Modelado matemático

Función de transferencia del proceso:

Esta función es una herramienta que describe como un sistema dinámico responde a una entrada. Sirve para representar como la salida cambia en respuesta a una entrada. En la ecuación a continuación, T(s) es la temperatura actual en la transformada de Laplace, Td(s) es la temperatura deseada en la transformada de Laplace, K es la ganancia del sistema (representa la relación entre la señal de entrada y la de salida), y τ (tau) es la velocidad de respuesta del sistema a los cambios en la señal de entrada.

$$T(s)/Td(s) = K/\tau$$

Error del sistema:

Esto se refiere a la diferencia entre la temperatura deseada y la temperatura real del departamento. La ecuación queda de la siguiente manera:

$$E(t) = Td(t) - T(t)$$

Donde E(t) corresponde al error, Td(t) corresponde a la temperatura deseada y T(t) a la temperatura real del departamento, todo esto evaluado en el tiempo t.

Cálculo de términos PID:

Término proporcional (P): se calcula multiplicando el error por una constante proporcional (Kp).

$$P(t) = Kp * e(t)$$

Término integral (I): se calcula sumando el error a lo largo del tiempo y multiplicándolo por una constante integral (Ki).

$$I(t) = Ki * \int_0^T e(t)\, dt$$

Término derivativo (D): es el producto la tasa de cambio del error por una constante derivativa (Kd):

$$D(t) = Kd * \frac{de(t)}{dt}$$

PID:

La suma de los tres componentes da como resultado el controlador PID, quedando de la siguiente manera:

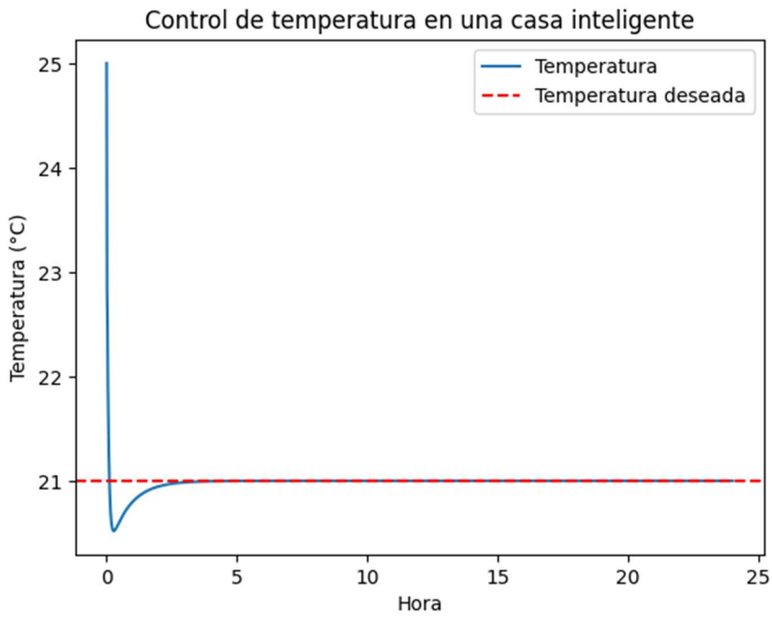
$$g(t) = P(t) + I(t) + D(t)$$

Donde g(t) es la señal de control enviada al sistema de control de temperatura.

El código de modelado del control PID queda de la siguiente manera:

```
import numpy as np
import matplotlib.pyplot as plt

class CasaInteligente:
    def __init__(self, temperatura_inicial, K, tau):
        self.temperatura_actual = temperatura_inicial
        self.temperatura_deseada = temperatura_inicial
        self.historial_temperaturas = [temperatura_inicial]
        self.K = K
        self.tau = tau
        self.dt = 1.0 / 60  # Paso de tiempo en horas (1 minuto)
        self.error_integral = 0
        self.error_anterior = 0

    def actualizar_temperatura(self, nueva_temperatura):
        self.temperatura_actual = nueva_temperatura
        self.historial_temperaturas.append(nueva_temperatura)

    def controlador_pid(self, Kp, Ki, Kd):
        error = self.temperatura_deseada - self.temperatura_actual
        self.error_integral += error * self.dt
        error_derivada = (error - self.error_anterior) / self.dt

        u = Kp * error + Ki * self.error_integral + Kd * error_derivada
        self.error_anterior = error
        return u

    def simular_dia(self, temperatura_deseada, Kp, Ki, Kd):
        self.temperatura_deseada = temperatura_deseada
        for _ in range(24 * 60):  # Simular durante 24 horas en intervalos
 de 1 minuto
            u = self.controlador_pid(Kp, Ki, Kd)
            dydt = (self.K * u - self.temperatura_actual) / self.tau
            nueva_temperatura = self.temperatura_actual + dydt * self.dt
            self.actualizar_temperatura(nueva_temperatura)

    def graficar_temperaturas(self):
        plt.plot(np.arange(len(self.historial_temperaturas)) / 60, self.hi
storial_temperaturas, label="Temperatura")
        plt.axhline(self.temperatura_deseada, color="r", linestyle="--
        ", label="Temperatura deseada")
        plt.xlabel("Hora")
        plt.ylabel("Temperatura (°C)")
        plt.title("Control de temperatura en una casa inteligente")
        plt.legend()
        plt.ylim(20,22)
        plt.xlim(0,1)
        plt.show()

def main():
    casa = CasaInteligente(25, K=7, tau=3)
    casa.simular_dia(21, Kp=8, Ki=10, Kd=0.08)
    casa.graficar_temperaturas()

if __name__ == "__main__":
    main()
```

Luego de probar valores para Kp, Ki y Kd, se concluyó que los valores adecuados son los expresados en el código (Kp=8, Ki=10, Kd=0.08). Reduciendo el componente derivativo y aumentando los otros parámetros, se logra el comportamiento deseado. Mostrando un comportamiento de la siguiente manera:



En el plazo de una hora: