GROUP MEMBERS
Nandyala Rama Sanjeevini - 19085057
Naveen Kumar Meena - 19085058
Nehal Rane - 19085059

TOPIC: Temperature Regulation Process

SHORT DESCRIPTION:

Temperature regulation refers to the process of maintaining a constant temperature in a device or a place. It is used in almost every field, including household, industrial, research, and others. Temperature regulation systems are also used in air conditioners, refrigerators, and geysers. Temperatures are automatically adjusted in accordance with inputs. Basically, we find an error signal based on the current temperature measured by the sensors for a given temperature, and then use a control algorithm to determine the actuating signal for the heat source. Our goal is to use PID control as a controller.

Papers:

1) PID Control Design for a Temperature Control System
2) Real Time Temperature Control of Oven Using MATLAB-SIMULINK
3) Temperature Control System and its Control using PID Controller

PAPER-1 SUMMARY:

The purpose of this paper is to design a PID controller for the Temperature Control System. The paper begins by thoroughly explaining the P (proportional) controller, the PI (proportional integral) controller, and the PID (proportional integral derivative) controller. Then it describes how to tune each of the controllers mentioned using the Ziegler- Nicholas tuning method. Finally, the authors present their observations and findings, as well as a comparison of the P, PI, and PID controllers.

PAPER-2 SUMMARY:

The goal of this paper is to create a MATLAB-Simulink model that will use a PID controller to control the real-time temperature of an oven. The paper begins by explaining the PID controller's reliability and performance before delving into the auto tuning methods for PID controllers, which include the Ziegler-Nichols Step Response Method, Relay Tuning Method, and Integral Square Time Error (ISTE)

Tuning Method. The PID parameters obtained using the methods described above are used to control the oven temperature. Using hardware, the real-time temperature of the oven is obtained in MATLAB-Simulink. Finally, the authors discuss their observations and findings, as well as a comparison of the three tuning methods.

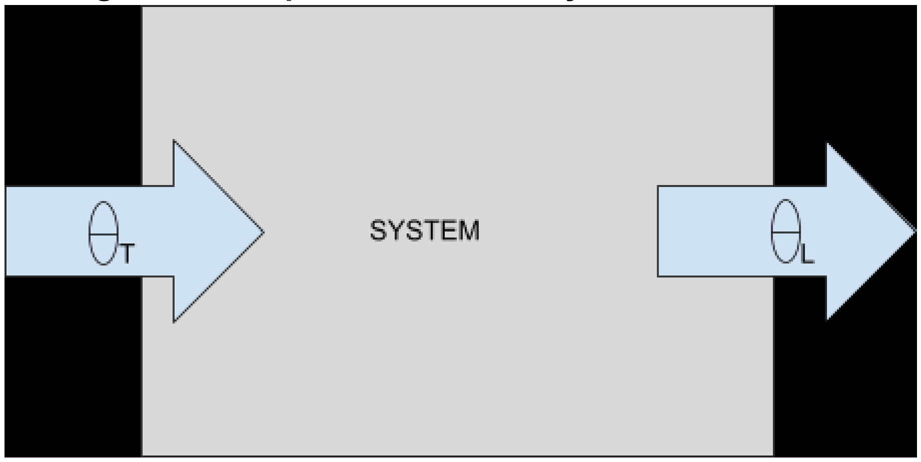
PAPER-3 SUMMARY:

The aim of this paper is to design a temperature control system and its control using a PID controller for an oven. In this, the plant is an electric oven or a heater whose temperature is to be controlled with respect to the reference input and control.The paper starts with introducing the basics of heat transfer and develops a mathematical model of the oven. And then it explains the basics of the PID controller and how it becomes a part of the overall regulation process. Then it obtains a transfer function model of the entire system using the experimental data. And then it presents a Ziegler–Nicholas tuning method to tune the parameters of the PID controller. And finally, in conclusion the authors provide their observations and results.

Objective:

To regulate the temperature of a system using PID Control technique.

Control Design for a Temperature Control System:



The heat transfer of the temperature system is given by :

$$\theta = \alpha\ \Delta T$$

Where,

θ is the rate of heat flow in joule/secθ
ΔT=Temperature difference in
α=constant

Under assumptions of linearity , the thermal resistance is defined as :

R= temperature difference / rate of heat flow = (θ / ΔT)

This is analogous to electrical resistance defined by I=V/R.

In the similar pattern thermal capacitance of the mass is given by equation :

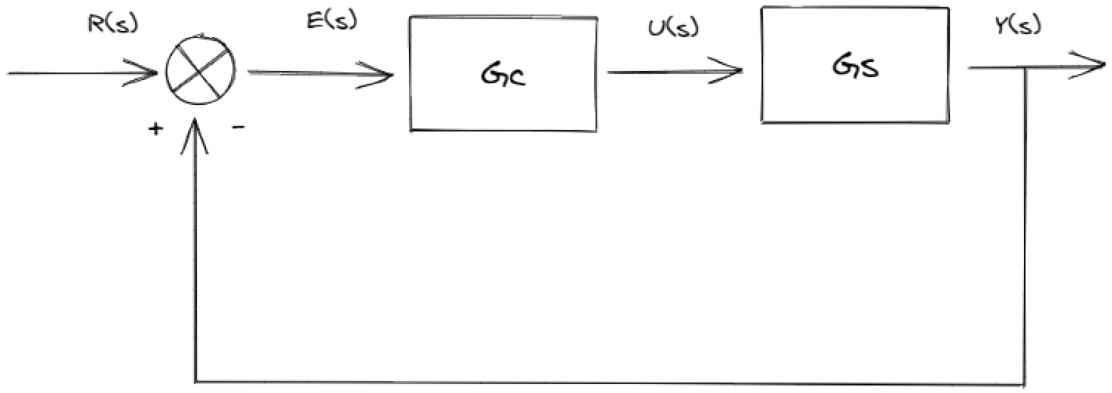
$$\theta = C\ d(\Delta T)/dt\theta$$

which is analogous to the V-I relationship of a capacitor namely I=C dV/dt. In the case of heat, C= rate of heat flow / rate of temperature change

Transfer function of the temperature system:

$$T(s)/\theta(s) = R/(1 + sCR)$$

Where: R = temperature difference / rate of heat flow = (θ/ ΔT)
C = rate of heat flow / rate of temperature change.
θ = the rate of heat flow in joule/sec



The transfer function of the PID controller and is given by equation(Gc) :

$$G_c = K_p\left[1 + \frac{1}{T_i s} + T_d s\right]$$

Where,

Kp - the controller path gain
Ti - the controller's integrator time constant
Td - the controller's derivative time constant

PID CONTROLLER:

Proportional-Integral-Derivative (PID) control is the most common control algorithm used in industry and has been universally accepted in industrial control.The basic idea behind a PID controller is to read a sensor, then compute the desired actuator output by calculating proportional, integral, and derivative responses and summing those three components to compute the output.

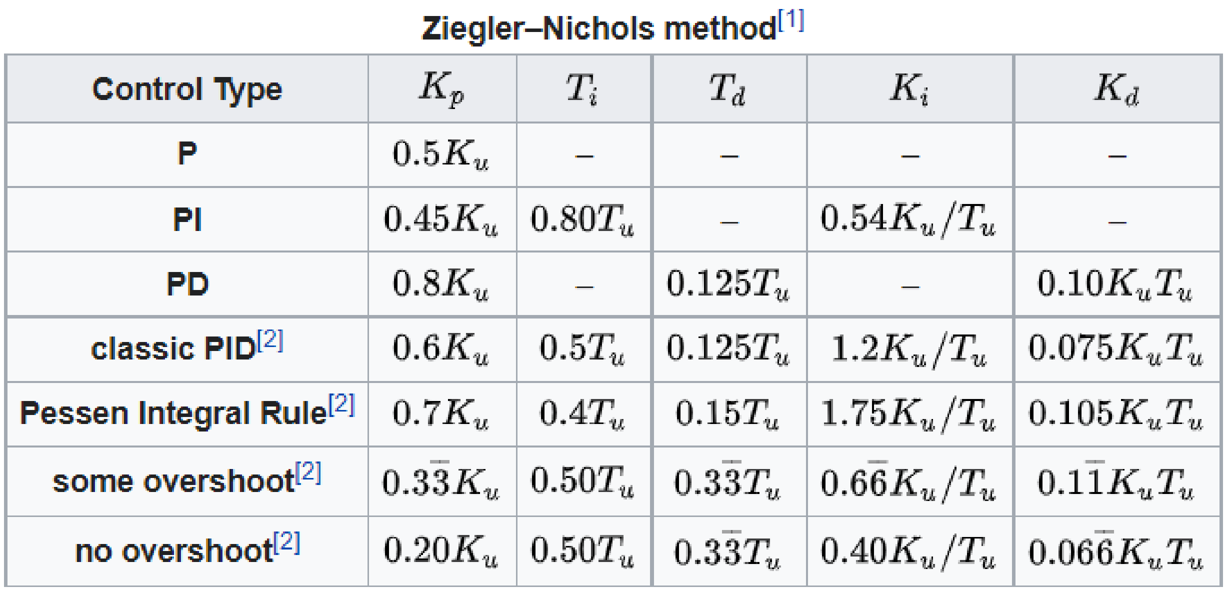
$$A(t) = K_p e(t) + K_i \int_0^t e(t)dt + K_d \frac{de(t)}{dt}$$

ZEIGLER-NICHOLS METHOD:

The Ziegler-Nichols rule is a heuristic PID tuning rule that attempts to produce good values for the three PID gain parameters: K_p , T_i and T_d.
given two measured feedback loop parameters derived from measurements:

1. the period T_u of the oscillation frequency at the stability limit
2. the gain margin K_u for loop stability

with the goal of achieving good regulation (disturbance rejection).



Ziegler–Nichols method[1]

Control Type	K_p	T_i	T_d	K_i	K_d
P	$0.5K_u$	–	–	–	–
PI	$0.45K_u$	$0.80T_u$	–	$0.54K_u/T_u$	–
PD	$0.8K_u$	–	$0.125T_u$	–	$0.10K_uT_u$
classic PID[2]	$0.6K_u$	$0.5T_u$	$0.125T_u$	$1.2K_u/T_u$	$0.075K_uT_u$
Pessen Integral Rule[2]	$0.7K_u$	$0.4T_u$	$0.15T_u$	$1.75K_u/T_u$	$0.105K_uT_u$
some overshoot[2]	$0.3\overline{3}K_u$	$0.50T_u$	$0.3\overline{3}T_u$	$0.6\overline{6}K_u/T_u$	$0.1\overline{1}K_uT_u$
no overshoot[2]	$0.20K_u$	$0.50T_u$	$0.3\overline{3}T_u$	$0.40K_u/T_u$	$0.06\overline{6}K_uT_u$

PID TUNING:

Tuning the PID controller involved determination of Kp, Ki, Kd that best suit our needs. We started with the Zeigler-Nichols method. First, we set Ki and Kd to 0. Using the proportional control action only, we increase Kp from 0 to critical value Kcr at which the output exhibits sustained oscillations. Thus, the critical gain Kcr and the corresponding period Pcr are experimentally determined. Ziegler and Nicholas suggested that we set the values of the parameters Kp, Ki, Kd according to the formula shown below:

Kp= 0.6 Kcr

Ki= 1.2 Kcr/ Pcr

Kd= 0.075 $Kcr\ Pcr$

But we found these values are overshooting and causing the measured temperature to diverge. So, we used the formulas from the No Overshooting method.

Kp= 0.2 Kcr

K_i = 0.4 Kcr/ Pcr

Kd= 0.06 $Kcr\ Pcr$

CODE STRUCTURE:

The code is available here

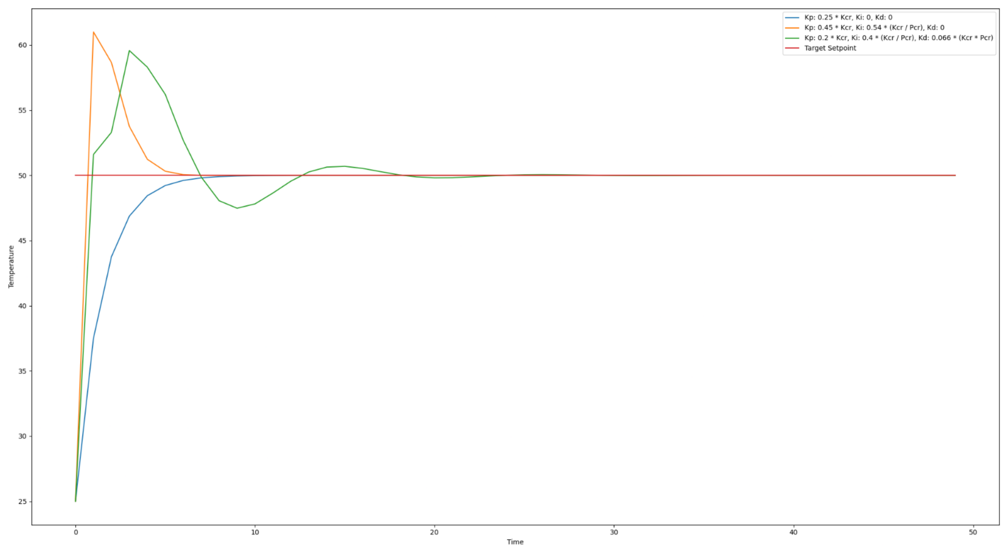
1. room.py: It implements a class, Room with init and step methods.

2. pid.py: It implements a class PID with init, step and reset methods.

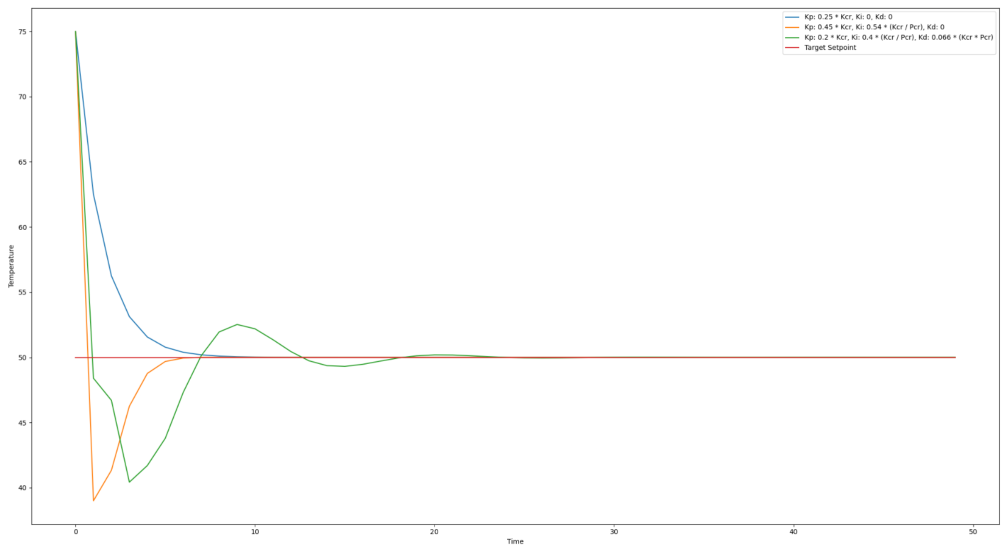
3. main.py: It runs the simulation and contains the flow of the code. And plots the results of the simulation.

<u>OUTPUTS AND SIMULATION:</u>

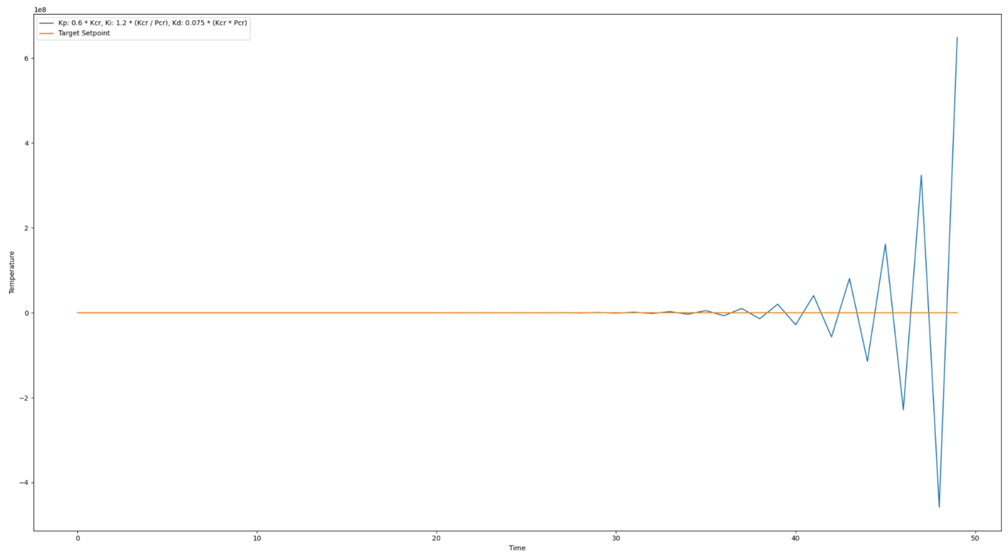
Simulation with P, PI, PID controllers (with starting temperature of 25^{o}C)



Simulation with P, PI, PID controllers (with starting temperature of 75^{o}C)



Simulation of PID controller with parameters obtained from ZN method (Diverges)



CONCLUSION:

We can manage temperature under many circumstances by using a PID controller, as demonstrated by simulations and measurements. However, special consideration should be given to PID tuning since in our situation, using the Ziegler-Nicholas second method settings, we discovered that the response is diverging from the setpoint and to overcome.

REFERENCES:

1) https://www.atlantis-press.com/article/5187.pdf
2)https://students.iitk.ac.in/robocon/docs/doku.php?id=robocon16:programming:pid_controller
3) https://en.wikipedia.org/wiki/Ziegler%E2%80%93Nichols_method
4) https://www.naun.org/main/NAUN/circuitssystemssignal/cssp-10.pdf