

Proyecto HOME IO



Realizado por:

Juan Luque Girón

Andrés Martínez Márquez

Fernando Román Hidalgo

Índice

1. Introducción y Objetivos	3
2. Implementación en Connect I/O	3
3. Archivos necesarios para el control mediante Matlab	5
4. Pasos iniciales previos al control	5
4.1. Configuración del Escenario y Parámetros Globales	6
4.2. Interfaz de Señales y Actuadores	6
4.2.1. Selección de Sensores (Variables de Proceso)	6
4.2.2. Selección del Actuador (Variable de Control)	7
4.2.3. Medición de perturbaciones	7
4.2.4. Métricas de desempeño	7
5. Identificación de Sistemas Monovariables	8
5.1. Punto de operación del sistema	8
5.2. Modelo Calefactor - Temperatura de Habitación	9
5.2.1. Ensayo de Respuesta al Escalón	9
5.2.2. Resultados de la Identificación	9
5.3. Modelo Temperatura Ambiente - Temperatura de Habitación	11
5.4. Modelo Luminosidad - Temperatura de Habitación	14
6. Identificación de modelos multivariable	16
6.1. Suma de contribuciones individuales	16
6.2. Modelo en espacio de estados	17
7. Diseño de controladores	18
7.1. Diseño de PID para el control de la temperatura	18
7.2. Gestión del Windup	19
7.3. Control Feedforward para el rechazo de perturbaciones	19
8. Parametrización para MPC	20
9. Comparativa entre controladores	22

1. Introducción y Objetivos

Esta memoria detalla el desarrollo de soluciones domóticas utilizando el conjunto de herramientas formado por Home I/O, Connect I/O y MATLAB Simulink. El trabajo se desglosa en dos fases diferenciadas:

En la primera parte, se describen los procesos automatizados implementados mediante Connect I/O. En la segunda parte, el foco se desplaza al control climático, donde se analiza y regula la temperatura de una habitación utilizando diferentes algoritmos de control diseñados en Simulink.

2. Implementación en Connect I/O

En el módulo de Connect I/O, se ha desarrollado la lógica de automatización para el subsistema de seguridad de la vivienda. Esta implementación integra diversos dispositivos de campo, tales como sensores de humo, detectores de apertura y sistemas de iluminación, para garantizar tanto la seguridad técnica como la protección anti-intrusión.

A continuación, se detalla la configuración de los elementos en el entorno de simulación HOME I/O:

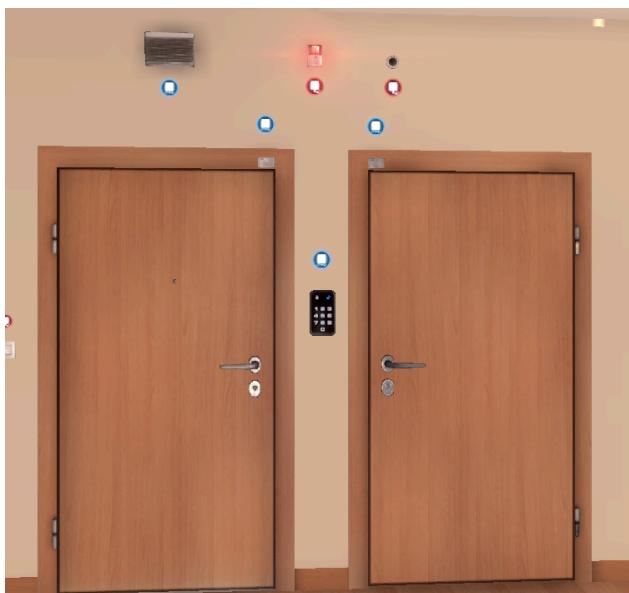


Figura 1: Configuración de sensores en el acceso principal (HOME I/O)

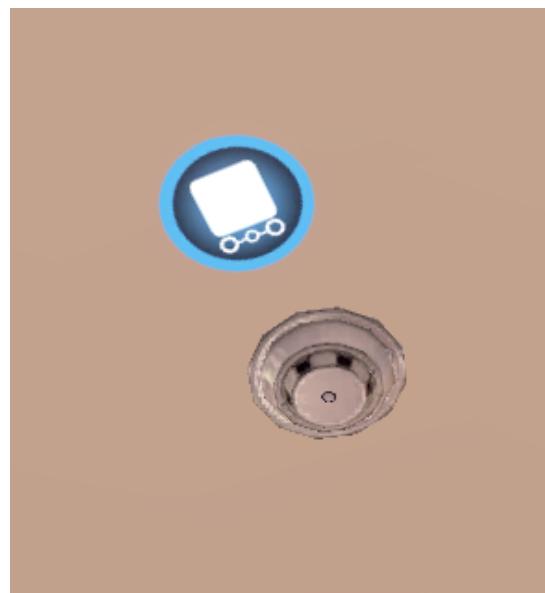


Figura 2: Ubicación y parametrización de detectores de humo (HOME I/O)

El algoritmo de control diseñado gestiona tres eventos principales de forma simultánea:

- Iluminación de Acceso: Al detectar la apertura de la puerta de entrada, el sistema activa automáticamente las luces del recibidor. Se ha programado un temporizador TOFF de 30 segundos para mantener la iluminación activa durante el ingreso a la vivienda
- Sistema Anti-Intrusión: Vinculado a la apertura de la puerta, se inicia un temporizador de seguridad de 20 segundos. Si el usuario no introduce el código de desactivación en el panel de la alarma dentro de este intervalo, el sistema interpretará una intrusión y activará la sirena de forma indefinida hasta su desactivación manual.
- Detección de Incendios: Se ha implementado un filtrado de señal para los sensores de humo. La alarma de incendios se disparará únicamente si los sensores detectan la presencia de humo de forma continua durante más de 20 segundos, evitando así falsos positivos y alertando eficazmente sobre un posible conato de incendio.

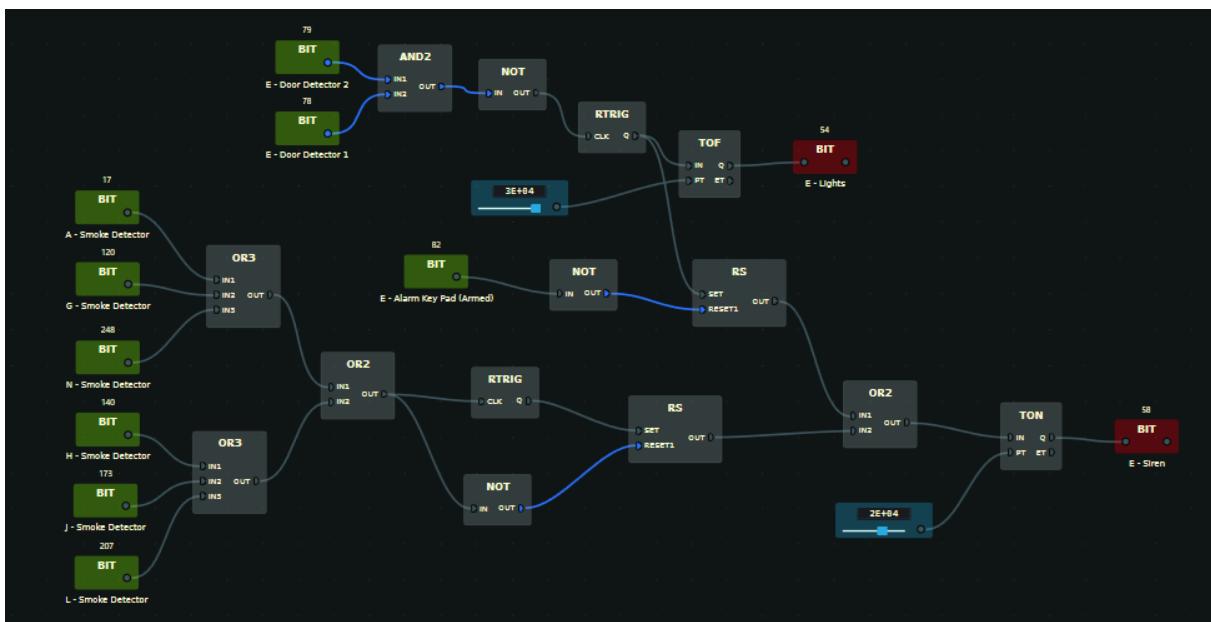


Figura 3: Diagrama de bloques lógicos y conexiones en Connect I/O

3. Archivos necesarios para el control mediante Matlab

En el fichero .zip entregado del proyecto se incluyen varios archivos importantes que se han utilizado para su desarrollo:

- [proyecto_homeIO.slx](#): archivo de Simulink que contiene módulos para ensayos de identificación, de tuning del PID, de control PID con y sin feedforward, de ajuste del MPC y de control con MPC. Aparece todos comentados, es necesario descomentarlos para acceder a cada uno.
- [graficas.m](#): archivo utilizado para representar resultados (las gráficas no se representarán porque los datos no fueron guardados, sin embargo se adjuntan los .fig correspondientes a las más relevantes (estas imágenes se encuentran también en esta memoria)).
- [ensayos/](#): archivos .fig de los ensayos realizados, se incluyen:
 - [comparativacontrol.fig](#): comparativa de PID con antiwindup, PID con Feedforward y antiwindup y MPC. En condiciones de temperatura ambiente entre 10 y 23°C.
 - [graficafeedforward.fig](#): gráfica de controlador feedforward en condiciones de temperatura ambiente entre 10 y 23°C.
 - [graficacontrolmpc.fig](#): gráfica de controlador MPC en condiciones de temperatura ambiente entre 10 y 23°C.
 - [controlwindup.fig](#): gráfica de controlador con antiwindup y sin feedforward en condiciones de temperatura ambiente entre 0 y 30°C.
 - [controlpidbasico.fig](#): gráfica de controlador sin antiwindup ni feedforward en condiciones de temperatura ambiente entre 0 y 30°C.
 - [ensayodistintosmodeloss.fig](#): gráfica comparativa de los modelos en espacio de estados.
 - [ensayodistintosmodelosmonovariable.fig](#): gráfica comparativa de los modelos monovariable.
- [datos_entrega.mat](#): archivo que contiene variables necesarias como las matrices del espacio de estados ya calculadas.
- [identificacion_ss.m](#): utilizado para obtener el modelo de espacio de estados de 2 y 3 variables.
- [mpc.m](#): archivo para ajustar el controlador MPC. Es necesario ejecutar antes [datos_relevantes.mat](#)
- [calculo_metricas.m](#): archivo para calcular métricas e índices de desempeño. Es necesario ejecutar antes [datos_relevantes.mat](#).

4. Pasos iniciales previos al control

El objetivo de esta parte del proyecto consiste en modelar la variación de temperatura en una habitación de la casa para posteriormente controlarla mediante diversos algoritmos usados en la asignatura, y compararlos entre sí para encontrar la mejor solución posible respecto al control y eficiencia energética de la vivienda.

4.1. Configuración del Escenario y Parámetros Globales

Para garantizar la consistencia de los ensayos y definir las perturbaciones externas (como la radiación solar y la temperatura exterior), se han fijado condiciones iniciales estrictas en el simulador Home I/O. La elección de la Sala L como objeto de estudio se debe a sus características particulares de aislamiento e iluminación, ya que tiene una gran cristalera que da al exterior de la casa, por lo que la variación de la temperatura ambiente influirá en gran medida en la temperatura de la habitación.



Figura 4: Habitación elegida en HOME I/O

Los parámetros espacio-temporales definidos para la simulación son:

Habitación: Sala L

Ubicación: Latitud 38, Longitud -6

Fecha de inicio: 15/10/2025

Estas coordenadas y fecha determinan el perfil de temperatura ambiente y la incidencia solar que actuarán como perturbaciones sobre el sistema.

4.2. Interfaz de Señales y Actuadores

Para establecer el lazo de control cerrado, es imprescindible definir la interfaz de comunicación con la vivienda virtual. Esto implica identificar las direcciones de memoria correspondientes a los sensores (entradas al controlador) y actuadores (salidas del controlador).

4.2.1. Selección de Sensores (Variables de Proceso)

Se ha realizado un barrido de las señales disponibles en Home I/O para localizar los sensores ubicados en la Sala L.

Input	Float	103	L	Brightness Sensor (Analogue)	-	0
Input	Float	104	L	Thermostat (Room Temperature)	-	0
Input	Float	105	L	Thermostat (Set Point)	-	0
Input	Float	106	L	Roller Shades (Openness)	-	0

Figura 5: Mapeo de señales de entrada/salida disponibles en la vivienda

Tras el análisis de la lista de variables, se han seleccionado las señales con ID 104 y 105. Estas variables proporcionan la lectura en tiempo real de la temperatura interior de la habitación, permitiendo modelar y controlar el sistema.

4.2.2. Selección del Actuador (Variable de Control)

La regulación de la temperatura se ejercerá mediante el control de la potencia del calefactor eléctrico instalado en la estancia. Se ha identificado la dirección de memoria que permite controlar este dispositivo:

Output	Float	124	L	Heater (Analogue)	-	750
--------	-------	-----	---	-------------------	---	-----

Figura 6: Direccionamiento del actuador del calefactor

Esta señal actuará como la variable manipulada o señal de control del sistema, permitiendo inyectar potencia calorífica para elevar la temperatura según los requerimientos del controlador.

4.2.3. Medición de perturbaciones

El tercer y último elemento que compone un sistema de control, además de la señal a controlar y la señal de control, son las perturbaciones. En este caso, estas resultan medibles mediante sensores que incluye Home I/O.

Las perturbaciones que serán tomadas en cuenta para este sistema son la temperatura ambiente y la luminosidad captada.

Para ello disponemos de las siguientes señales:

Input	Float	103	L	Brightness Sensor (Analogue)	-	0
-------	-------	-----	---	------------------------------	---	---

Figura 7: Dirección asociada al sensor de luminosidad

Memory	Float	132	-	Air Temperature [K]	-	0
--------	-------	-----	---	---------------------	---	---

Figura 8: Dirección asociada a la temperatura ambiente

4.2.4. Métricas de desempeño

Uno de los objetivos de este proyecto, además del control, consiste en perseguir la eficiencia energética de la vivienda. Para ello es necesario tomar datos acerca del consumo de la misma. Para ello, hay una señal disponible que permite medir la potencia instantánea consumida por la vivienda.

Memory	Float	141	-	Instant Power [W]	-	0
--------	-------	-----	---	-------------------	---	---

Figura 9: Dirección asociada a la potencia instantánea total consumida por la vivienda

5. Identificación de Sistemas Monovariantes

El primer paso en el camino hasta controlar un sistema es obtener un modelo del mismo. El objetivo fundamental de este primer paso es obtener un modelo en tiempo continuo que describa de la manera más parecida posible la dinámica térmica de la habitación objeto de estudio. Para diseñar un controlador robusto y eficiente, es imperativo comprender primero cómo evoluciona la temperatura interior en función de la energía suministrada y las perturbaciones del entorno.

A continuación, se procede al modelado de los subsistemas térmicos de forma individual, para observar de qué forma afectan a la temperatura de la habitación.

5.1. Punto de operación del sistema

Previo a la identificación dinámica de la planta, es indispensable establecer un Punto de Operación alrededor del cual el sistema trabajará la mayor parte del tiempo. Dado que los sistemas térmicos reales presentan comportamientos no lineales, el diseño del controlador se basará en un modelo linealizado en torno a este punto de equilibrio.

Para obtener este punto experimentalmente, se ha procedido a realizar un ensayo en lazo abierto. Manteniendo la señal de control fija a 0V y reduciendo las perturbaciones al mínimo. Esta reducción de perturbaciones se realiza manteniendo el sistema con una temperatura ambiente entre 16 y 17°C y subiendo la nubosidad al máximo, para que la luminosidad incidente difiera en la menor cantidad posible con la que hay de noche.

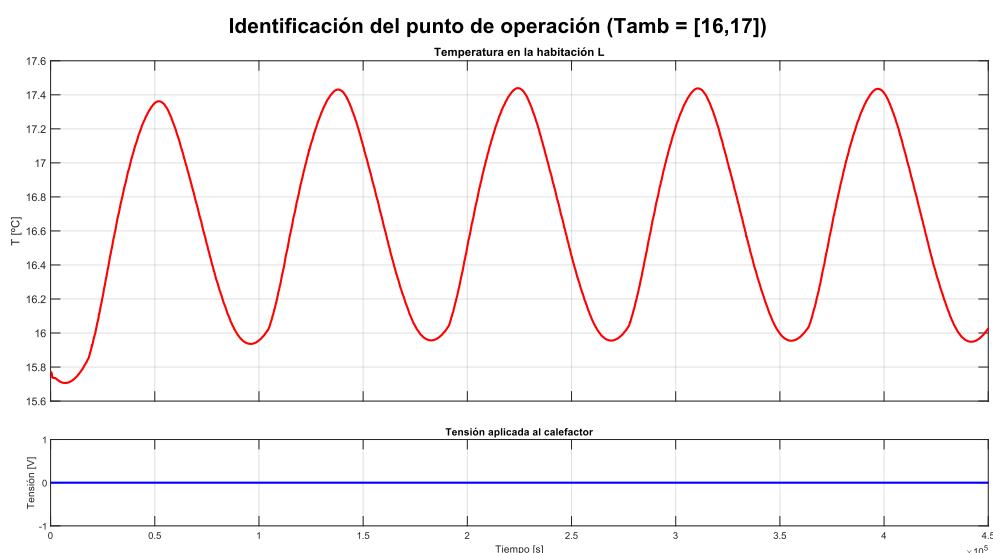


Figura 10: Identificación del punto de operación

Como punto de operación se tomará la media, por tanto será 16.5°C y 0V.

5.2. Modelo Calefactor - Temperatura de Habitación

5.2.1. Ensayo de Respuesta al Escalón

Para identificar la dinámica, se realiza un ensayo limitando la variación de la temperatura ambiente al rango de 16 a 17°C. Al minimizar la variación de temperatura ambiente, la respuesta de escalón se deberá casi en su totalidad a la variación de potencia del calefactor.

Se excita el sistema con saltos en escalón en el calefactor:

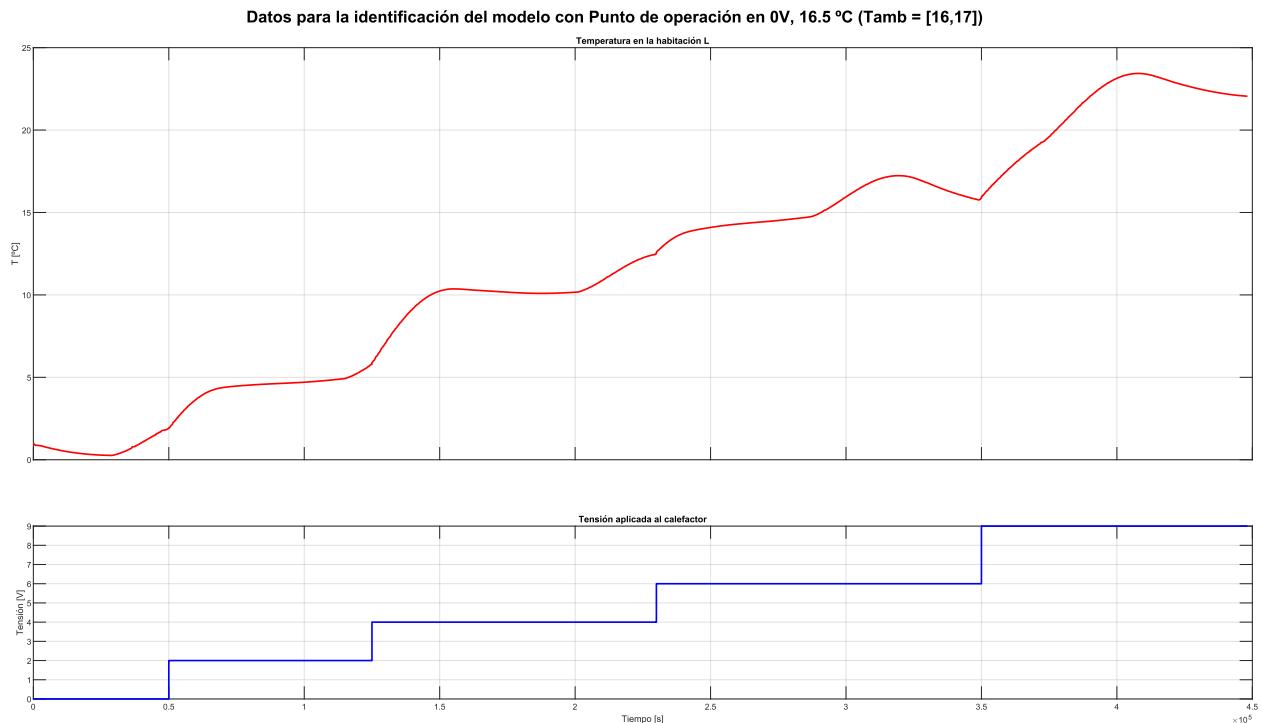


Figura 11: Respuesta temporal

5.2.2. Resultados de la Identificación

Utilizando la herramienta System Identification, se puede identificar el modelo mediante la respuesta generada. Para ello basta únicamente con indicar que el tiempo de muestreo es de 200 segundos y que se pretende obtener un modelo de primer orden sin ceros y con un solo polo. Tras esto, la herramienta itera entre varios modelos, de manera que en la tercera iteración ya obtiene un modelo que se ajuste al experimento realizado con una precisión del 87.05%.

El modelo obtenido es el siguiente:

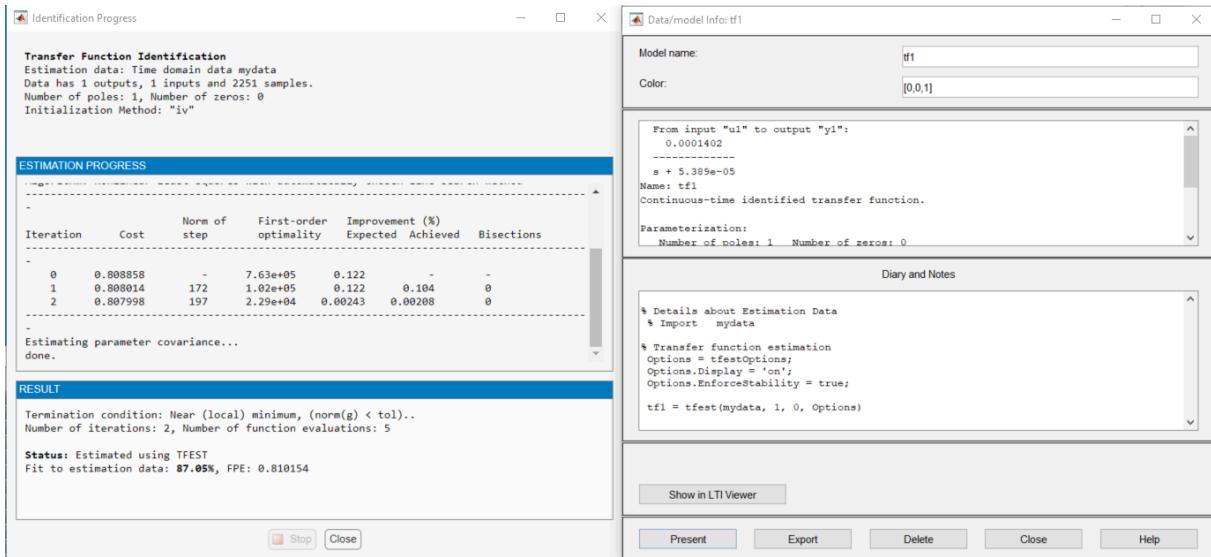


Figura 12: Resultado de la identificación del modelo

```
1  >> tf1
matlab
2
3  tf1 =
4      From input "u1" to output "y1":
5          0.0001402
6  -----
7      s + 5.389e-05
8
9  Name: tf1
10 Continuous-time identified transfer function.
11
12 Parameterization:
13     Number of poles: 1    Number of zeros: 0
14     Number of free coefficients: 2
15     Use "tfdata", "getpvec", "getcov" for parameters and their uncertainties.
16
17 Status:
18 Estimated using TFEST on time domain data "mydata".
19 Fit to estimation data: 87.05% (stability enforced)
20 FPE: 0.8102, MSE: 0.808
```

Tras esto, se excita el modelo con el mismo ensayo que se realizó antes, de manera que se pueda observar su diferencia con el sistema real.

La gráfica obtenida se muestra a continuación.

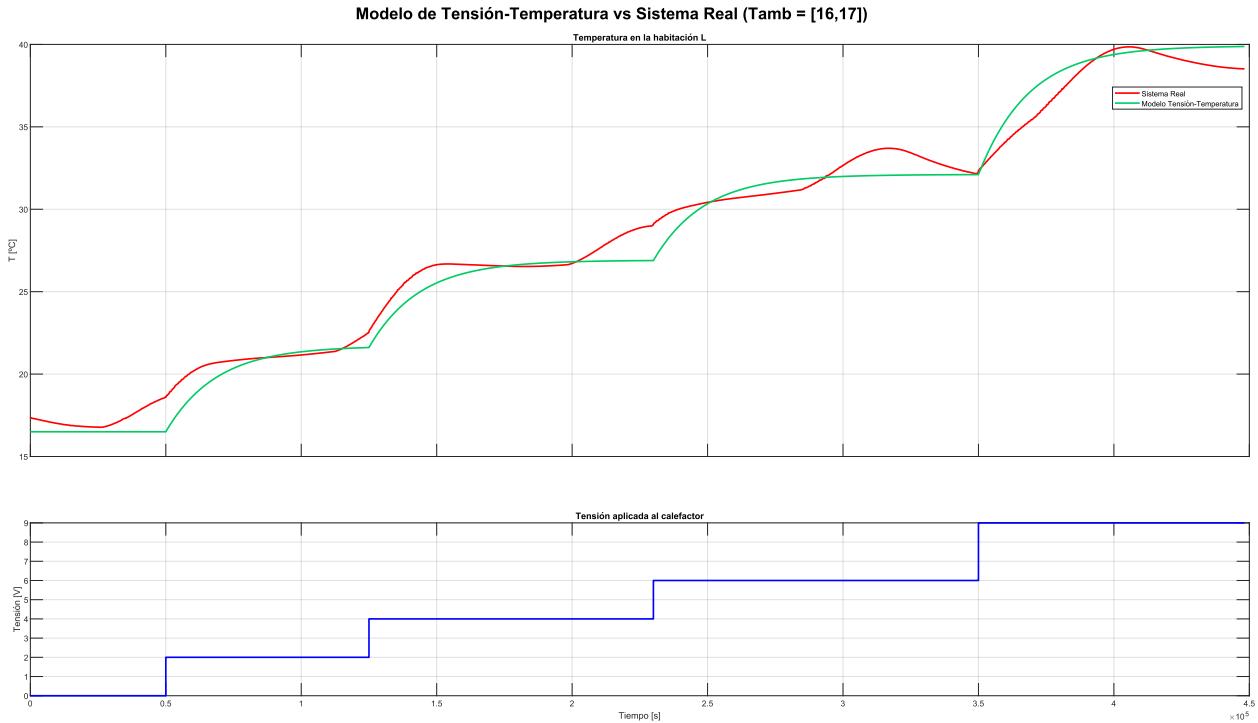


Figura 13: Modelo de Tensión-Temperatura frente a la evolución del sistema real

Se puede comprobar entonces que el modelo sigue al sistema con cierta fiabilidad, aunque no es capaz de modelar por completo todas las dinámicas del sistema, debido a su limitado orden. Además, al no tener en cuenta las perturbaciones, el sistema se hace aún más difícil de seguir para el modelo.

5.3. Modelo Temperatura Ambiente - Temperatura de Habitación

Aprovechando que las perturbaciones son medibles, se pueden realizar modelos también para estas, de manera que describan la evolución del sistema según las mismas. Estos modelos servirán tanto para caracterizar el sistema de forma matemática, así como para aprovecharlos en compensación de perturbaciones, por ejemplo en controles de tipo feed-forward.

Se procede entonces a identificar la influencia de la temperatura ambiente. Se desconecta el calefactor y se somete el sistema a una variación de temperatura ambiente entre 0 y 30°C.

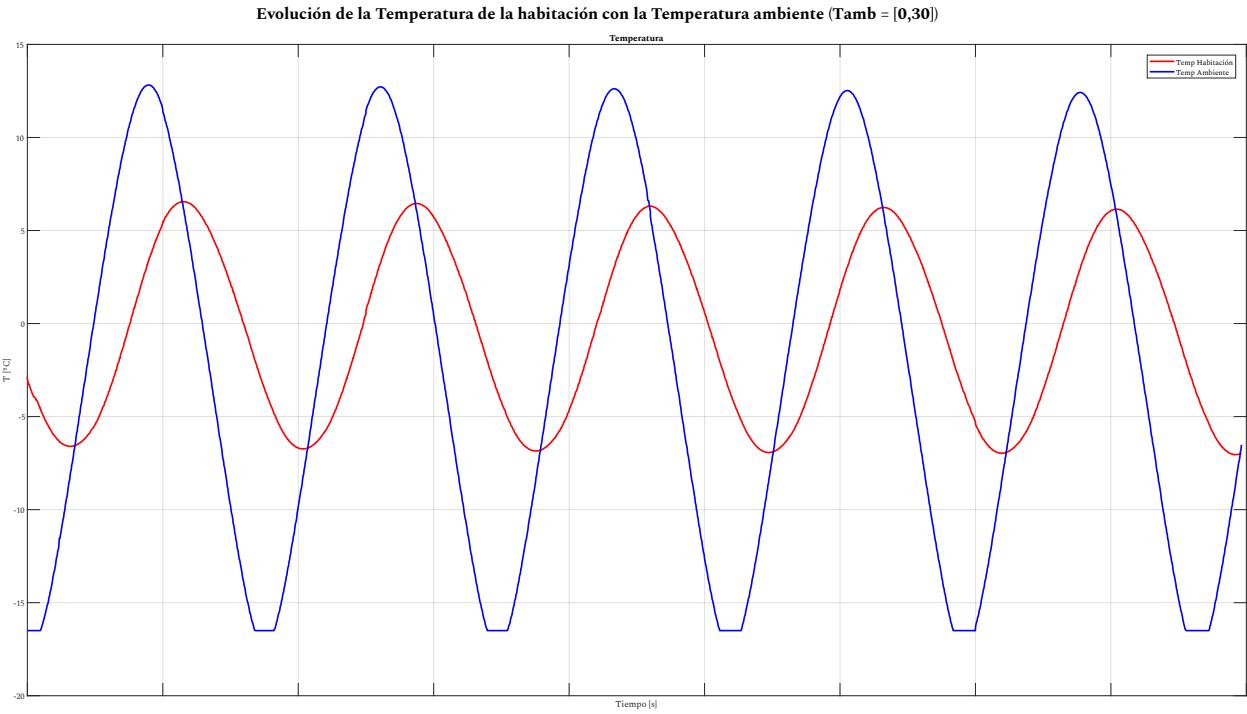


Figura 14: Ensayo de variación de temperatura con la temperatura ambiente
Estimando un modelo de primer orden de nuevo con la herramienta utilizada anteriormente.

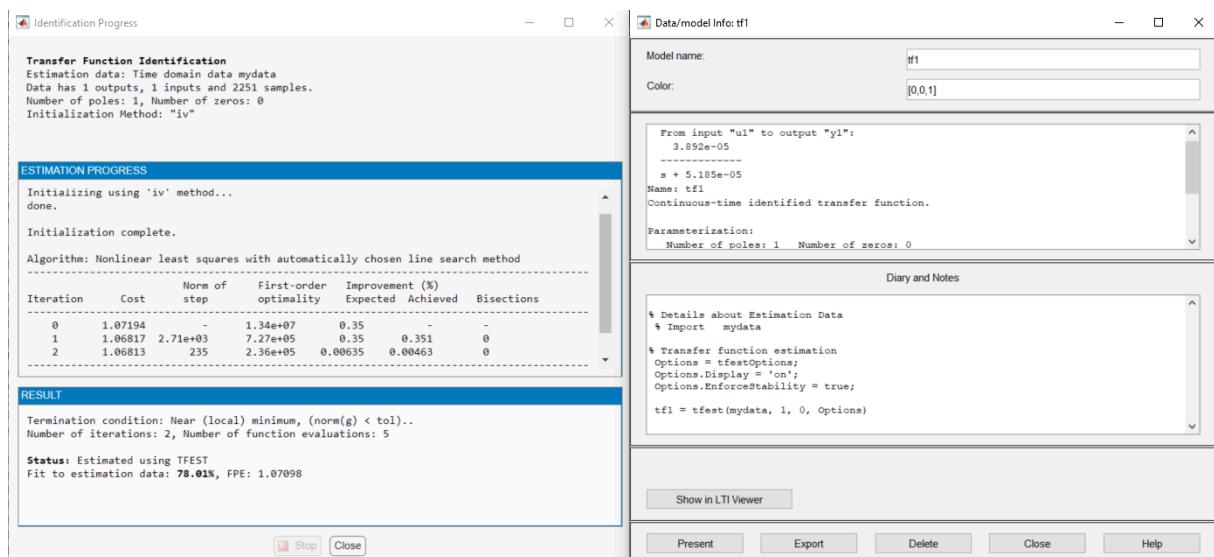


Figura 15: Estimación del modelo de perturbación

```

1  >> tf2
2
3  tf2 =
4      From input "u1" to output "y1":
5          3.892e-05
6          -----
7          s + 5.185e-05
8
9  Name: tf2
10 Continuous-time identified transfer function.
11
12 Parameterization:
13     Number of poles: 1    Number of zeros: 0
14     Number of free coefficients: 2
15     Use "tfdata", "getpvec", "getcov" for parameters and their uncertainties.
16
17 Status:
18 Estimated using TFEST on time domain data "mydata".
19 Fit to estimation data: 78.01% (stability enforced)
20 FPE: 1.071, MSE: 1.068

```

matlab

Validación del modelo comparándolo con la simulación de HOMER I/O:

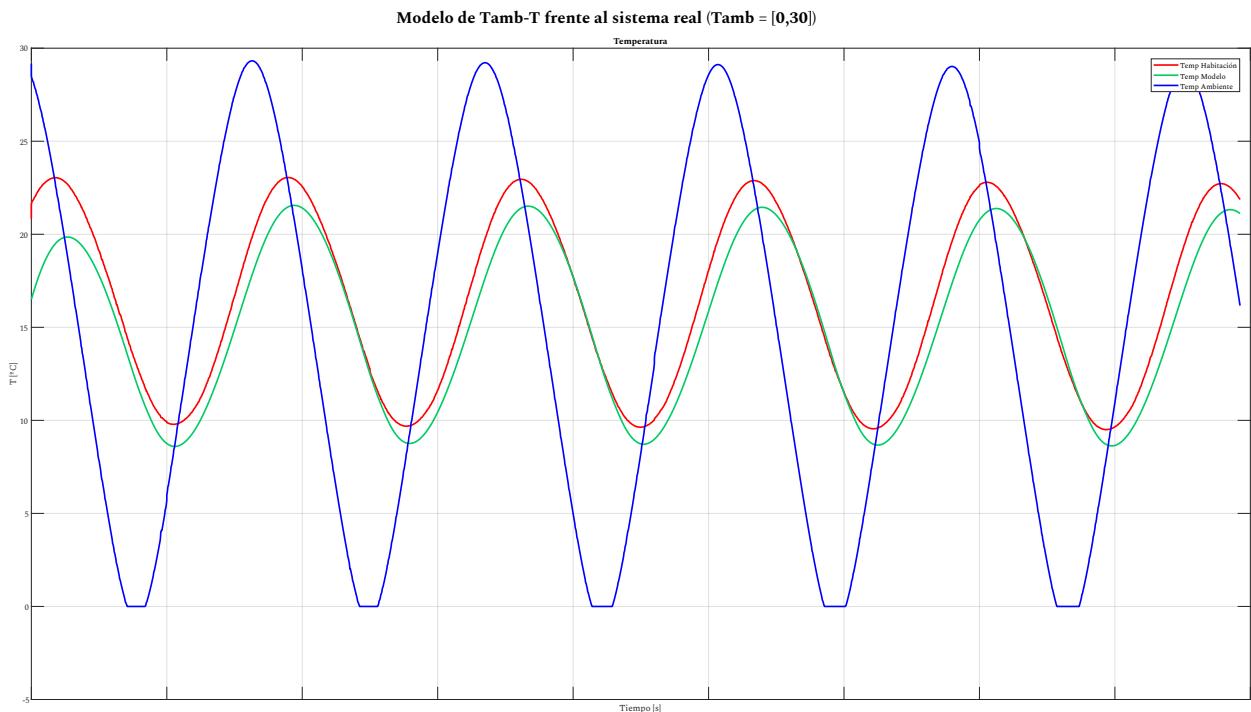


Figura 16: Modelo de Temperatura ambiente-Temperatura frente a la evolución del sistema real

5.4. Modelo Luminosidad - Temperatura de Habitación

Se procede a identificar la influencia de la luminosidad en la temperatura de la habitación. Se desconecta el calefactor y se deja evolucionar el sistema únicamente con bajo la influencia de la luminosidad.

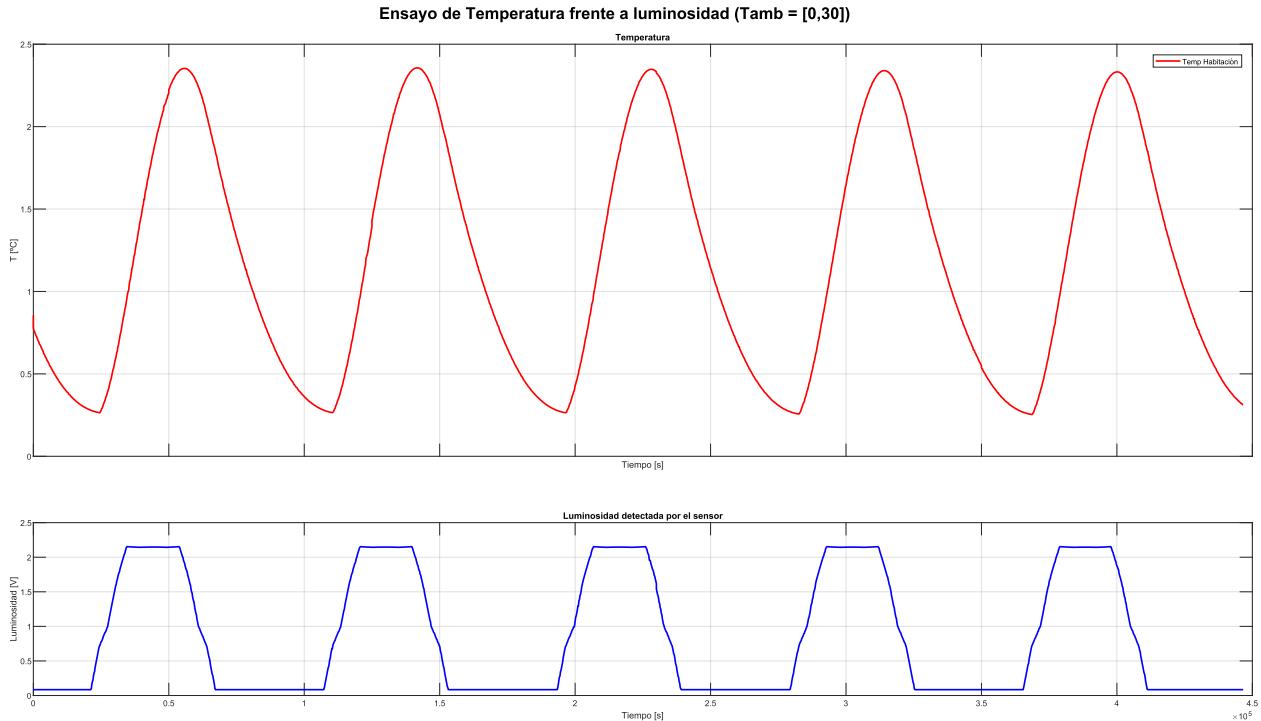


Figura 17: Ensayo de variación de temperatura con la luminosidad

Estimación y función de transferencia:

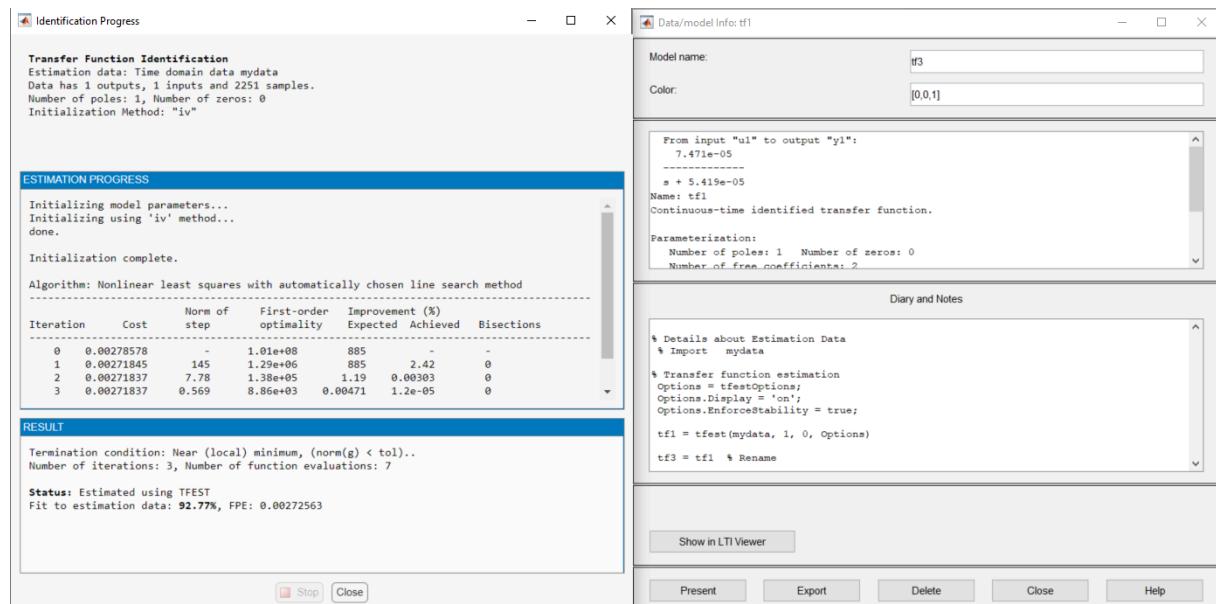


Figura 18: Estimación del modelo de perturbación

```

1  >> tf3
2
3  tf3 =
4      From input "u1" to output "y1":
5          7.471e-05
6  -----
7  s + 5.419e-05
8
9  Name: tf3
10 Continuous-time identified transfer function.
11
12 Parameterization:
13     Number of poles: 1    Number of zeros: 0
14     Number of free coefficients: 2
15     Use "tfdata", "getpvec", "getcov" for parameters and their uncertainties.
16
17 Status:
18 Estimated using TFEST on time domain data "mydata".
19 Fit to estimation data: 92.77% (stability enforced)
20 FPE: 0.002726, MSE: 0.002718

```

Validación del modelo comparándolo con la simulación de HOMER I/O:

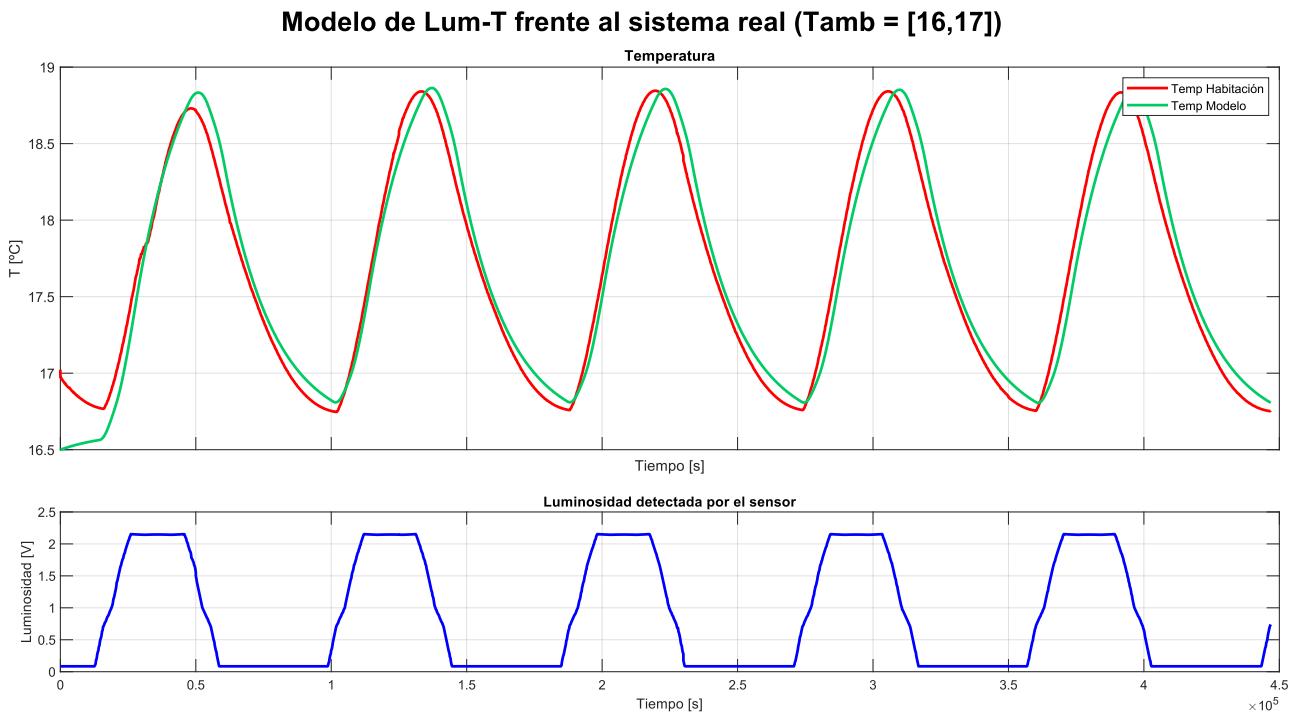


Figura 19: Modelo de Luminosidad-Temperatura frente a la evolución del sistema real

6. Identificación de modelos multivariable

Se pretenden realizar ahora modelos que tengan en cuenta la contribución tanto de la señal de control como de las perturbaciones.

6.1. Suma de contribuciones individuales

Una primera idea o aproximación sería sumar varias contribuciones a la vez, es decir, sumar las salidas de los modelos anteriores. Si se hace esto tanto sumando el modelo del calentador y el de temperatura ambiente (excluyendo luminosidad), como sumando los tres, se obtiene lo siguiente:

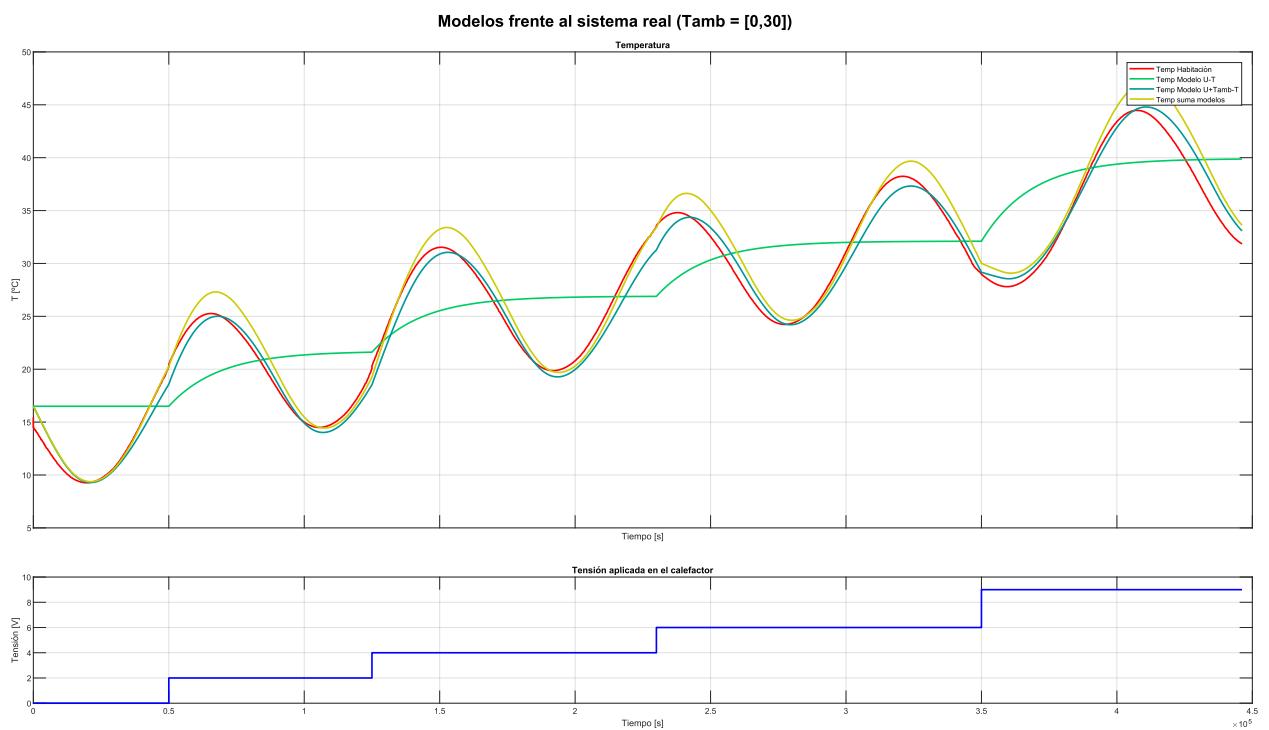


Figura 20: Validación del Principio de Superposición: La suma de las respuestas individuales reconstruye la salida real del sistema.

Observamos que estos modelos de suma funcionan mejor que únicamente el modelo del calentador, sin embargo sucede algo que a primera vista resulta confuso: la suma de dos contribuciones se ajusta mejor que la suma de tres. Sin embargo, esto tiene una explicación lógica, la correlación entre ambas perturbaciones. En el simulador, la influencia de la temperatura exterior nunca puede suprimirse del todo, igual que la de luminosidad, ya que la temperatura ambiente se puede dejar como máximo, en una diferencia de 1°C entre el máximo y el mínimo, y por mucho que la nubosidad se suba al máximo, siempre habrá algo más de luminosidad de día que de noche. Por tanto, como el momento en que la temperatura ambiente es mayor, y la luminosidad es máxima, lo cual tiene como consecuencia que parte de la contribución de cada perturbación se sume en la otra. Esto hace que la suma de las tres contribuciones siempre vaya por encima a la de dos, y se ajuste peor al sistema real.

6.2. Modelo en espacio de estados

También se pretende realizar un modelo en espacio de estados para el sistema. Esto se realiza a partir del código adjunto en el programa [identificacion_ss.m](#). En este código se obtiene un espacio de estados en función de un ensayo realizado. Esto se realiza tanto para dos como para tres variables. En este caso no debería suceder que el modelo de dos variables sea más preciso que el de tres, ya que se le está pasando el experimento completo al programa, de manera que este puede identificar si existe correlación entre estas variables. El resultado obtenido demuestra esta hipótesis:

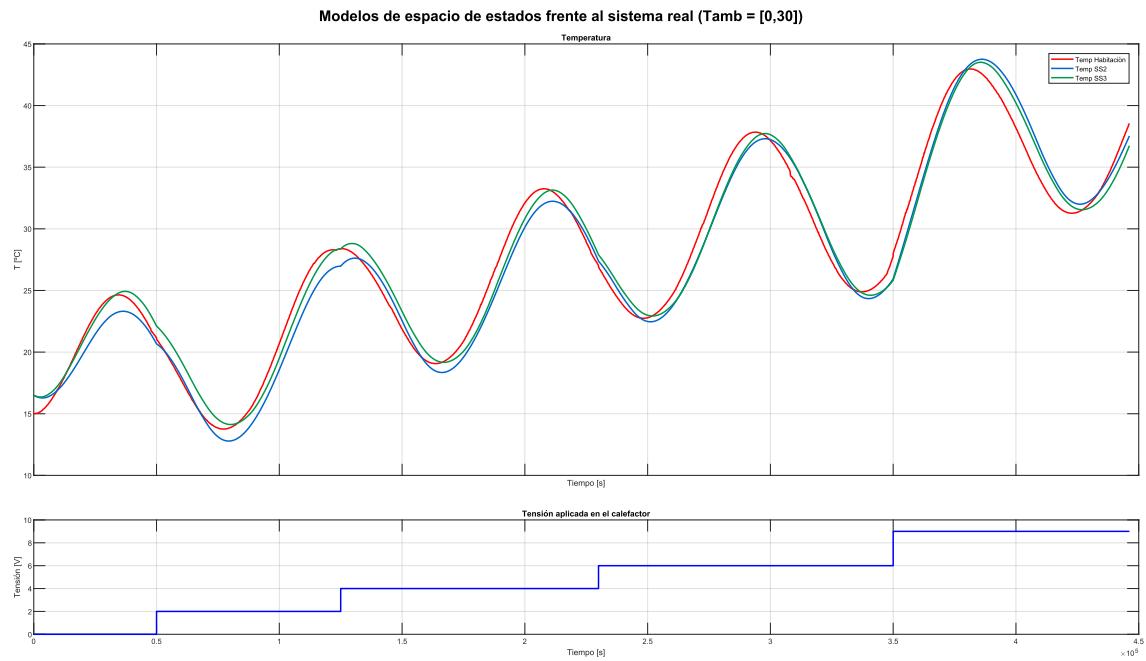


Figura 21: Comparativa de validación: El modelo con 3 entradas (verde) sigue con mayor fidelidad los datos reales (rojo) que el modelo de 2 entradas (azul)

El sistema final se elige el de 3 entradas de manera que convertido a la representación en Espacio de Estados definida por las ecuaciones:

$$x[k+1] = Ax[k] + Bu[k]$$

$$y[k] = Cx[k] + Du[k]$$

Donde el vector de entradas $u[k]$ está compuesto por la entrada de control y las perturbaciones medidas: $u = [U_{\text{cal}}, T_{\text{amb}}, \text{Luz}]^T$.

Las matrices resultantes que definen el comportamiento térmico de la casa son:

$$A = (-4.313e - 05)$$

$$B = (1.1539e - 6 \ 3.7128e - 7 \ 8.1024e - 7)$$

$$C = (89.70)$$

$$D = (0 \ 0 \ 0)$$

Esta representación de estado captura la dinámica completa del sistema y servirá como modelo de predicción interno para el controlador MPC.

7. Diseño de controladores

7.1. Diseño de PID para el control de la temperatura

El primer enfoque consiste en la implementación de un controlador PID (Proporcional-Integral-Derivativo). Este algoritmo calcula la acción de control en función del error presente (P), la acumulación de errores pasados (I) y la predicción de errores futuros (D).

Para la determinación óptima de las ganancias (K_p, K_i, K_d), se ha prescindido de métodos de tanteo manual o empíricos clásicos. En su lugar, se ha utilizado la funcionalidad de Sintonización («Tune») integrada en el propio bloque PID Controller de Simulink.

El procedimiento realizado ha sido el siguiente:

1. Se ha modelado en simulink un bucle cerrado de control con el modelo anteriormente obtenido.
1. Desde la interfaz del bloque PID, se ha ejecutado la herramienta PID Tuner.
2. El software linealiza automáticamente la planta identificada en el punto de operación de equilibrio ($16.5^{\circ}C$).
3. Mediante el ajuste iterativo de los parámetros de diseño (Response Time y Transient Behavior), se ha buscado un compromiso entre velocidad y estabilidad.

Se ha priorizado un diseño robusto sobre uno excesivamente agresivo, ajustando la respuesta para minimizar las sobreoscilaciones, dado que en sistemas de climatización los cambios bruscos de potencia son ineficientes y pueden dañar el actuador.

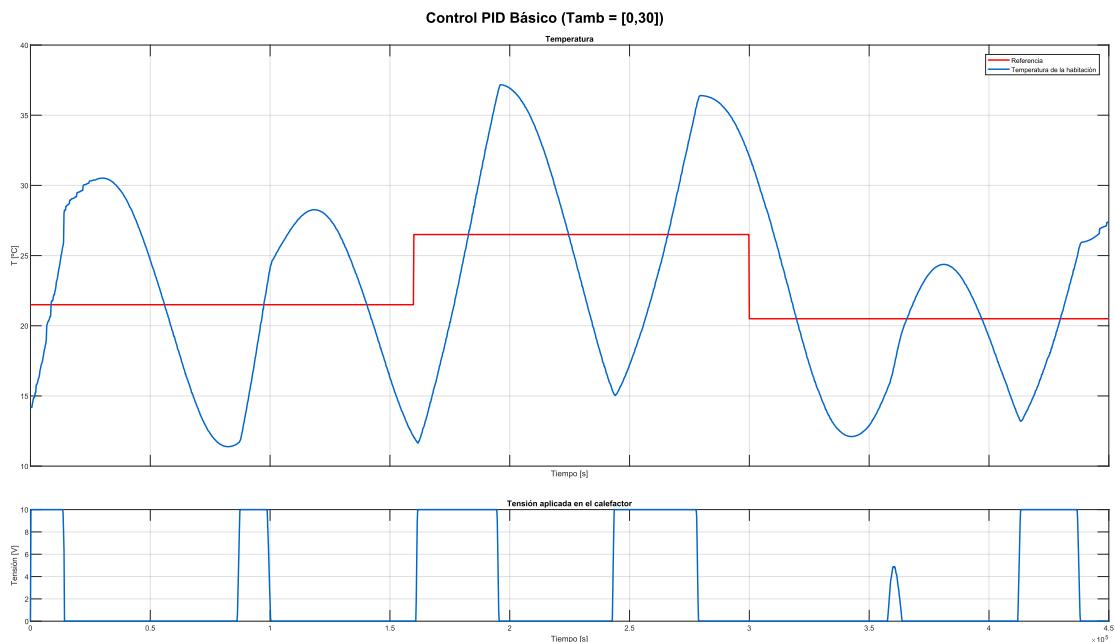


Figura 22: Controlador PID Básico

El resultado sin embargo resulta bastante decepcionante. Las perturbaciones tienen un gran efecto en el controlador, además de estar muy condicionado por el windup, lo cual se observa ya que la señal de control aparece en la mayoría del tiempo saturada, dando lugar a un controlador que está lejos de ser aceptable.

7.2. Gestión del Windup

A pesar de la sintonización automática, el PID lineal no contempla por sí solo los límites físicos del actuador (0-100%). Por ello, se ha configurado el método de clamping (sujeción) dentro de las opciones avanzadas del bloque. Esto detiene la acción integral cuando el actuador se satura, evitando que el error acumulado crezca de forma desmedida y garantizando una recuperación rápida cuando el error cambia de signo.

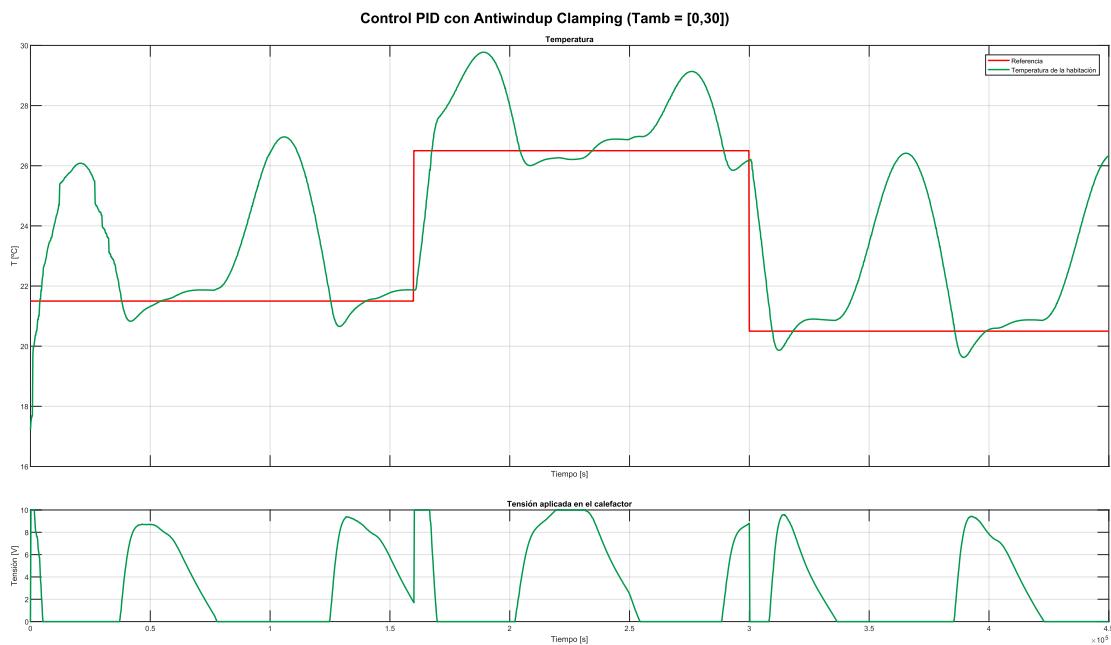


Figura 23: Controlador PID Básico con efecto anti-windup

Este resultado sí es más interesante. Se observa que el control está claramente condicionado por la perturbación de temperatura ambiente, de manera que este no puede hacer mucho cuando la perturbación crece. Sin embargo, es capaz de controlar a pesar de ella. Todo esto ha sido realizado en un experimento con una gran variación de temperatura ambiente, de 0 a 30°C, algo irreal en un entorno cotidiano.

7.3. Control Feedforward para el rechazo de perturbaciones

Finalmente, concluyendo con los controladores no predictivos se procede a implementar un controlador de tipo Feedforward. El objetivo principal de implementar un lazo de control por prealimentación es dotar al sistema de capacidad de anticipación frente a las perturbaciones medibles, específicamente la temperatura ambiente y la luminosidad incidente. A diferencia del control por realimentación, que reacciona a posteriori una vez que la perturbación ya ha desviado la temperatura de la habitación de la referencia, el controlador feedforward actúa a priori: mide la magnitud de la perturbación

entrante y calcula la acción de control exacta necesaria para cancelar su efecto antes de que este llegue a alterar la dinámica de la temperatura.

Este ensayo se realiza en un rango de temperatura ambiente más reducido y realista, entre 10 y 23°C

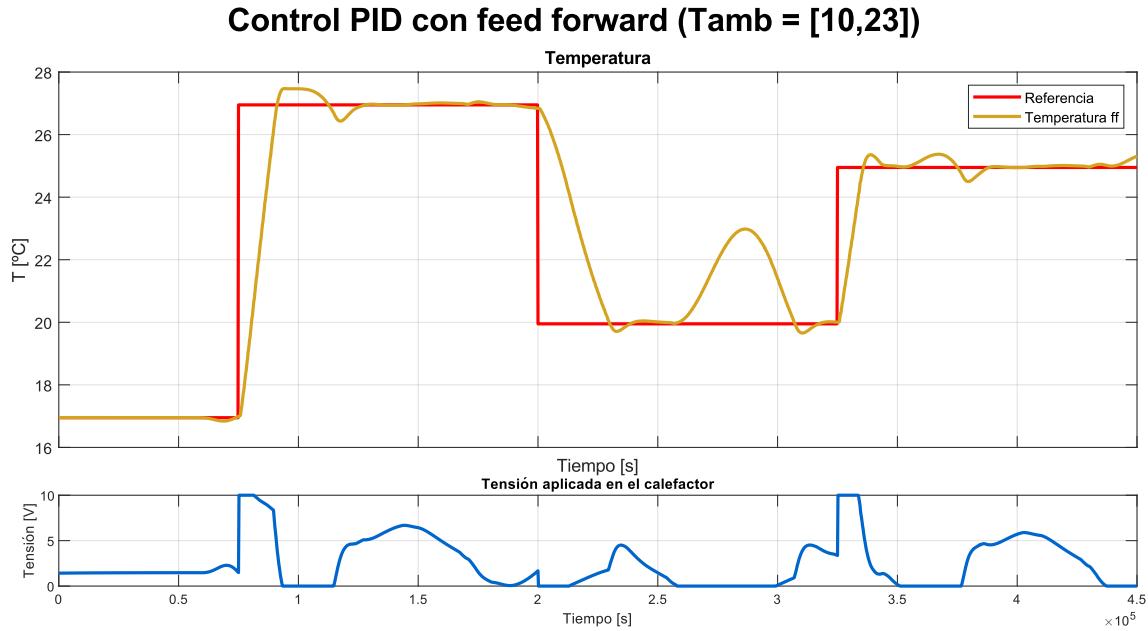


Figura 24: Controlador PID Básico con efecto anti-windup y feedforward

8. Parametrización para MPC

Como base para la predicción, se utiliza el modelo en espacio de estados de tercer orden identificado en la Sección 6. Este modelo multivariable integra no solo la acción de control, sino también las perturbaciones medibles de temperatura ambiente y luminosidad. Las ecuaciones de estado discretas utilizadas por el controlador son las vistas en la Sección 6.2., donde las matrices A, B, C capturan la inercia térmica de la Sala L, permitiendo al controlador estimar cómo evolucionará la temperatura ante cambios en la calefacción y el entorno.

El objetivo del controlador se define mediante una función de coste cuadrática (J) que se minimiza en tiempo real. Este controlador se ha configurado para priorizar la minimización del error de seguimiento frente al esfuerzo de control. Esto garantiza que el sistema busque siempre alcanzar la referencia, asumiendo el coste energético necesario. La función de coste es la siguiente:

$$J(u) = \sum_{i=1}^P w_y (r[k+i] - y[k+i])^2 + \sum_{i=0}^{M-1} w_u (\Delta u[k+i])^2$$

Donde se busca minimizar la desviación frente a la referencia y suavizar los cambios en la señal de control.

Para el diseño del controlador MPC, ha sido necesario modelar tanto la temperatura ambiente como la luminosidad como señales periódicas. Para modelar estas perturbaciones, a partir de los ensayos realizados previamente, se extraen los parámetros temporales.

El periodo medido es:

$$T = 86482 \text{ s}$$

Esto corresponde a una frecuencia:

$$f = 11.5\mu \text{ Hz}$$

Para la configuración del bloque de control, se requiere la frecuencia angular ω :

$$\omega = 72.65 \cdot 10^{-6} \text{ rad/s}$$

Para el diseño del MPC, después de varias iteraciones, se elige un Horizonte de Predicción (P) de 25 muestras, lo que otorga una visión a futuro de 5000 segundos, es decir, 1 hora y 23 minutos, ya que el tiempo de muestreo del sistema es de 200 segundos. Este es suficiente para predecir la tendencia de calentamiento sin incurrir en costes computacionales elevados. El Horizonte de Control (M) se fija en 4 muestras, una estrategia conservadora que favorece acciones de control suaves y evita cambios bruscos e agresivos innecesarios.

Respecto a la función de coste, se priorizó la minimización del error de seguimiento frente al consumo energético garantizando que el sistema siempre intente alcanzar la referencia.

Para crear este controlador y elegir de manera correcta los valores del horizonte de predicción y del horizonte de control, así como los costes del sistema, se ha partido de un controlador MPC inicial generado a través del script `mpc.m`, donde se han configurado los horizontes y los pesos iniciales. Sobre este `MPCOBJ` generado se ha trabajado a través del bloque MPC controller de simulink para configurarlo de forma que se ajuste de una forma más correcta al sistema. Para ello, se ha configurado para que el control sea algo más agresivo y más rápido. Este objeto se añade en `datos_relevantes.mat`.

En la Figura 25 se puede observar la respuesta de este controlador ante un ensayo entre 10 °C y 23 °C de temperatura ambiente.

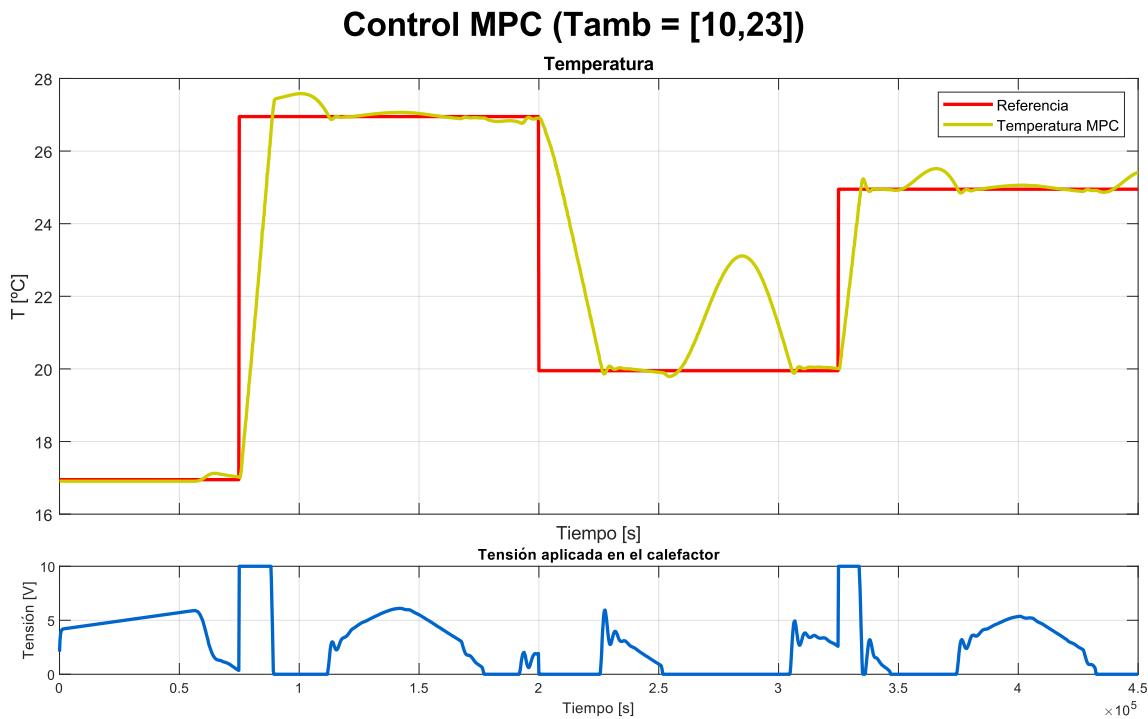


Figura 25: Controlador MPC

9. Comparativa entre controladores

Se procede a comparar los tres principales controladores diseñados: el PID con antiwindup, el PID con antiwindup y feedforward y el MPC. Para ello, como es necesario realizar 3 ensayos distintos, se define una metodología estricta a seguir para realizar los ensayos, y así no penalizar a ninguno de ellos respecto a los demás. La metodología es la siguiente:

- Se coloca el sistema en el día 13/10/2025 con una temperatura ambiente entre 16 y 17°C. Se acelera la evolución al máximo de velocidad
- Se detiene el sistema el día 15/10/2025 a las 2:00. Para entonces el sistema se ralentiza a x1 de velocidad y deberá estar estable en torno a los 16.9°C.
- Se cambia entonces la temperatura ambiente a un rango de 10 a 23°C. Al estar ejecutándose al mínimo de velocidad, esta no cambiará casi nada la temperatura de la habitación hasta dentro de bastante tiempo.
- Se lanza entonces la simulación, y cuando esta llegue a 50000 segundos, se acelera el sistema al máximo de velocidad. Esto hará que a partir de los 58000 segundos, todos los ensayos arranquen desde un punto muy similar y a la máxima velocidad

Como es obvio, las métricas solo se calculan a partir de esos 58000 segundos (se dan 8000 segundos de margen).

El resultado de estos ensayos es el siguiente:

Comparativa diferentes controladores (Tamb = [10,23])

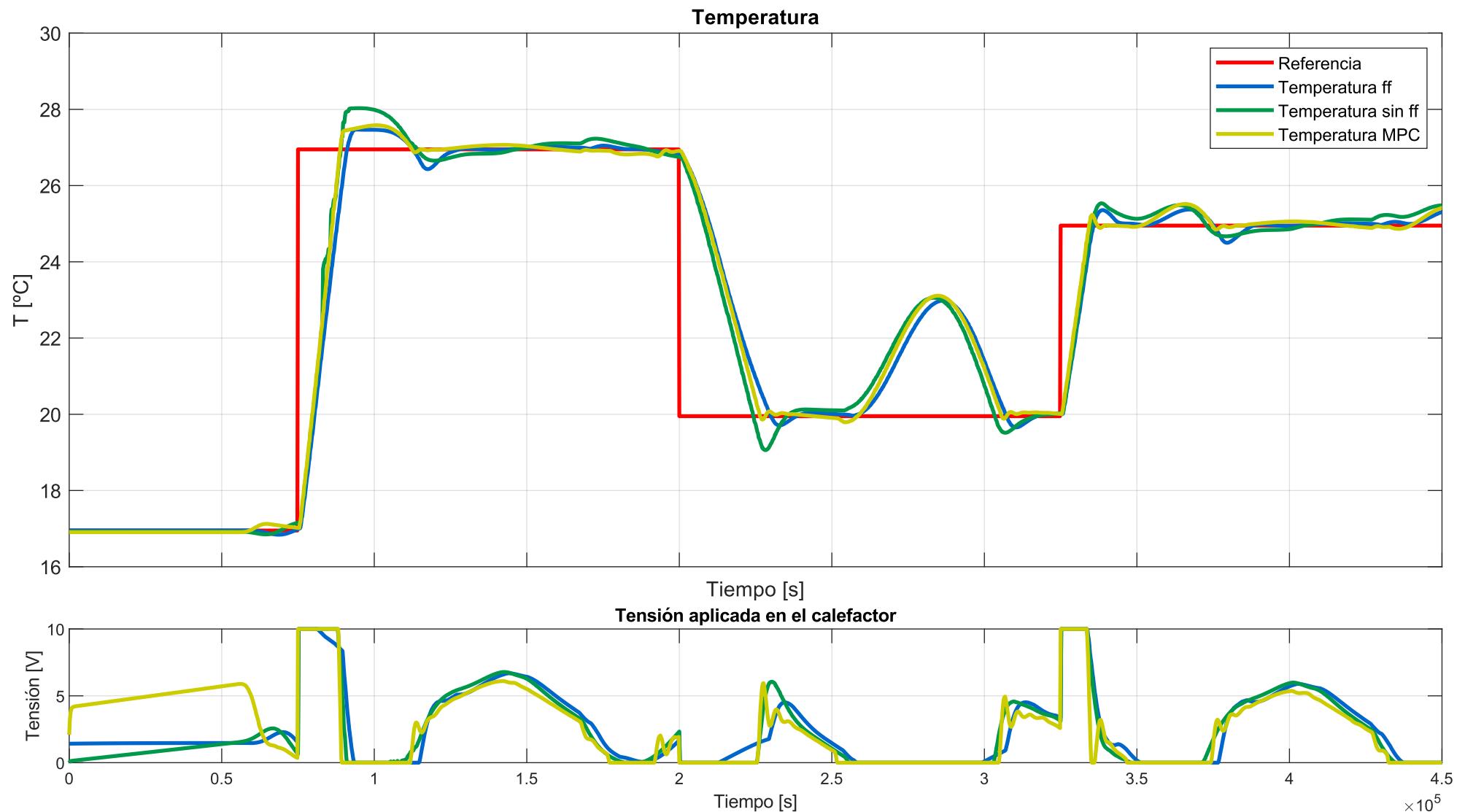


Figura 26: Comparativa entre los controladores finales

A la vista de los resultados obtenidos en la simulación, Tabla 1 y Figura 27, se observa:

- El MPC se destaca indiscutiblemente como la opción más eficiente, con un consumo medio de 176.30 W. A diferencia del PID, que es reactivo, el MPC anticipa la inercia térmica. Esto le permite realizar cortes de energía estratégicos que el PID no puede ver, ahorrando aproximadamente un 9.4% de energía respecto a los PID. Entre ambos modelos de PID, la diferencia es mínima, consumiendo el que tiene feedforward más al tener inyecciones más agresivas para compensar perturbaciones de antemano.
- Al estudiar el error medio, se observa que el que mejor se ajusta a la referencia es el MPC, siendo este el que traza la trayectoria más optima. Éste modelo está seguido por el modelo PID con feedforward, que ayuda a desacoplar las perturbaciones externas permitiendo que el lazo PID se centre únicamente en corregir la dinámica interna y consiguiendo así mejores resultados que con el PID normal.

Tipo de Controlador	Potencia Media Consumida (W)	Error Medio (MSE, °C)
PID + Clamping	194.66	0.8955
PID + Clamping + Feedforward	195.69	0.8681
MPC	176.30	0.8302

Tabla 1: Comparativa de desempeño de las estrategias de control

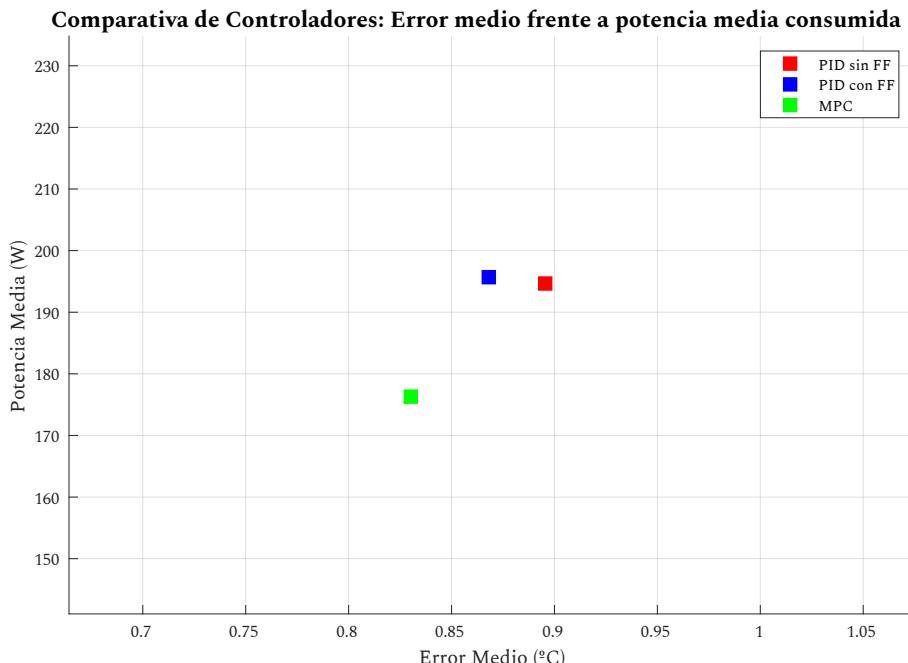


Figura 27: Error medio frente a potencia media consumida. Comparativa entre PID sin Feedforward, PID con feedforward y MPC.

Además, se han calculado diferentes índices de desempeño que permiten analizar el correcto funcionamiento de los controladores:

- IAE, suma total del error acumulado durante toda la simulación. En este caso, el MPC es el que tiene un valor más bajo, y por lo tanto es mejor, ya que ha mantenido la temperatura más cerca de la referencia durante más tiempo en total. El PID estándar acumula más error porque tarda más en reaccionar ante los cambios.
- ITAE, suma total del error acumulado multiplicado por el tiempo. El hecho de que el MPC tenga el valor más bajo confirma que es el controlador que más rápido elimina el error y estabiliza la temperatura. El PID clásico tarda mucho más en eliminar ese error, y por eso el tiempo lo castiga severamente.
- ITSE, suma total de los errores al cuadrado multiplicados por el tiempo. En este caso, aunque el PID es más lento y acumula más error total, su respuesta es muy suave. No tiene cambios bruscos. Por el contrario, el PID con Feedforward y el MPC son más agresivos, teniendo así más error cuadrático.

Tipo de Controlador	IAE $(\int e(t) dt)$	ITAE $(\int t * e(t) dt)$	ITSE $(\int t * e(t)^2 dt)$
PID + Clamping	3.53×10^5	7.62×10^{10}	2.29×10^{11}
PID + Clamping + Feedforward	3.41×10^5	7.16×10^{10}	2.48×10^{11}
MPC	3.28×10^5	6.93×10^{10}	2.38×10^{11}

Tabla 2: Comparativa de índices de desempeño: IAE, ITAE e ITSE

El MPC demuestra ser la estrategia superior globalmente, teniendo mejores resultados de consumo energético, de error medio, y además dominando en IAE e ITAE. Sin embargo, el PID básico obtiene el mejor ITSE, lo que indica que, a pesar de su lentitud, ofrece una respuesta muy suave y libre de picos de error bruscos en el estado estacionario, mientras que las estrategias más complejas (FF y MPC) pueden inducir pequeñas oscilaciones transitorias que son penalizadas por el término cuadrático.