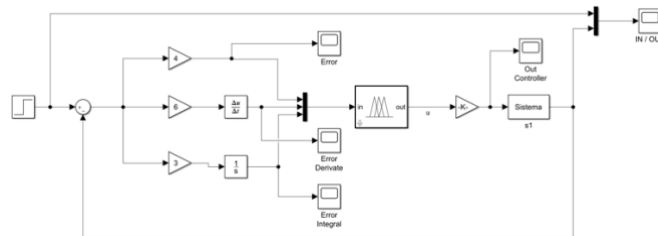




Universidad Complutense de Madrid.
Facultad de Ingeniería Informática.
Inteligencia Artificial Aplicada al Control.



Práctica 4

Lógica Fuzzy

Alumno:
Borges Noronha, Frederick Ernesto

Tabla de contenido

| | |
|---|-----------------|
| <i>Introducción a la práctica.....</i> | <i>3</i> |
| <i>Pasos para reproducir la práctica</i> | <i>3</i> |
| Control PD Borroso | 3 |
| Control PID Borroso. | 7 |
| <i>Resultados obtenidos.....</i> | <i>8</i> |
| <i>Análisis de resultados</i> | <i>9</i> |

Introducción a la práctica

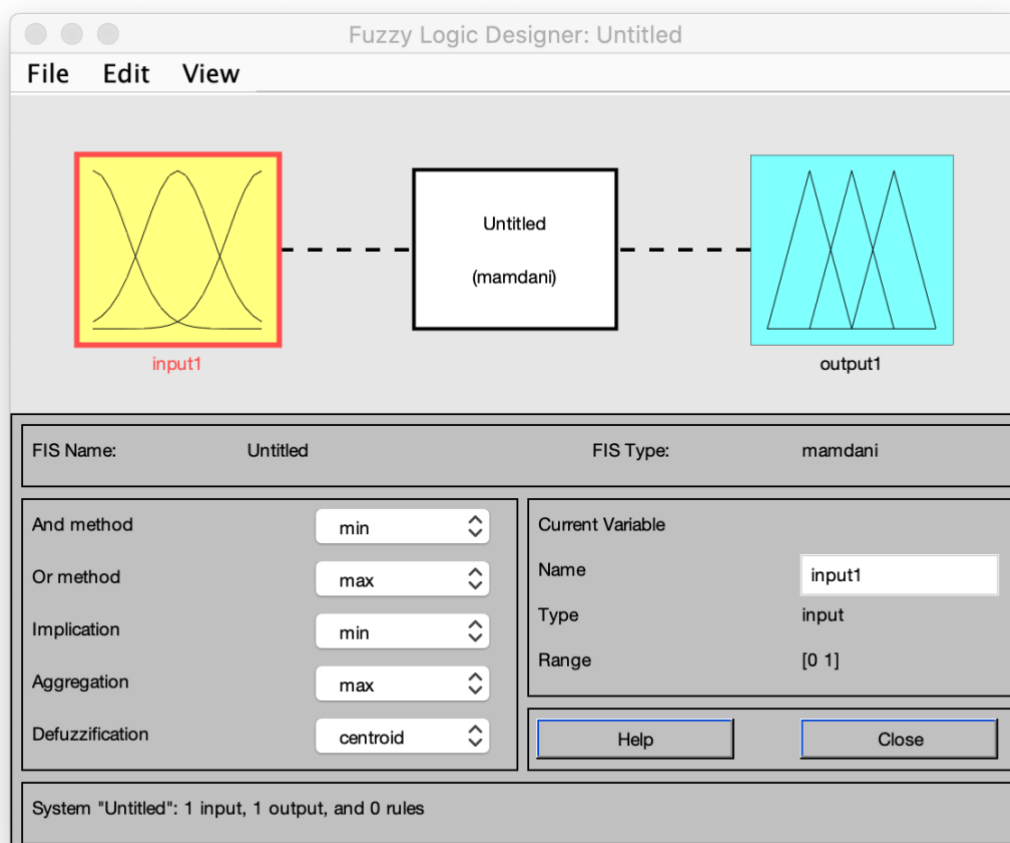
En esta práctica se pedía crear un modelo de lógica fuzzy para estabilizar el sistema introducido en la practica 2, el cual contiene una planta de tipo 2. Para lograr esto se debe tener instalado en MATLAB el modulo de Fuzzy Logic para poder diseñar los ficheros “.fis” que vamos a utilizar para programar el controlador borroso.

Pasos para reproducir la práctica

Control PD Borroso

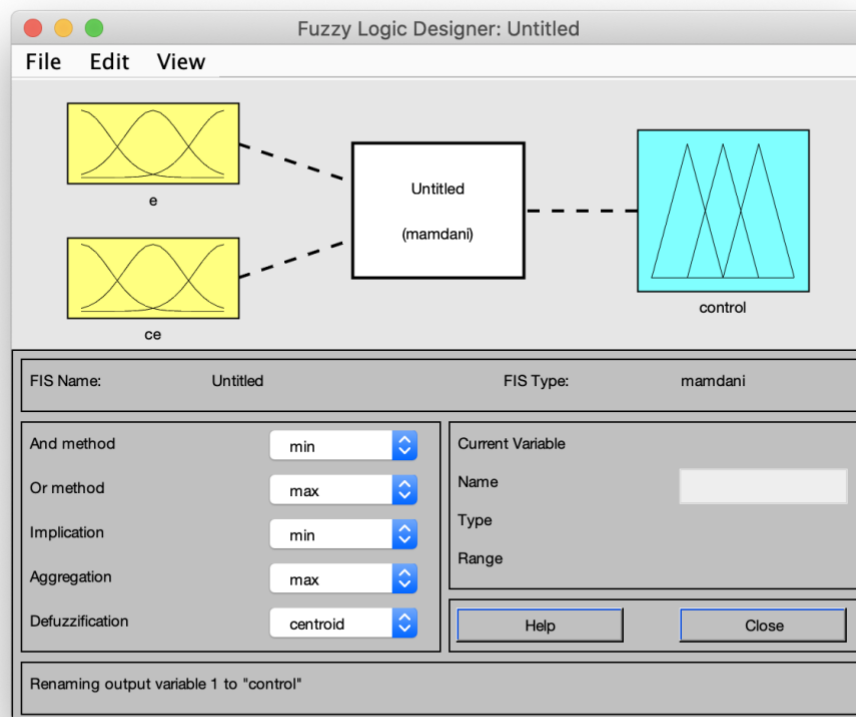
Para realizar el control borroso se deben seguir los siguientes pasos:

1. Crear el fichero pd_fuzzy.fis para el cual en la línea de comandos de MATLAB se debe escribir el comando `fuzzy` el cual mostrará la siguiente ventana:

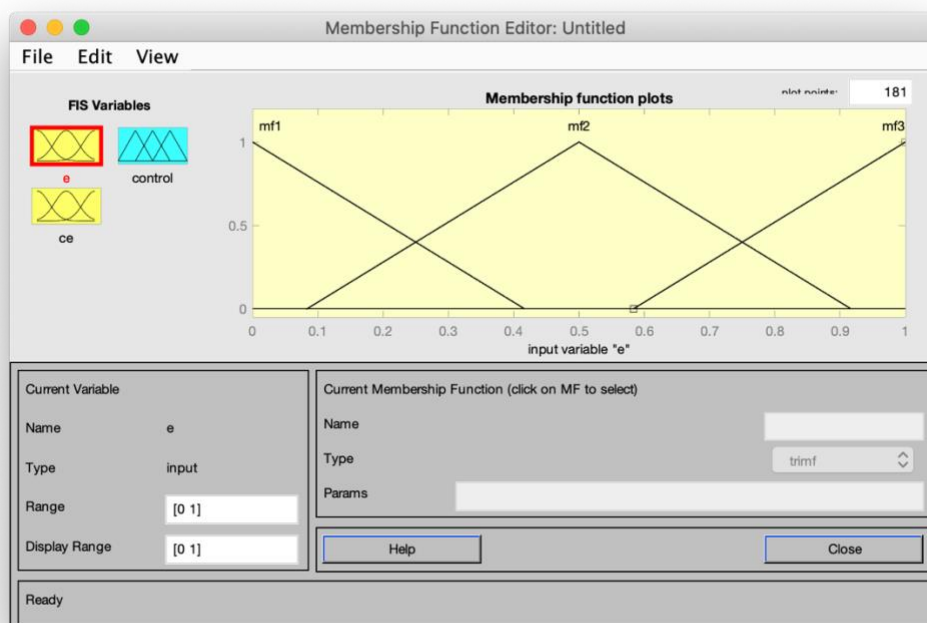


En esta ventana podremos crear las entradas necesarias para construir nuestro fichero “.fis”.

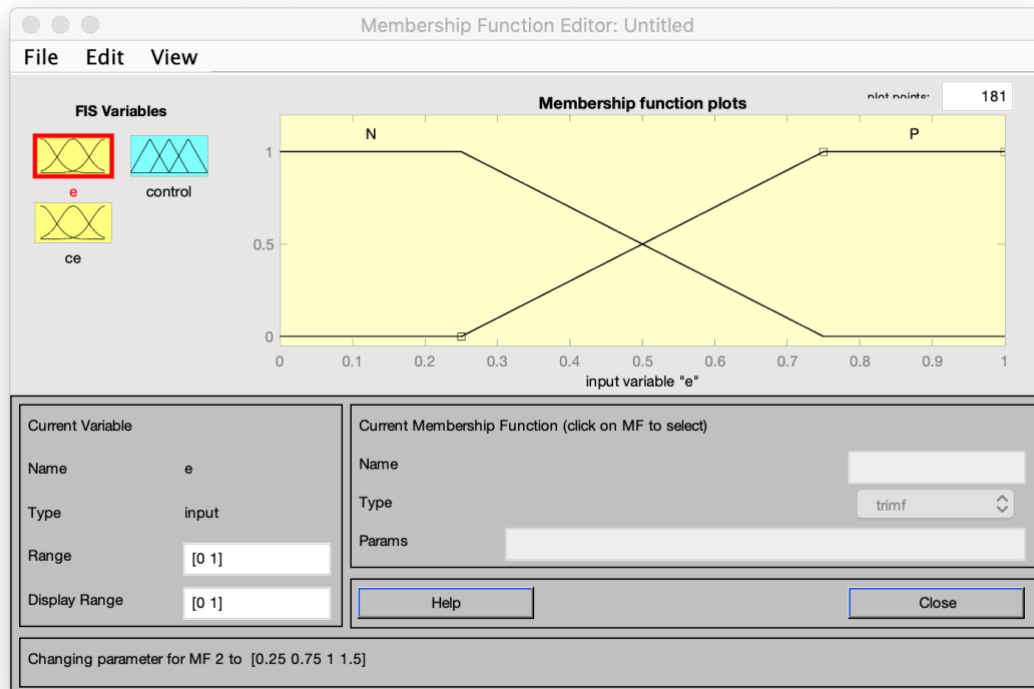
2. Para el caso del control PD vamos a necesitar 2 entradas por tanto agregaremos una haciendo clic en `Edit -> Add variable -> Input`. Deberíamos tener ahora algo de la siguiente forma:



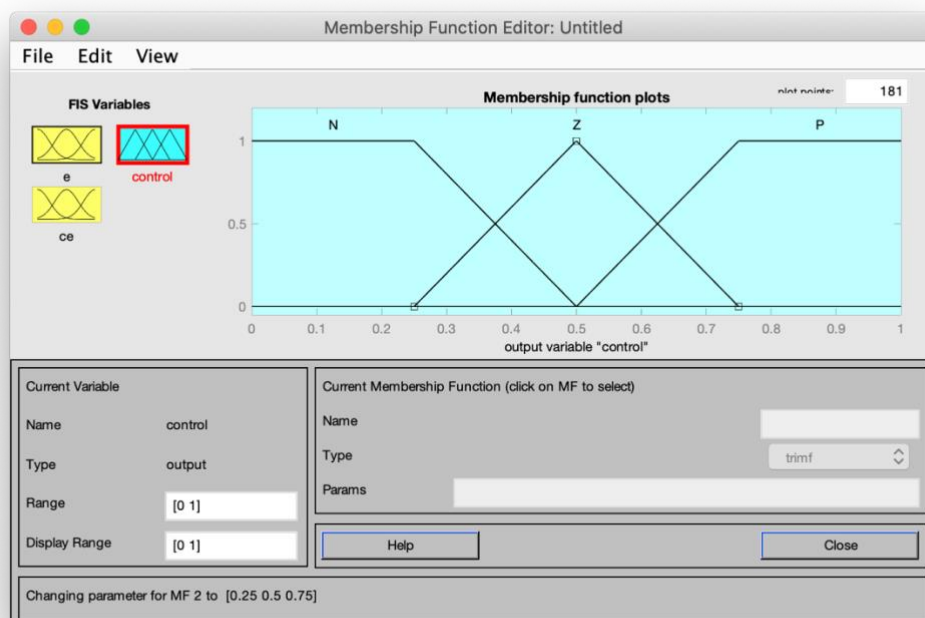
3. Ahora procedemos a hacer doble clic en uno de los recuadros amarillos para realizar modificaciones sobre las variables de los inputs. Esta acción abrirá una ventana como esta:



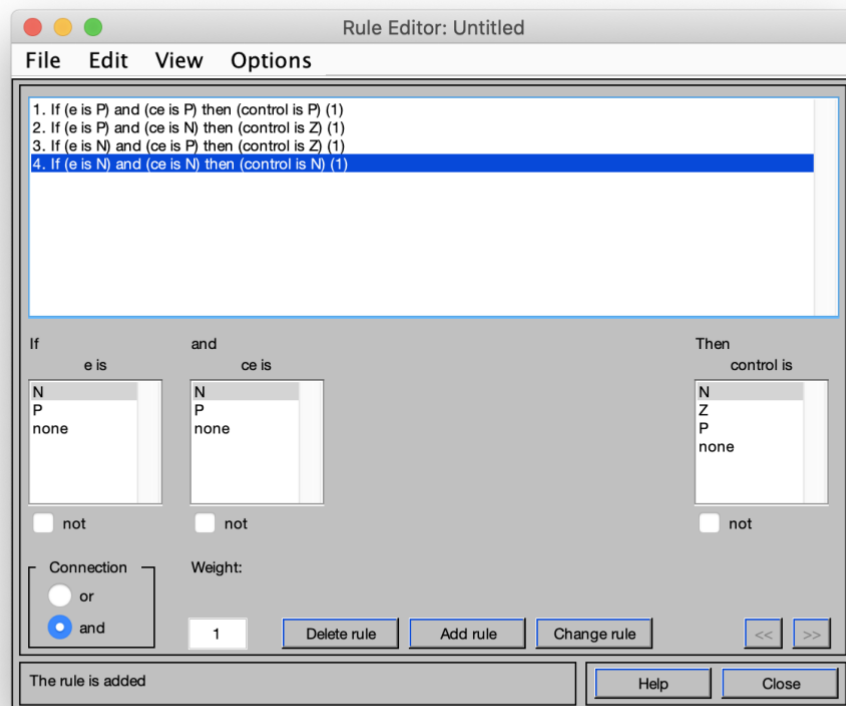
4. En la ventana abierta en el paso anterior vamos a eliminar una de las variables, a ponerlas como `trapmf` y a proceder a dejarlas de la siguiente forma:
- Inputs:



- Output:

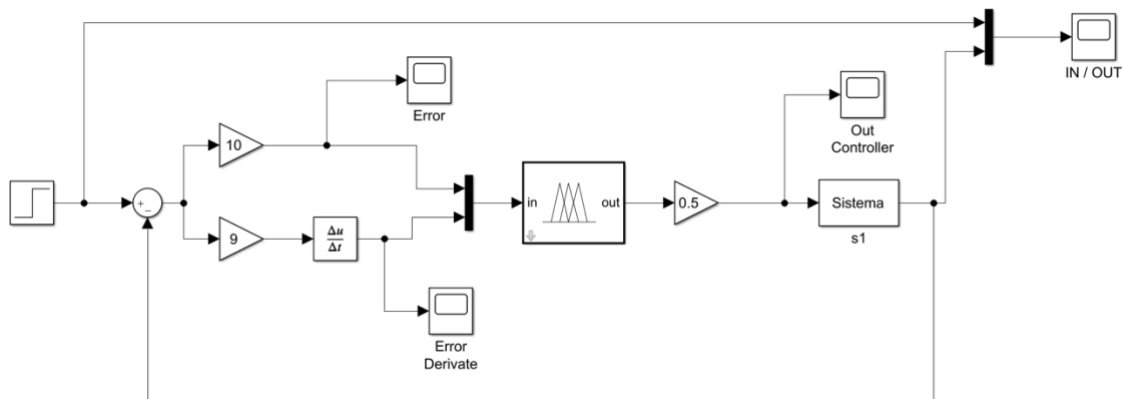


5. Una vez configurado esto, cerramos la ventana y hacemos clic en `Edit -> Rules` para generar las reglas del controlador fuzzy.



6. Cerramos la ventana anterior luego de definir las reglas y procedemos a exportar las reglas a un fichero `.fis` haciendo clic en `File -> Export -> To File...`.

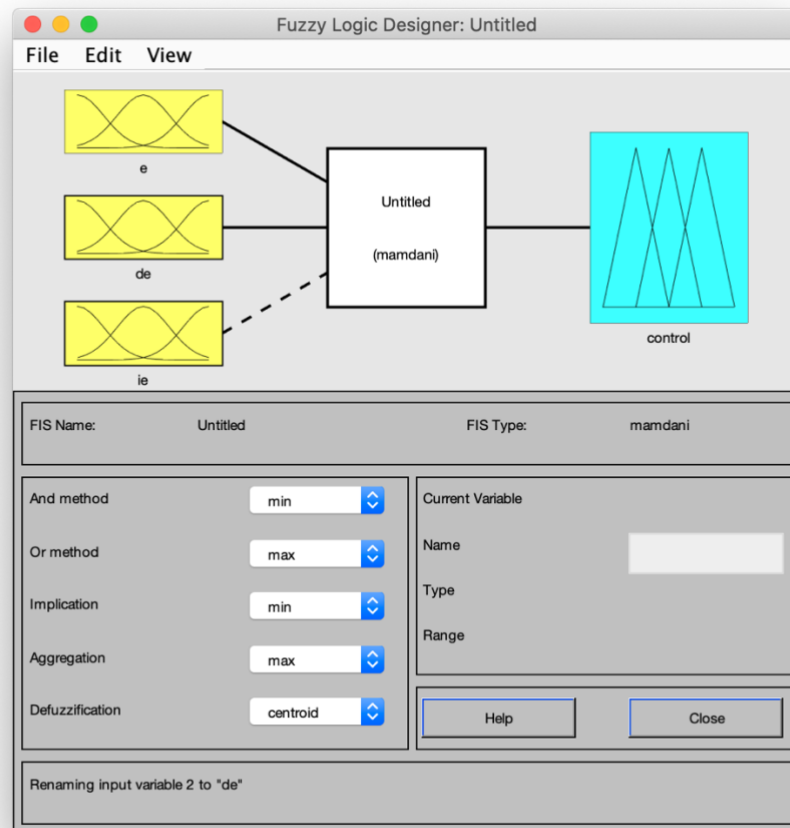
7. Creamos el siguiente modelo en simulink:



8. Dentro del `Fuzzy Logic Controller` en el input de `FIS name:` debemos colocar el nombre de nuestro fichero `.fis` recién generado.

Control PID Borroso.

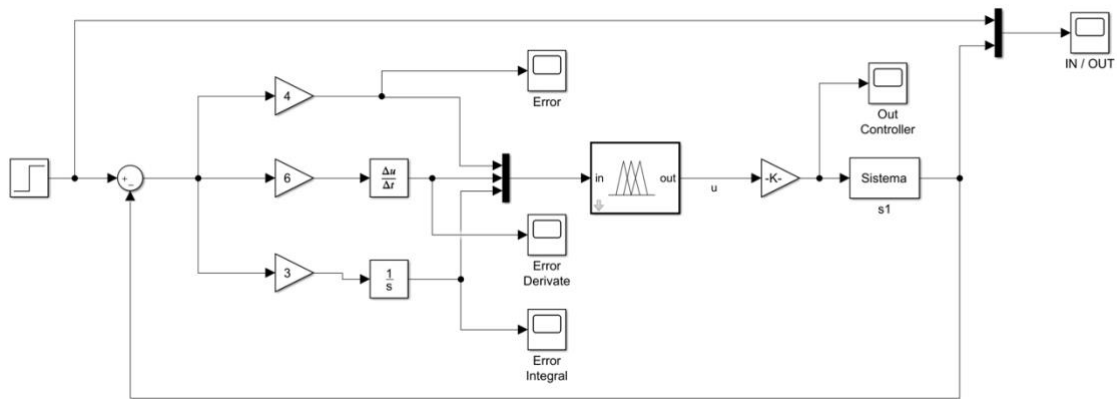
Para realizar este controlador realizamos lo mismo que en el anterior, pero en el editor del controlador fuzzy creamos una tercera variable para la acción integral, tal y como se muestra a continuación:



Las reglas para colocar en el controlador pueden ser:

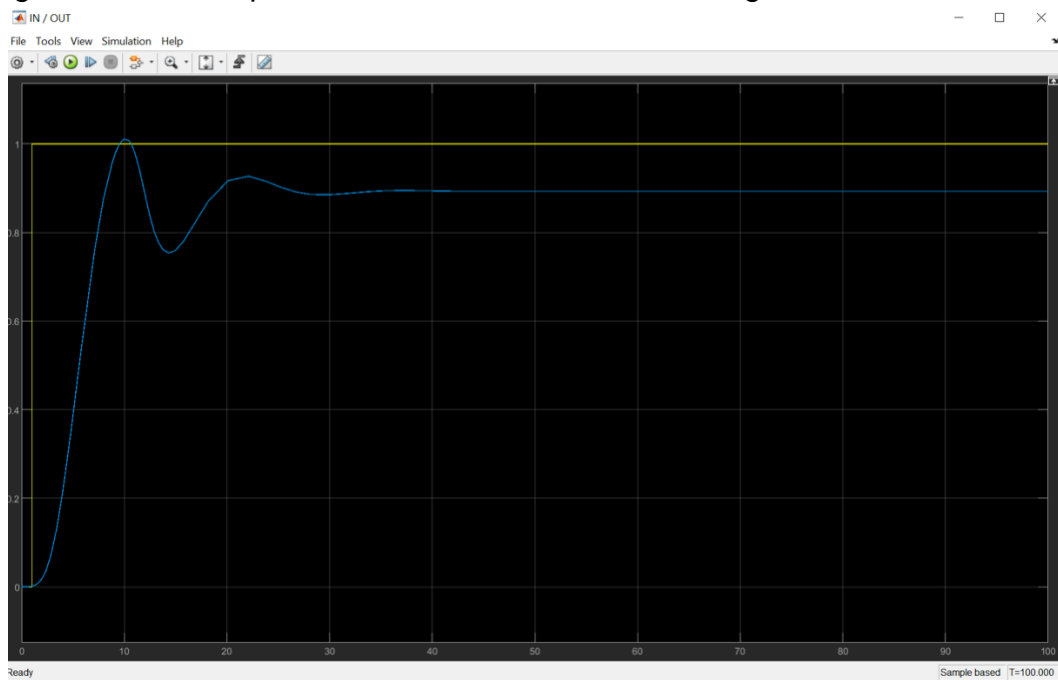
| Error | Integral del Error | Derivada del Error | Control |
|-------|--------------------|--------------------|---------|
| N | N | N | N |
| N | N | P | N |
| N | P | N | N |
| N | P | P | Z |
| P | N | N | Z |
| P | P | N | P |
| P | N | P | P |
| P | P | P | P |

Y el modelo a generar con simulink es el siguiente:

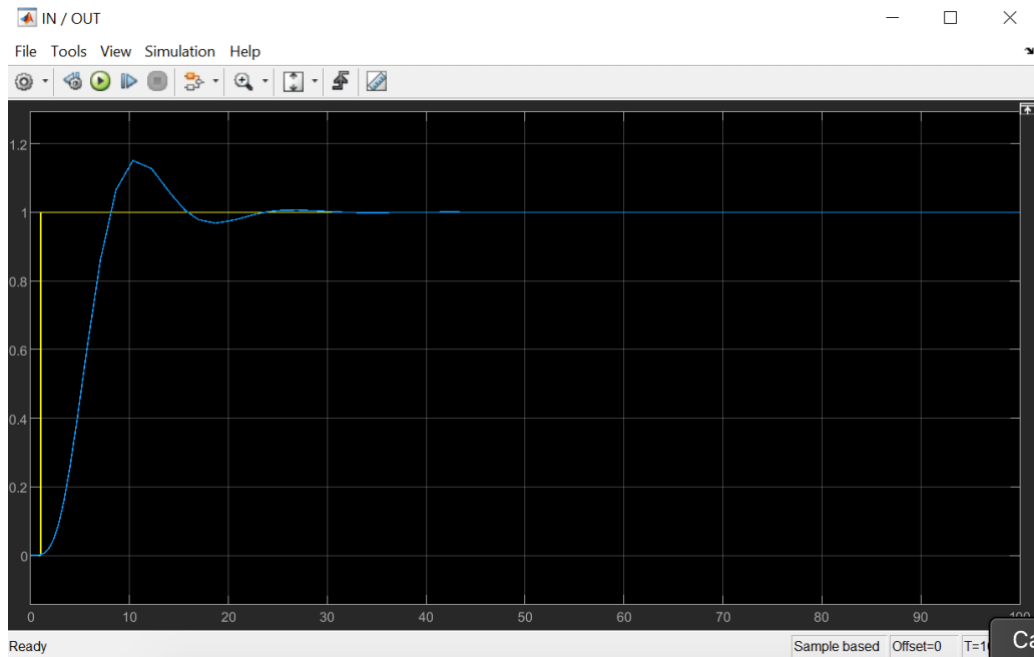


Resultados obtenidos

La gráfica obtenida para el control PD borroso fue la siguiente:



La gráfica obtenida para el control borroso fue la siguiente:



Análisis de resultados

Gracias a las diversas pruebas realizadas para ajustar las ganancias de los controladores podemos concluir que la modificación que tiene mas peso a la hora de ajustar la gráfica obtenida es la ganancia que se encuentra a la salida del controlador borroso, ya que al ajustar las otras producen cambios mínimos en la salida del sistema, debido a esto, se puede observar que esta ganancia es la que mas se ha modificado en relación con las otras.