

ESCUELA POLITÉCNICA NACIONAL

FACULTAD DE INGENIERÍA ELÉCTRICA Y ELECTRÓNICA

ESTUDIO, DISEÑO E IMPLEMENTACIÓN DE ESQUEMAS DE CONTROL PARA PROCESOS NO LINEALES DISEÑO Y SIMULACIÓN DE CONTROLADORES APLICADOS AL MODELO DE UN INVERNADERO

**TRABAJO DE INTEGRACIÓN CURRICULAR PRESENTADO COMO REQUISITO
PARA LA OBTENCIÓN DEL TÍTULO DE INGENIERO ELECTRÓNICO Y
AUTOMATIZACIÓN**

RAÚL ANDRÉS ALTAMIRANO SAMANIEGO

raul.altamirano@epn.edu.ec

DIRECTOR: GEOVANNY DANILO CHÁVEZ GARCÍA

danilo.chavez@epn.edu.ec

DMQ, Septiembre 2024

CERTIFICACIONES

Yo, RAÚL ANDRÉS ALTAMIRANO SAMANIEGO declaro que el trabajo de integración curricular aquí descrito es de mi autoría; que no ha sido previamente presentado para ningún grado o calificación profesional; y, que he consultado las referencias bibliográficas que se incluyen en este documento.

RAÚL ANDRÉS ALTAMIRANO SAMANIEGO

Certifico que el presente trabajo de integración curricular fue desarrollado por RAÚL ANDRÉS ALTAMIRANO SAMANIEGO, bajo mi supervisión.

GEOVANNY DANILO CHÁVEZ GARCÍA

DECLARACIÓN DE AUTORÍA

A través de la presente declaración, afirmamos que el trabajo de integración curricular aquí descrito, así como el (los) producto(s) resultante(s) del mismo, son públicos y estarán a disposición de la comunidad a través del repositorio institucional de la Escuela Politécnica Nacional; sin embargo, la titularidad de los derechos patrimoniales nos corresponde a los autores que hemos contribuido en el desarrollo del presente trabajo; observando para el efecto las disposiciones establecidas por el órgano competente en propiedad intelectual, la normativa interna y demás normas.

RAÚL ANDRÉS ALTAMIRANO SAMANIEGO

GEOVANNY DANILO CHÁVEZ GARCÍA

AGRADECIMIENTO

A mis padres, quienes siempre han permanecido a mi lado guiándome y apoyándome incondicionalmente, gracias por creer en mis capacidades y brindarme su amor.

A mis hermanos, por su compañía en mi vida, que me ha llenado de momentos felices y siempre me han empujado a superarme en diferentes aspectos de mi vida.

A Stephanie, cuyo cariño y apoyo han sido mi baluarte en los momentos difíciles.

A mi tutor, cuya guía y conocimientos han hecho posible el desarrollo de este trabajo.

ÍNDICE DE CONTENIDOS

CERTIFICACIONES.....	I
DECLARACIÓN DE AUTORÍA.....	II
AGRADECIMIENTO	III
RESUMEN	VI
ABSTRACT.....	VII
3 INTRODUCCIÓN	1
3.1 Objetivo general	1
3.2 Objetivos específicos.....	2
3.3 Alcance.....	2
3.4 Marco teórico.....	2
3.4.1 Invernadero	2
3.4.2 Aproximación de sistemas no lineales	3
3.4.3 Controlador tipo PID	4
3.4.3.1 Sintonización de controladores PID.....	5
3.4.4 Controlador Difuso.....	6
3.4.4.1 Fuzzyficación.....	7
3.4.4.2 Inferencia Fuzzy	7
3.4.4.3 Defuzzificación	8
3.4.5 Controlador por modos deslizantes	8
3.4.6 Índices de Desempeño	10
3.4.6.1 Índice Integral del error cuadrático (ISE).....	10
3.4.6.2 Índice Integral del error absoluto (IAE)	10
4 METODOLOGÍA	12
4.1 Obtención del modelo matemático.....	12
4.2 Sintonización de los controladores	15
4.2.1 Controlador PID de calefactores.....	16
4.2.2 Controlador PID de nebulizador.....	18
4.2.3 Controlador SCM-difuso	20
4.2.4 SCM para Calefactores	21
4.2.5 Controlador Difuso del SMC para Calefactores	23
4.2.6 Controlador SMC para Nebulizador	27
4.2.7 Controlador Difuso para SMC del Nebulizador	28
4.3 Interfaz Modelo del Invernadero	31
4.3.1 Modelo del Invernadero	34

5	RESULTADOS, CONCLUSIONES Y rECOMENDACIONES	35
5.1	Resultados	35
5.1.1	Respuesta del controlador PID	35
5.1.2	Respuesta del controlador SCM-difuso.....	39
5.1.3	Desempeño de los controladores	42
5.2	Conclusiones	43
5.3	Recomendaciones.....	43
6	REFERENCIAS BIBLIOGRÁFICAS	44
7	ANEXOS.....	46

RESUMEN

En el presente trabajo tiene como objetivo principal la comparación del desempeño de dos tipos de controlador: PID y SMC-Difuso, aplicados al control del microclima de un invernadero y respecto al Déficit de Presión de Vapor (DPV) como variable de interés. Mediante la herramienta Simulink de Matlab se implementó el modelo matemático del invernadero, y se diseñaron los controladores. Los resultados de la simulación se presentaron en una interfaz creada a partir de Guide de Matlab.

Los controladores se compararon mediante índices de rendimiento ISE e IAE para cuantificar el desempeño, ambos fueron puesto a prueba en varios escenarios: sin presencia de perturbaciones y con diferentes conjuntos de grupo de perturbaciones (temperatura externa, humedad externa y radiación externa), obteniendo como resultado un mejor desempeño del controlador SMC-Difuso ya que sus índices de desempeño ISE e IAE presentaban mejores resultados.

PALABRAS CLAVE: Invernadero, VPD, PID, SMC-Difuso, ISE, IEA.

ABSTRACT

The main objective of this work is to compare the performance of two types of controllers: PID and Fuzzy SMC, applied to the control of the microclimate of a greenhouse with respect to the Vapor Pressure Deficit (VPD) as the variable of interest. The mathematical model of the greenhouse was implemented using Matlab's Simulink tool, and the controllers were designed. The simulation results were presented in an interface created using Matlab's Guide.

The controllers were compared using the ISE and IAE performance indices to quantify performance. Both were tested in various scenarios: without disturbances and with different sets of disturbances (external temperature, external humidity, and external radiation), resulting in better performance of the Fuzzy SMC controller as its ISE and IAE performance indices showed better results.

KEYWORDS: Greenhouse, VPD, PID, Fuzzy SMC, ISE, IAE.

3 INTRODUCCIÓN

Ecuador a lo largo de su historia se ha caracterizado por ser un país exportador agrícola, siendo esta una de sus principales actividades económicas. Dentro de sus exportaciones se tienen cultivos de alto valor agregado como son: hortalizas, flores y frutas. [1]Sin embargo, no siempre se tiene condiciones ambientales adecuadas para un óptimo crecimiento de los cultivos; bajo este contexto, los invernaderos son una herramienta útil para mantener condiciones ambientales adecuadas, lo que mejora tanto la eficiencia como la calidad del producto final sin tener en cuenta los cambios climáticos presente en las diferentes regiones del país. Esto permite obtener productos alta calidad durante todo el año porque se evitan las temporadas donde las condiciones no son óptimas para el cultivo.

Para mantener un microclima adecuado (entiéndase microclima como la interacción entre temperatura, humedad, déficit de presión de vapor, radiación solar, etc.) dentro de un invernadero, es necesario realizar un sistema de control avanzando para garantizar condiciones óptimas de crecimiento en los cultivos. Para lograr este objetivo existen varios modelos de sistema de controladores, entre ellos se tiene: el controlador PID (Proporcionar-Integral-Derivativo) que es uno de los más utilizados dentro de la industria debido a su versatilidad y simplicidad; por otro lado, el SMC-Difuso, un controlador híbrido que es resultado de las ventajas que aporta el control por modos deslizantes y la lógica difusa, dando como resultado un controlador robusto frente a las incertidumbres de pueden darse dentro de un invernadero.

Con el objetivo de generar una métrica para evaluar la eficiencia de los controladores se utilizan los índices de rendimiento IAE (Integral of Absolute Error) e ISE (Integral of Squared Error), los cuales permiten cuantificar el desempeño de los controladores. El índice ISE brinda información sobre los errores grandes que se presenta en el sistema dando una idea de la estabilidad del controlador; mientras que el índice IAE se centra en errores pequeños y proporciona una idea global del desempeño del controlador.

3.1 Objetivo general

Evaluar y comparar el desempeño de los controladores PID y SMC-difuso para el control de condiciones ambientales dentro de un invernadero.

3.2 Objetivos específicos

1. Definir un modelo matemático que describa el comportamiento de un invernadero e implementarlo en Matlab Simulink.
2. Diseñar dos controladores: PID y SMC-Difuso.
3. Realizar una interfaz gráfica en Matlab Simulink del invernadero y los controles aplicados.

3.3 Alcance

Para evaluar el desempeño de un controlador PID frente a un controlador SMC-difuso dentro de un invernadero, se definirá un modelo matemático que represente el intercambio de energía del invernadero con el exterior, sin considerar la humedad del suelo como una variable que influye en dicho intercambio. Luego, se implementará el modelo en Matlab Simulink. Para controlar el déficit de presión de vapor que rige el invernadero, se diseñarán y sintonizarán dos controles, uno de tipo PID y un controlador híbrido SMC-difuso. Se realizará una interfaz gráfica en Matlab Simulink del invernadero y los controladores aplicados.

3.4 Marco teórico

3.4.1 Invernadero

Los invernaderos son sistemas aislados del medio ambiente que tienen como objetivo producir un microclima controlado para garantizar un óptimo crecimiento del cultivo, sin tomar en cuenta los cambios climáticos que se presentan en el exterior, como, por ejemplo, el cambio de estaciones. [2]

Dentro de un invernadero interactúan varias variables que dan como el resultado un microclima. Aquellas que tienen una mayor influencia son: humedad, radiación solar, temperatura, déficit de presión de vapor; siendo esta última mediante la cual se desarrolla el presente trabajo.[2]

Déficit de Presión de Vapor (DPV)

El DPV es la variable que indica la diferencia de la presión de saturación de vapor y la presión de vapor presente en el invernadero, lo cual aporta utilidad al momento del control climático debido que relaciona tanto la temperatura como la humedad relativa proporcionando un equilibrio óptimo de las dos variables a favor del cultivo. [2]

Las ventajas que se presentan al tener un DPV óptimo son:

- Mejora la absorción de nutrientes
- Disminución de agentes patógenos
- Mejora en el crecimiento del cultivo
- Control del estrés hídrico

En [3] se presentan rangos óptimos de Deficit de Presion de Vapor (DPV) que se deben manejar dentro de un invernadero

Tabla 3.1. Rangos Óptimos DPV [3]

VPD (kPa)	T (°C)	RH (%)
0.75-1.06	15-34	40-85
0.5-1.2	15-35	35-90
0.4-0.79	15-30	55-90
0.47-1.27	15-30	60-85

3.4.2 Aproximación de sistemas no lineales

La forma más común de sintonizar controladores es emplear los parámetros de un modelo de orden reducido. Uno de los más utilizados es el de primer orden con tiempo muerto, como el que describe la Ecuación 3.1. [4]

$$G(s) = \frac{ke^{-t_0s}}{\tau s + 1} \quad (3.1)$$

Donde:

$G(s)$: Función de Transferencia de primer orden con retardo

k : Ganancia del Sistema

t_0 : Retardo de Tiempo

τ : Constante de Tiempo

Para determinar los parámetros del modelo aproximado se emplea el método de dos puntos de Smith. En el cual, al sistema en lazo abierto se realiza un cambio de referencia tipo escalón, generalmente 10%, para obtener información de la curva de reacción del proceso en los puntos 28.3% y 63.2%. La Figura 3.1 ilustra este método. [4]

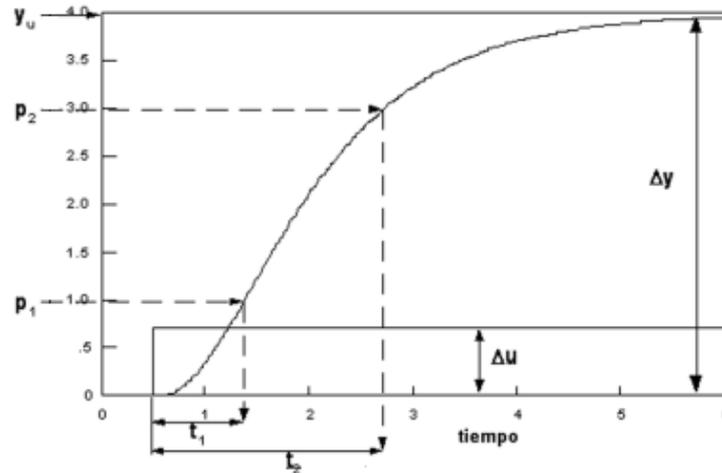


Figura 3.1. Método de dos puntos de Smith [4]

A partir de la Figura 1, se obtiene la relación que se muestra en la Ecuación 3.2.

$$k = \frac{\Delta Y}{\Delta u} \quad (3.2)$$

Donde:

ΔY : Diferencia entre el valor final y el valor inicial del sistema.

Δu : Diferencia entre el valor final y el valor inicial del escalón.

$$\tau = 1.5 * (t_{63} - t_{28}) \quad (3.3)$$

$$t_0 = t_{63} - \tau \quad (3.4)$$

3.4.3 Controlador tipo PID

Los controladores PID, desde su desarrollo en el siglo pasado, han sido los más utilizados dentro de los temas de control industrial debido a su buena respuesta a una gran cantidad de escenarios, además de su sencilla implementación respecto a otros controladores.

Un controlador PID es el resultado de la unión de tres controladores: proporcional, integral y derivativo, tal como se muestra en la Ecuación 3.5. [5]

$$PID = \underbrace{K_p e(t)}_{\text{Proporcional}} + \underbrace{\frac{1}{K_i} \int e dt}_{\text{Integral}} + \underbrace{k_d \frac{de}{dt}}_{\text{Derivativo}} \quad (3.5)$$

- El termino *Proporcional*, como su nombre lo indica, genera una acción proporcional al error. Esto influye en la velocidad con la que se corrige el error. Puede generar picos u oscilaciones en el sistema.
- El termino *Integral* disminuye el error en estado estable por medio de la suma de errores pasados.
- El termino *Derivativo* disminuye el error en estado transitorio utilizando la variación del error.

La Figura 3.2 describe el esquema de un controlador PID de un invernadero.

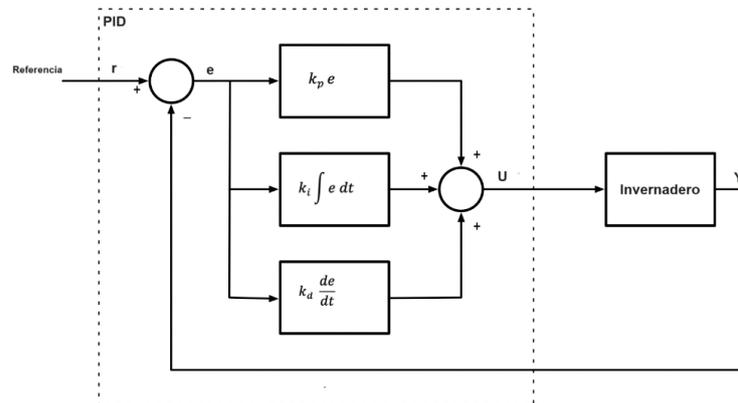


Figura 3.2. Controlador PID para un invernadero

3.4.3.1 Sintonización de controladores PID

Existen varios métodos de sintonización basados en las características que definen a un modelo de primer orden con retraso, por ejemplo, el método de Dahlin, quien desarrolló las fórmulas que se presentan en las Ecuaciones 3.6, 3.7 y 3.8, para sintonizar PID: [6]

$$K_p = \frac{1}{2k} \left(\frac{t_0}{\tau} \right)^{-1} \quad (3.6)$$

Donde:

K_p : Parámetro Proporcional

k : Ganancia del Sistema

t_0 : Tiempo muerto

τ : Constante de Tiempo del Sistema

$$t_i = \tau \quad (3.7)$$

Donde:

t_i : Parámetro Integral

$$t_d = \frac{t_0}{2} \quad (3.8)$$

Donde:

t_d : Parámetro Derivativo

3.4.4 Controlador Difuso

Un control difuso es capaz de trabajar sin la necesidad de información exacta. Se suele utilizar para sistemas con modelos complejos o que carecen de un modelo específico, ya que este tipo de controlador funciona desde la experiencia. Esto lo vuelve más flexible frente a controles convencionales que requieren un modelo exacto.[7]

La Figura 3.3 muestra la estructura de un controlador difuso.

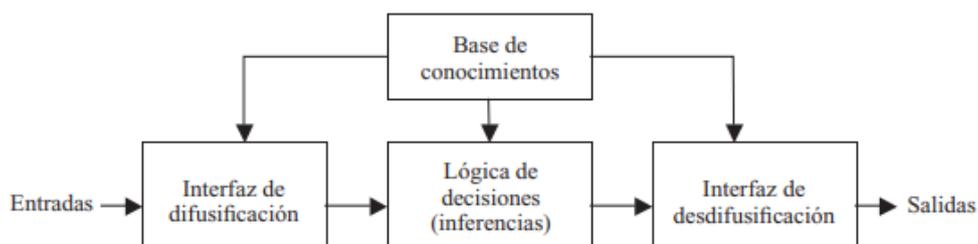


Figura 3.3. Estructura de un controlador difuso [7]

Algunos términos para considerar durante el desarrollo de un controlador difuso son:

- *Función de Pertenencia*: también conocida como membresía, es un conjunto difuso que otorga un grado de pertenencia a las entradas abruptas de su dominio. [7]
- *Grado de Pertenencia*: hace referencia al nivel de compatibilidad de una entrada a una función de pertenencia. Puede tomar valores de 0 a 1, donde 0 implica que no son compatibles y 1 que son totalmente compatibles.[7]

- *Entrada Abrupta*: son los valores variables para evaluar el sistema.[7]

Para desarrollar un controlador difuso se debe seguir 3 etapas: fuzzyficación, inferencia fuzzy y defuzzyficación.

3.4.4.1 Fuzzyficación

Es la primera etapa del control difuso. Durante esta etapa se transforman las entradas abruptas en entradas fuzzy al darle un grado de pertenencia a una o varias funciones de membresía. La Figura 3.4 resume esta etapa. [7]

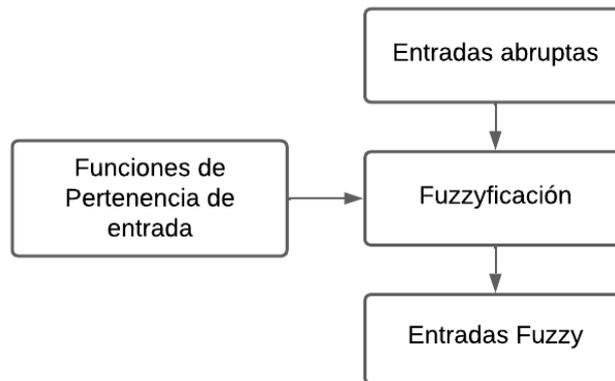


Figura 3.4. Esquema de la etapa de fuzzyficación [7]

3.4.4.2 Inferencia Fuzzy

La Figura 3.5 describe la segunda etapa de la lógica difusa, en donde el controlador difuso emplea reglas gramaticales sobre las entradas fuzzy, estas reglas son de tipo *IF-THEN* y sirven para decidir la acción que tomara el controlador en respuesta a las entradas fuzzy.[7]

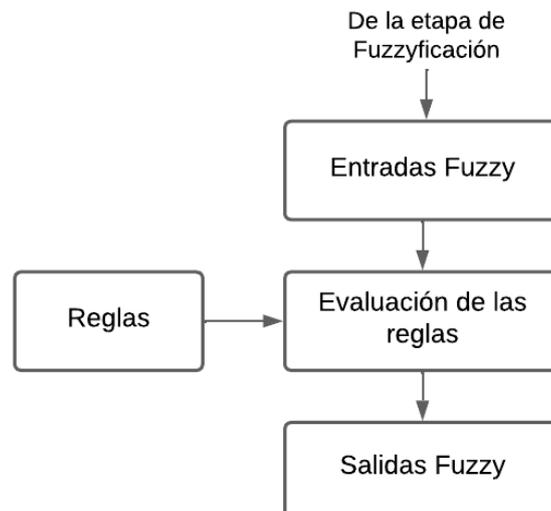


Figura 3.5. Esquema de la etapa de inferencia Fuzzy [7]

El objetivo de las reglas es examinar los antecedentes (entradas fuzzy) por medio de operadores lógicos como AND/OR, dando como resultado los consecuentes. Estos serán las acciones de control que realizará la lógica difusa. Lo anterior descrito se puede expresar siguiendo la forma:

IF Antecedente 1 AND Antecedente 2 THEN Consecuente

Estas reglas se desarrollan por medio del conocimiento del proceso y se escriben con las etiquetas usados en los conjuntos difusos de las funciones de membresía. [7]

3.4.4.3 Defuzzificación

Es el último paso del control difuso. El objetivo de esta etapa es encontrar las salidas del controlador; los consecuentes definidos en las reglas de evaluación cambiarán a su respectiva función de membresía de salida obteniendo un valor exacto como acción de control. Esta etapa se ilustra en la Figura 3.6.[7]

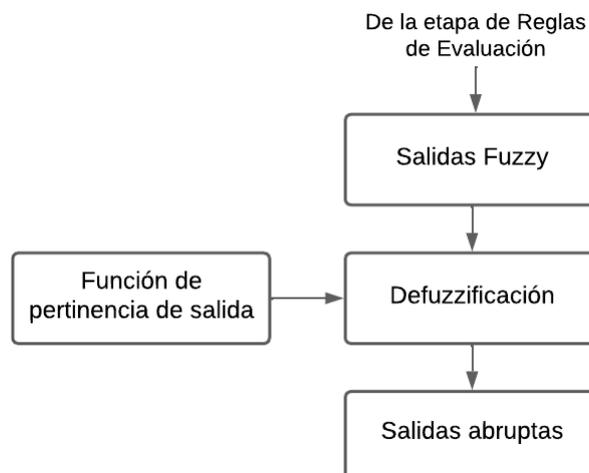


Figura 3.6. Esquema de la etapa de Defuzzyficación [7]

3.4.5 Controlador por modos deslizantes

El controlador SMC (Sliding Mode Control) es una técnica robusta de estructura variable, esto lo hace óptimo para sistemas dinámicos garantizando un alto rendimiento frente a perturbaciones y variaciones en el tiempo [8].

La Figura 3.7 muestra de forma gráfica el funcionamiento del SMC, en donde el controlador toma como base la creación de una superficie deslizante en la cual el proceso pueda moverse de forma controlada hacia el valor de referencia.

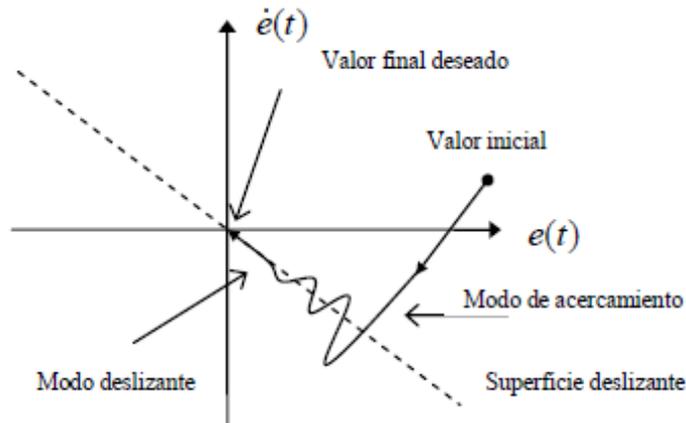


Figura 3.7. Funcionamiento de un controlador SCM [8]

En la Ecuación 3.9 puede observarse un modelo de estructura de deslizamiento:

$$S(t) = \left(\frac{de(t)}{dt} + \lambda \right)^n \int e(t) \quad (3.9)$$

Donde:

n : es orden del sistema

e : es el error del sistema

λ : es un parámetro de sintonización

Una vez definida la superficie de deslizamiento, el SMC se enfoca en la ley de control del sistema descrita en la Ecuación 3.10 y que consta de dos partes: (1) una parte continua (U_c) que tiene como objetivo mantener al sistema dentro de la superficie de deslizamiento y (2) una discontinua (U_d) que añade un elemento no lineal con el objetivo de llevar al sistema a la superficie de deslizamiento[9] .

$$U_{smc} = U_c + U_d \quad (3.10)$$

De la ley de control, la parte continua se diseña a partir de las características del sistema que se desea controlar mientras que la parte discontinua viene dada por la Ecuación 3.11.

$$U_d = \alpha * \text{sing}(S) \quad (3.11)$$

Donde:

α : Parámetro de Sintonización

Esta función no lineal tipo relé incorpora un problema denominado *chattering*. Este produce la reducción de vida útil del elemento final de control, por lo que una solución es modificar la parte discontinua por una función sigmoide como se muestra en la Ecuación 3.12.

$$U_d = k_d \frac{s}{|s| + \delta} \quad (3.12)$$

Donde el parámetro δ es el encargado de evitar el *chattering* mientras que término k_d tiene como determina la velocidad con la que el sistema se acerca a la superficie de deslizamiento [9].

3.4.6 Índices de Desempeño

El desempeño de un controlador frente al de otro puede ser evaluado y comparado al establecer índices de desempeño.

3.4.6.1 Índice Integral del error cuadrático (ISE)

El índice ISE permite evaluar sistemas de control priorizando el análisis los errores de mayor tamaño, generalmente producidos en un cambio de referencia; en tanto que los errores pequeños son concebidos como una aportación menor[10].

Este índice se define de acuerdo con la Ecuación 3.13:

$$ISE = \int e(t)^2 dt \quad (3.13)$$

El controlador presenta un mejor desempeño mientras el valor del índice ISE tiende a 0.

El índice ISE considera el error elevado al cuadrado, por lo que magnifica errores más grandes. En relación al invernadero, un índice ISE alto implica grandes desviaciones del valor deseado resultando en condiciones no óptimas para el cultivo, es decir, el crecimiento de los cultivos se ve afectando drásticamente. Los controladores tienen la intención de disminuir este índice.

3.4.6.2 Índice Integral del error absoluto (IAE)

Al contrario del ISE, el índice IAE se mayormente influenciado por la contribución de errores pequeños y constantes; mientras que los errores grandes tienen una menor aportación dentro del análisis [10].

Este índice se define de acuerdo con la Ecuación 3.14:

$$IAE = \int |e(t)| dt \quad (3.14)$$

El controlador presentara un mejor desempeño cuando el índice IAE se acerque a un valor nulo.

El índice IAE, toma en cuenta la magnitud total del error, castigando las desviaciones del valor deseado. En un invernadero en donde el sistema esta activo durante largos periodos, un índice IAE alto da como resultado condiciones ambientales inestable perjudicando drásticamente el crecimiento de los cultivos, mantener este índice lo mas bajo posible da como resultado un ambiente estable y favorable para el desarrollo de plantas.

4 METODOLOGÍA

La evaluación del desempeño de los controladores PID y SMC-difuso para el control de condiciones ambientales dentro de un invernadero hace referencia a un estudio de enfoque mixto y de carácter descriptivo y explicativo. Las ecuaciones algebraicas y diferenciales del modelo matemático que describen el intercambio de masa y energía dentro del invernadero se obtuvieron de análisis documental, mientras que los datos de las perturbaciones con los que los que se validaron los controlados se obtuvieron de los resultados presentados en [11]

4.1 Obtención del modelo matemático

Para el modelo del invernadero se establecieron 2 variables de entrada: una de enfriamiento y una de calefacción; y 1 variable de salida que relaciona la temperatura y humedad del invernadero, esta variable se denominó Déficit de Presión de Vapor, tal como se muestra en la Figura 4.1.

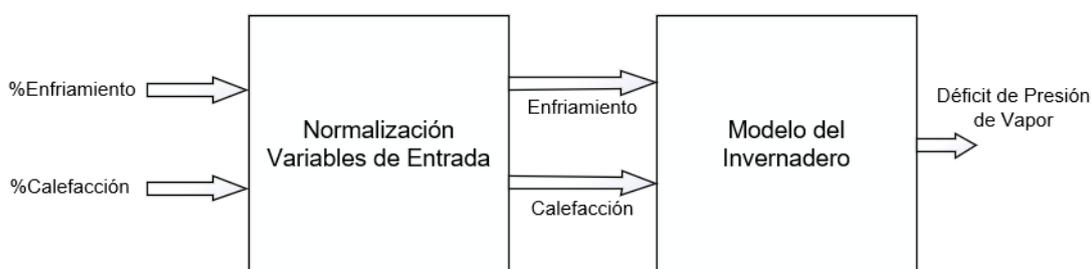


Figura 4.1. Esquematización del modelo matemático

La variable de calefacción permitió aumentar el Déficit de Presión de Vapor, mientras que la de enfriamiento contribuyó a disminuir el valor del Déficit de Presión de Vapor.

Las condiciones climáticas dentro del invernadero estuvieron dadas por los balances de masa y energía descritos en las Ecuaciones 4.1 y 4.2, y que fueron obtenidas de [12]

$$\frac{dT_{in}}{dt} = \frac{1}{C_p * \rho * H} * \left(Q_{GRin} + Q_{Heater} - L - E - (T_{in} - T_{out}) * (q_v * C_p * \rho + w * k) \right) \quad (4.1)$$

Donde:

$$C_p: \text{Calor específico del aire húmedo} \left[\frac{J}{kg \cdot ^\circ C} \right]$$

$$\rho: \text{Masa específica del aire} \left[\frac{kg}{m^3} \right]$$

H: Altura Media del Invernadero [m]

Q_{GRin} : Radiación solar interna $\left[\frac{W}{m^2}\right]$

Q_{Heater} : Radiación calorífica del sistema de calefacción $\left[\frac{W}{m^2}\right]$

L : Calor Latente de Vaporización del Agua $\left[\frac{J}{kg}\right]$

E: Evaporación y transpiración en el interior del invernadero $\left[\frac{kg}{m^2 s}\right]$

T_{in} : Temperatura Interna [°C]

T_{out} : Temperatura Externa [°C]

q_v : Ventilación $\left[\frac{m}{s}\right]$

w: Relación entre superficie acristalada y superficie del suelo

k: Coeficiente de Transmisión de Calor $\left[\frac{J}{m^2 °C s}\right]$

$$\frac{dW_{in}}{dt} = \frac{1}{H \cdot \rho} * (E - (W_{in} - W_{out}) * q_v * \rho) \quad (4.2)$$

Donde:

W_{in} : Humedad Absoluta Interna $\left[\frac{g_{agua}}{kg_{aire seco}}\right]$

W_{out} : Humedad Absoluta Externa $\left[\frac{g_{agua}}{kg_{aire seco}}\right]$

La radiación solar Q_{GRin} que adquiere el invernadero del exterior se describió por medio de la Ecuación 4.3:

$$Q_{GRin} = \tau_c * (1 - p_g) * G_{GRout} \quad (4.3)$$

La calefacción (Q_{Heater}) se expresó de acuerdo con las Ecuaciones 4.4 y 4.5.

$$Q_{Heater} = N_H * \frac{H_{cap}}{A_{fl}} \quad (4.4)$$

$$H_{cap} = \left(\frac{200000}{100}\right) * \%H_{cap} \quad (4.5)$$

La humedad del invernadero es afectada directamente por la transpiración del cultivo (E_t) y por el sistema de enfriamiento (E_c), tal como se muestra en la Ecuación 4.6.

$$E = E_t + E_c \quad (4.6)$$

Donde:

$$E_t = \begin{cases} 0.0003 * \tau_c * Q_{GRout} + 0.0021 & \text{mucho cultivo} \\ 0.00006 * \tau_c * Q_{GrouT} + 0.004 & \text{poco cultivo} \\ 0 & \text{sin cultivo} \end{cases} \quad (4.7)$$

$$E_c = \left(\frac{140.8}{100}\right) * \%E_c \quad (4.8)$$

Donde:

$$E_t: \text{Transpiración de las plantas} \left[\frac{kg}{m^2 \text{ dia}} \right]$$

$$E_c: \text{Evaporización del sistema de enfriamiento} \left[\frac{kg}{m^2 \text{ dia}} \right]$$

En la Ecuación 4.9, el déficit de presión de vapor se definió como la diferencia entre la presión de vapor de saturación (V_{SAT}), que está relacionado con la temperatura del invernadero; y la presión de vapor de aire (V_{AIRE}) que está estrechamente relacionada con la humedad presente en el invernadero.

$$VPD = V_{SAT} - V_{AIRE} \quad (4.9)$$

V_{SAT} y V_{AIRE} se obtuvieron de [13].

$$V_{SAT} = 0.61078 * \left(e^{\frac{17.27 * T_{\circ}C}{237.3 + T_{\circ}C}} \right) \quad (4.10)$$

$$V_{AIRE} = 0.61078 * \left(e^{\frac{17.27 * T_{\circ}C}{237.3 + T_{\circ}C}} \right) * \frac{RH}{100} \quad (4.11)$$

A partir de las Ecuaciones 4.10 y 4.11, la Ecuación 4.9 se reescribió como se muestra en la Ecuación 4.12.

$$VPD = 0.61078 * \left(e^{\frac{17.27 * T_{\circ}C}{237.3 + T_{\circ}C}} \right) * \left(1 - \frac{RH}{100} \right) \quad (4.12)$$

Donde:

$$VPD: \text{Déficit de presión de Vapor [kPa]}$$

Para una mejor comprensión de la humedad, se realizó un cambio de humedad absoluta del invernadero a humedad relativa ($HR\%$) de acuerdo con la Ecuación 4.13.

$$HR\% = \left(\frac{P_v}{P_{sat}} \right) * 100\% \quad (4.13)$$

Donde:

P_v : Presión Parcial de Agua

P_{sat} : Presión Saturada de Agua

Además,

$$P_v = \frac{W * P_{atm}}{0.622 + W} \quad (4.14)$$

Donde:

P_{atm} :

$$P_{atm} = (1 - (2.25577 * 10^{-5}) * H_{NM})^{5.2559} \quad (4.15)$$

Donde:

H_{NM} : Altura al Nivel del Mar

$$P_{sat} = 0.61078 * \left(e^{\frac{17.27 * T_{°C}}{273.3 + T_{°C}}} \right) * \left(\frac{1000}{101325} \right) \quad (4.16)$$

4.2 Sintonización de los controladores

Los valores constantes utilizados para la simulación se presentan en la Tabla 4.1.

Tabla 4.1. Constantes de las ecuaciones diferenciales y algebraicas

Constante	Valor
C_p	1013
ρ	1.2
H	5
L	2.5e6
q_v	0.093
w	2.3
k	6.2
τ_c	0.87
p_g	0.4

N_H	2
A_{fl}	162
A_{gl}	372.6

La obtención de los valores característicos del sistema se realizó en estado estable, es decir, cuando la tasa de cambio de las ecuaciones diferenciales fue nula. Este estado se alcanza con los valores presentes en la Tabla 4.2.

Tabla 4.2. Valor de las constantes en el estado estable

Contante	Valor
$RH\%_{in}$	60
T_{in}	20
VPD	0.94
$RH\%_{out}$	63.08
G_{GRout}	200
T_{out}	19.19

4.2.1 Controlador PID de calefactores

Para desarrollar el PID se utilizó el método de los dos puntos con el que se obtuvo un modelo POMTM del invernadero, la metodología se inició con un paso del 10% en la entrada, tal como se ilustra en la Figura 4.2.

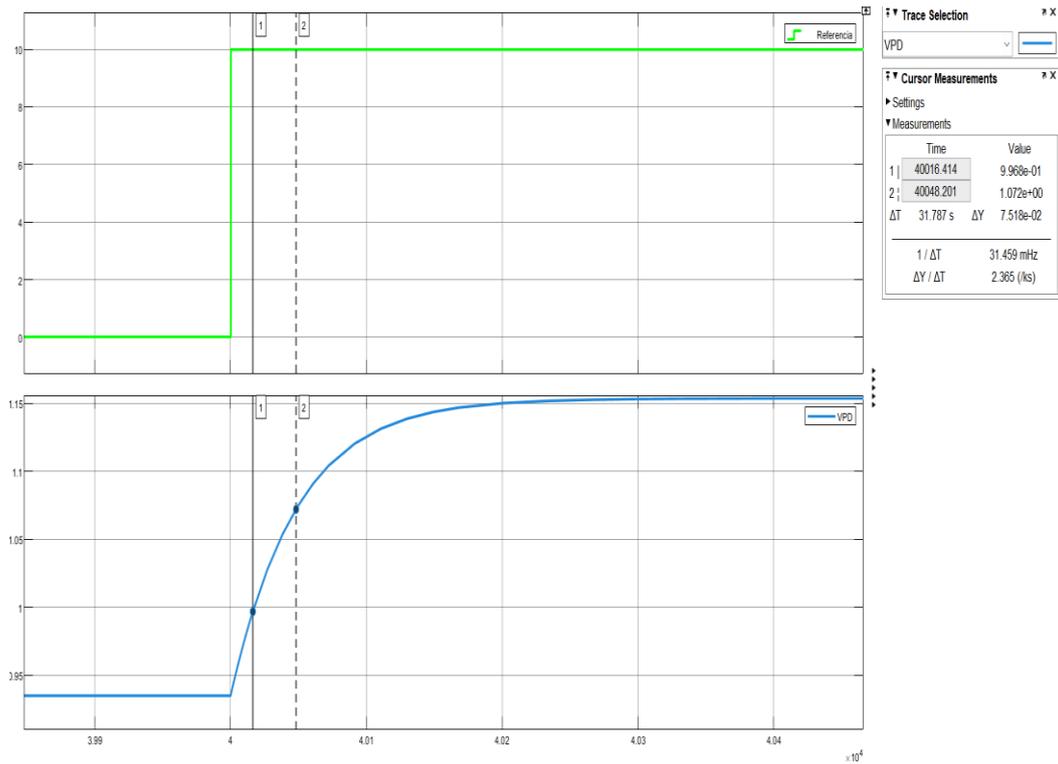


Figura 4.2. Curva de Reacción del Invernadero en Lazo Abierto al aumento de Calefacción

$$K = \frac{\Delta}{\delta} = \frac{1.154 - 0.9349}{10} = 0.02191 \left[\frac{KPa}{\%CO} \right] \quad (4.17)$$

$$\Delta_{28.3\%} = 0.9352 + \Delta * 0.283 = 0.9968 [KPa] \quad (4.18)$$

$$\Delta_{63.3\%} = 0.9352 + \Delta * 0.633 = 1.072 [KPa] \quad (4.19)$$

$$\tau = 1.5(\tau_2 - \tau_1) = 1.5(31.787) = 47.68 [s] \quad (4.20)$$

$$t_0 = t_2 - \tau = (40048.201 - 40000) - 47.68 = 0.521 [s] \quad (4.21)$$

Una vez obtenidos los parámetros dinámicos de la curva de reacción se validó el modelo aproximado.

$$G_S = \frac{K * e^{-t_0}}{\tau S + 1} = \frac{0.02191 e^{0.521 * S}}{47.68 S + 1} \quad (4.22)$$

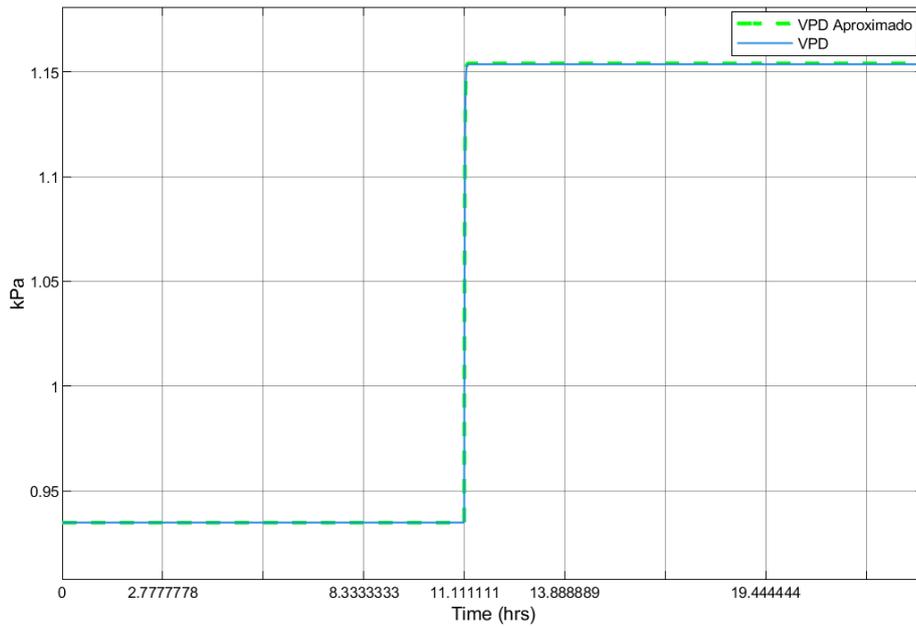


Figura 4.3. Validación del Modelo Aproximado de Calefacción

A partir de las ecuaciones (3.6) (3.7) (3.8) se diseñó el controlador PID de acuerdo

$$K_p = \frac{1}{2k} \left(\frac{t_0}{\tau} \right)^{-1} = 2088.45 \quad (4.23)$$

$$\tau_i = \tau = 47.68 \quad (4.24)$$

$$\tau_d = \frac{t_0}{2} = 0.261 \quad (4.25)$$

Se obtuvo como ley de control:

$$U = K_p \left(1 + \frac{1}{\tau_i s} + \tau_d s \right) \quad (4.26)$$

$$U = 2088.45 \left(1 + \frac{1}{47.68s} + 0.261s \right) \quad (4.27)$$

4.2.2 Controlador PID de nebulizador

Para la obtención de los parámetros del modelo FOPDT analizó la curva de reacción del sistema respecto a la variación de la entrada del sistema de enfriamiento.

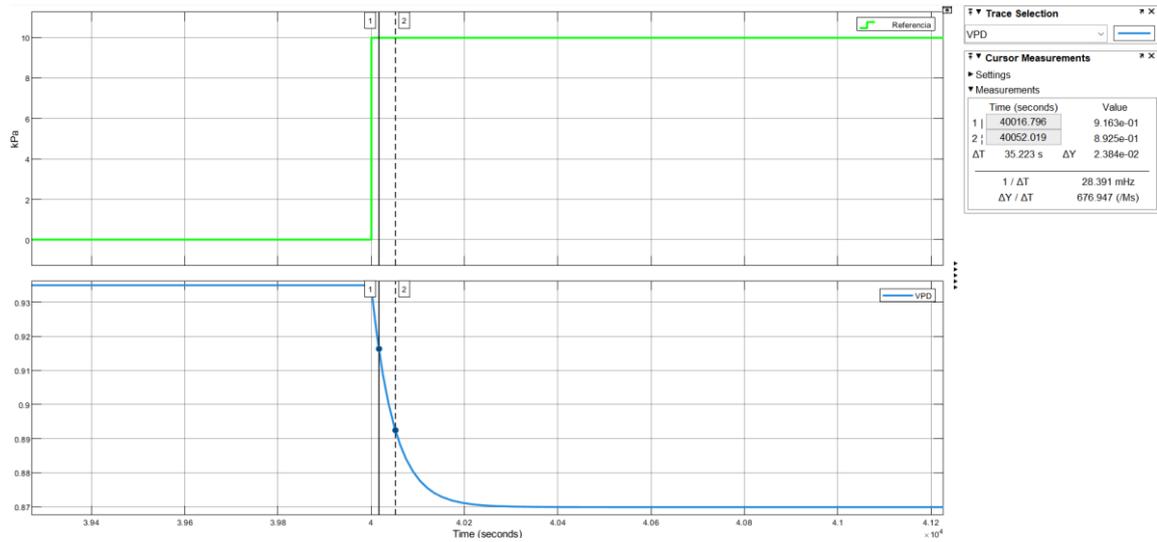


Figura 4.4. Curva de Reacción del Invernadero en Lazo Abierto al aumento de Enfriamiento

$$K = \frac{\Delta}{\delta} = \frac{0.87 - 0.9349}{10} = -0.0065 \left[\frac{KPa}{\%CO} \right] \quad (4.28)$$

$$\Delta_{28.3\%} = 0.9349 + \Delta * 0.283 = 0.9163 [KPa] \quad (4.29)$$

$$\Delta_{63.3\%} = 0.9349 + \Delta * 0.632 = 0.8925 [KPa] \quad (4.30)$$

$$\tau = 1.5(\tau_2 - \tau_1) = 1.5(35.233) = 52.84 [s] \quad (4.31)$$

$$t_0 = t_2 - \tau \approx 0 \quad (4.32)$$

Mediante los parámetros obtenidos de la curva de reacción se obtuvo el modelo aproximado del invernadero

$$G_S = \frac{K * e^{-t_0}}{\tau S + 1} = \frac{-0.0065}{52.84 S + 1} \quad (4.33)$$

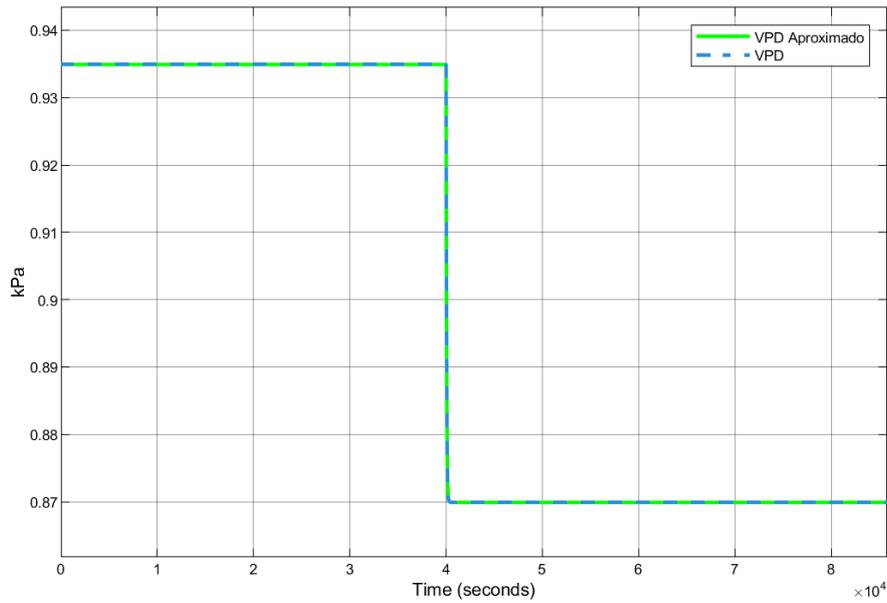


Figura 4.5. Validación del Modelo Aproximado de Enfriamiento

Con base a en las ecuaciones de la bibliografía [6] realizó la sintonización

$$\tau_i = \tau = 52.84 \quad (4.34)$$

$$\tau_c = 0.4\tau = 21.136 \quad (4.35)$$

$$K_p = \frac{\tau}{\tau_c * K} = 384.6153 \quad (4.36)$$

Se obtuvo como ley de control:

$$U = K_p \left(1 + \frac{1}{\tau_i s} \right) \quad (4.37)$$

$$U = 384.6153 \left(1 + \frac{1}{52.84s} \right) \quad (4.38)$$

4.2.3 Controlador SCM-difuso

El controlador SMC-FUZZY es una combinación de ambos tipos de controlador, se desarrollará tomando como base el controlador SMC y con el controlador Fuzzy se varió los parámetros de sintonización encontrados en el desarrollado del controlador SMC.

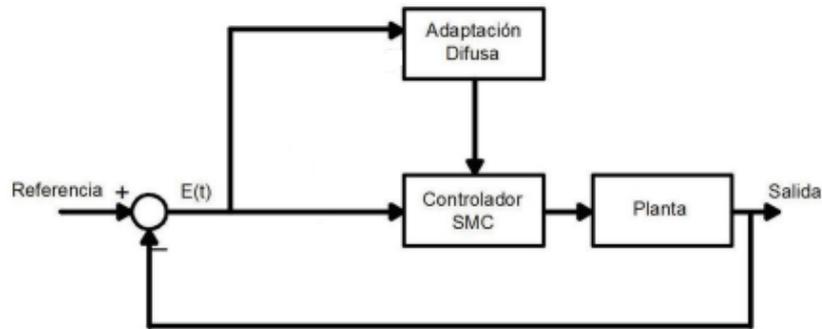


Figura 4.6. Estructura Base del Controlador SMC-Difuso

4.2.4 SCM para Calefactores

El controlador SMC se define en dos partes, una continua y una discontinua

$$U_{smc} = U_c + U_d \quad (4.39)$$

Se implementará tomando en cuenta que es un sistema de primer orden con retraso

$$\frac{Y(S)}{U(S)} = \frac{ke^{-t_0s}}{\tau s + 1} \quad (4.40)$$

Utilizando las series de Taylor para aproximar el retraso se obtiene

$$\frac{Y(S)}{U(S)} = \frac{k}{(\tau s + 1)(t_0 s + 1)} \quad (4.41)$$

Lo cual se puede expresar en forma diferencial como:

$$t_0 \tau \frac{d^2 Y(t)}{dt^2} + (t_0 + \tau) \frac{dY(t)}{dt} + Y(t) = kU(t) \quad (4.42)$$

Para la parte discontinua del controlador se debe seleccionar una superficie la cual será integral tomando en cuenta que el grado del sistema es $n=2$

$$S(t) = \left(\frac{d}{dt} + \lambda \right)^n \int e(t) dt \quad (4.43)$$

$$S(t) = \frac{de(t)}{dt} + 2\lambda e(t) + \lambda^2 \int e(t) dt \quad (4.44)$$

Donde $\lambda_1 = 2\lambda$ y $\lambda_0 = \lambda^2$

$$S(t) = \frac{de(t)}{dt} + \lambda_1 e(t) + \lambda_0 \int e(t) dt \quad (4.45)$$

Se debe tener en cuenta que el objetivo del controlador es mantener el error en 0 por lo tanto

$$\frac{dS(t)}{dt} = 0 \quad (4.46)$$

Obteniendo como superficie de deslizamiento:

$$\frac{dS(t)}{dt} = \frac{d^2e(t)}{dt^2} + \lambda_1 \frac{de(t)}{dt} + \lambda_0 e(t) = 0 \quad (4.47)$$

El error se puede expresar como la referencia menos la salida:

$$e(t) = R(t) - Y(t) \quad (4.48)$$

$$\frac{d^2R(t)}{dt^2} - \frac{d^2Y(t)}{dt^2} + \lambda_1 \left(\frac{dR(t)}{dt} - \frac{dY(t)}{dt} \right) + \lambda_0 e(t) = 0 \quad (4.49)$$

Remplazando en (4.39) en (4.32)

$$U(t) = \frac{t_0\tau}{k} \left(\left(\frac{t_0+\tau}{t_0\tau} - \lambda_1 \right) \frac{dY(t)}{dt} + \frac{1}{t_0\tau} Y(t) + \lambda_0 e(t) + \frac{d^2R(t)}{dt^2} + \lambda_1 \frac{dR(t)}{dt} \right) \quad (4.50)$$

Las derivadas de la referencia se pueden descartar sin producir mayor cambio

$$U_c(t) = \frac{t_0\tau}{k} \left(\left(\frac{t_0+\tau}{t_0\tau} - \lambda_1 \right) \frac{dY(t)}{dt} + \frac{1}{t_0\tau} Y(t) + \lambda_0 e(t) \right) \quad (4.51)$$

La parte discontinua del controlador viene definida por:

$$U_d = K_d \frac{S(t)}{|S(t)|+\delta} \quad (4.52)$$

$$S(t) = \text{signo}(k) * \left(\frac{de(t)}{dt} + \lambda_1 e(t) + \lambda_0 \int e(t) dt \right) \quad (4.53)$$

Los parámetros de sintonización de la parte discontinua [14]:

$$K_d = \frac{0.51}{k} \left(\frac{\tau}{t_0} \right)^{0.76} \quad (4.54)$$

$$\delta = 0.68 + 0.12|k|K_d\lambda_1 \quad (4.55)$$

Los parámetros de sintonización de la parte continua [14]:

$$\lambda_1 = \frac{\tau+t_0}{\tau t_0} \quad (4.56)$$

$$\lambda_0 = \frac{\lambda_1^2}{4} \quad (4.57)$$

Tomando en cuenta los valores obtenidos en (4.17) (4.20) y (4.21)

$$K = 0.02191 \left[\frac{KPa}{\%CO} \right] \quad (4.58)$$

$$\tau = 47.68 [s] \quad (4.59)$$

$$t_0 = 0.521 [s] \quad (4.60)$$

Obteniendo como resultado:

$$\lambda_1 = 1.94 \quad (4.61)$$

$$\lambda_0 = 0.94 \quad (4.62)$$

$$k_D = 720.55 \quad (4.63)$$

$$\delta = 4.35 \quad (4.64)$$

Se obtuvo como resultado la siguiente ley de control:

$$U_{smc} = (1133.79) \left[\frac{X(t)}{24.84} + 1.88 * e(t) \right] + 720.55 * \frac{S(t)}{|S(t)|+4.35} \quad (4.65)$$

$$S(t) = sing(0.02191) \left[\frac{e(t)}{dt} + 1.94e(t) + 1.88 \int e(t)dt \right] \quad (4.66)$$

4.2.5 Controlador Difuso del SMC para Calefactores

La implementación del controlador difuso se desarrolló por medio de variar los parámetros de sintonización del controlador SMC

La parte difusa del controlador se desarrolló bajo las siguientes consignas:

- La entrada del error tendrá un rango entre [-0.5 0.5] y sus funciones de membresías se observan en la Figura 4.7

- La variación de los parámetros de sintonización del controlador SMC tomara como base los valores encontrados en (4.61) hasta (4.64) y se modificara su magnitud alrededor de estos valores.

$$e = [-0.5 \ 0.5]$$

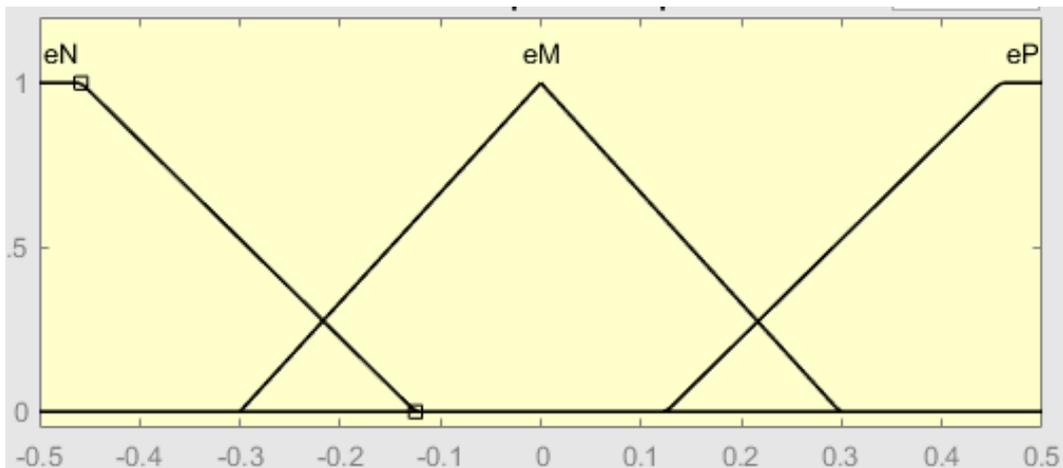


Figura 4.7. Funciones Membresías de Entrada

$$K_d = [-720.55 \ 720.55]$$

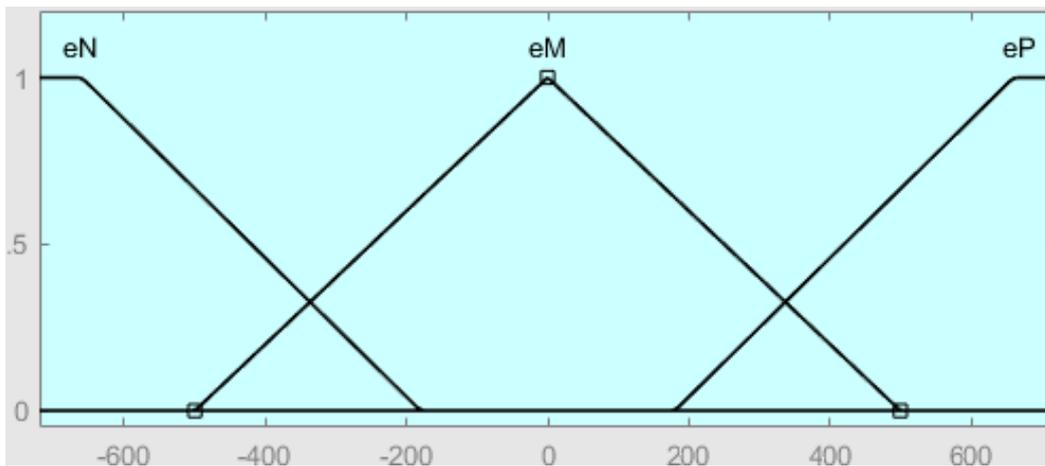


Figura 4.8. Funciones Membresías de K_d

$$\delta = [-4.35 \quad 4.34]$$

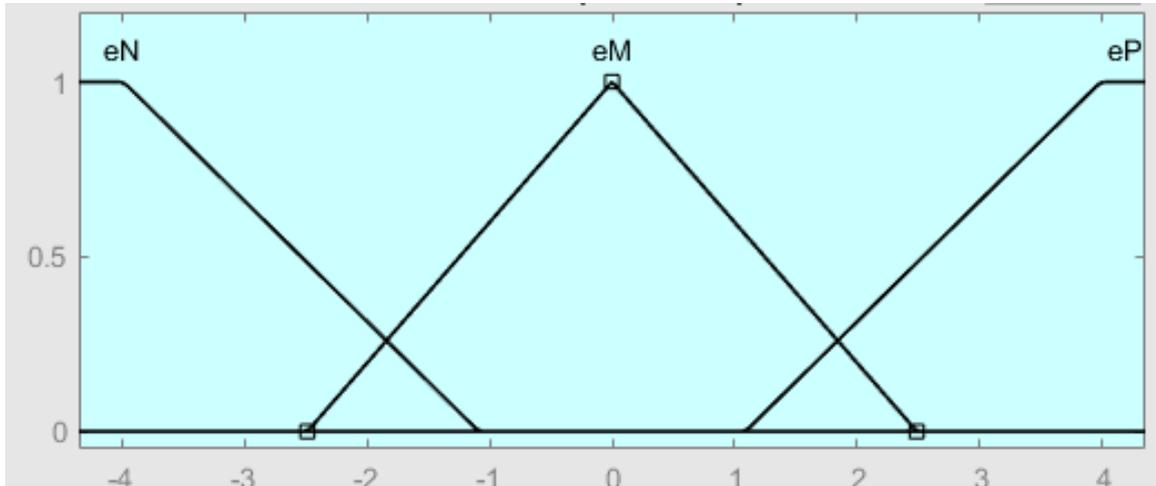


Figura 4.9. Funciones Membresías de δ

$$\lambda_0 = [-0.94 \quad 0.94]$$

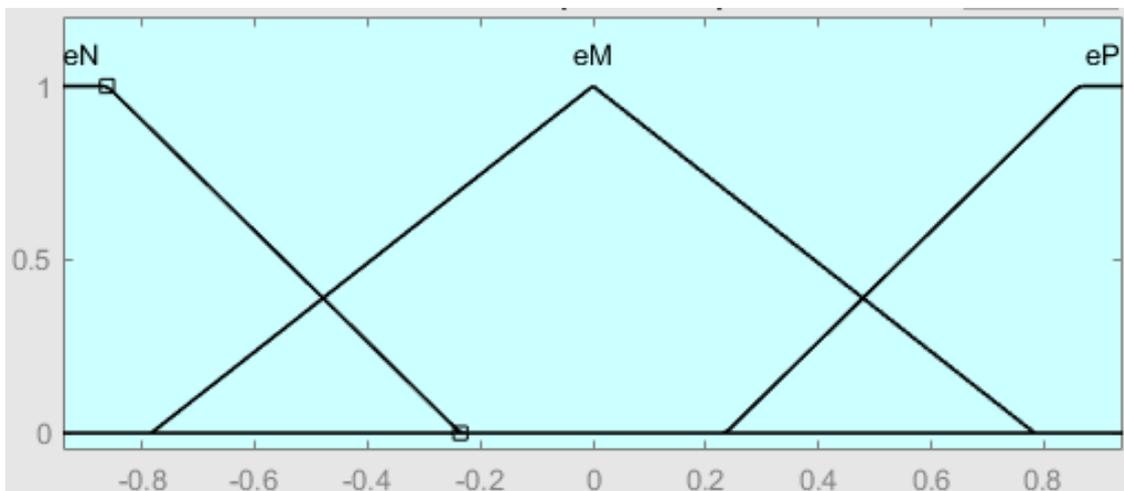


Figura 4.10. Funciones Membresías de λ_0

$$\lambda_1 = [-1.94 \quad 1.94]$$

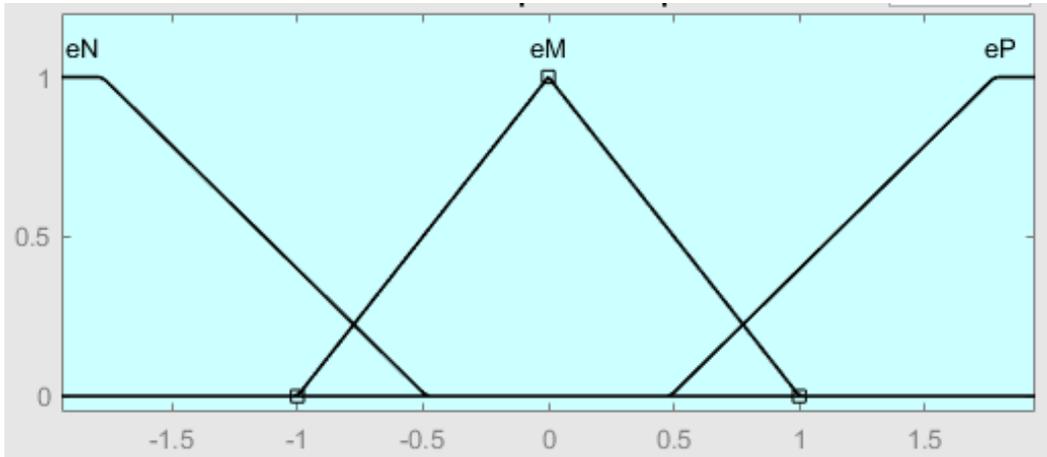


Figura 4.11. Funciones Membresías de λ_1

Las reglas que se utilizaron para la implantación del controlador difuso se presentan en la Tabla 4.4.

Tabla 4.4. Reglas Fuzzy para Calefactor

Parámetro	Entrada	Salida
K_d	eN	eP
	eM	eM
	eP	eN
δ	eN	eP
	eM	eM
	eP	eN
λ_0	eN	eP
	eM	eM
	eP	eN
λ_1	eN	eP
	eM	eM
	eP	eN

4.2.6 Controlador SMC para Nebulizador

El controlador SMC se diseñará con la premisa que el sistema es de primer orden sin retraso

$$\frac{Y(S)}{U(S)} = \frac{k}{\tau s + 1} \quad (4.67)$$

Expresando de forma diferencial se obtiene:

$$\tau \frac{dY(t)}{dt} + Y(t) = kU(t) \quad (4.68)$$

Se desarrolla la superficie de deslizamiento con la observación que se presenta un sistema de primer grado

$$S(t) = \left(\frac{de(t)}{dt} + \lambda \right)^n \int e(t) dt \quad (4.69)$$

$$S(t) = e(t) + \lambda \int e(t) dt \quad (4.70)$$

El objetivo del controlador es que el error permanezca en cero por ende se obtiene:

$$\frac{dS(t)}{dt} = 0 \quad (4.71)$$

$$\frac{dS(t)}{dt} = \frac{de(t)}{dt} + \lambda e(t) = 0 \quad (4.72)$$

El error puede ser expresado como la diferencia entre la referencia y la salida

$$e(t) = R(t) - Y(t) \quad (4.73)$$

Reemplazando se obtiene

$$\frac{dR(t)}{dt} - \frac{dY(t)}{dt} + \lambda e(t) = 0 \quad (4.74)$$

La derivada de la referencia puede ser eliminada sin que represente un mayor cambio

$$-\frac{dY(t)}{dt} + \lambda e(t) = 0 \quad (4.75)$$

Reemplazando (4.75) en (4.68)

$$U_c = \frac{\tau}{k} \left(\lambda e(t) + \frac{Y(t)}{\tau} \right) \quad (4.76)$$

La parte discontinua queda definida por

$$U_d = K_d \frac{S(t)}{|S(t)| + \delta} \quad (4.77)$$

$$S(t) = \text{signo}(k) * (e(t) + \lambda \int e(t)dt) \quad (4.78)$$

Tomando en cuenta los valores obtenidos en (4.34) y (4.36)

$$K = -0.0065 \quad (4.79)$$

$$\tau = 52.84 \quad (4.80)$$

Se obtiene

$$U_c = -8129.23 \left(\lambda e(t) + \frac{Y(t)}{52.84} \right) \quad (4.81)$$

$$U_d = K_d \frac{S(t)}{|S(t)| + \delta} \quad (4.82)$$

$$S(t) = \text{signo}(-0.0065) * (e(t) + \lambda \int e(t)dt) \quad (4.83)$$

Debido a la falta de ecuaciones para obtener λ , k_d y δ en este tipo de sistema, se procede a sintonizar estos parámetros de forma heurística, teniendo en cuenta disminuir los índices ISA e IAE, obteniendo los siguientes resultados

$$\lambda = 1.3 \quad (4.84)$$

$$k_d = 400 \quad (4.85)$$

$$\delta = 1.9 \quad (4.86)$$

Dando como resultado la ecuación de control del SMC:

$$U_c = -8129.23 \left(1.9e(t) + \frac{Y(t)}{52.84} \right) \quad (4.87)$$

$$U_d = 400 \frac{S(t)}{|S(t)| + 2} \quad (4.88)$$

$$S(t) = \text{signo}(-0.0065) * (e(t) + 1.3 \int e(t)dt) \quad (4.89)$$

4.2.7 Controlador Difuso para SMC del Nebulizador

La implementación del controlador difuso se desarrolló por medio de hacer variar los parámetros de sintonización del controlador SMC.

La parte difusa del controlador se desarrolló bajo las siguientes consignas:

- La entrada del error tendrá un rango entre [-0.5 0.5] y sus funciones de membresías se observan en la Figura 4.7

- La variación de los parámetros de sintonización del controlador SMC tomara como base el valor encontrado en (4.84) (4.85) (4.86) y se modificara su magnitud alrededor de estos valores.

$$k_d = [-400 \ 400]$$

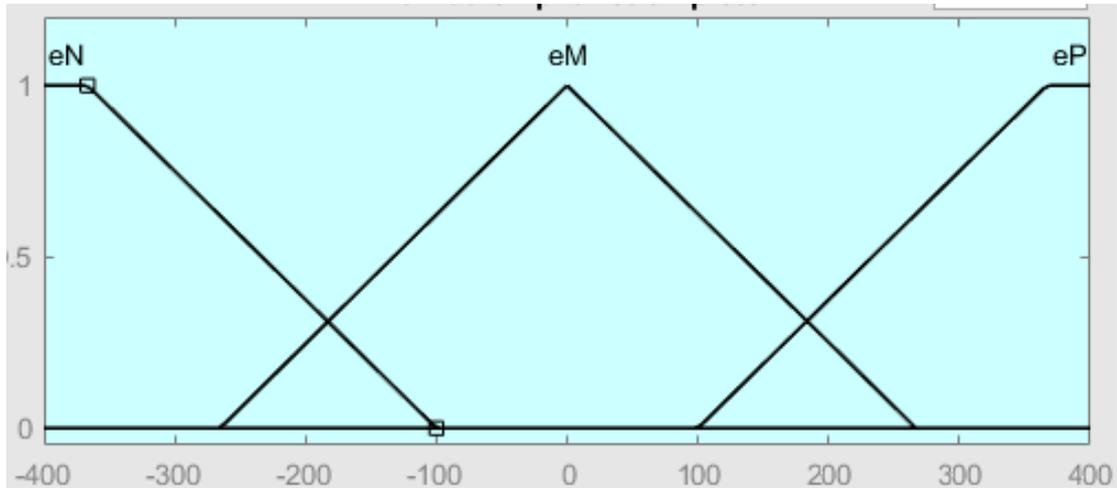


Figura 4.12. Funciones Membresías de k_d

$$\delta = [-1.9 \ 1.9]$$

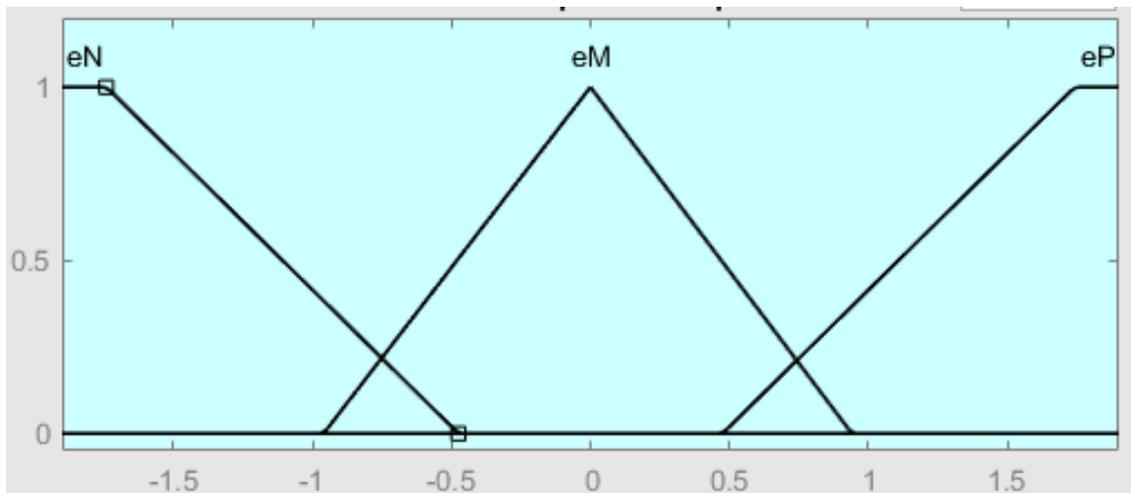


Figura 4.13. Funciones Membresías de δ

$$\lambda = [-1.3 \ 1.3]$$

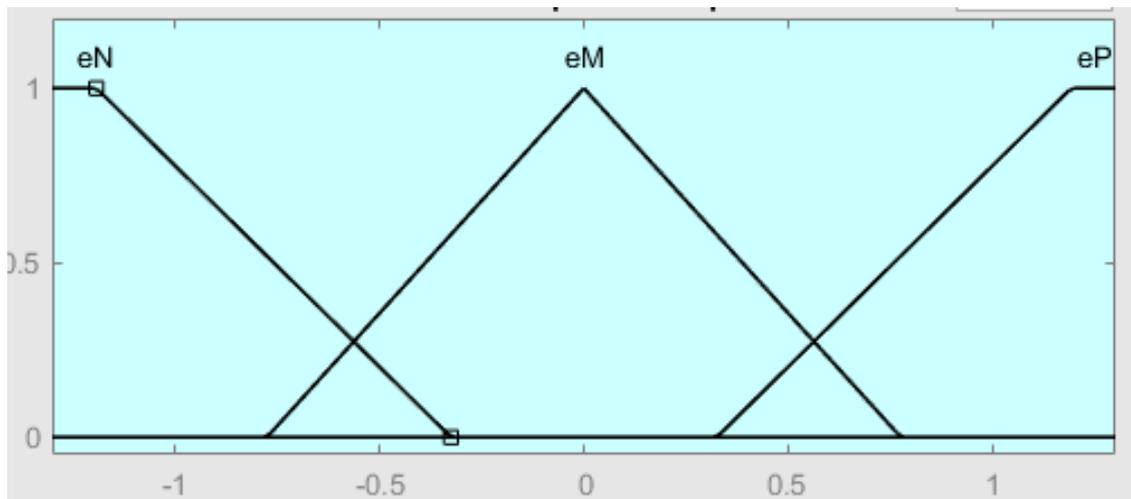


Figura 4.14. Funciones Membresías de λ

Las reglas del controlador difuso de enfriamiento se presentan en la Tabla 4.5.

Tabla 4.5. Reglas Fuzzy para Enfriamiento

Parámetro	Entrada	Salida
K_d	eN	eP
	eM	eM
	eP	eN
δ	eN	eP
	eM	eM
	eP	eN
λ	eN	eP
	eM	eM
	eP	eN

4.3 Interfaz Modelo del Invernadero

La interfaz se desarrolla con el aplicativo de Guide de Matlab mientras que la implementación del modelo del invernadero y los controladores se utiliza Simulink para su simulación.

La interfaz cuenta con 3 diferentes ventanas: pantalla inicial, pantalla de simulación y pantalla de parámetros.

a) Pantalla inicial:



Figura 5.1. Pantalla Inicial

La primera ventana permite interactuar con tres botones

- *Simulación:* el objetivo de este botón es redirigir hacia la ventana de simulación.
- *Datos:* este botón abre una ventana en la cual se puede modificar parámetros generales del modelo del invernadero.
- *Salir:* cierra la interfaz.

b) Pantalla de simulación

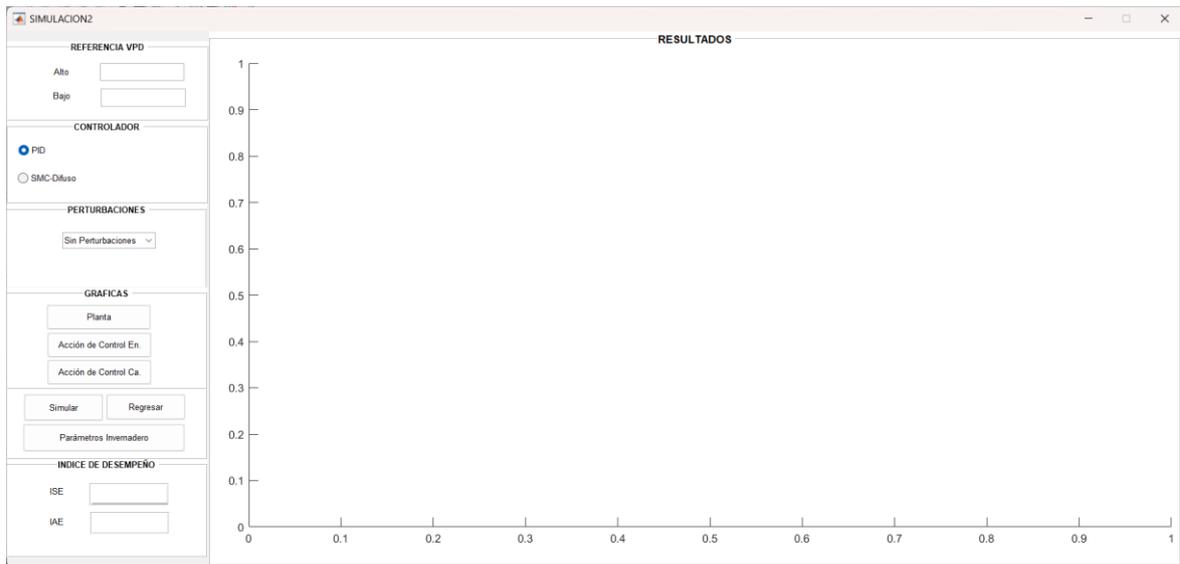


Figura 5.2. Pantalla de Simulación

La ventana de simulación está compuesta por varias secciones con las cuales se pueden interactuar, descritas a continuación:

- *Resultados:* esta sección presenta las respuestas del modelo del invernadero
- *Referencia VPD:* son cuadros de texto que permiten la entrada del valor referencia que el invernadero debe estar, existen dos opciones un valor de referencia alto y otro valor bajo, estos se colocan dependiendo los rangos óptimos de los cultivos.
- *Controlador:* esta sección de la posibilidad de intercambiar entre los controladores PID y SMC-Difuso.
- *Perturbaciones:* se puede seleccionar entre 5 opciones, una sin perturbaciones y otras cuatro opciones de datos reales obtenidos en un invernadero situado en Venezuela.
- *Gráficas:* permite alternar entre diferentes gráficas, mostrando la respuesta del invernadero, la acción de control de enfriamiento y la acción de control de calefacción.
- *Índice de desempeño:* muestra los índices de desempeño ISA e IAE de los controladores dependiendo cual este seleccionado.
- *Simular:* envía los datos seleccionados y establecidos en la ventana al modelo del invernadero e inicia con la simulación
- *Regresar:* redirige a la ventana inicial de la interfaz
- *Parámetros Invernadero:* redirige a la ventana de modificación de los parámetros generales del modelo del invernadero.

c) Pantalla de Parámetros del Invernadero

Figura 5.3. Pantalla Parámetros del Invernadero

Permite modificar los parámetros que se muestran en la Figura 5.3., los materiales de cubierta tienen valores preestablecidos como se observa en la Tabla 5.1.

Tabla 5.1. Material de Cubierta del Invernadero[12]

Material	Transmisión de Calor	Transmitancia del Material
Vidrio Simple	6.2	90%
Polietileno Simple	6.2	87%
Policarbonato Simple	6.2	87%
Polietileno Doble	4	76%
Policarbonato Doble	3.3	79%

Además, la ventana cuenta con 3 botones

- *Simulación:* redirigí a la ventana de simulación
- *Regresar:* redirigí a la ventana inicial de la interfaz
- *Restablecer parámetros iniciales:* permite colocar los valores con los que se desarrolló el presente trabajo.

4.3.1 Modelo del Invernadero

El modelo constituido por las ecuaciones (4.1) hasta (4.16) se desarrolla en forma de bloques en el software Matlab/Simulink como se muestra en la Figura 5.4.

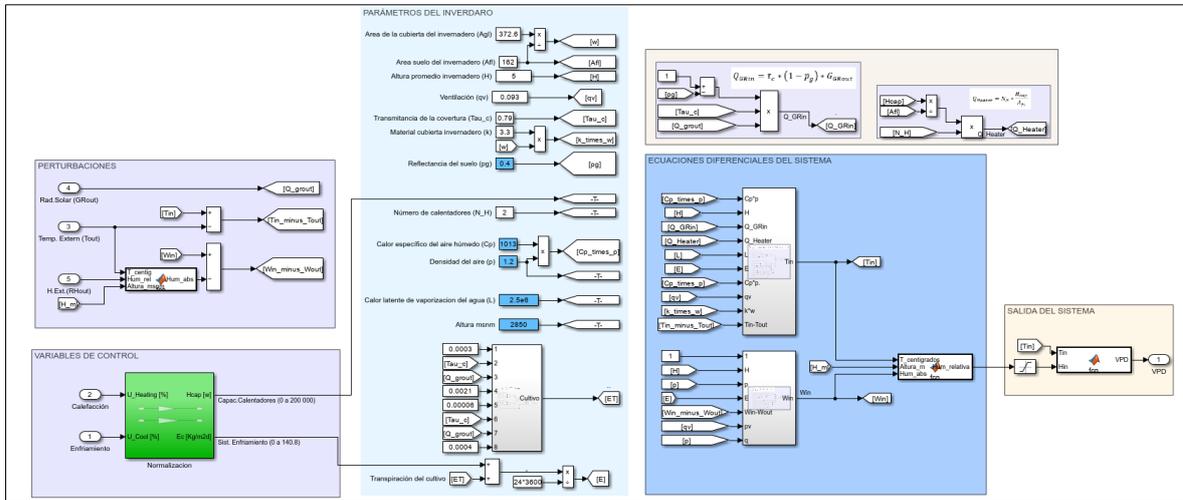


Figura 5.4. Modelo del Invernadero

5 RESULTADOS, CONCLUSIONES Y RECOMENDACIONES

5.1 Resultados

5.1.1 Respuesta del controlador PID

El controlador PID consta de dos partes, una controlador PID de calefactor y un controlador PID de enfriamiento.

En la Figura 5.4. se observa los resultados de la simulación para una entrada de 1.2 kPa y 0.5 kPa sin presencia de perturbaciones.

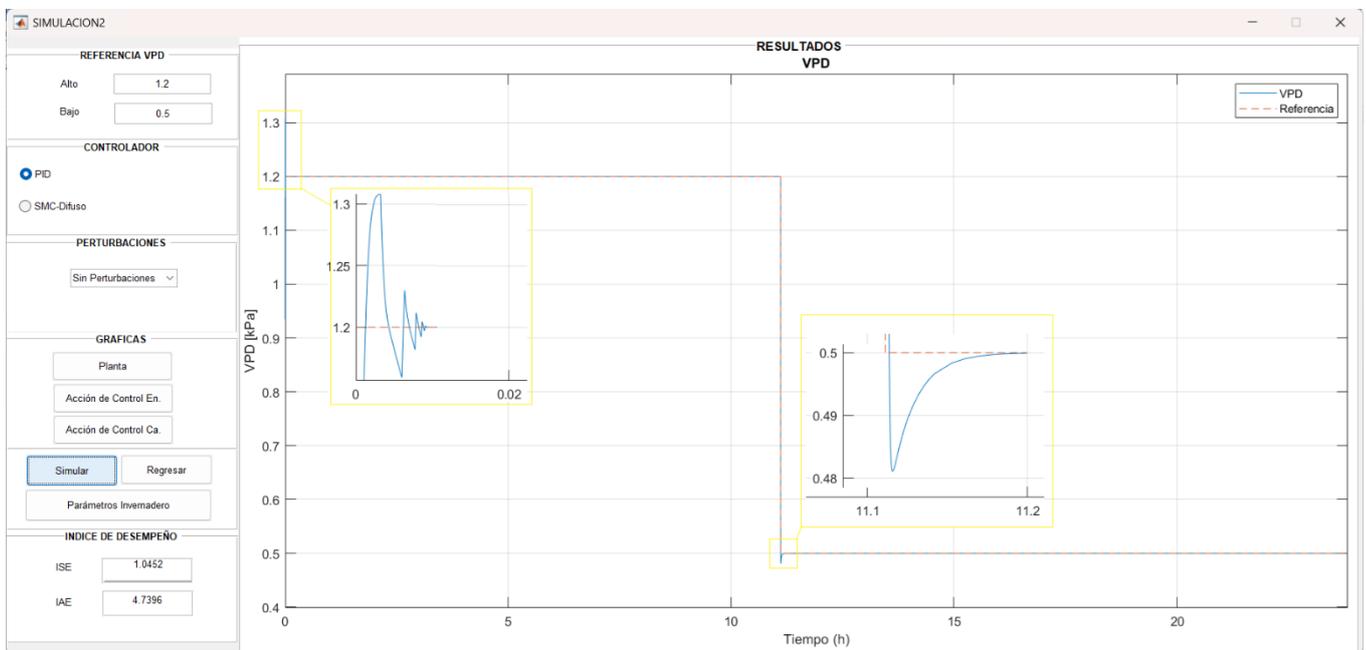


Figura 5.4. Respuesta del Invernadero sin Perturbaciones (PID)

El controlador PID es capaz de seguir la referencia establecida presentando un sobrepico y unas oscilaciones como se observa en la Figura 5.4 lo cual aumenta su tiempo de establecimiento mientras que en el cambio de referencia a 0.5 kPa se presenta un subpico, pero los corrige y sigue la referencia sin dificultades mostrando un buen desempeño

Las acciones de control tanto de enfriamiento como calefacción solo presentan permutaciones en el momento del cambio de referencia debido a que no existen perturbaciones en el sistema. En la Figura 5.6. y Figura 5.7. se observan modificaciones bruscas durante este cambio de referencia. Esto se debe a que el cambio es en forma de paso, lo que genera un error considerable, provocando que el controlador realice una acción intensa para corregir dicho error.

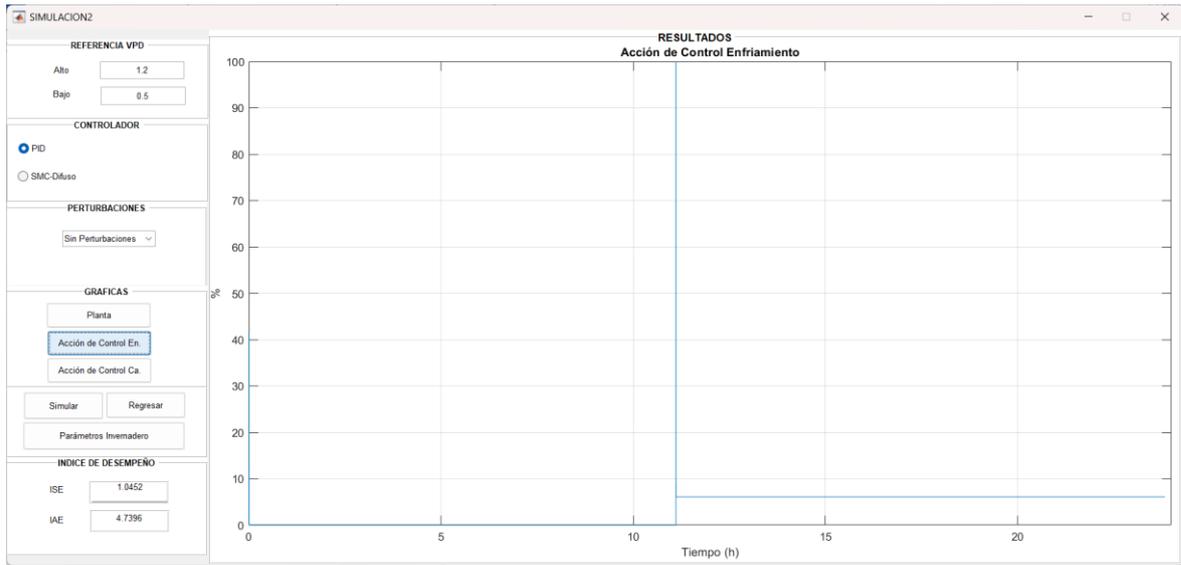


Figura 5.6. Acción de Control Enfriamiento

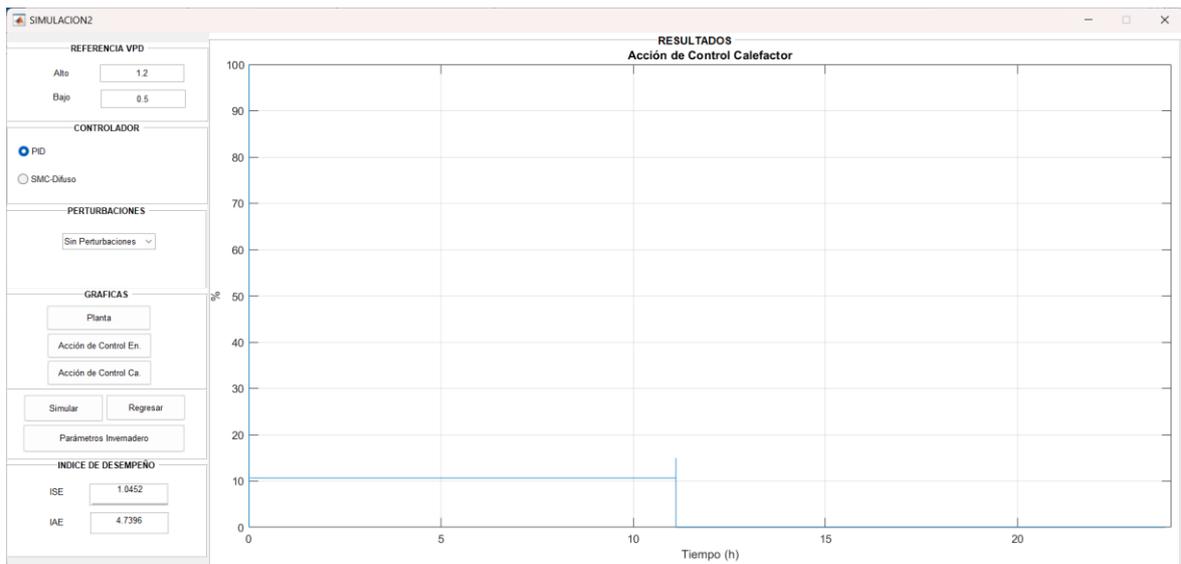


Figura 5.7. Acción de Control Calefactor

En la Figura 5.8. se observa los resultados de la simulación para una entrada de 1.2 kPa y 0.5kPa con la presencia de perturbaciones Figura 5.9.

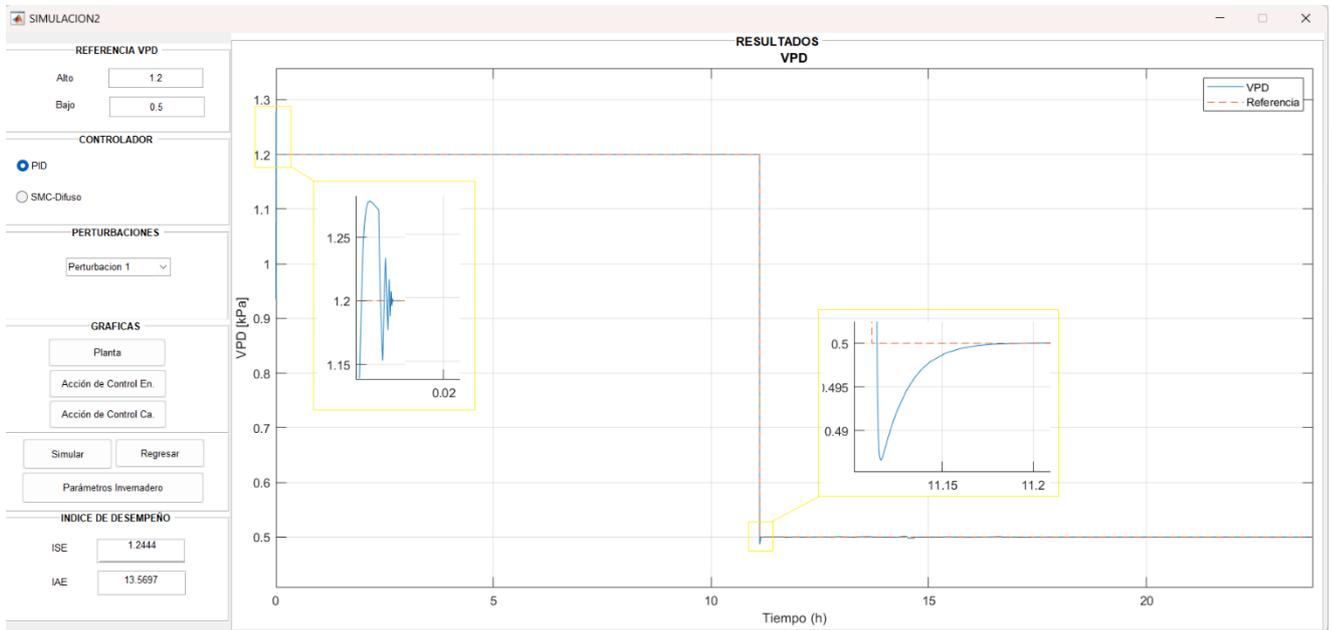


Figura 5.8. Respuesta del Invernadero con Perturbaciones (PID)

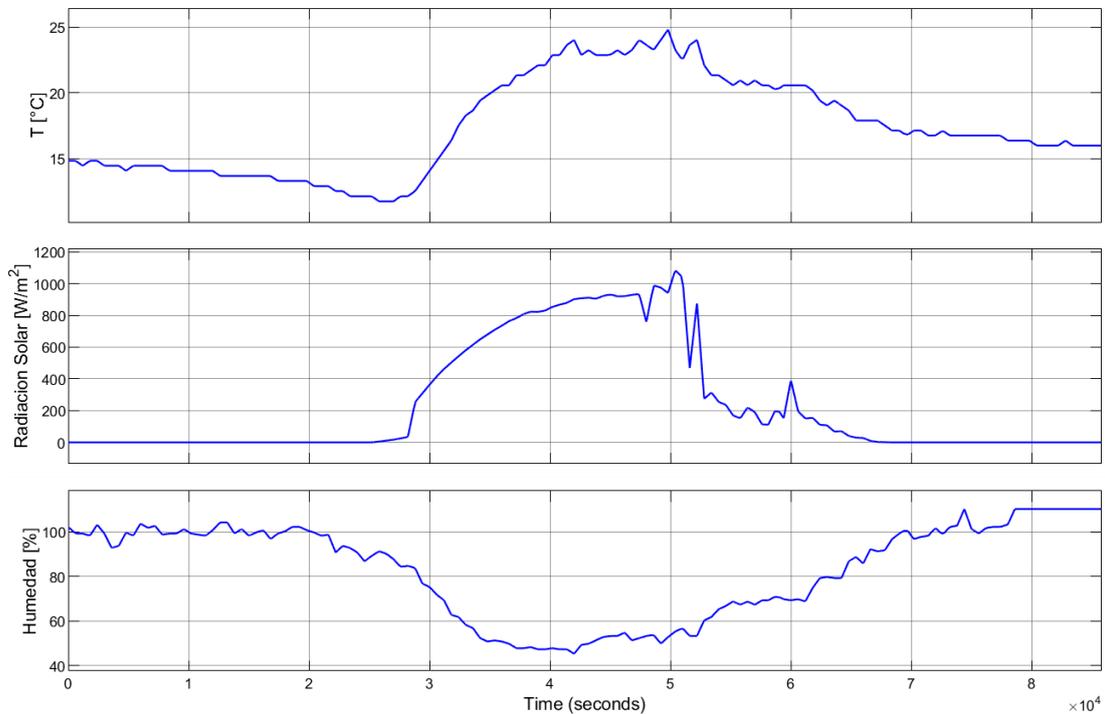


Figura 5.9. Perturbaciones

Con el aumento de las perturbaciones, se puede observar en la Figura 5.8 que el VPD alto experimenta un sobrepico seguido de oscilaciones, mientras que el VPD bajo muestra un subpico, ambos transitorios el controlador los corrige. Sin embargo, en el caso del VPD bajo, se observan además dificultades del controlador para seguir la referencia, produciendo inestabilidad dentro del sistema.

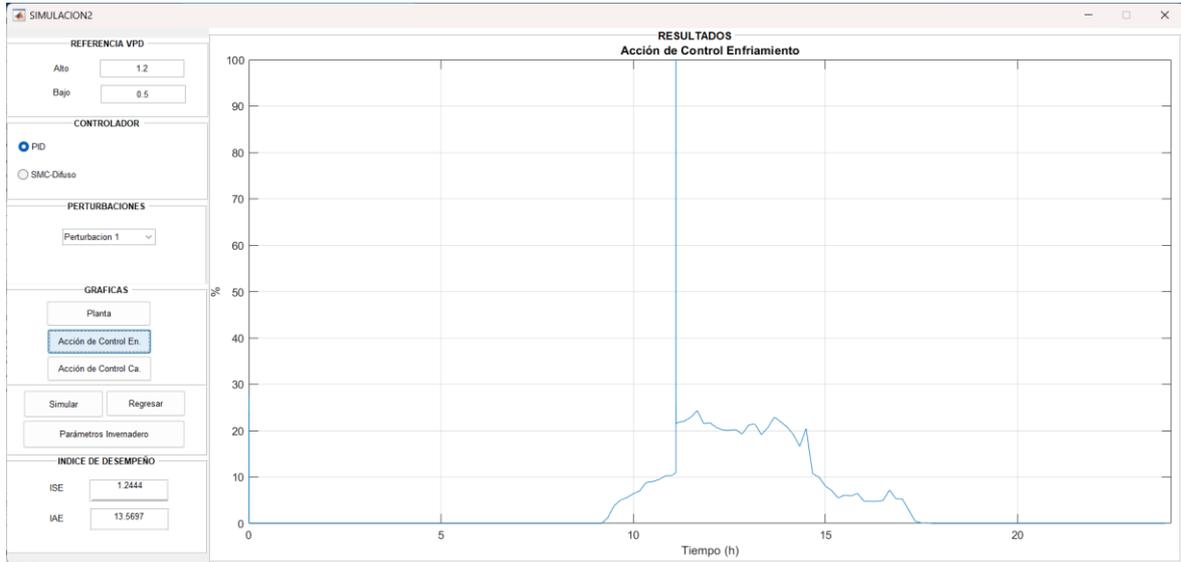


Figura 5.10. Acción de Control Enfriamiento con Perturbaciones

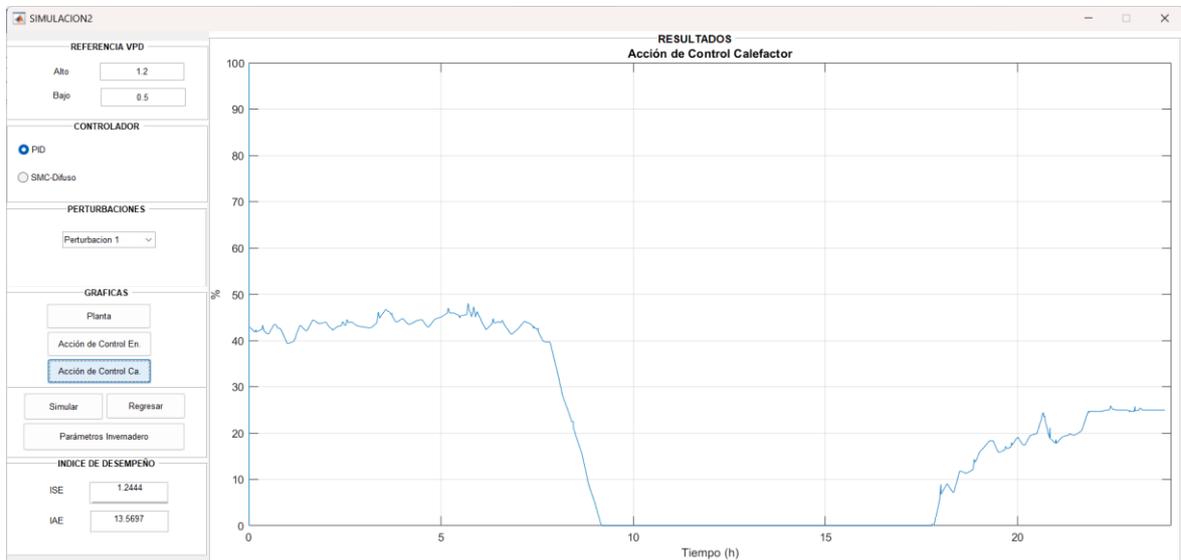


Figura 5.11. Acción de Control Enfriamiento con Perturbaciones

Las acciones de control tanto del enfriamiento como de la calefacción ya no solo trabajan durante el cambio de referencia sino trabajan constantemente para tratar de seguir la referencia.

5.1.2 Respuesta del controlador SCM-difuso

De igual manera que en el apartado anterior se toma como referencia los valores de VPD para alto de 1.2 kPa y de bajo de 0.5 kPa sin perturbaciones como se observa en la Figura 5.12.

De los resultados obtenidos de la simulación se puede determinar que el controlador SMC-Difuso cumple con su objetivo de mantener al invernadero en la referencia deseada, en la Figura 5.12 se observa un sobrepico que ocurre al momento del cambio de referencia que se estabiliza de forma rápida de igual manera en el cambio de referencia bajo se presenta el mismo caso.

Las acciones de control tanto de enfriamiento como calefacción responden para estabilizar de forma eficaz los cambios de referencia, se puede determinar que el sistema no presenta perturbaciones debido a que las acciones de control son constantes.

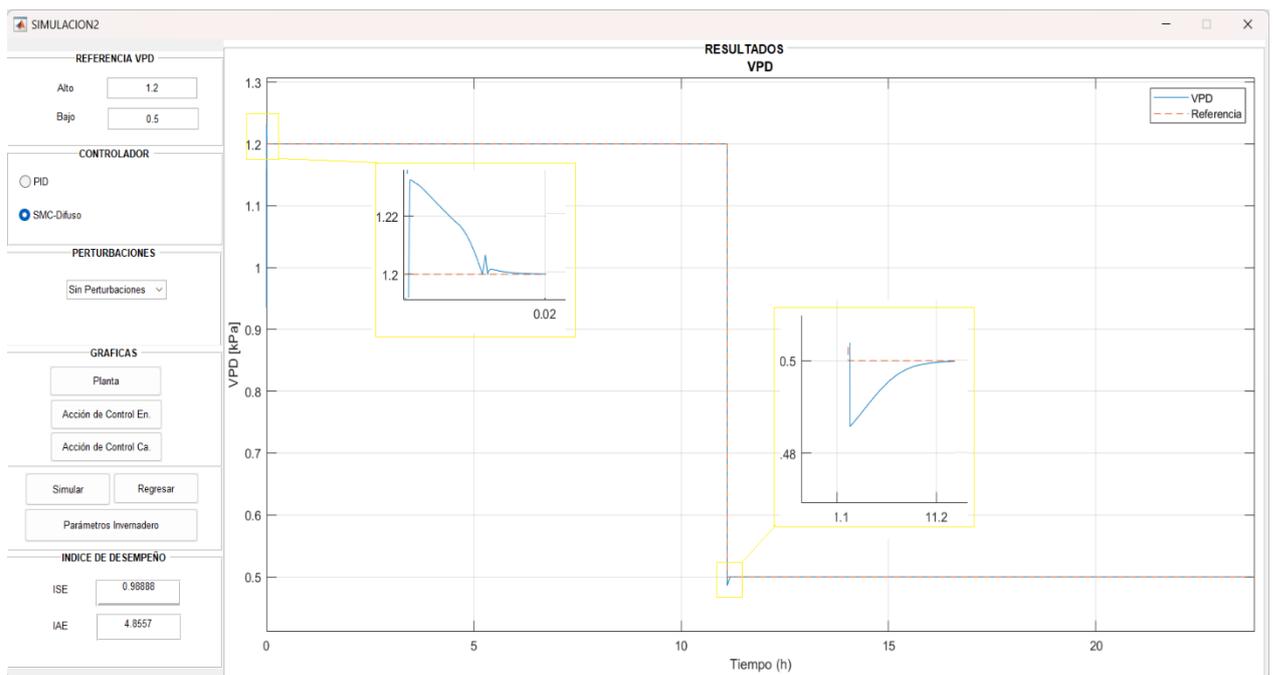


Figura 5.12. Respuesta del Invernadero sin Perturbaciones (smc-difuso)

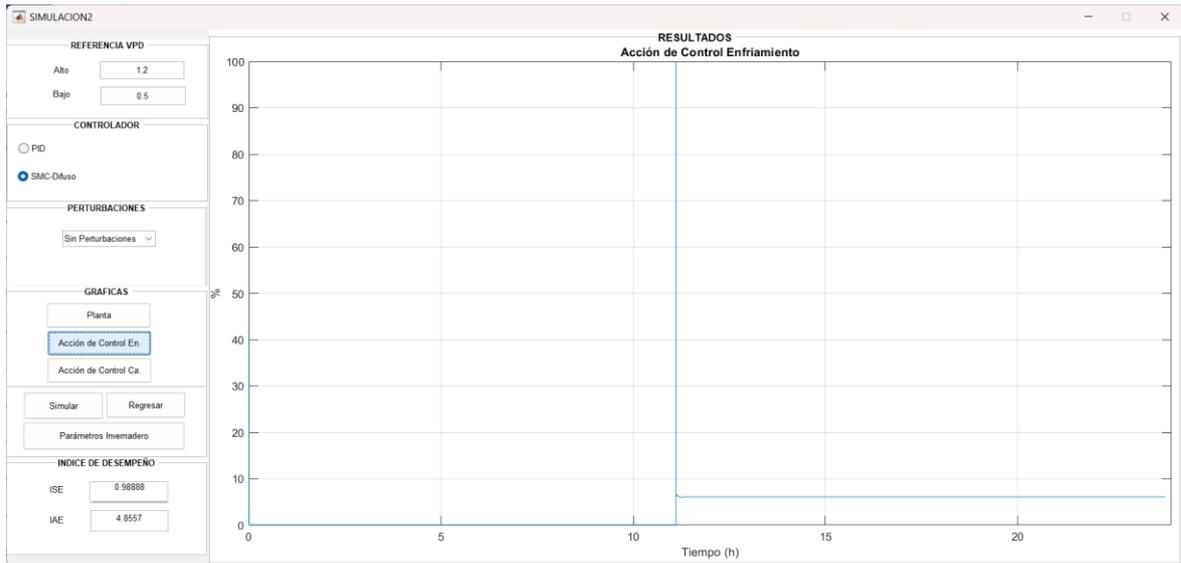


Figura 5.13. Acción de Control de Enfriamiento sin Perturbaciones

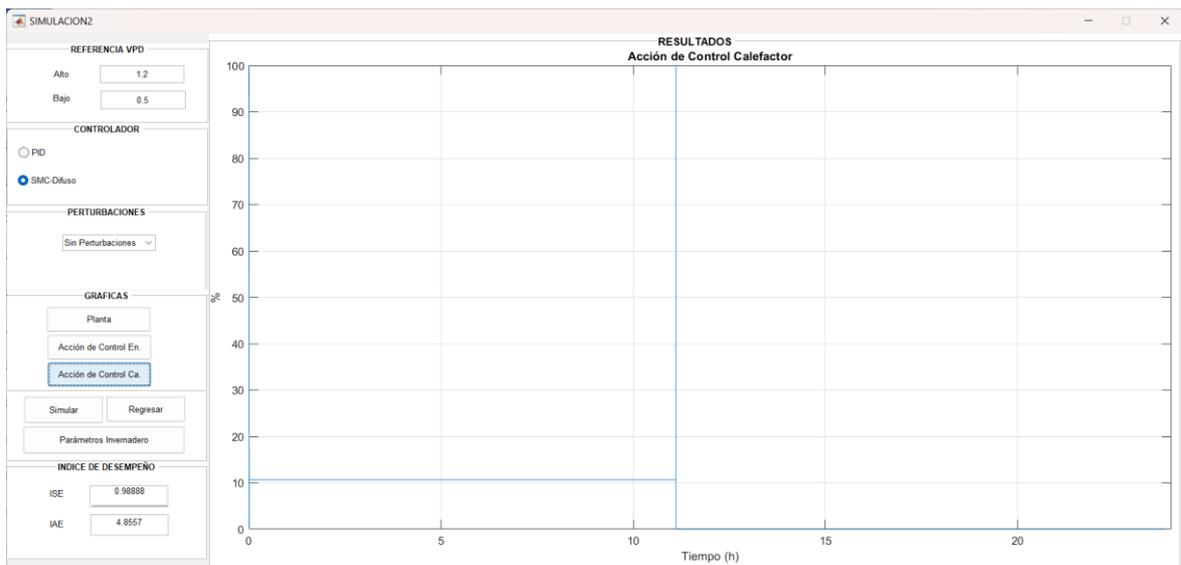


Figura 5.14. Acción de Control de Calefactor sin Perturbaciones

En la Figura 5.15. se observa la respuesta del modelo del invernadero a las referencias 1.2 kPa y 0.5 kPa en presencia de perturbaciones mostrada en la Figura 5.9.

El controlador SMC-Difuso presenta un buen desempeño siguiendo la referencia a pesar de la presencia de las perturbaciones, pero como se puede observar en la Figura 5.15. se presenta un sobrepico en el cambio de referencia, pero además se observa otro pico producto de las perturbaciones, en la referencia baja se puede observar el mismo comportamiento dando como resultado un peor desempeño como muestran los índices de desempeño ISE e IAE

Por otro lado, las acciones de control se encuentran más activas debido a la presencia de las perturbaciones, estas realizan trabajos más fuertes con el fin de mantener estable el déficit de presión de vapor del invernadero.

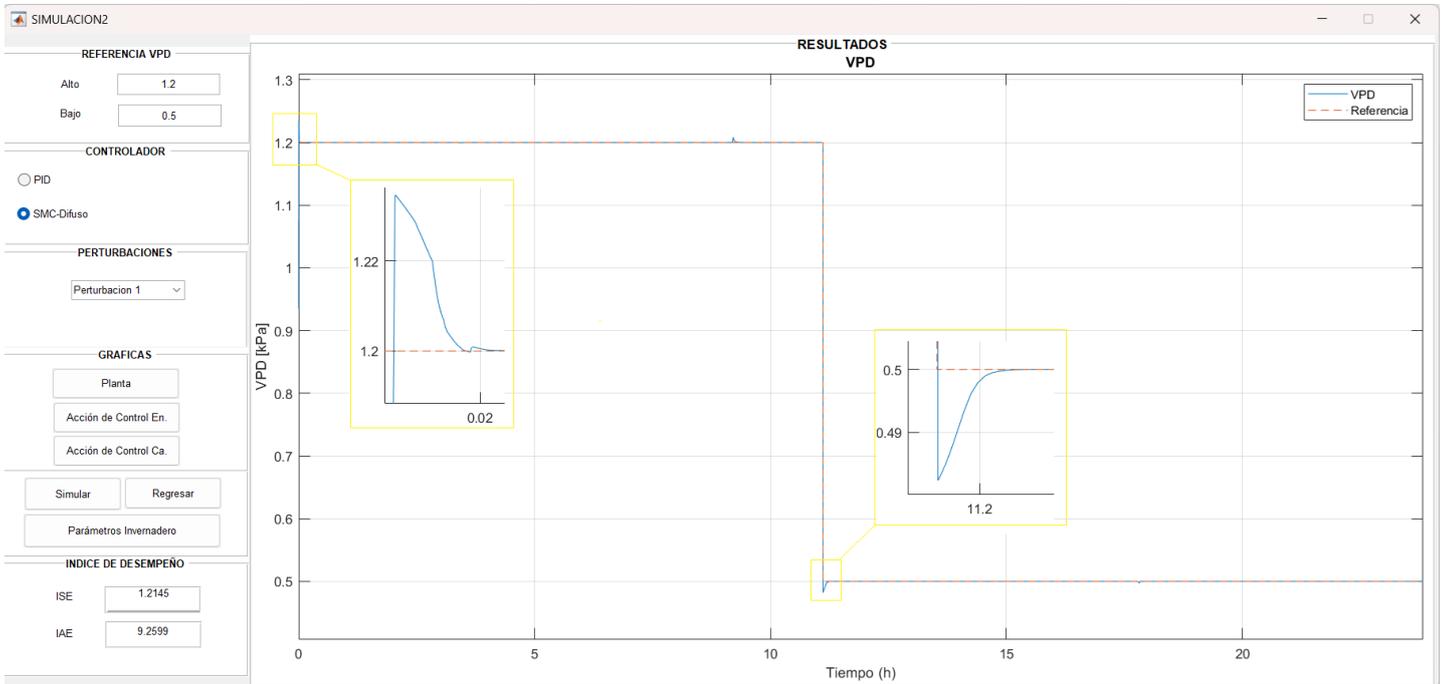


Figura 5.15. Respuesta del Invernadero con Perturbaciones (smc-difuso)

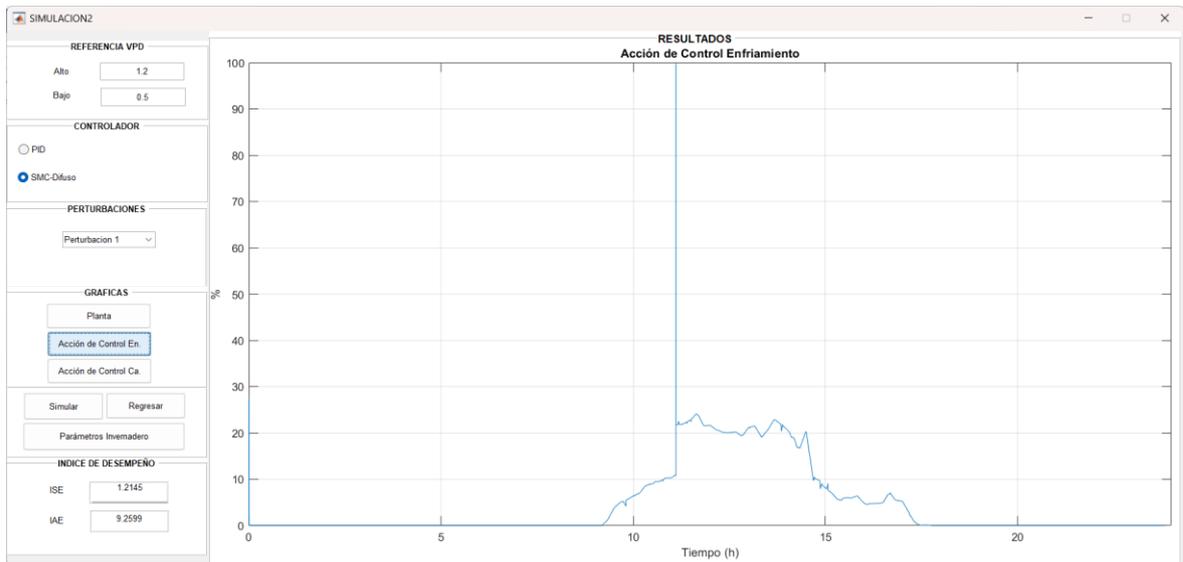


Figura 5.16. Acción de Control Enfriamiento con Perturbaciones

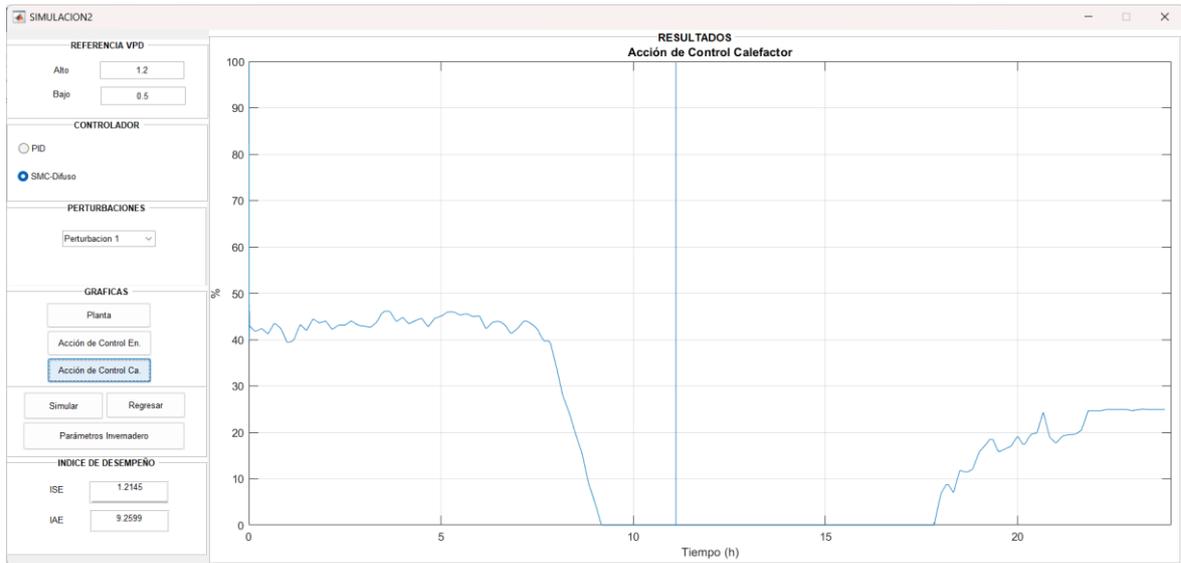


Figura 5.17. Acción de Control Calefactor con Perturbaciones

5.1.3 Desempeño de los controladores

El desempeño de los controladores se los analiza por medio de los índices ISE e IAE los cuales interactúan directamente con el error presente en el modelo y brindan un valor cuantitativo lo que facilita realizar una comparación entre los controladores

Para comparar los controladores PID y SMC-Difuso se realizó las mismas pruebas de simulación para obtener sus índices de funcionamiento ante situaciones de no tener presente perturbaciones y ante diferentes perturbaciones obteniendo como resultado lo presente en la Tabla 5.2.

Tabla 5.2. Índices de Desempeño PID y SMC-Difuso

Perturbaciones	PID		SMC-Difuso		Comparación	
	ISE	IAE	ISE	IAE	% ISE	% IAE
Sin Perturbaciones	1.0452	4.7396	0.9888	4.8557	5,70%	-2,39%
Perturbación 1	1.2444	13.5697	1.2145	9.2599	2,46%	46,54%
Perturbación 2	1.2396	13.8152	1.2142	9.7017	2,09%	42,40%
Perturbación 3	1.3059	16.5427	1.2757	9.7998	2,37%	68,81%
Perturbación 4	1.1367	18.6076	1.0931	8.9091	3,99%	108,86%

El desempeño de los controladores sin perturbaciones fue similar. El SMC-Difuso presentó un ISE ligeramente menor que el PID, con una diferencia del 5.7%. Sin embargo, el índice

IEA del PID fue levemente menor, con un valor del 2.39% más bajo. Dado que ambos presentaron valores similares, ninguno de los dos presenta ventaja sobre el otro.

Por otro lado, al incluir las diferentes perturbaciones, ambos controladores presentaron un ISE parecido siendo ligeramente mejor el del controlador SMC-Difuso, con el PID mostrando un valor hasta un 3.99% mayor. Esto muestra que ambos controladores corrigen de forma satisfactoria los sobrepicos y subpicos que se producen en el sistema. Sin embargo, al comparar el índice IEA se observó que el SMC-Difuso es superior al PID, presentando el PID valores hasta 108.86% mayores. El controlador SMC-Difuso muestra superioridad al controlador perturbaciones dando como resultado una mayor estabilidad y robustez frente al PID.

5.2 Conclusiones

Se definió e implementó un modelo matemático para describir el comportamiento del microclima de un invernadero.

Se diseñaron dos controladores: un PID y un SMC-Difuso, así como una interfaz gráfica que permitió observar el comportamiento de ambos controladores bajo diferentes perturbaciones

El controlador SMC-Difuso es una mejor opción ya que demostró tener mejor capacidad de manejar sistemas con incertidumbre, lo que lo vuelve idóneo para trabajar en entornos donde se presentan cambios constantes, como el caso particular de un invernadero; mientras que el PID es un controlador sencillo de implementar que puede ser una buena opción cuando el sistema no presenta perturbaciones fuertes

5.3 Recomendaciones

Se recomienda comparar al controlador SMC-Difuso contra otros controladores avanzados como podrían ser controladores predictivos, adaptativos, etc. para refinar las técnicas de control en este campo.

Se recomienda poner a prueba a los controladores PID y SMC-Difuso en otros campos industriales.

Se recomienda, en caso de implementación, aplicar un cambio de referencia gradual mediante una rampa, esto evita picos bruscos en la respuesta del sistema, lo que puede prevenir posibles daños en los elementos de control.

6 REFERENCIAS BIBLIOGRÁFICAS

- [1] Banco Central del Ecuador, “Evolución de la Balanza Comercial por Productos”, mar. 2024.
- [2] F. Beluzán, “Déficit de presión de vapor (DPV) y factores microclimáticos como herramientas de pronóstico”, Universidad Austral de Chile, Chile, 2013.
- [3] M. Sultan, H. Ashraf, T. Miyazaki, R. R. Shamsiri, y I. A. Hameed, “Temperature and Humidity Control for the Next Generation Greenhouses: Overview of Desiccant and Evaporative Cooling Systems”, en *Next-Generation Greenhouses for Food Security*, IntechOpen, 2021. doi: 10.5772/intechopen.97273.
- [4] V. Alfaro, “Identificación de procesos sobreamortiguados utilizando técnicas de lazo abierto”, *Revista de la Universidad de Costa Rica*, vol. 11, núm. 1,2, pp. 12–25, 2001.
- [5] K. Tehrani y A. Mpanda, “PID Control Theory”, en *Introduction to PID Controllers - Theory, Tuning and Application to Frontier Areas*, France: InTech, 2012. doi: 10.5772/34364.
- [6] O. Camacho, A. Rosales, y F. Rivas, *Control de Procesos*, Primera. Quito: EPN Editorial, 2020.
- [7] G. Tejada, “Lógica Fuzzy”, Lima, 2000.
- [8] D. Baquero y C. Ochoa, “Diseño y simulación de un control adaptativo difuso basado en modos deslizantes para sistemas aproximables a un sistema de primer orden con retardo”, Escuela Politécnica Nacional, Quito, 2017.
- [9] O. Camacho, C. Smith, y W. Moreno, “Development of an internal model sliding mode controller”, *Ind Eng Chem Res*, vol. 42, núm. 3, pp. 568–573, feb. 2003, doi: 10.1021/ie010481a.
- [10] V. Orosco, “Desarrollo de una herramienta computacional para la sintonización de parámetros de controladores PID y SMC para el seguimiento de trayectoria de un cuadricóptero basado en algoritmos genéticos”, Quito, 2018.
- [11] Ramón E. Jaimez, Leyde Y. Briceño, y Wilmer E. Espinoza, “INFLUENCIA DE LA CONDICIÓN CLIMÁTICA DE DIFERENTES LOCALIDADES EN EL MICROCLIMA DEL INVERNADERO: REGIÓN ANDINA Y CENTRAL DE VENEZUELA”, *Interciencia*, vol. VOL. 35, pp. 380–384, 2010.
- [12] E. Fitz-Rodríguez, C. Kubota, G. A. Giacomelli, M. E. Tignor, S. B. Wilson, y M. McMahon, “Dynamic modeling and simulation of greenhouse environments under several scenarios: A web-based application”, *Comput Electron Agric*, vol. 70, núm. 1, pp. 105–116, ene. 2010, doi: 10.1016/j.compag.2009.09.010.
- [13] R. Pahuja, H. K. Verma, y M. Uddin, “Implementation of greenhouse climate control simulator based on dynamic model and vapor pressure deficit controller”, *Engineering in Agriculture, Environment and Food*, vol. 8, núm. 4, pp. 273–288, oct. 2015, doi: 10.1016/j.eaef.2015.04.009.
- [14] O. Camacho y A. Rosales, “Ecuaciones de Sintonización para Controlador por Modos Deslizantes y Control de Matriz Dinámica a partir de un Módulo Difuso”, 2015. [En línea]. Disponible en: <https://www.researchgate.net/publication/286450559>

- [15] J. Slotine y W. Li, *Applied nonlinear control*. Prentice Hall, 1991.
- [16] B. Cajamarca, O. Camacho, D. Chávez, P. Leica, y M. Pozo, “Sliding Mode Control Based on Internal Model for a Non-minimum phase Buck and Boost Converter”, *Enfoque UTE*, vol. 10, núm. 1, pp. 41–53, 2019, doi: 10.29019/enfoqueute.v10n1.420.
- [17] A. Bastidas, “Diseño y simulación de un control dual por modos deslizantes para control de un tanque de neutralización de pH”, Escuela Politécnica Nacional, Quito, 2022.
- [18] J. Gutiérrez, “Control de déficit de presión de vapor (DPV) en cultivos de invernadero con sistemas de nebulización”, Universidad Autónoma Chapingo, México, 2020.

7 ANEXOS

ANEXO I. Modelo Matemático del Invernadero en Matlab.

ANEXO II. Código Interfaz

ANEXO I

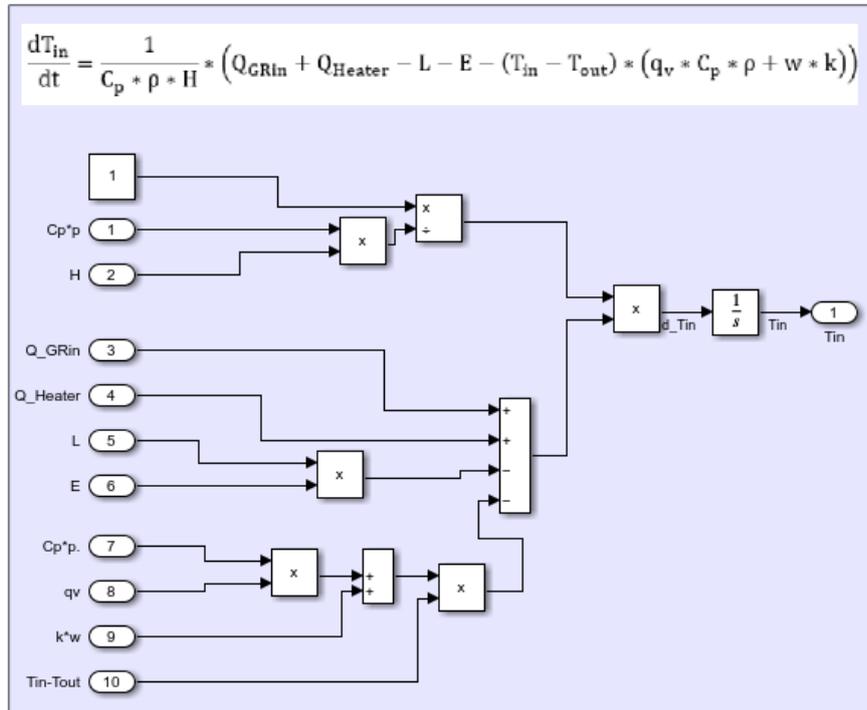


Figura A.1. Temperatura del Invernadero

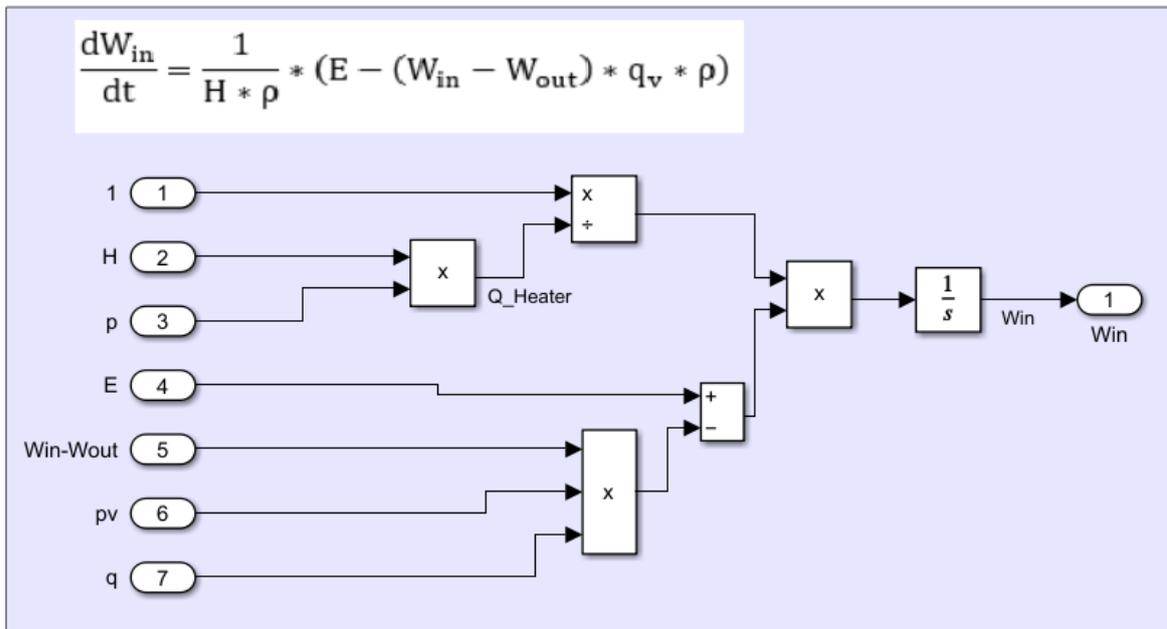


Figura A.2. Humedad del Invernadero

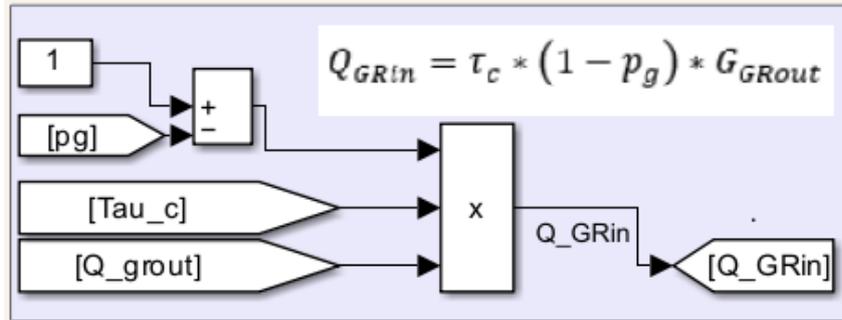


Figura A.3. Radiación Solder Interna del Invernadero

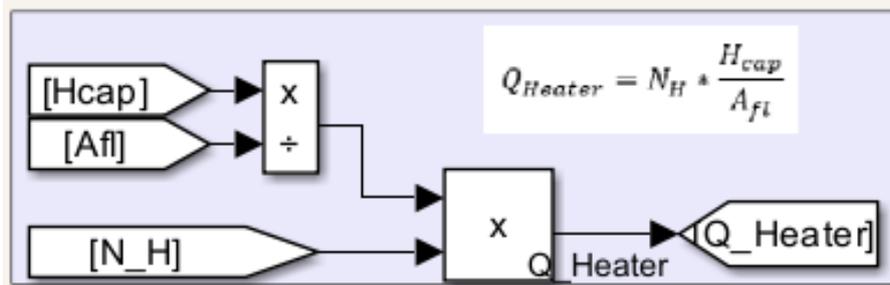


Figura A.4. Calefacción

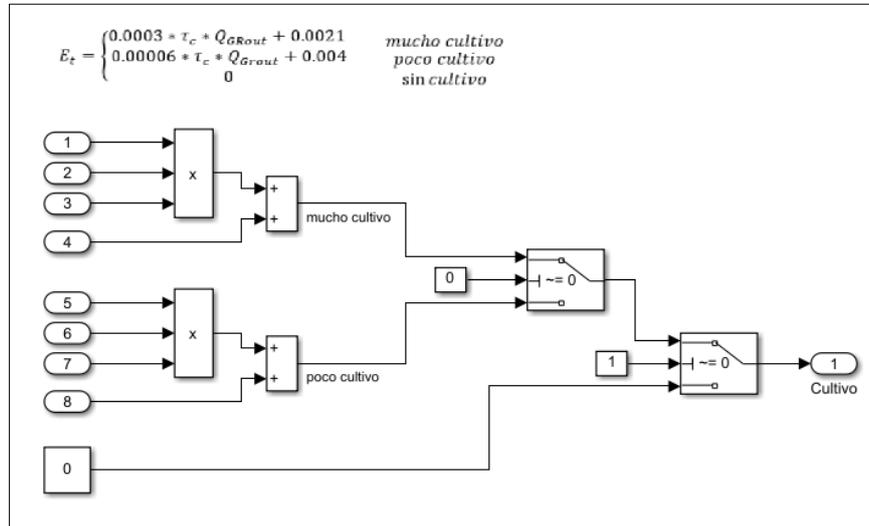


Figura A.5. Cantidad de Cultivo

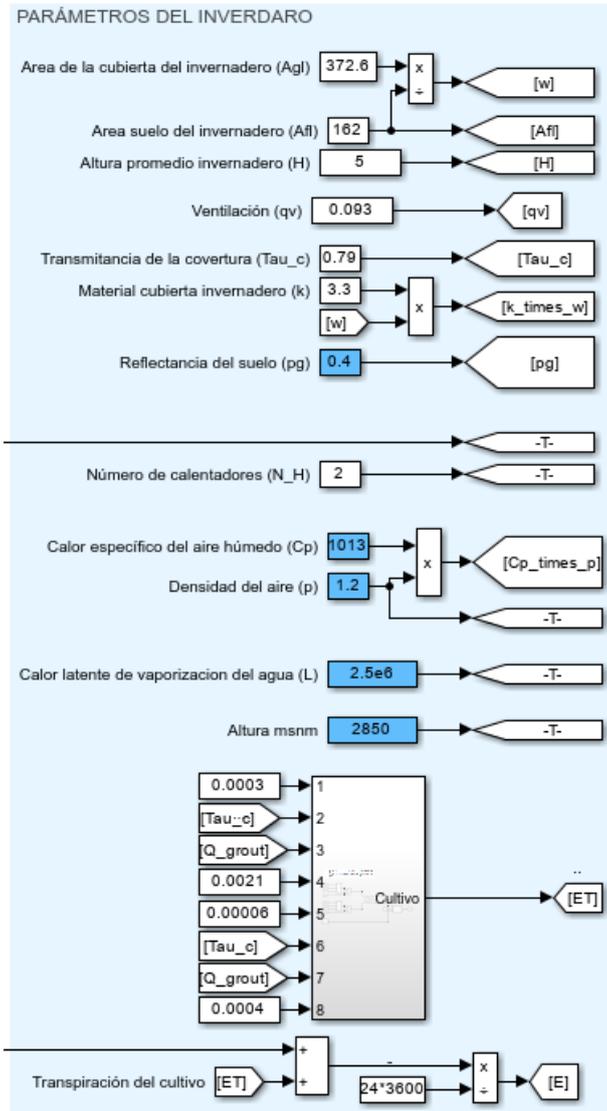


Figura A.6. Parámetros del Invernadero

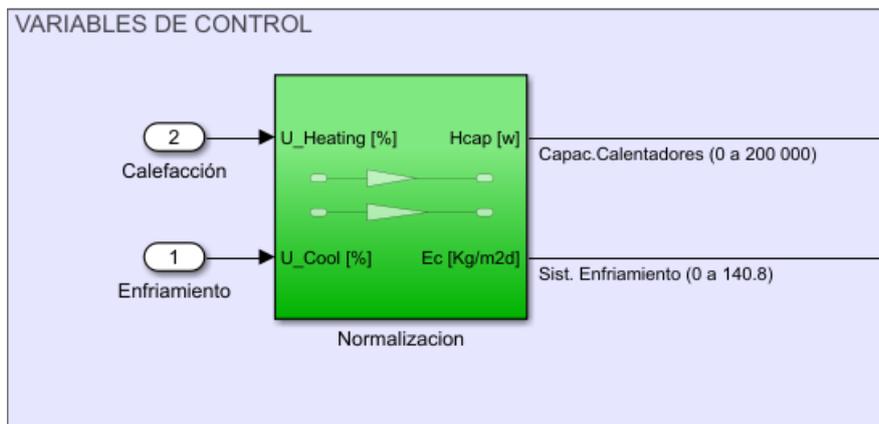


Figura A.7. Variables de Control

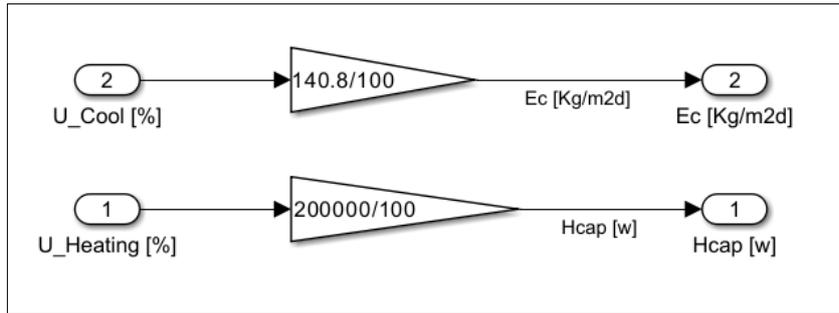


Figura A.8. Normalización de las Variables de Control

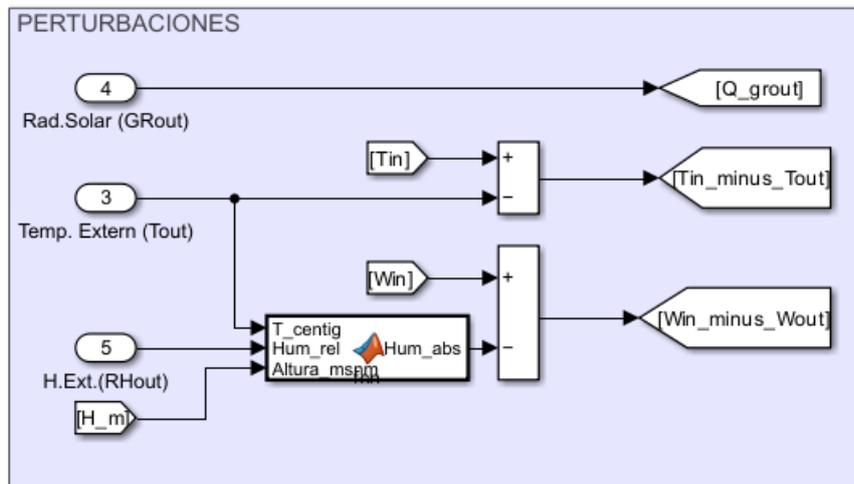


Figura A.8. Perturbaciones del Sistema

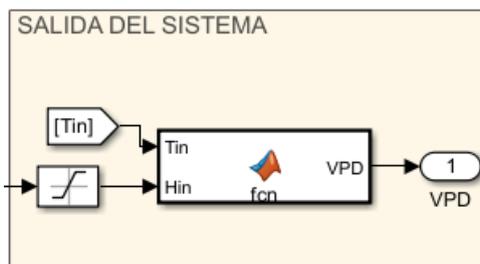


Figura A.9. Salida del Sistema VPD

ANEXO II.

Pantalla Inicial

```
function varargout = Interfaz(varargin)
gui_Singleton = 1;
gui_State = struct('gui_Name',       mfilename, ...
                  'gui_Singleton',   gui_Singleton, ...
                  'gui_OpeningFcn',  @Interfaz_OpeningFcn, ...
                  'gui_OutputFcn',  @Interfaz_OutputFcn, ...
                  'gui_LayoutFcn',   [], ...
                  'gui_Callback',    []);
if nargin && ischar(varargin{1})
    gui_State.gui_Callback = str2func(varargin{1});
end

if nargout
    [varargout{1:nargout}] = gui_mainfcn(gui_State, varargin{:});
else
    gui_mainfcn(gui_State, varargin{:});
end

function Interfaz_OpeningFcn(hObject, ~, handles, varargin)

axes(handles.axes1);
background = imread('Invernadero.png');
axis off;
imshow(background);

handles.output = hObject;
guidata(hObject, handles);

function varargout = Interfaz_OutputFcn(~, eventdata, handles)
varargout{1} = handles.output;

function pushbutton2_Callback(hObject, eventdata, handles)
SIMULACION2
close Interfaz

function pushbutton3_Callback(hObject, eventdata, handles)
DATOS
close Interfaz

function pushbutton4_Callback(hObject, eventdata, handles)
clc,close all
```

Pantalla Simulación

```
function varargout = SIMULACION2(varargin)
gui_Singleton = 1;
gui_State = struct('gui_Name',       mfilename, ...
                  'gui_Singleton',   gui_Singleton, ...
                  'gui_OpeningFcn',  @SIMULACION2_OpeningFcn, ...
                  'gui_OutputFcn',   @SIMULACION2_OutputFcn, ...
                  'gui_LayoutFcn',   [], ...
                  'gui_Callback',    []);
if nargin && ischar(varargin{1})
    gui_State.gui_Callback = str2func(varargin{1});
end

if nargout
    [varargout{1:nargout}] = gui_mainfcn(gui_State, varargin{:});
else
    gui_mainfcn(gui_State, varargin{:});
end

function SIMULACION2_OpeningFcn(hObject, eventdata, handles, varargin)
handles.output = hObject;
guidata(hObject, handles);
set(handles.popupmenu1, 'String', {'Sin Perturbaciones', 'Perturbacion 1',
    'Perturbacion 2', 'Perturbacion 3', 'Perturbacion 4',});
load_system("Invernadero");
find_system;
selector='0';
set_param('Invernadero/Selector', 'Value', selector);
set(handles radiobutton1, 'Value', 1);
fullPath = mfilename('fullpath');
[pathname, ~, ~] = fileparts(fullPath);
filename = 'FUZZY.fis';
filename2 = 'FUZZY2.fis';
fisPath = fullfile(pathname, filename);
fisPath2 = fullfile(pathname, filename2);
FUZZY = readfis(fisPath);
FUZZY2= readfis(fisPath2);
assignin('base', 'FUZZY', FUZZY);
assignin('base', 'FUZZY2', FUZZY2);
    data = readtable("Perturbacion 1.xlsx");
    tiempo = data(:, 1);
    temp=19.19 * ones(height(data), 1);
    rad=200 * ones(height(data), 1);
    humedad=63.08 * ones(height(data), 1);
    tiempo = table2array(data(:, 1));
    TEMP = [tiempo, temp];
    RAD = [tiempo, rad];
    HUMEDAD = [tiempo, humedad];
assignin('base', 'TEMP', TEMP);
assignin('base', 'RAD', RAD);
assignin('base', 'HUMEDAD', HUMEDAD);

function varargout = SIMULACION2_OutputFcn(hObject, eventdata, handles)
varargout{1} = handles.output;

function pushbutton1_Callback(hObject, eventdata, handles)
```

```

global smc1 smce smcc ref ise iae

initialValue = str2double(get(handles.edit1,'String'));
finalValue = str2double(get(handles.edit2,'String'));
set_param('Invernadero/Ref', 'Before', num2str(initialValue));
set_param('Invernadero/Ref', 'After', num2str(finalValue));

set(handles.axes1, 'Visible', 'off');
cla(handles.axes1, 'reset');

salida=sim('Invernadero', 'ReturnWorkspaceOutputs', 'on');

smc1=salida.VPD;
smce=salida.SMCE;
smcc=salida.SMCC;
ref=salida.Ref;
ise=salida.ISE.signals.values;
iae=salida.IAE.signals.values;

axes(handles.axes1);
cla(handles.axes1); % Limpiar el eje
plot(smc1.time/3600,smc1.signals.values,'MarkerSize',16);
hold on
plot(ref.time/3600,ref.signals.values,'--','MarkerSize',10);
hold off
title('VPD');
xlabel('Tiempo (h)');
ylabel('VPD [kPa]');
legend('VPD', 'Referencia')
grid on
axis tight;
% Obtener los límites actuales
current_xlim = xlim();
current_ylim = ylim();

% Añadir un pequeño margen alrededor de los datos (por ejemplo, 10%)
x_margin = 0 * (current_xlim(2) - current_xlim(1));
y_margin = 0.05 * (current_ylim(2) - current_ylim(1));

% Ajustar los límites con el margen
xlim([current_xlim(1) - x_margin, current_xlim(2) + x_margin]);
ylim([current_ylim(1) - y_margin, current_ylim(2) + y_margin]);

handles = guidata(hObject)

set(handles.textISE, 'String', num2str(ise));
set(handles.textIAE, 'String', num2str(iae));

guidata(hObject, handles);

function edit1_Callback(hObject, eventdata, handles)

function edit1_CreateFcn(hObject, eventdata, handles)
if ispc && isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUicontrolBackgroundColor'))

```

```

        set(hObject, 'BackgroundColor', 'white');
    end

function edit2_Callback(hObject, eventdata, handles)

function edit2_CreateFcn(hObject, eventdata, handles)
if ispc && isequal(get(hObject, 'BackgroundColor'),
get(0, 'defaultUiControlBackgroundColor'))
    set(hObject, 'BackgroundColor', 'white');
end

function popupmenu1_Callback(hObject, eventdata, handles)
global TEMP RAD HUMEDAD;

contents = cellstr(get(hObject, 'String'));
selectedFile = contents{get(hObject, 'Value')};

if strcmp(selectedFile, 'Sin Perturbaciones')
    data = readtable('Perturbacion 1.xlsx');
    tiempo = data(:, 1);
    temp=19.19 * ones(height(data), 1);
    rad=200 * ones(height(data), 1);
    humedad=63.08 * ones(height(data), 1);
    tiempo = table2array(data(:, 1));
    TEMP = [tiempo, temp];
    RAD = [tiempo, rad];
    HUMEDAD = [tiempo, humedad];
    assignin('base', 'TEMP', TEMP);
    assignin('base', 'RAD', RAD);
    assignin('base', 'HUMEDAD', HUMEDAD);
else
    data = readtable(selectedFile);
    TEMP = data{:, {'Tiempo', 'TEMP'}};
    RAD = data{:, {'Tiempo', 'RAD'}};
    HUMEDAD = data{:, {'Tiempo', 'HUMEDAD'}};
    assignin('base', 'TEMP', TEMP);
    assignin('base', 'RAD', RAD);
    assignin('base', 'HUMEDAD', HUMEDAD);
    msgbox('Datos cargados correctamente desde el archivo seleccionado.');
```

```

end

function popupmenu1_CreateFcn(hObject, eventdata, handles)
if ispc && isequal(get(hObject, 'BackgroundColor'),
get(0, 'defaultUiControlBackgroundColor'))
    set(hObject, 'BackgroundColor', 'white');
end

function pushbutton2_Callback(hObject, eventdata, handles)
Interfaz
close SIMULACION2

function pushbutton3_Callback(hObject, eventdata, handles)
DATOS
close SIMULACION2

```

```

function radiobutton1_Callback(hObject, eventdata, handles)
selector='0';
set_param('Invernadero/Selector', 'Value', selector);

function radiobutton2_Callback(hObject, eventdata, handles)
selector='1';
set_param('Invernadero/Selector', 'Value', selector);

function progressBar_Callback(hObject, eventdata, handles)

function progressBar_CreateFcn(hObject, eventdata, handles)
if isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUiControlBackgroundColor'))
    set(hObject,'BackgroundColor',[.9 .9 .9]);
end

function edit3_Callback(hObject, eventdata, handles)

function edit3_CreateFcn(hObject, eventdata, handles)
if ispc && isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUiControlBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end

function edit4_Callback(hObject, eventdata, handles)

function edit4_CreateFcn(hObject, eventdata, handles)
if ispc && isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUiControlBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end

% --- Executes on button press in pushbutton7.
function pushbutton7_Callback(hObject, eventdata, handles)
global smc1 smce smcc ref
axes(handles.axes1);
cla(handles.axes1); % Limpiar el eje
plot(smc1.time/3600,smc1.signals.values,'MarkerSize',16);
hold on
plot(ref.time/3600,ref.signals.values,'--','MarkerSize',10);
hold off
title('VPD');
xlabel('Tiempo (h)');
ylabel('VPD [kPa]');
legend('VPD', 'Referencia')
grid on
axis tight;
% Obtener los límites actuales
current_xlim = xlim();
current_ylim = ylim();

% Añadir un pequeño margen alrededor de los datos (por ejemplo, 10%)
x_margin = 0 * (current_xlim(2) - current_xlim(1));
y_margin = 0.1 * (current_ylim(2) - current_ylim(1));

% Ajustar los límites con el margen
xlim([current_xlim(1) - x_margin, current_xlim(2) + x_margin]);

```

```

ylim([current_ylim(1) - y_margin, current_ylim(2) + y_margin]);
;

% --- Executes on button press in pushbutton8.
function pushbutton8_Callback(hObject, eventdata, handles)
global smc1 smce smcc ref
axes(handles.axes1);
cla(handles.axes1); % Limpiar el eje
plot(smce.time/3600,smce.signals.values,'MarkerSize',16);
title('Acción de Control Enfriamiento');
xlabel('Tiempo (h)');
ylabel('%')
grid on
ylim([0 100]);
xlim([0 24]);

% --- Executes on button press in pushbutton9.
function pushbutton9_Callback(hObject, eventdata, handles)
global smc1 smce smcc ref
axes(handles.axes1);
cla(handles.axes1);
plot(smcc.time/3600,smcc.signals.values,'MarkerSize',16);
title('Acción de Control Calefactor');
xlabel('Tiempo (h)');
ylabel('%')
grid on
ylim([0 100]);
xlim([0 24]);

% --- Executes on button press in pushbutton4.
function pushbutton4_Callback(hObject, eventdata, handles)
% hObject    handle to pushbutton4 (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)

% --- Executes on button press in pushbutton5.
function pushbutton5_Callback(hObject, eventdata, handles)
% hObject    handle to pushbutton5 (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)

% --- Executes on button press in pushbutton6.
function pushbutton6_Callback(hObject, eventdata, handles)
% hObject    handle to pushbutton6 (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)

```

Pantalla Parámetros del Invernadero

```
function varargout = DATOS(varargin)
gui_Singleton = 1;
gui_State = struct('gui_Name',       mfilename, ...
                  'gui_Singleton',   gui_Singleton, ...
                  'gui_OpeningFcn',  @DATOS_OpeningFcn, ...
                  'gui_OutputFcn',   @DATOS_OutputFcn, ...
                  'gui_LayoutFcn',   [], ...
                  'gui_Callback',    []);
if nargin && ischar(varargin{1})
    gui_State.gui_Callback = str2func(varargin{1});
end

if nargout
    [varargout{1:nargout}] = gui_mainfcn(gui_State, varargin{:});
else
    gui_mainfcn(gui_State, varargin{:});
end

function DATOS_OpeningFcn(hObject, eventdata, handles, varargin)
global k tau Af1 Agl H Hnm c1 c2
load_system('Invernadero');
find_system;
Af1=162;
Agl=372.6;
k= 6.2;
tau= 0.87;
H=5;
Hnm=2850;
c1=0;
c2=1;
set(handles.edit5,'String',tau)
set(handles.edit4,'String',k)
set(handles.edit6,'String',Af1)
set(handles.edit7,'String',Agl)
set(handles.edit8,'String',H)
set(handles.edit9,'String',Hnm)
set_param('Invernadero/MODELO INVERNADERO/Material cubierta invernadero (k)',
'Value', num2str(k))
set_param('Invernadero/MODELO INVERNADERO/Transmitancia de la cobertura
(Tau_c)', 'Value', num2str(tau))
set_param('Invernadero/MODELO INVERNADERO/Area suelo del invernadero (Af1)',
'Value', num2str(Af1));
set_param('Invernadero/MODELO INVERNADERO/Area de la cubierta del invernadero
(Agl)', 'Value', num2str(Agl));
set_param('Invernadero/MODELO INVERNADERO/Altura promedio invernadero (H)',
'Value', num2str(H))
set_param('Invernadero/MODELO INVERNADERO/Altura msnm', 'Value', num2str(Hnm))
set_param('Invernadero/MODELO INVERNADERO/Subsystem1/Constant1', 'Value',
num2str(c1))
set_param('Invernadero/MODELO INVERNADERO/Subsystem1/Constant2', 'Value',
num2str(c2))
set(handles.radiobutton9, 'Value', 1);
set(handles.radiobutton14, 'Value', 1);
handles.output = hObject;
```

```

guidata(hObject, handles);

function varargout = DATOS_OutputFcn(hObject, eventdata, handles)
varargout{1} = handles.output;

function radiobutton2_Callback(hObject, eventdata, handles)

function radiobutton3_Callback(hObject, eventdata, handles)

function radiobutton5_Callback(hObject, eventdata, handles)

function radiobutton6_Callback(hObject, eventdata, handles)

function radiobutton7_Callback(hObject, eventdata, handles)

function edit2_Callback(hObject, eventdata, handles)

function edit2_CreateFcn(hObject, eventdata, handles)

if ispc && isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end

function edit3_Callback(hObject, eventdata, handles)

function edit3_CreateFcn(hObject, eventdata, handles)

if ispc && isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end

function edit4_Callback(hObject, eventdata, handles)

function edit4_CreateFcn(hObject, eventdata, handles)

if ispc && isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end

function edit5_Callback(hObject, ~, handles)

function edit5_CreateFcn(hObject, eventdata, handles)

if ispc && isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end

function radiobutton8_Callback(hObject, eventdata, handles)
k=6.2;
tau=0.90;

```

```

set(handles.edit5,'String',tau)
set(handles.edit4,'String',k)
set_param('Invernadero/MODELO INVERNADERO/Material cubierta invernadero (k)',
'Value', num2str(k))
set_param('Invernadero/MODELO INVERNADERO/Transmitancia de la cobertura
(Tau_c)', 'Value', num2str(tau))

function radiobutton9_Callback(hObject, eventdata, handles)
k=6.2;
tau=0.87;
set(handles.edit5,'String',tau)
set(handles.edit4,'String',k)
set_param('Invernadero/MODELO INVERNADERO/Material cubierta invernadero (k)',
'Value', num2str(k))
set_param('Invernadero/MODELO INVERNADERO/Transmitancia de la cobertura
(Tau_c)', 'Value', num2str(tau))

function radiobutton10_Callback(hObject, eventdata, handles)

k=6.2;
tau=0.87;
set(handles.edit5,'String',tau)
set(handles.edit4,'String',k)
set_param('Invernadero/MODELO INVERNADERO/Material cubierta invernadero (k)',
'Value', num2str(k))
set_param('Invernadero/MODELO INVERNADERO/Transmitancia de la cobertura
(Tau_c)', 'Value', num2str(tau))

function radiobutton11_Callback(hObject, eventdata, handles)
k=4;
tau=0.76;
set(handles.edit5,'String',tau)
set(handles.edit4,'String',k)
set_param('Invernadero/MODELO INVERNADERO/Material cubierta invernadero (k)',
'Value', num2str(k))
set_param('Invernadero/MODELO INVERNADERO/Transmitancia de la cobertura
(Tau_c)', 'Value', num2str(tau))

function radiobutton12_Callback(hObject, eventdata, handles)
k=3.3;
tau=0.79;
set(handles.edit5,'String',tau)
set(handles.edit4,'String',k)
set_param('Invernadero/MODELO INVERNADERO/Material cubierta invernadero (k)',
'Value', num2str(k))
set_param('Invernadero/MODELO INVERNADERO/Transmitancia de la cobertura
(Tau_c)', 'Value', num2str(tau))

function edit6_Callback(hObject, eventdata, handles)
Af1= str2double(get(hObject, 'String'));
    if isnan(Af1)
        errordlg('Por favor, ingrese un número válido', 'Error');
        return;
    end
    set_param('Invernadero/MODELO INVERNADERO/Area suelo del invernadero
(Af1)', 'Value', num2str(Af1));

```

```

function edit6_CreateFcn(hObject, eventdata, handles)

if ispc && isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end

function edit7_Callback(hObject, eventdata, handles)
Agl= str2double(get(hObject, 'String'));
    if isnan(Agl)
        errordlg('Por favor, ingrese un número válido', 'Error');
        return;
    end
    set_param('Invernadero/MODELO INVERNADERO/Area de la cubierta del
invernadero (Agl)', 'Value', num2str(Agl));

function edit7_CreateFcn(hObject, eventdata, handles)
if ispc && isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end

function edit8_Callback(hObject, eventdata, handles)
H=str2double(get(hObject, 'String'));
    if isnan(H)
        errordlg('Por favor, ingrese un número válido', 'Error');
        return;
    end
    set_param('Invernadero/MODELO INVERNADERO/Altura promedio invernadero (H)',
'Value', num2str(H))

function edit8_CreateFcn(hObject, eventdata, handles)
if ispc && isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end

function edit9_Callback(hObject, eventdata, handles)
Hnm=str2double(get(hObject, 'String'));
    if isnan(Hnm)
        errordlg('Por favor, ingrese un número válido', 'Error');
        return;
    end
    set_param('Invernadero/MODELO INVERNADERO/Altura msnm', 'Value',
num2str(Hnm))

function edit9_CreateFcn(hObject, eventdata, handles)
if ispc && isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end

function pushbutton1_Callback(hObject, eventdata, handles)
SIMULACION2

```

```

close DATOS

function pushbutton2_Callback(hObject, eventdata, handles)
Interfaz
close DATOS

function radiobutton13_Callback(hObject, eventdata, handles)
c1=0;
c2=0;
set_param('Invernadero/MODELO INVERNADERO/Subsystem1/Constant1', 'Value',
num2str(c1))
set_param('Invernadero/MODELO INVERNADERO/Subsystem1/Constant2', 'Value',
num2str(c2))

function radiobutton14_Callback(hObject, eventdata, handles)
c1=0;
c2=1;
set_param('Invernadero/MODELO INVERNADERO/Subsystem1/Constant1', 'Value',
num2str(c1))
set_param('Invernadero/MODELO INVERNADERO/Subsystem1/Constant2', 'Value',
num2str(c2))

function radiobutton15_Callback(hObject, eventdata, handles)
c1=1;
c2=1;
set_param('Invernadero/MODELO INVERNADERO/Subsystem1/Constant1', 'Value',
num2str(c1))
set_param('Invernadero/MODELO INVERNADERO/Subsystem1/Constant2', 'Value',
num2str(c2))

function pushbutton3_Callback(hObject, eventdata, handles)
Af1=162;
Ag1=372.6;
k= 6.2;
tau= 0.87;
H=5;
Hnm=2850;
c1=0;
c2=1;
set(handles.edit5,'String',tau)
set(handles.edit4,'String',k)
set(handles.edit6,'String',Af1)
set(handles.edit7,'String',Ag1)
set(handles.edit8,'String',H)
set(handles.edit9,'String',Hnm)
set_param('Invernadero/MODELO INVERNADERO/Material cubierta invernadero (k)',
'Value', num2str(k))
set_param('Invernadero/MODELO INVERNADERO/Transmitancia de la cobertura
(Tau_c)', 'Value', num2str(tau))
set_param('Invernadero/MODELO INVERNADERO/Area suelo del invernadero (Af1)',
'Value', num2str(Af1));
set_param('Invernadero/MODELO INVERNADERO/Area de la cubierta del invernadero
(Ag1)', 'Value', num2str(Ag1));
set_param('Invernadero/MODELO INVERNADERO/Altura promedio invernadero (H)',
'Value', num2str(H))
set_param('Invernadero/MODELO INVERNADERO/Altura msnm', 'Value', num2str(Hnm))
set_param('Invernadero/MODELO INVERNADERO/Subsystem1/Constant1', 'Value',
num2str(c1))

```

```
set_param('Invernadero/MODELO INVERNADERO/Subsystem1/Constant2', 'Value',  
num2str(c2))  
set(handles.radiobutton9, 'Value', 1);  
set(handles.radiobutton14, 'Value', 1);
```