico. Il problema che ne deriva dipende dalle caratteristiche chimiche della colla usata per incollare uno sopra l'altro i vari fogli che nel complesso formano il vero e proprio pannello. La colla infatti è un fluido non newtoniano e quindi ha una densità e una viscosità variabili in base ai cambiamenti di temperatura. Durante il normale funzionamento della macchina la colla che circola nell'impianto acquista calore e quindi la quantità di colla che viene stesa sui vari fogli di legno non è costante durante il giorno e ciò rischia di compromettere gli standard tecnici che il prodotto finito deve soddisfare. Lo scopo di questo progetto è di costruire un modello fedele al caso reale che controlli questo comportamento e che lo corregga automaticamente in corso d'opera.

Abstract

Nowadays the production of plywood panels in the company who commissioned the work, the Fabinmec Europe S.R.L, has been done by using a gluing machine that is not provided with automatic control. The problem that arises also depends on the chemical features of the glue used to stick on top of each other the sheets which as a whole form the actual panel. The glue indeed is a non-Newtonian fluid and so it has density and viscosity variables with temperature changes. During the normal work of the machine, the glue circulating in the plant achieves heat, so the amount of glue that is spread on the wood sheets is not constant in the day and this could compromise the technical standards that the finite product has to guarantee. The aim of this project is to build a model faithful to the real case which controls this behavior and fixes it automatically at runtime.

Table of Contents

List of Figures

Chapter 1

Introduction

1.1 Field of research

The research performed in this thesis is collocated in the field of Automatic Control. The control object is highly non-linear and time-varying, hence it is difficult to establish a precise mathematical model [1]. This work is related to a project in collaboration with Fabinmec Europe S.R.L, which operates in the field of woodworking machinery; in particular, they produce plywood panels [2]. This type of wood is built by gluing sheets of wood overlapped. Glue is a non-Newtonian fluid and so, it presents physical and chemical features not constant in time; in particular, it has density and viscosity that are variables with temperature. So, the control of the temperature is a crucial aspect to consider within this context.

1.2 Objective and structure

The aim of this thesis is to build a model that is faithful to the real case of a glue spreading machine used for the production of plywood panels, that controls and optimize the amount of spread on each wood sheet given a setpoint and considering the changes of temperature.

The thesis is articulated in four chapters: in the first one there is a scheme of the plant within a description of the various components; the second one is the control part in which it is explained the model in all its features; in the third one there are some graphs to show the results of the model; the last one is a conclusive chapter and provides connections between the model written in MATLAB & SIMULINK [3] and the implementation of some features industrial language of PLC by using the software CODESYS [4].

1.3   Workflow

The plant work in this way: a tank, initially empty, is filled with glue; once turn on the machine a pump starts to withdraw the glue from the bottom of the tank; then, flowing through the pipes, the glue arrives at the extruder and starts to come out passing through many holes placed in a straight line that is between two conveyor belts over which the wood sheets flow; the glue that is not spread on the wood is recycled and comes back to the tank.

If the level in the tank is lower than the minimum, it must be refilled with fresh glue since the level reaches the maximum.

The part dedicated to the water for washing is just to know that in the real case there is also that feature but is not relevant for our study.
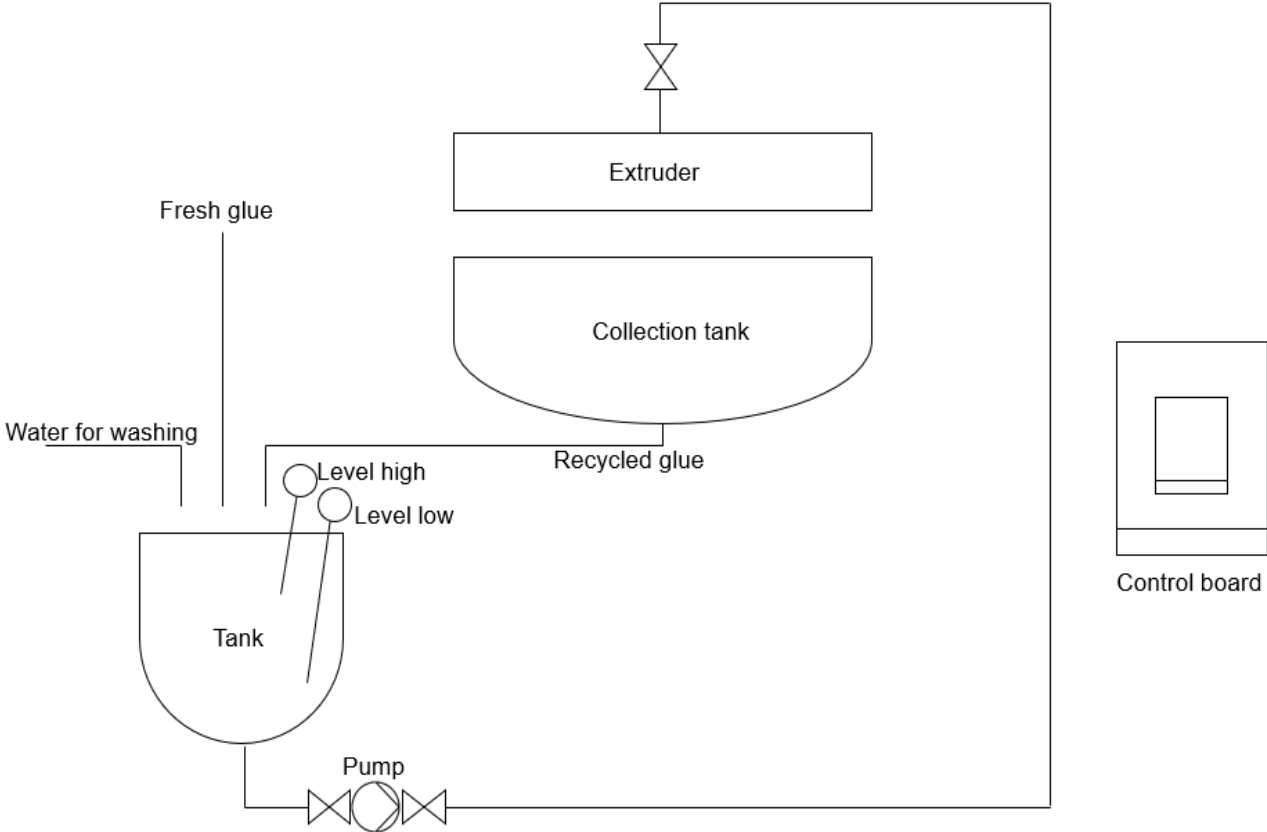
Figure 1. Workflow [5]

1.4 Components

1.4.1 Tank

In the tank both the fresh and the recycled glue converge and the glue that is withdrawn by the pump goes out. The capacity of the tank is 50kg so if we consider the glue with a density equal to its medium value the capacity is 35.71 liter. To prevent that the tank gets empty there are two indicators of level, one for the maximum and one for the minimum; when the level is lower than the minimum value the tank has to be refilled with fresh glue to retake the level to the maximum level.

1.4.2 Pump

The pump is a volumetric one and it withdraws the glue from the bottom of the tank and pushes it through the pipes until it reaches the extruder and flows out. Knowing the setpoint we can regulate the power of the pump and so the number of rounds per minute in a way that the actual glue moved by the pump and then spread on each sheet of wood is more similar as possible to the desired value (setpoint).

To see some more detailed model and to investigate further on the pump I visited the website of the provider of the pump [6].

I DATI RIPORTATI SONO RIFERITI A:
 Temperatura: 40 °C (104 °F)
 Viscosità: 6 °E
 Pressione: 3 bar
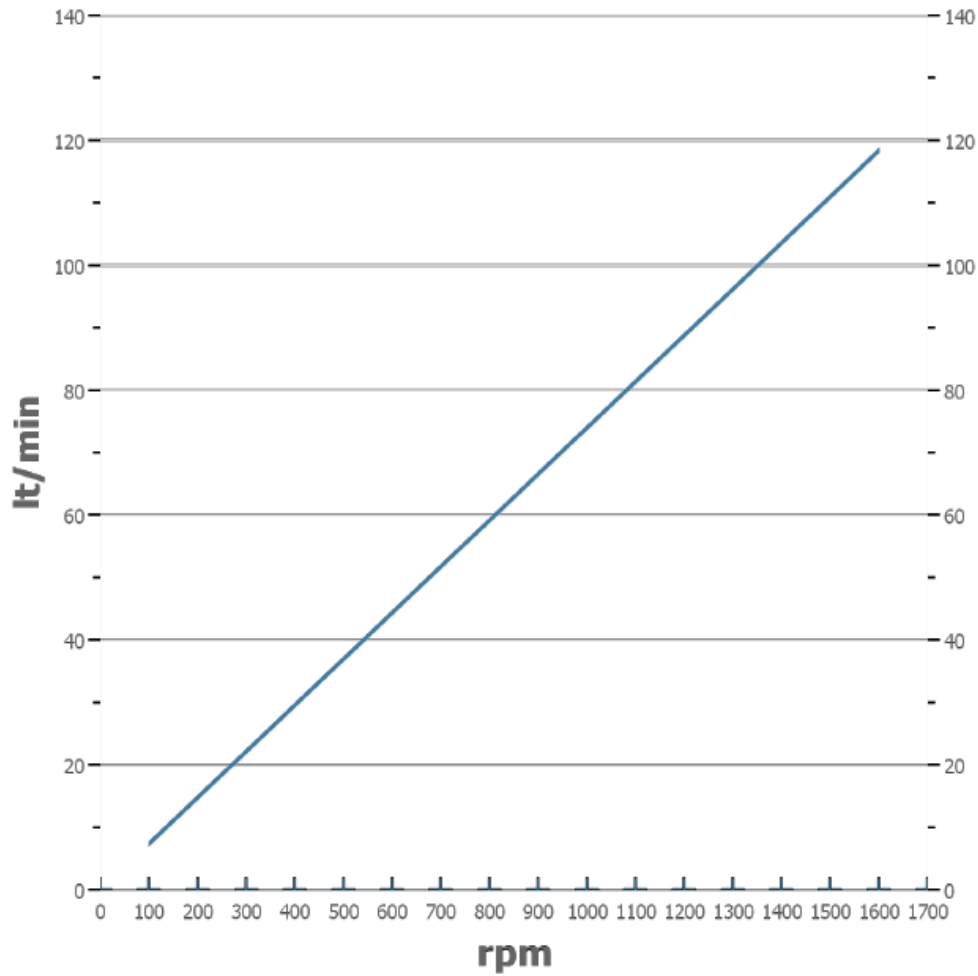
Figure 2. Performance of the pump at a fixed temperature, viscosity, and pressure [7]

This graph is provided by the company Pompe Morgan that produce the pump of the real case [6]; it is a useful graph because it links the round per minute, that is the variable we want to manipulate, to the liter per minute at fixed values of pressure, density, and viscosity.

1.4.3 Extruder

The collector is composed of thousand holes that allow the glue to come out; for simplicity of modeling, we consider all the holes as only one which is the sum of everyone.

Figure 3. Real machine's extruder

This picture was taken by me during the test of the real machine; it shows well how the distribution of the glue is done. Glue arrives at the extruder by a branch of the pipes and the extruder is composed of many holes that put out the same amount of glue. Wood sheets pass right under those holes and flow along the green conveyor belt.

1.5 Glue

The glue is the main issue of the case; it is a non-Newtonian fluid, so it presents particular features such as density and viscosity variable with the changes of temperatures. The glues used in the process are two: urea and phenolic. They present characteristics similar, so we did not consider this variability of the choice of the glue in our model. We have considered only one kind of glue which has a

density that varies in a range of 1.2÷1.6 kg/l and viscosity that varies in a range of 0.8÷1.2 mPas [8], [9].

In our model we also perform a control of the temperature of the glue in the tank; to do this we have simulated a sort of jacket around the tank through which it passes some fluid in such a way as to maintain the temperature of the glue inside the tank as close as possible to the desired value which is 20 degrees centigrade.

Chapter 2

Control Part

The control part is the part that took most of the time; it is the one in which I actually built the model of the plant. For this purpose, I used MATLAB & SIMULINK [3], a software useful for modeling, simulation, and analysis of dynamic systems.

The first thing to do when you have to model a plant is to have in mind how the plant works, so it was organized a meeting in Fabinmec Europe S.R.L [2] to test the gluing machine and to take measurements useful for the work.

Secondly, the company provides us the right dimension of all the components of the machine, like the capacity of the tank, the range of the two indicators of level, the dimensions of the wood sheets and of the extruder, the refill logic, and the datasheets of the glue and the one of the pump.
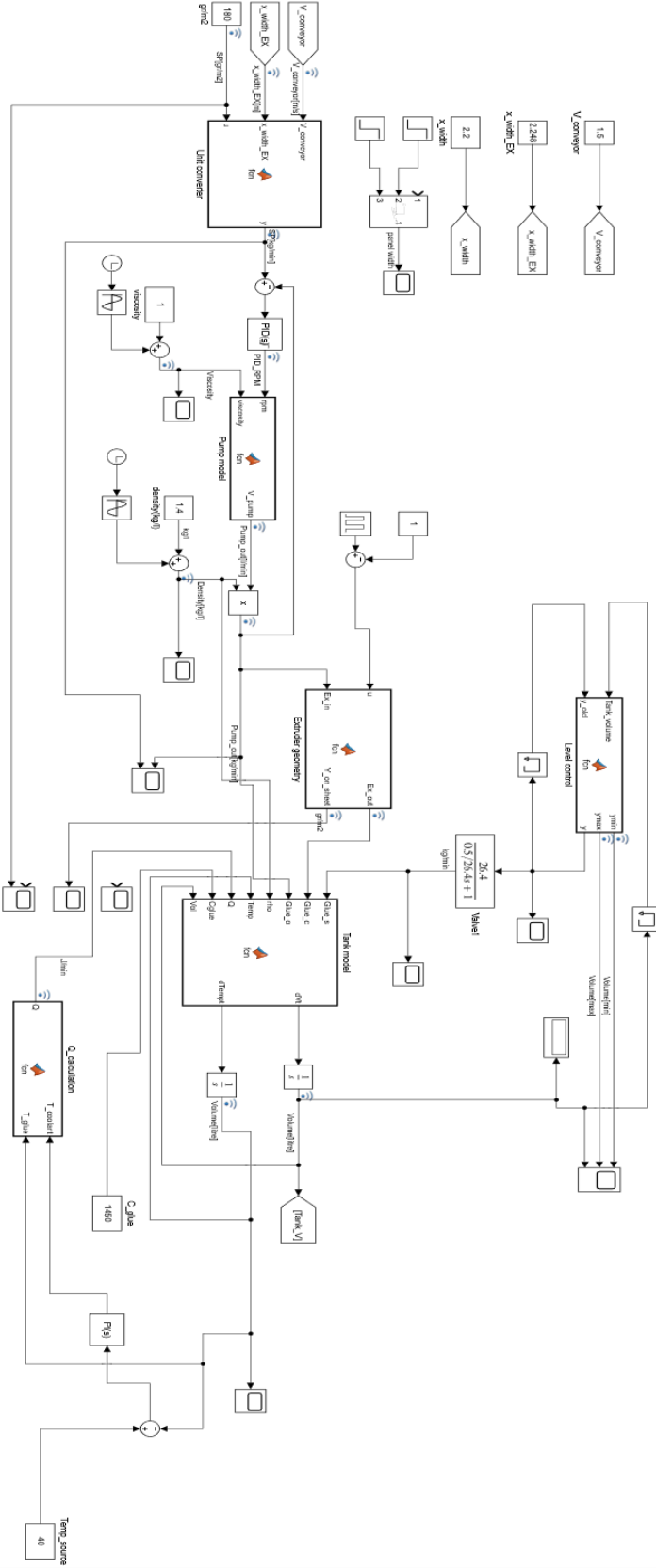
Figure 4. Complete Model

Now I'm going to split the model in order to analyze the single parts:

2.1 Unit Converter

The first block of the model is the unit converter, it's important because I know that the setpoint is 180 gr/m^2 but all the other units of measure are kg/min or l/min; so, knowing the speed of the conveyor, the width of the extruder and the setpoint, it calculates automatically the setpoint in kg/min.
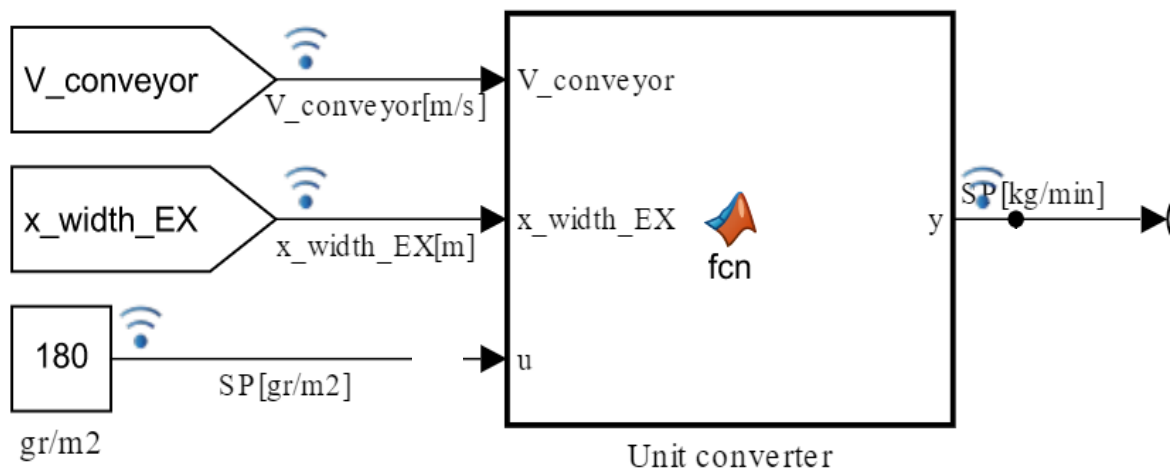
Figure 5. Unit Converter Block

2.2 Pump Model

By looking at Figure 2, the logic of the pump can be extrapolated; it depends on the viscosity and the round per minute; the output is the amount of glue per minute the pump moves in l/min. The problem, as already known, is that the viscosity is variable with temperature. Since there is no information about the correct relationship between viscosity and temperature, I simulated the variation during the day by adding to the medium value of the viscosity (1 mPas) a sinusoid with a day as period and amplitude 0.2 so I have different values of the viscosity but always in the range indicated. The round per minute is the variable I want to manipulate so I have to control it. The logic is that I want to increase or decrease the round per minute in a way to make the output of the pump as close as possible

to the setpoint already converted in kg/min at the previous step. So, I used a PID controller to regulate this variable; SIMULINK is very useful because it has the function of tuning, and automatically calculates the parameters of the controller (proportional, integral, and derivative coefficients) [3].

It's important to pay attention to the unit of measure, because the setpoint is in kg/min while the output of the pump is in l/min, so before the retroaction, the output must be converted by multiplying it for the density in kg/l. The logic I have adopted for calculating the variability of the viscosity is valid also for the density.

The formula retrieved to calculate the volume moved by the pump comes by looking at Figure 2 and is the following:

$$V_{pump} = VISCOSITY \cdot rpm \cdot 0.0733 + 1.66 \qquad (1.1)$$
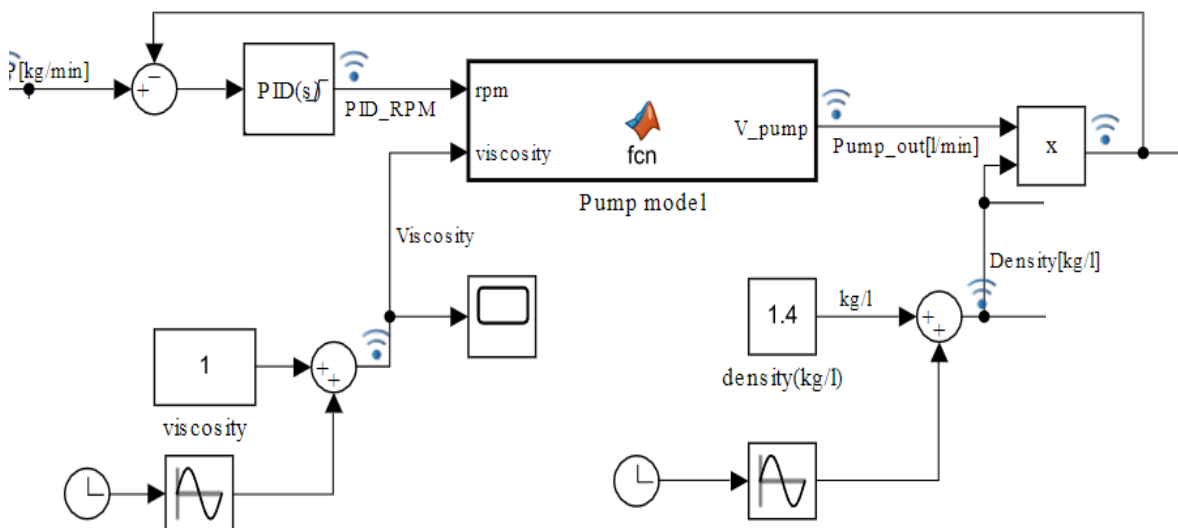
Figure 6. Pump Model, Viscosity and Density block

2.3   Extruder geometry

The next block of the model is the one in which it is calculated the amount of glue that does not fall on the wood and then is recycled; knowing the flow rate of the

extruder, the dimensions of the sheet incoming, the setpoint, and the amount of glue moved by the pump, it is easy to calculate the percentage that returns to the tank as recycled glue. I have created the variable u that simulates the alternation of the type of sheet incoming. In general, it turned out that for a small sheet with dimension 1.3m x 1.5m there is 91% of recycled glue (this data is obtained by the information provided by the company, which said that the extruder flow rate was equal to 29.7 kg/min and that the frequency of sheet per minute was equal to 7.5 unit per minute); at the same way for a big sheet with dimension 2.2m x 4.2m the recycled glue is equal to the 58% of the glue moved by the pump [10].
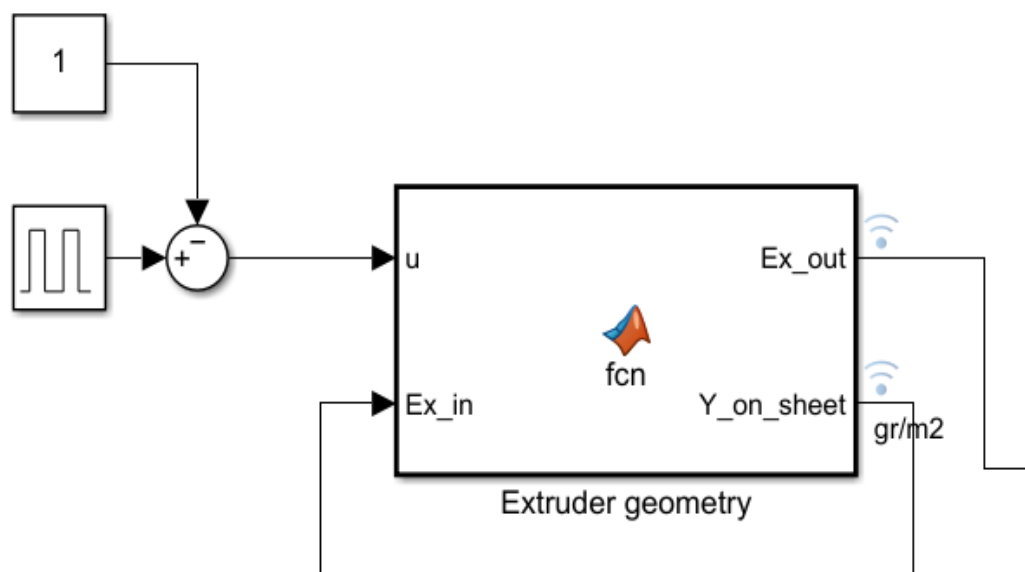
Figure 7. Extruder geometry block

2.4 Level Control

I focus now on the part dedicated to the control of the level of the glue in the tank. The task to satisfy is that the level must be between the two indicators level of minimum and maximum and if it reaches the lower, the tank must be refilled with fresh glue; this action of refilling has to be stopped when the level reaches the indicator higher. The block in question has two input and one output: it receives the current value of glue in the tank in a recursive way from the block "Tank Model" that I will analyze later and the information about the status of the refiller logic (if it is on or off); it gives as output a command to the refiller to turn on or turn off. This command will be the input of the block "valve 1" which represent

the refill logic. I know by the information provided by the company that the refilling of the tank is almost continuous; one action refills the tank of 6/7 kg every 15 seconds, so about 26.4 kg/min and this value is the gain of the refill logic that it can be seen as a transfer function [10]. So, the output of block "valve 1" represents the amount of fresh glue in kg/min that enters the tank.
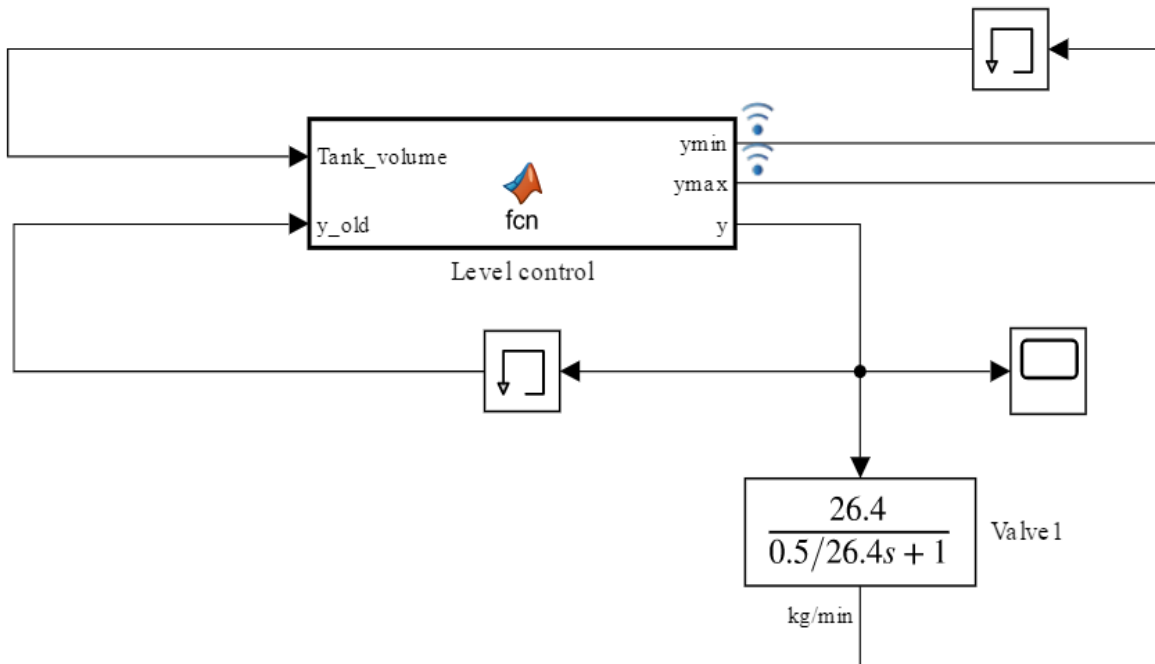
Figure 8. Level control and Refill logic block

2.5   Q Calculation

Parallel to the level control in this model is performed also a temperature control; the objective is to maintain the temperature inside the tank as close as possible to the operating temperature of 20 degrees. For this purpose, I need a function that calculates instantaneously the amount of heat to give to the tank to correct the temperature inside. The tank is wrapped by a jacket in which flows some fluid; we regulate the temperature of this fluid in a way to correct the temperature of the glue. The formula is $Q = 10 \cdot \Delta(T)$; the number 10 can be seen as a parameter; we set it to 10 and then with the tuning, all the others coefficients are set coherently; $\Delta(T)$ is the difference between the instantaneous value of the temperature of the glue in the tank (in a recursive way from the block "Tank Model" that I will

analyze later) and the temperature of the fluid in the jacket needed to maintain the operating temperature. To calculate this Coolant temperature, I used a PI controller that takes as input the difference of value between the instantaneous temperature of the glue in the tank and a Temperature fixed.
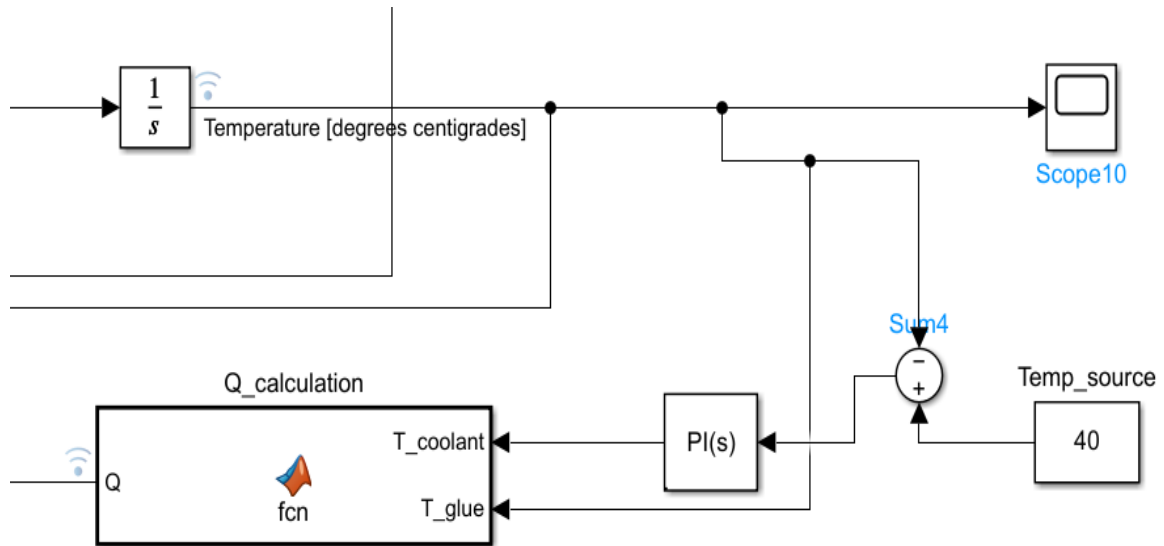
Figure 9. Q calculation block

2.6 Tank Model
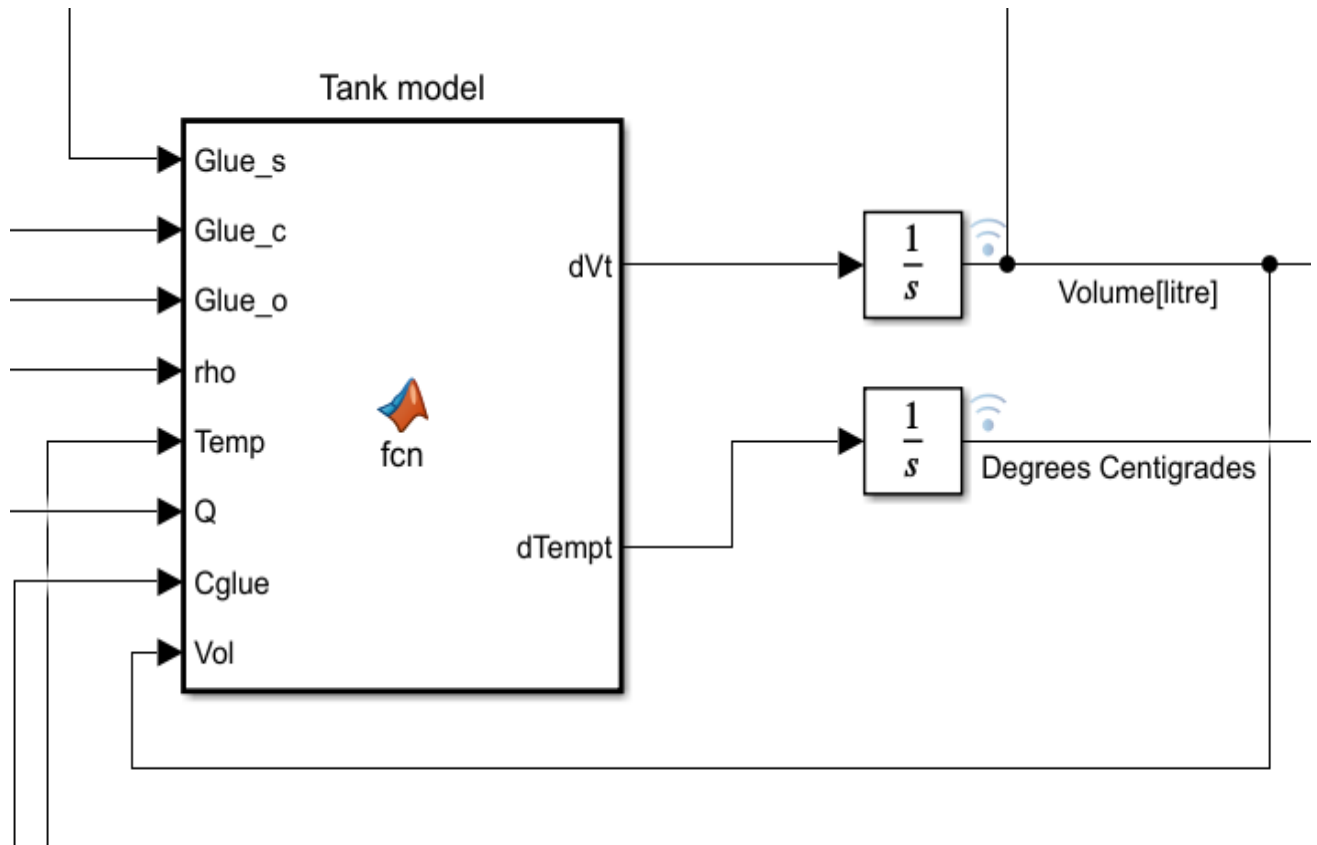
Figure 10. Tank model block

In this block, there is the calculation of the instantaneous volume of glue inside the tank and at the same time the calculation of the instantaneous value of temperature. As input, there are all the variables seen till now. Glue_s stands for Glue source, and it is the quantity per minute used to refill the tank when its level is lower than the minimum level and for this reason, there are moments in which the value of this quantity is zero. Glue_c stands for the Glue of the cycle, and it is the amount of glue that is recycled per minute. Glue_o stands for Glue out and it is the quantity of glue per minute that is withdrawn by the pump that arrives through the pipes to the extruder and for this reason, in the formula inside the block, this variable is subtracted from the sum of the other two. Rho is the density of the glue that I have already explained in 2.2. These first four variables

contribute to the calculation of the derivate of the volume of the glue in the tank. The formula is

$$\dot{V} = \frac{Glue_s + Glue_c - Glue_o}{\rho} \qquad (1.2)$$

However, this is the derivate of the volume, so I have to integrate this value to find out the volume of the glue in the tank (in liter since the density is in kg/liter).

At the same time is performed the calculation of the temperature made by using the other four inputs. Temp is the temperature of the glue in the tank at the previous step; it is connected to the output in a recursive way, so at the beginning is initialed to a proper value to make the code run. Q is the amount of heat to give to the tank in order to maintain the temperature of the glue at the desired value and is coming from the block "Q_calculation" at 2.5. Cglue is the specific heat of the glue; since I had no information about its variability, I took the value of the specific heat of a phenolic resin cast found on the internet, and I considered it as a constant [11]. Vol is the Volume of the glue in the tank calculated before and is connected here in a recursive way.

These second four inputs contribute to the calculation of the derivate of the temperature of the glue in the tank and the formula inside the block is

$$\dot{T} = \frac{Glue_s \cdot 20 \cdot C_{glue} + Glue_c \cdot Temp \cdot C_{glue} - Glue_o \cdot Temp \cdot C_{glue} - Q - C_{glue} \cdot Temp \cdot \rho \cdot \dot{V}}{\rho \cdot C_{glue} \cdot Vol} \qquad (1.3)$$

However, this is the derivate of the volume, so I have to integrate this value to find out the volume of the glue in the tank (in liter since the density is in degrees Centigrade).

By adding two scopes to our model, one for each output of this block, I have the possibility to check if the model created behaves correctly. It follows with some graphs that are shown in the next chapter.

Chapter 3

Results

This chapter discusses the results obtained by the model presented in Chapter 2. In the first section, there is a comparison between the setpoint we have, and the amount of glue actually moved, in both the measure units gr/m^2 and kg/min. In the second section is presented the behavior of the density and viscosity curves that contributes to the sensitivity analysis that will be presented in the last section. The third one briefly shows the refill logic. The last one presents two subsections, one for each of the main tasks of the project: the level control and the temperature control.

3.1   Setpoint

I know that the desired value to achieve is to distribute on the wood sheets an amount of glue equal to 180 gr/m^2. By looking at Figure 7 it can be seen that there is another output not explained in Chapter 2 because is useful not for the Control Part but just to compare that value associated with the amount of glue we are spreading on the sheets (in gr/m^2) with the desired value we have since the beginning.
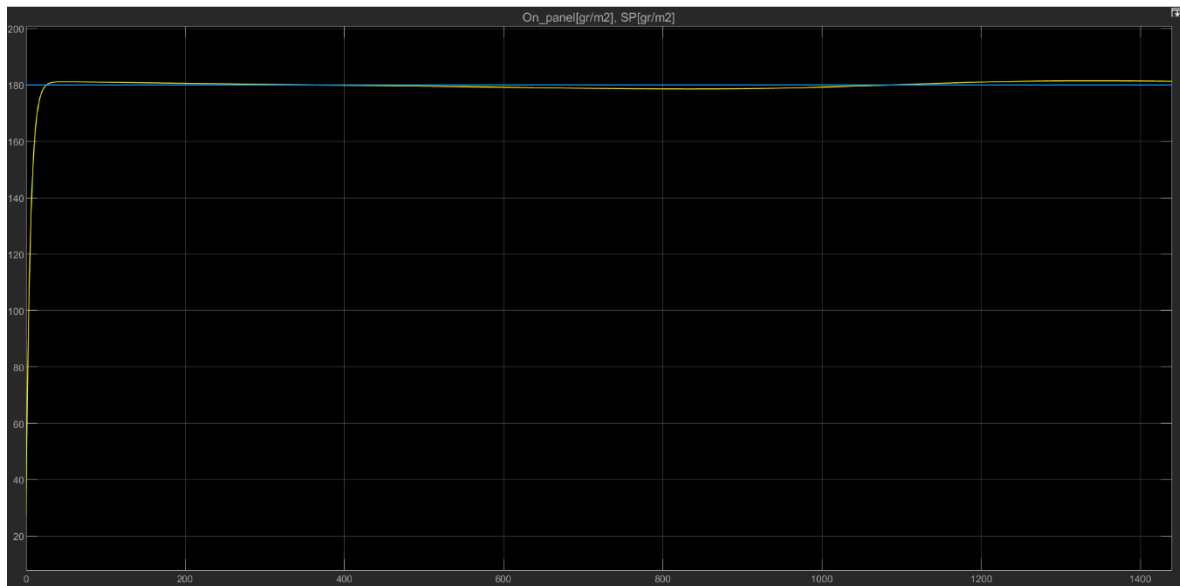
Figure 11. Comparison between Setpoint and Volume spread on each sheet in gr/m^2

As it can be seen the blue line is fixed at 180 gr/m^2 (our Setpoint) while the yellow line (associated with the glue spread on the panel) quickly reaches the desired value and then stays almost constantly there. So, our model works well in terms of the setpoint. Just to be sure about this sentence, since all the measurements are in kg/min or l/min, I compare the Setpoint converted after the block "Unit Converter", explained in section 2.1, with the amount of glue, withdrawn by the pump, that reaches the extruder and then gets spread on the wood sheets. So, after the adding of a scope, there is this new graph:
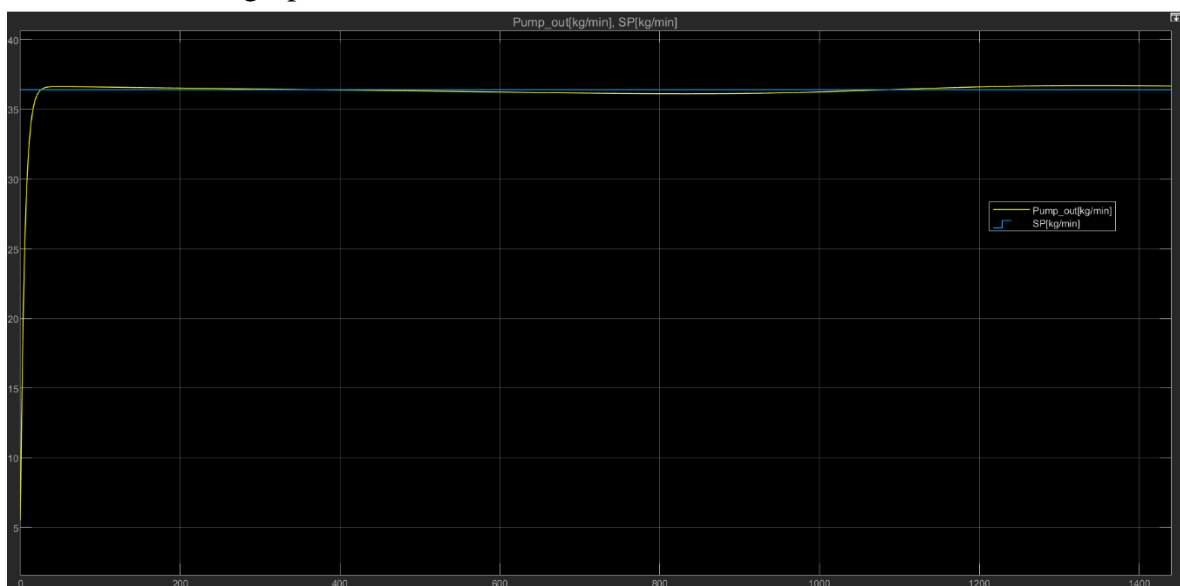
Figure 12. Comparison between Setpoint converted and volume moved by the pump in kg/min

Also this time the behavior of our model is correct in terms of Setpoint so I can move on to see the next results.

3.2   Density and Viscosity

This section is very simple and presents the values of density and viscosity in our model. I know that these physical quantities are not constant for the glue since it is a non-Newtonian fluid; it is known that their values change with respect to the temperature and in general when temperature increases density and viscosity decrease.

Adding a sinusoid to both the quantities' medium value with amplitude equal to half the difference between their maximum and minimum values, and with a period of 1 day, I have a rudimental simulation of the variation of density and viscosity during a day.

The next two graphs show right this behavior I have simulated.

The first is for the viscosity and it can be seen that the range of values seen in 1.5 is fulfilled (0.8÷1.2 mPas):
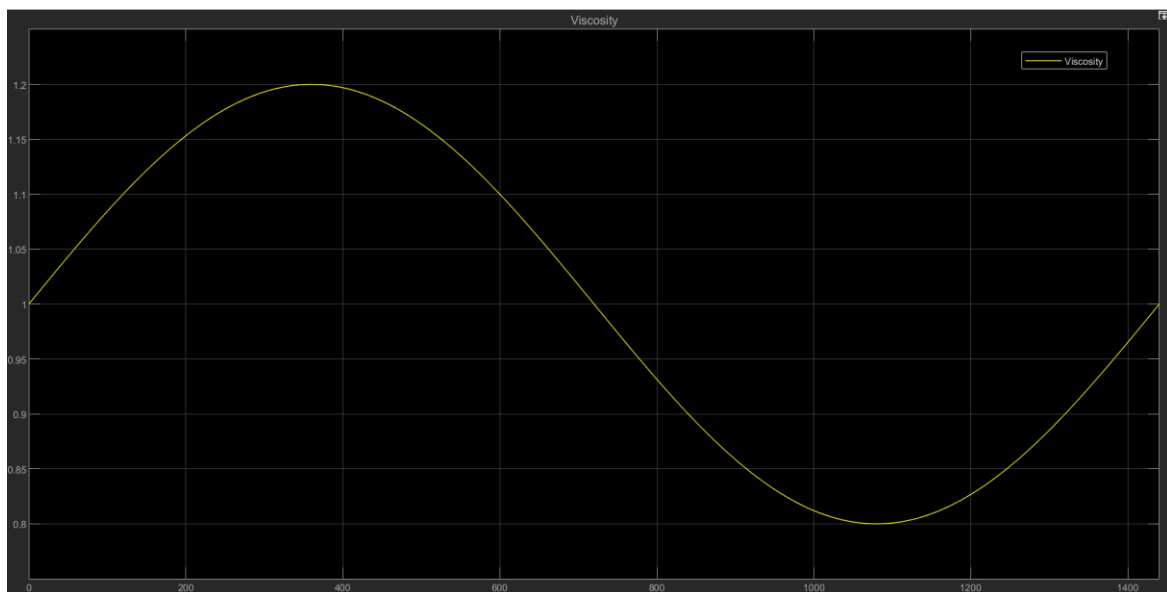
Figure 13. Value of viscosity in a period of one day

The second one is for the density and also here the range of values previously seen is fulfilled (1.2÷1.6 kg/l):
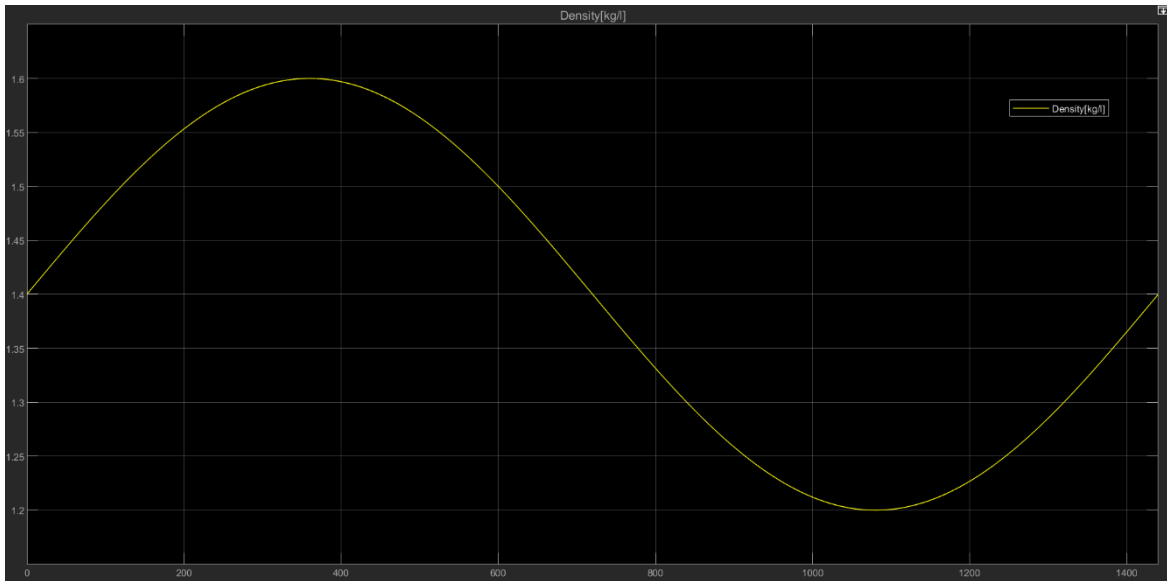
Figure 14. Value of density in a period of one day

3.3 Refiller logic

Now the last graph to analyze before the proper results of the model is the one of the refill logic. As already seen in 2.4, when the amount of glue in the tank is lower than a fixed level, the model gives a command that activates the block "Valve 1" that refills the tank in need.
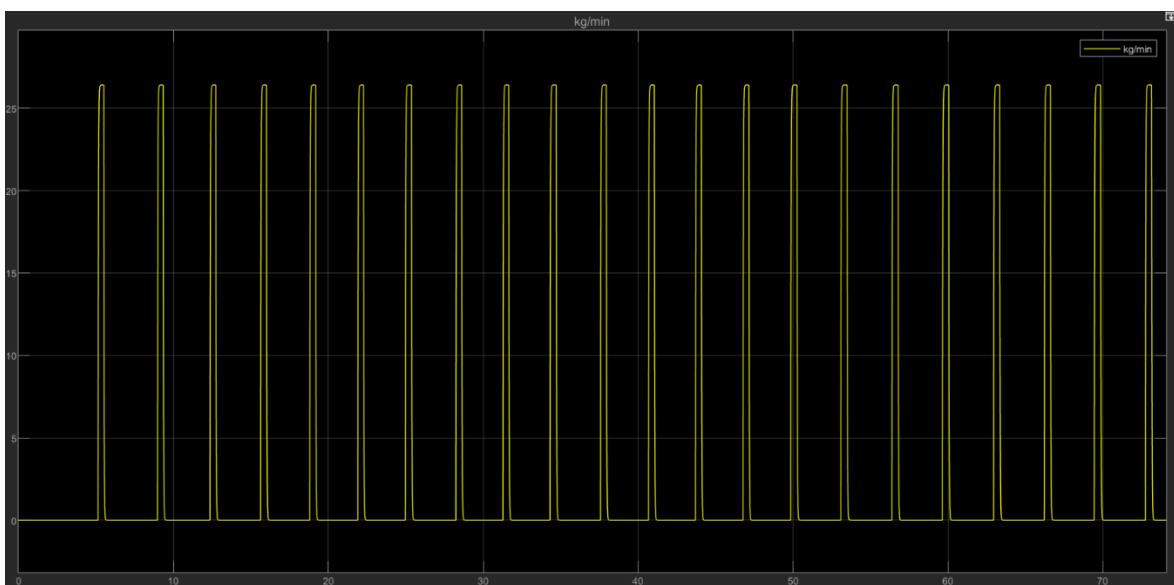
Figure 15. Refill logic

It can be seen that refilling is very quick. In fact, it is known that the tank is refilled by 6/7 kg of glue every 15 seconds which is equivalent to 26.4 kg/min. As soon as the level in the tank reaches the high indicator, the command that turns on the refiller is interrupted and this one goes instantaneously to zero waiting for another "turn on" command.

3.4 Control Tasks

3.4.1 Level Control

Now I am going to check if the model works correctly in terms of the level of glue in the tank. The task to fulfill is to maintain the level between two indicators of maximum and minimum. Of course, there will be variability during the simulation of a day of work due to the trick used to calculate the values of density and viscosity. I expect similar and accurate results in the neighbor of the medium values of density and viscosity (so at the beginning and at the half of the simulation), because of the periodicity of the sinusoid, while a little bit different results in the region corresponding to the two apexes of the sinusoid, so at the maximum and minimum levels of density and viscosity.
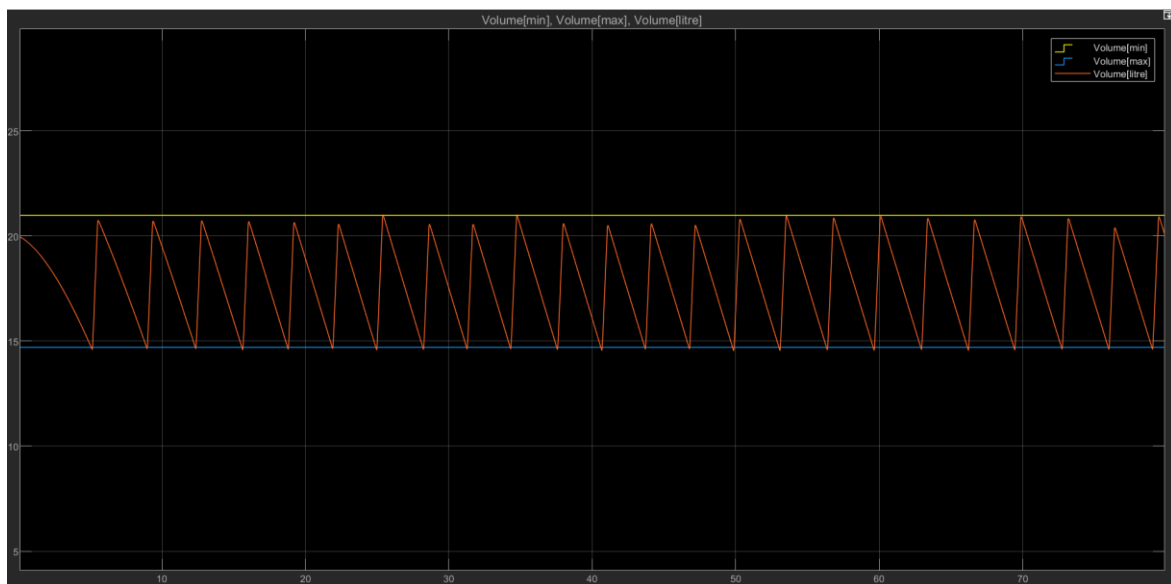
Figure 16. Level control at the beginning of the simulation

This picture shows the dynamics of the model at the beginning of the working day, so when density and viscosity are close to their medium value; I have initialized the

volume in the tank to 20 liters and it can be seen that the model behaves correctly with the refill logic almost always accurate.

The next graphs show the dynamics of the model in the neighborhoods of the most significant moments of the simulation.

Figure 17. Level control in a neighbor of the maximum value of density and viscosity

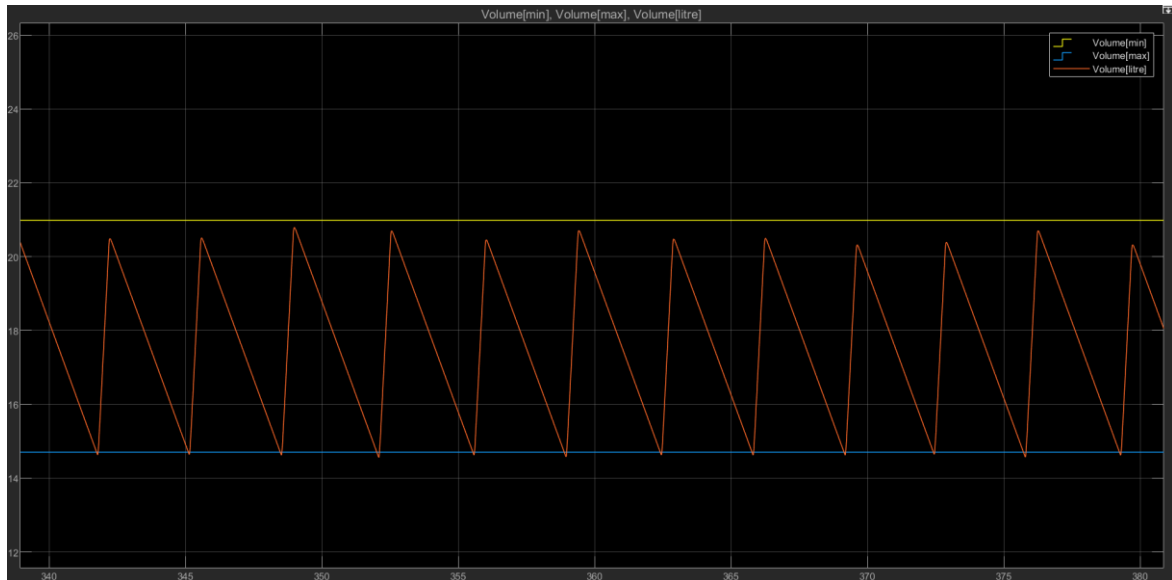This picture is taken when density and viscosity were at their maximum value; as I expected the model works a little bit worse than before with the refill logic less accurate.
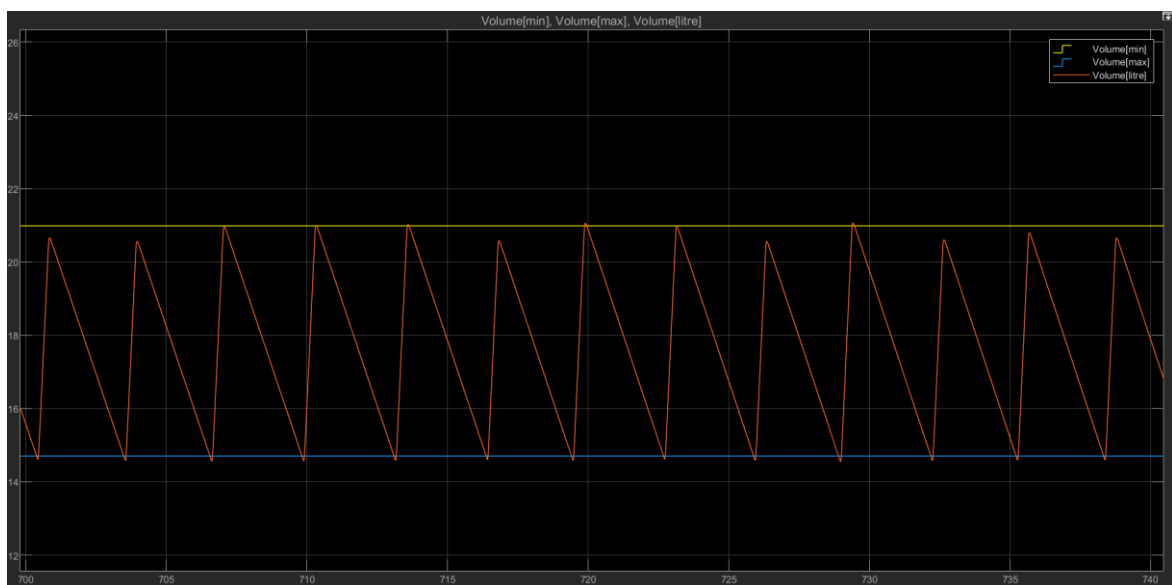
Figure 18. Level control at the half of the simulation

This one shows the dynamics when the sinusoid is in a neighborhood of π and in fact, the behavior of the model returns similar to the beginning when density and viscosity were at their medium values.
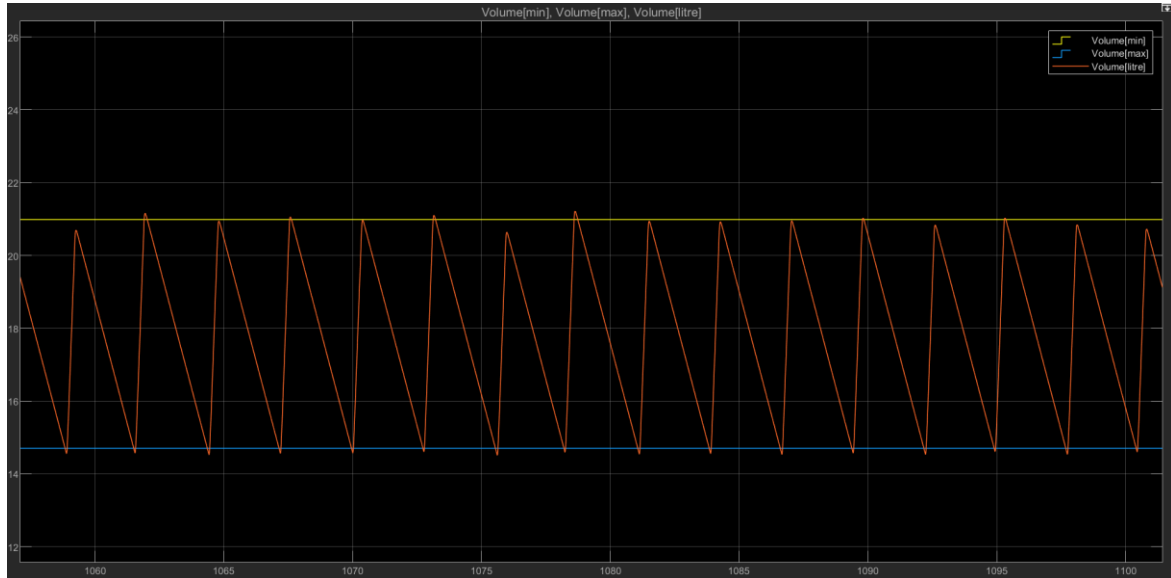
Figure 19. Level control in a neighbor of the minimum value of density and viscosity

This last one is taken in a neighborhood of the minimum level of density and viscosity, and we can see that the model behaves even better than in Figure 16 and in Figure 18.

So, I can conclude that the system works better when density and viscosity are low, and this can be converted in the sentence: "the system works better at high temperatures".

3.4.2 Temperature Control

This section shows the behavior of the system from the temperature point of view. I have imagined that the tank containing glue is surrounded by a jacket in which passes some kind of fluid in order to keep the temperature as close as possible to a fixed value that we set to 20 degrees Centigrade.

The fluid inside the jacket flows at a constant flow rate, we modulate its temperature to keep the glue close to 20 degrees Centigrade.
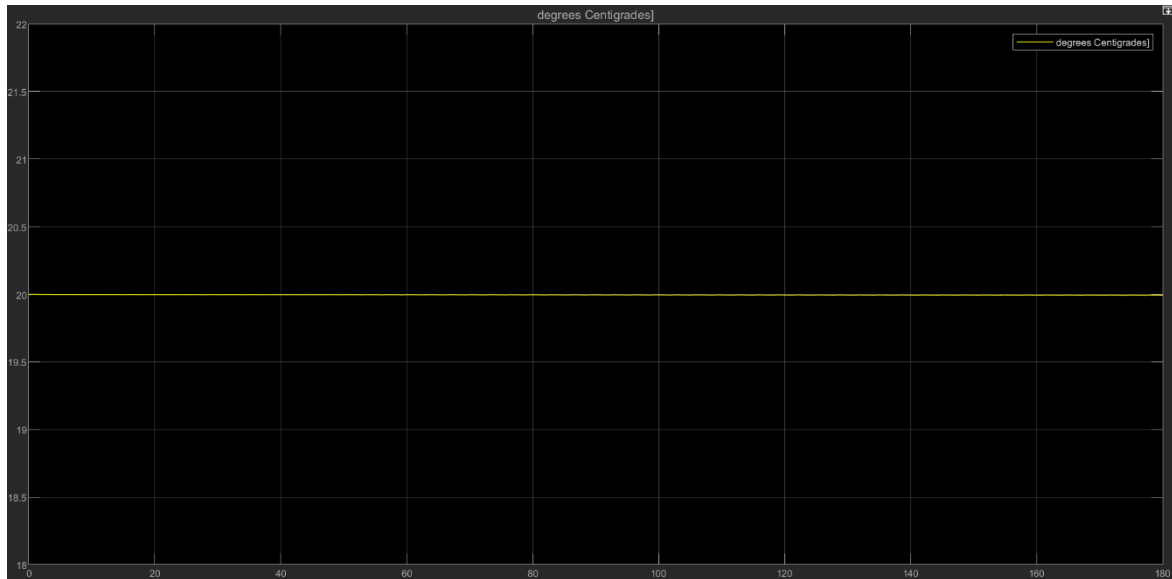
Figure 20. Temperature Control

As it can be seen, the temperature control works well, in fact, it stays constant at the desired value.

In the next picture, there is a zoom-in of Figure 20 to better view the goodness of the temperature controller.

Figure 21. Temperature control zoomed-in

As shown in this graph, the temperature of the glue tends to go down, but I constantly give it heat in order to bring it back close to the desired value.

Chapter 4

Implementation of Controllers in PLC

This chapter presents the implementation in the PLC programming language of the controllers simulated in SIMULINK. To do this I used C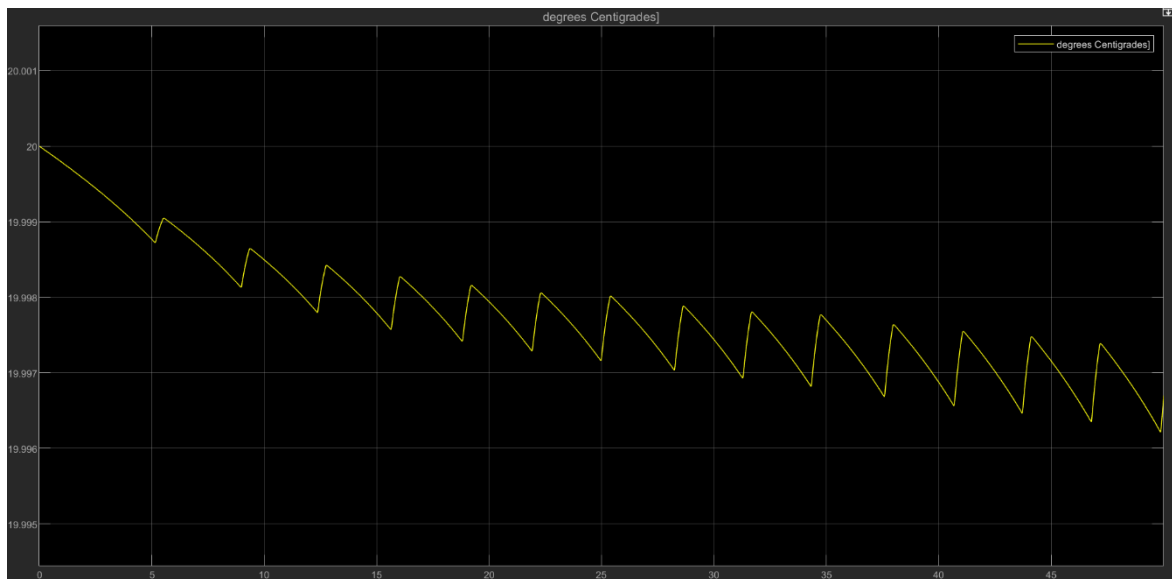ODESYS, a platform-independent development environment that is compatible with PLC hardware and many other automation components [4], [12].

A Programmable Logic Controller (PLC) is an industrial computer control system that continuously monitors the state of input devices and makes decisions based upon a custom program to control the state of output devices [13].

The most popular types of PLC programming languages are:

- Ladder Diagram (LD)
- Sequential Function Charts (SFC)
- Function Block Diagram (SBD)
- Structured Text (ST)
- Instruction List (IL)

In order to implement the controllers for Pump, Level control, and Temperature the language chosen is Ladder Diagram because this language features an excellent graphical representation based on well-understood circuit design concepts [14].

In parallel to the main program written in Ladder Diagram, for the realization of the PI controllers of the pump and of the temperature control, I have implemented some functions to be called in the main program. I have written these functions in Structured Text because I found it easier since I did not care to also have a graphical representation for the single function, and it allows me to do the same things of LD in much less space [15].

4.1 Level Control

To implement a controller for the Level in the tank I Imaginate a logic like this:

1. Turn On the Motor.
2. Set the Auto Mode to make the Pump work.
3. If the level is lower than the minimum value, press the low-Level switch.
4. At this point, the Manual Mode is set, and so Motor and Pump are turned off.
5. When the level reaches the maximum level, press the high-Level switch; the system goes in Auto Mode, and Motor and Pump restart working.

Figure 22. Level controller – Ladder diagram

Figure 23. Level controller - Visualization

4.2 Pump Controller

The Pump Controller is needed to set the number of rounds per minute in order to pump the right quantity of glue to the extruder.

In this implementation, the languages used are the Ladder Diagram for the main program and the Structured Text for the functions needed for the calculation of the rounds per minute and the volume of glue moved by the pump. In particular, the function to calculate the round per minute is the proper PI controller already seen in 2.2; this function follows the theory of PI and PID controller. A PID calculates the error by calculating the difference between the actual value and the desired value and then sets the deciding parameters accordingly. This error is continuously being calculated until the process stops [16].
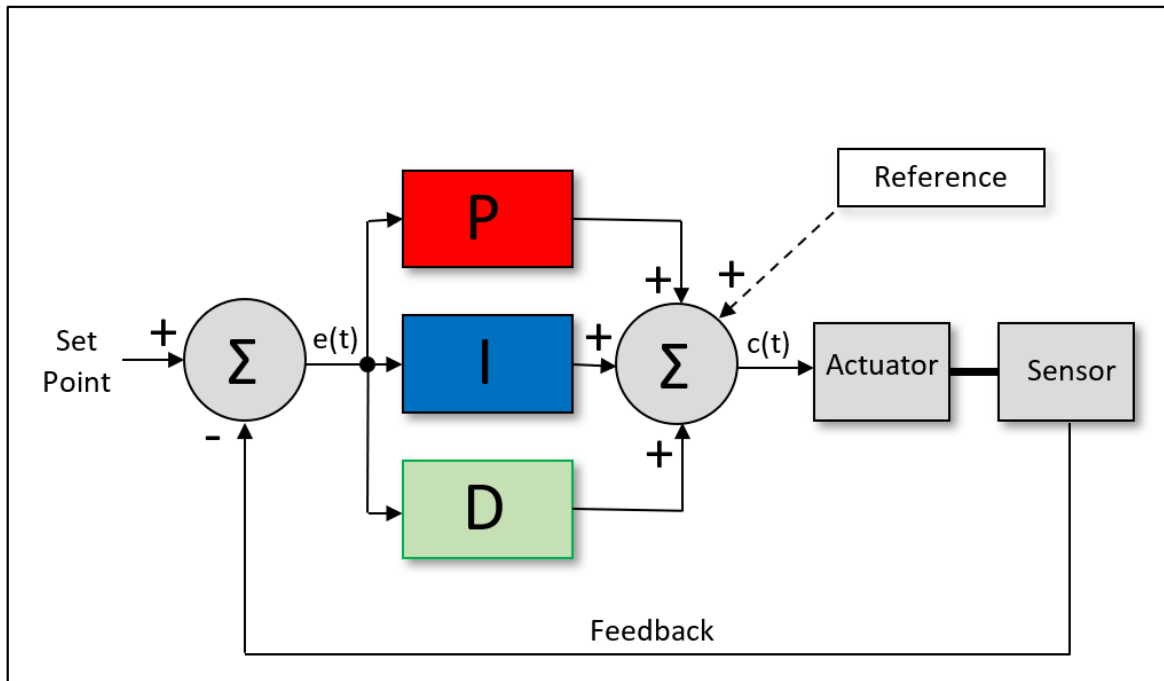
Figure 24. Scheme of a PID controller [17]

The derivative part can be considered as a prediction of the error in the future based on the current slope of the error and so it can be easily negligible for the study case.

The next picture shows how I implemented a PI controller as a code.

```
1       e :=  SetPoint – V_pump_curr ;
2       up := kp * e ;
3       ui := ( Tk / Ti ) * up + ui ;
4       IF  ui > UMAX  THEN
5            ui := UMAX ;
6             ELSIF  ui < UMIN  THEN
7                  ui := UMIN ;
8       END_IF
9       rpm := up + ui ;
```

Figure 25. Pump controller - PI controller

It can be seen that in the first place the calculation of the error is performed. Then it is calculated the proportional contribution by multiplying the error to the proportional coefficients that are found after the tuning of parameters in the SIMULINK model. The next step is to calculate the integral contribution; in this part, the variables Tk and Ti correspond to the sample time and to integral time (this last variable is found in the same way as the proportional coefficient). The next step is the limitation of the integral action by imposing a maximum and a minimum value. And finally, the rounds per minute can be calculated as a sum of the two contributes.

The other function implemented for this controller, and is just a copy of the equation (1.1) multiplied for the density in order to have coherence in the unit of measure.

The next picture shows the main program in the Ladder diagram that encloses both the functions just explained.
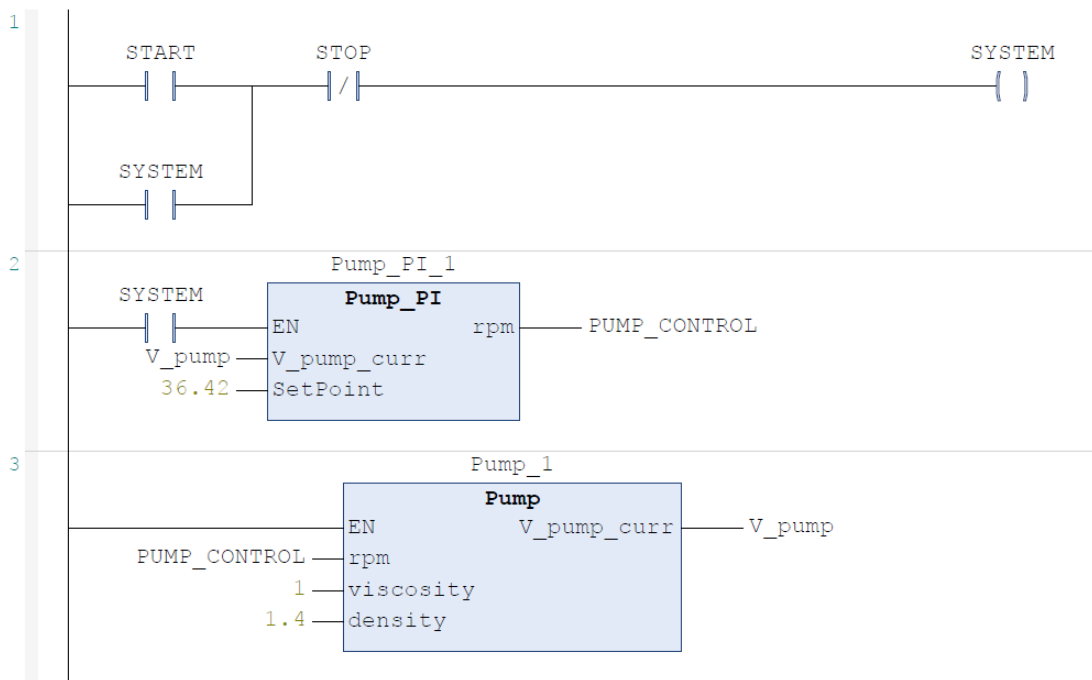

Figure 26. Pump controller - Ladder diagram

As can be seen by this figure, when the Start button is pressed, the system is turned On. In networks 2 and 3, there is the recalling of the two functions implemented. By setting a setpoint, viscosity, and density coherent to their ranges of values and 0 as the initial value of the volume moved by the pump, the program starts to calculate the correct number of rounds per minute that is put as input for the function of the pump that calculates the actual volume of glue moved by the pump that will be used as a new initial value of volume moved by the tank for the following step of the simulation.

Figure 27. Pump controller - Visualization

4.3 Temperature Controller

Also for this controller, I have written two functions: the first one is the implementation of the PI controller and it has the same structure of the function used for the Pump controller; in the second there is a copy of the formula used to calculate the amount of heat to give to the tank in order to maintain the temperature of the glue inside as close as possible to the desired value (already seen in Section 2.5), and the calculation of the derivate of the current temperature (similar to the equation (1.3)) plus an integration formula to remove the dependence to time.

```
1    e := SetPoint - T_curr ;
2    up := kp * e ;
3    ui_out := ( Tk / T_int ) * up + ui_in ;
4    T_cool := up + ui_out ;
```

Figure 28. Temperature controller - PI controller

```
1    Q := 10 * ( T_cool - Temp_curr_in ) ;
2    dTemp := Q / ( rho * Vol * Cglue ) ;
3    Temp_curr_out := dTemp * Tk + Temp_curr_in ;
```

Figure 29. Temperature controller - Tank model

The next picture shows the main program in the Ladder diagram that encloses both the functions just explained.
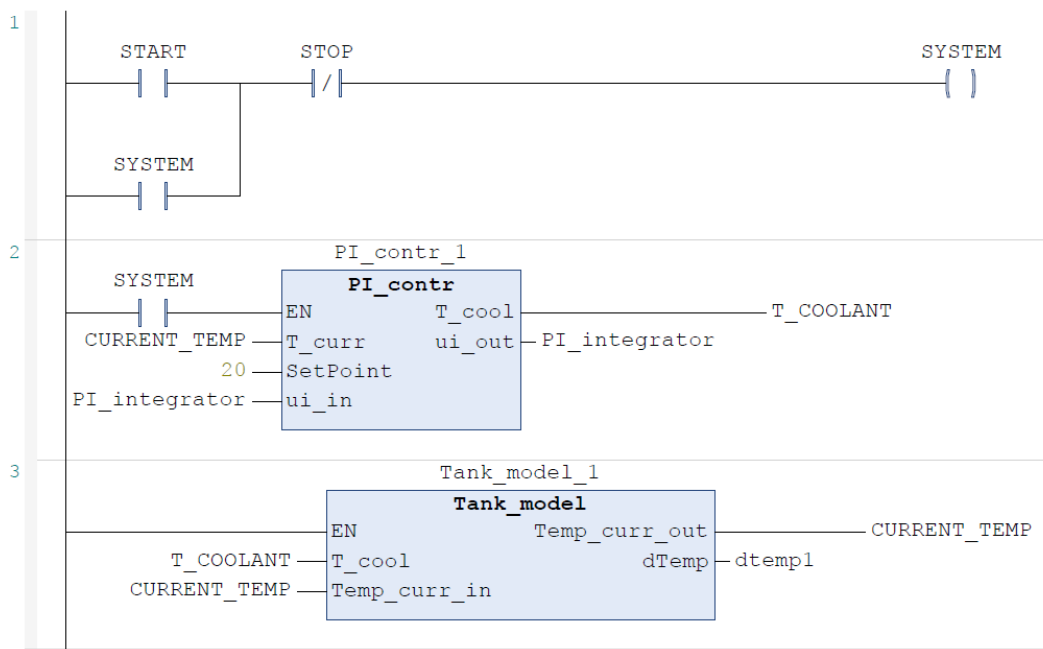
Figure 30. Temperature controller - Ladder diagram

When the Start button is pressed, the system is turned On. In networks 2 and 3, the two functions just explained are recalled. By setting a Setpoint for temperature to the temperature and an initial temperature to the glue inside the tank (just to make the code run), the program starts to calculate the value to set to the temperature of the coolant, which is put as an input for the tank model function in order to know the current temperature that will be used as a new starting temperature for the following step of the simulation.

Figure 31. Temperature controller - Visualization

4.4   Future works

The possible thing to do to make the work complete is to create a connection between the Model written in SIMULINK and the controllers written in CODESYS; basically, the idea is to find a way to just substitute the controllers simulated in SIMULINK with the ones implemented more accurately in PLC language

Bibliography

[1] F. H. Meng, L. P. Sun, and L. K. Zhu, "The Intelligent Controller Design of Parallel Online Mixing and Supplying Glue System," *Appl. Mech. Mater.*, vol. 44–47, pp. 4089–4093, 2011, doi: 10.4028/www.scientific.net/AMM.44-47.4089.

[2] "Fabinmec Europe S.R.L." Via Cantarana, San Cipriano Po, Italia. [Online]. Available: http://www.fabinmeceurope.it/

[3] S. Documentation, "Simulation and Model-Based Design." MathWorks, 2020. [Online]. Available: https://www.mathworks.com/products/simulink.html

[4] 3S-Smart Software Solutions GmbH, *CODESYS*. 2018. [Online]. Available: https://www.codesys.com

[5] "Scheme of the Extrusion system." Fabinmec Europe S.R.L, 2021.

[6] "Pompe Morgan." Via Carolina Romani, 59 Bresso (MI). [Online]. Available: https://www.pompemorgan.com/

[7] Pompe Morgan, "Model PQ100PB - Technical data sheet." 2021.

[8] "Prefere 14J350 - Technical data sheet." Prefere, Mar. 10, 2021. [Online]. Available: https://prefere.com/

[9] "SADECOL L 3084 - Technical data sheet." SADEPAN CHIMICA, Jul. 31, 2018. [Online]. Available: https://www.sadepan.com/

[10] "Description of the gluing machine." Fabinmec Europe S.R.L, Mar. 2021.

[11] "Specific heat of some common substances." [Online]. Available: https://www.engineeringtoolbox.com

[12] "PLC programming." [Online]. Available: https://ammc.com/codesys-why-should-you-use-it/

[13] "Programmable Logic Controller (PLC)." [Online]. Available: https://www.amci.com/industrial-automation-resources/plc-automation-tutorials/what-plc/

[14] "Ladder Diagram (LD) - Strenght and Weakness." [Online]. Available: https://www.controleng.com/articles/ladder-logic-strengths-weaknesses/

[15] "Structured Text (ST) - Why to use it." [Online]. Available: https://www.myplctraining.com/blog/learn-structured-text-in-2020

[16] "PID controller - Theory." [Online]. Available: https://www.ssla.co.uk/pid-controller/

[17] "PID controller - Scheme." [Online]. Available: https://www.rs-online.com/designspark/give-your-robot-the-mobility-control-of-a-real-mars-rover-part-1