



**UNIVERSIDAD  
CARLOS III DE MADRID**

# **CONTROL INTELIGENTE**

## **CONTROL DE ÁNGULO PITCH DE UN HELICÓPTERO CON CONTROL BORROSO**

**Profesores: Luis Enrique Moreno Lorente  
Luis Santiago Garrido Bullón  
Dorin Sabin Copaci**

## CONTROL DE ÁNGULO PITCH DE UN HELICÓPTERO CON CONTROL BORROSO

El objetivo de este trabajo será conseguir controlar el ángulo *Pitch* de un helicóptero de manera que pueda seguir a una señal de referencia mediante técnicas de control borroso.

### 1. Modelo de simulación

Se parte del modelo del trabajo anterior, donde de igual manera se controlaba el ángulo *Pitch* de un helicóptero por optimización de los parámetros de un controlador PID con el algoritmo *Differential Evolution*, y donde también se comprobó la necesidad de incorporar una segunda realimentación a través de una matriz estática (bloque SOF) para lograr una respuesta estable. El modelo usado en ese primer trabajo se recuerda a continuación (figura 1).

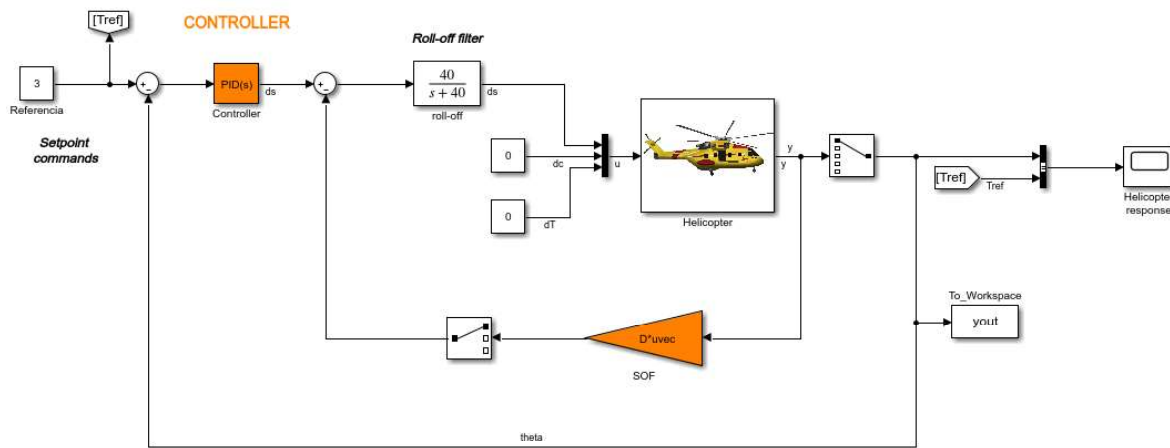


Figura 1: Modelo de partida

Para este segundo trabajo se tendrá que cambiar ese controlador PID por un controlador borroso. La matriz que se utiliza en el bloque SOF será la misma que se obtuvo en el trabajo anterior, que se recuerda es:

$$D = \begin{bmatrix} 4,59 & -0,0003554 & 2,531e - 5 & 0,6735 & -0,002247 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

## 2. Control borroso PD+I

Para este trabajo se va a implementar un controlador borroso PD+I. El modelo creado es el que se muestra en la figura 2 y nombrado como *rct\_fuzzy\_SOF.slx*.

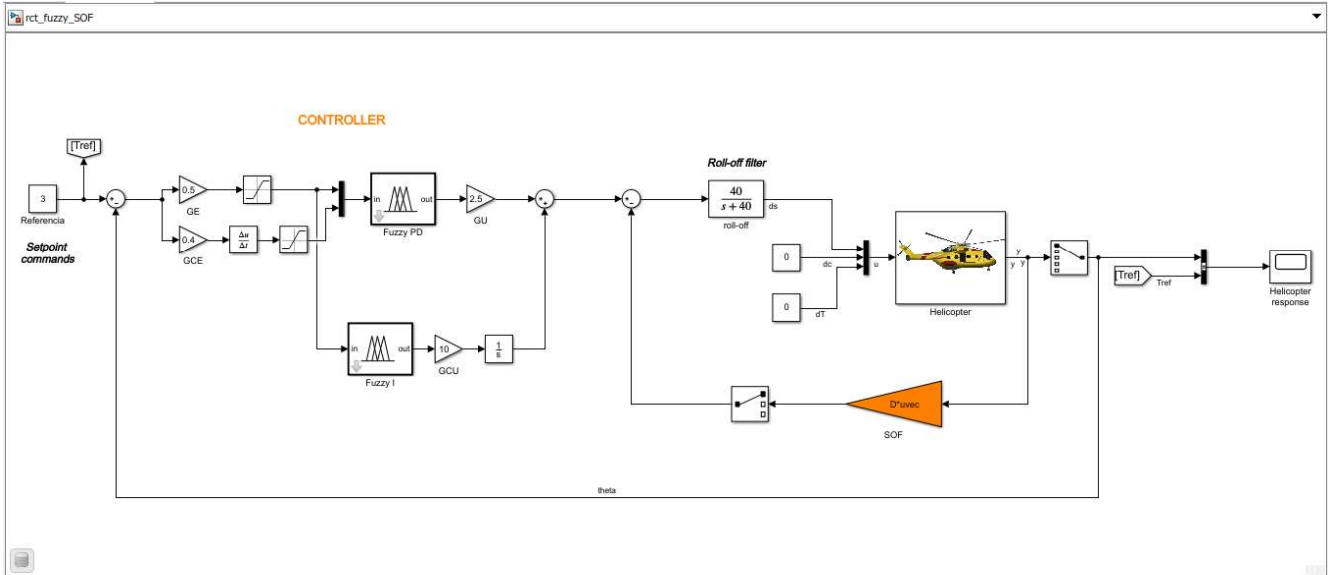


Figura 2: Modelo de simulación para control borroso PD+I

Como se vio en clase, se utiliza un controlador borroso PD+I y no un PID para reducir el número de reglas necesarias para el control *fuzzy* y así la complejidad del problema. A continuación, se detalla cómo se han diseñado ambos bloques *fuzzy*.

- **Bloque Fuzzy PD.**

Este bloque utiliza el fichero *rct.fis*, cuya configuración es la siguiente (figura 3):

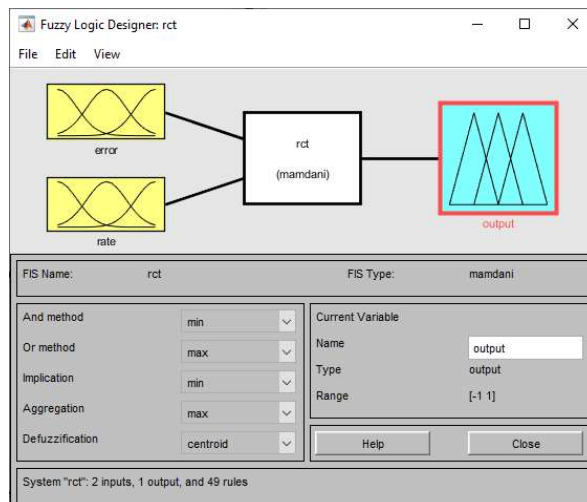
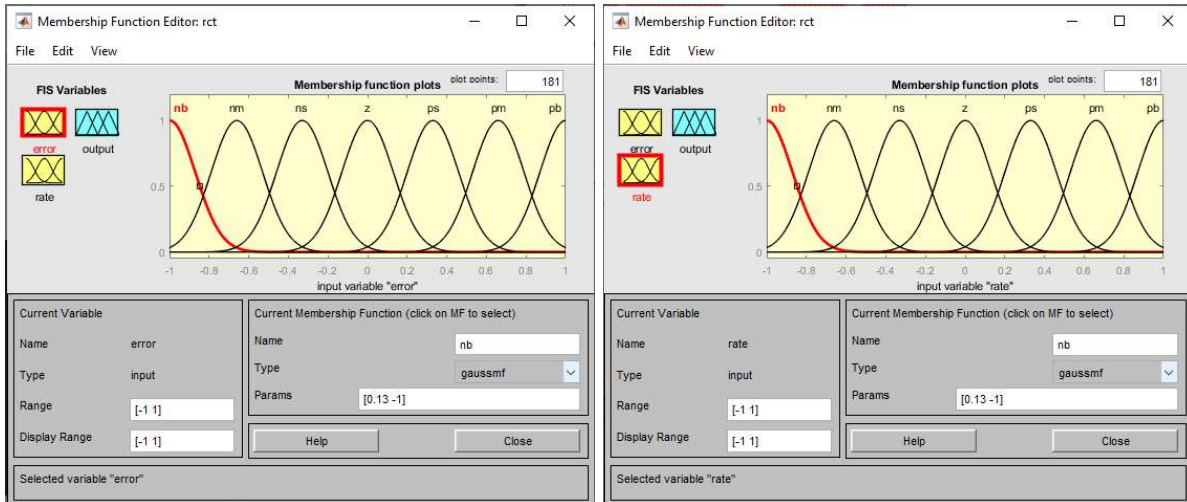
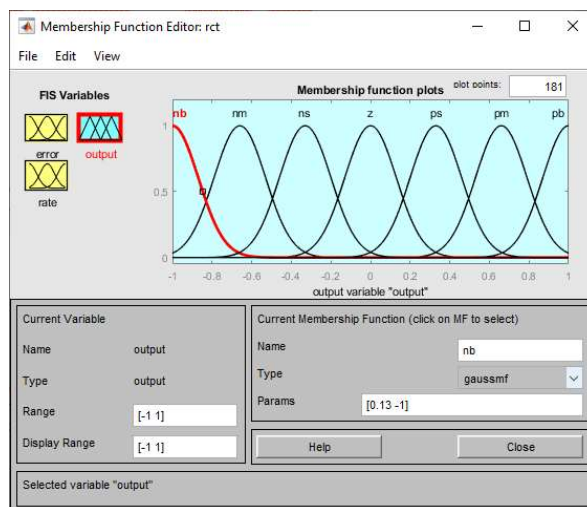


Figura 3: Configuración del fichero *rct.fis*

Como se puede observar, tiene dos entradas: *error* y *rate*, que es la derivada del error. Ambas están definidas en un rango de valores de -1 a 1. La salida *output*, también está comprendida en ese rango de valores. Las funciones de pertenencia tienen la misma distribución para ambas entradas y también la salida, siendo esta distribución la que se muestra en la figura 4.

(a) Funciones de pertenencia entrada *error*(b) Funciones de pertenencia entrada *rate*(c) Funciones de pertenencia salida *output*Figura 4: Funciones de pertenencia del fichero *rct.fis*

Estas funciones de pertenencia son funciones gaussianas cuyos centros están separados una distancia de 0.33. Los conjuntos que se han establecido son: negative big (nb), negative medium (nm), negative small (ns), zero (z), positive small (ps), positive medium (pm) y positive big (pb).

Las reglas que relacionan las entradas con la salida son las que se expresan en la tabla 1.

		Rate						
		nb	nm	ns	z	ps	pm	pb
Error	nb	nb	nb	nb	nb	nm	ns	z
	nm	nb	nb	nb	nm	ns	z	ps
	ns	nb	nb	nm	ns	z	ps	pm
	z	nb	nm	ns	z	ps	pm	pb
	ps	nm	ns	z	ps	pm	pb	pb
	pm	ns	z	ps	pm	pb	pb	pb
	pb	z	ps	pm	pb	pb	pb	pb

Tabla 1: Reglas fuzzy del controlador fuzzy PD

■ **Bloque Fuzzy I.**

Este bloque utiliza el fichero *rct\_I.fis*, cuya configuración es la siguiente (figura 5):

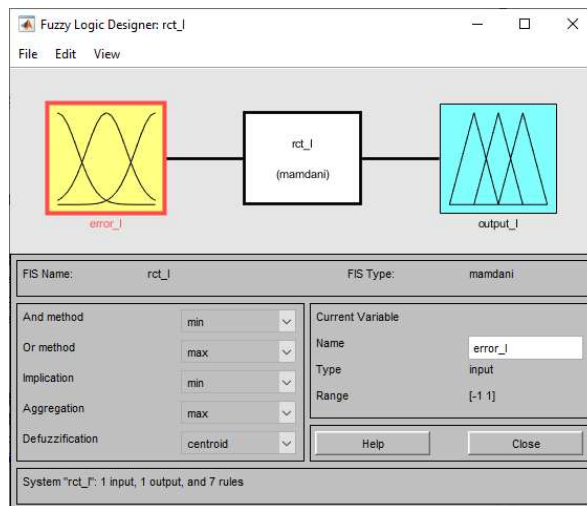
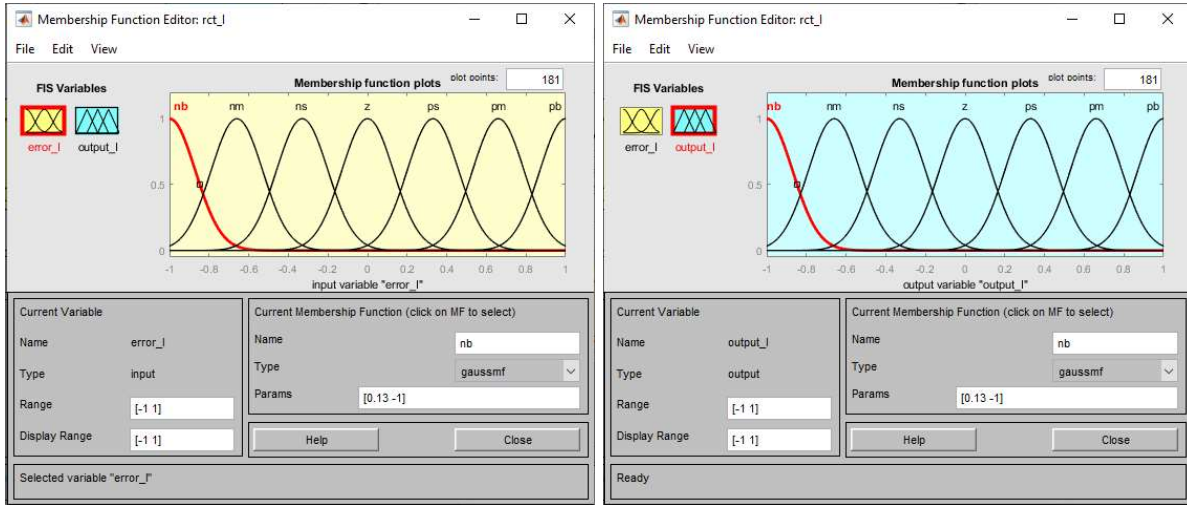


Figura 5: Configuración del fichero *rct\_I.fis*

En este caso el controlador sólo tiene una entrada, el error, con el nombre *error\_I*. Está definida en un rango de valores de -1 a 1. La salida *output\_I*, también está comprendida en ese rango de valores. Las funciones de pertenencia tienen la misma distribución para entrada y salida, y es la misma que la utilizada para el controlador anterior. Puede verse en la figura 6.



(a) Funciones de pertenencia entrada  $error_I$

(b) Funciones de pertenencia salida  $output_I$

Figura 6: Funciones de pertenencia del fichero  $rct_I.fis$

Las funciones de pertenencia son funciones gaussianas como en el controlador anterior, cuyos centros están separados también una distancia de 0.33. Los conjuntos que se han establecido son los mismos: negative big (nb), negative medium (nm), negative small (ns), zero (z), positive small (ps), positive medium (pm) y positive big (pb).

Las reglas que relacionan la entrada y la salida son las que se expresan en la tabla 2.

<b>Entrada <math>error_I</math></b>	<b>Salida <math>output_I</math></b>
nb	nb
nm	nm
ns	ns
z	z
ps	ps
pm	pm
pb	pb

Tabla 2: Reglas fuzzy del controlador fuzzy I

Del modelo diseñado también cabe destacar los bloques de saturación. Ambos están definidos con un límite inferior de -1 y un límite superior de 1, en consonancia con los rangos de entrada de los controladores *fuzzy*.

### 3. Pruebas y resultados

Con todo esto, lo único que falta ya es encontrar unos valores adecuados para los bloques de ganancia que hagan que la respuesta del sistema sea correcta: siga a la señal de referencia y sea estable y realizable. A base de prueba y error se encontraron los siguientes valores que dieron el resultado que se muestra en la figura 7, donde la señal azul es la señal de referencia y la señal amarilla es la salida del sistema

- **Ajuste de ganancias:  $GE = 0.5$ ,  $GCE = 0.4$ ,  $GU = 2.5$  y  $GCU = 10$ .**

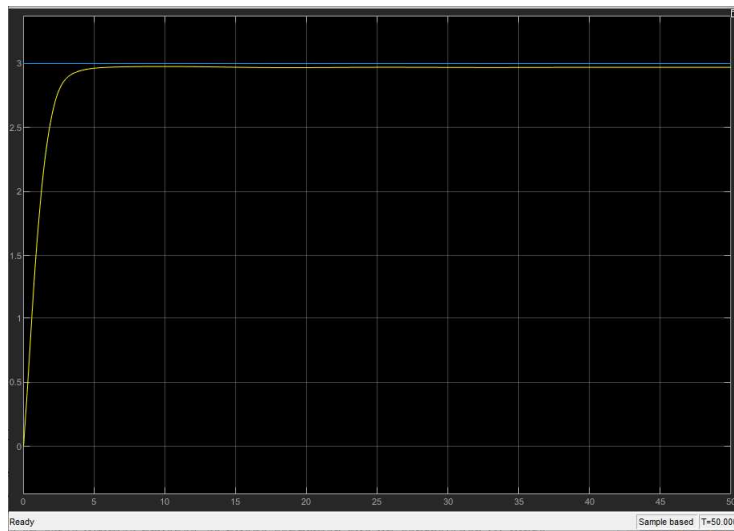


Figura 7: Resultados del control del ángulo *Pitch* mediante técnicas fuzzy

Se observa que se obtiene un resultado muy parecido al que se obtuvo mediante el algoritmo de *Differential Evolution* en el trabajo anterior: una respuesta suave sin sobreoscilaciones pero con un tiempo largo de establecimiento.

Por último se quiso también probar la respuesta del sistema ante entrada escalón, obteniéndose el siguiente resultado (figura 8).

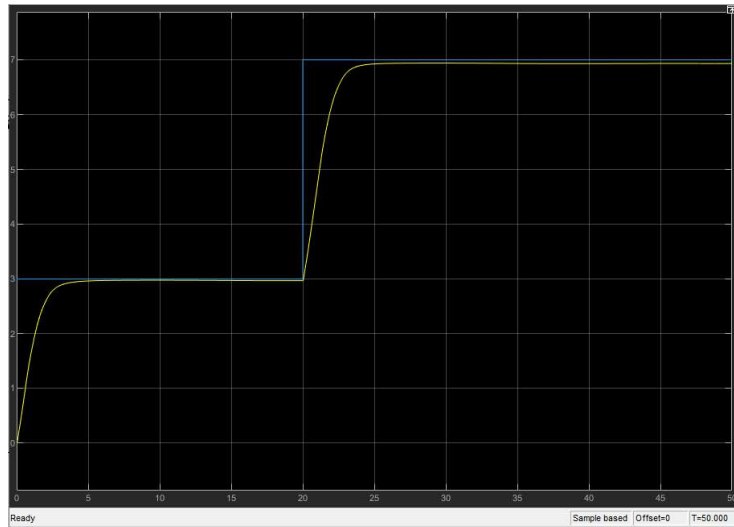


Figura 8: Respuesta ante escalón

Se observa que el sistema es capaz de seguir a la señal de referencia.

## 4. Conclusiones

Con el desarrollo de este trabajo se puede concluir que se ha conseguido controlar mediante técnicas de control borroso el ángulo *Pitch* de un helicóptero de manera sencilla y satisfactoria.