



UNIVERSITAT  
POLITÈCNICA  
DE VALÈNCIA



Escuela Técnica Superior de Ingeniería del Diseño

# UNIVERSITAT POLITÈCNICA DE VALÈNCIA

Escuela Técnica Superior de Ingeniería del Diseño

---

## DISEÑO Y VALIDACIÓN DEL CONTROL DE UN LEVITADOR MAGNÉTICO EN UN LABORATORIO VIRTUAL MEDIANTE EASY JAVA SIMULATIONS

*TRABAJO FINAL DEL*

**Grado en Ingeniería Electrónica Industrial y Automática**

*REALIZADO POR*

**Pau Mora García**

*TUTORIZADO POR*

**Antonio González Sorribes**

**CURSO ACADÉMICO: 2020/2021**



## **Resumen**

En este proyecto TFG se ha contribuido al desarrollo de métodos de diseño de controladores para el control de altura de una bola magnética en suspensión y posteriormente se ha validado e implementado en un laboratorio virtual mediante Easy Java Simulations. Cabe destacar que la dificultad principal en el control de un levitador magnético es su naturaleza inestable y no lineal, lo que implica una mayor dificultad para su estabilización en lazo cerrado.

## **Resum**

En aquest projecte TFG s'ha contribuït al desenvolupament de mètodes de disseny de controladors per al control d'altura d'una bola magnètica en suspensió i posteriorment s'ha validat i implementat en un laboratori virtual mitjançant Easy Java Simulations. Cal destacar que la dificultat principal en el control d'un levitador magnètic és la seua naturalesa inestable i no lineal, la qual cosa implica una major dificultat per a la seua estabilització en llaç tancat.

## **Abstract**

This TFG project has contributed to the development of controller design methods for the height control of a suspended magnetic ball and has been validated and implemented in a virtual laboratory using Easy Java Simulations. It should be noted that the main difficulty in the control of a magnetic levitator is its unstable and nonlinear nature, which implies a greater difficulty for its stabilization in closed loop.



## **Agradecimientos**

Me gustaría aprovechar este espacio para agradecer a mis padres, José y Amparo, por todos los sacrificios que han hecho por mí, y por animarme desde que soy un niño a que persiga mis sueños y apoyarme mientras cursaba mis estudios superiores. Sin ellos no me habría sido posible realizar la titulación y ser la persona que soy hoy a día de hoy. También me gustaría agradecer a todos los amigos que he hecho durante el grado, gracias a ellos los cuatro cursos han resultado una experiencia inolvidable.

Finalmente, agradecer a mi tutor, Antonio, por dedicar todo el tiempo necesario para realizar este trabajo, guiarme y ayudarme en todo lo que ha sido necesario durante el desarrollo del mismo.



## Índice de figuras

Fig. 1: Esquema eléctrico de un levitador magnético [2] .....	16
Fig. 2: Tren magnético MAGLEV (Japón) [4] .....	17
Fig. 3: Ejemplo de plataformas de levitación magnética.....	17
Fig. 4: Dispositivo de asistencia ventricular [6] .....	17
Fig. 5: Modelo de Simulink para la ecuación de la derivada de I .....	20
Fig. 6: Modelo de Simulink para la ecuación de la aceleración .....	21
Fig. 7: Parámetros del bloque Integrator .....	21
Fig. 8: Comparación de la ecuación no lineal y linealizada .....	24
Fig. 9: Modelo de Simulink para la comparación de los modelos .....	26
Fig. 10: Salida de los dos modelos ante una perturbación de 1 mV .....	27
Fig. 11: Esquema de control en bucle cerrado.....	28
Fig. 12: Señal discretizada mediante ZoH .....	29
Fig. 13: Modelo de Simulink para la implementación de controladores discretos.....	33
Fig. 14: Gráfica de la salida del modelo con el controlador discreto .....	34
Fig. 15: Acción de control del controlador discreto .....	34
Fig. 16: Gráfica de la salida del modelo con el controlador discreto desde $h = 0.31$ m. 35	
Fig. 17: Acción de control del controlador discreto desde $h = 0.31$ m.....	36
Fig. 18: Gráfica de la salida del modelo con el controlador discreto con acción integral .....	37
Fig. 19: Acción de control del controlador discreto con acción integral.....	37
Fig. 20: Gráfica de la salida del modelo con el controlador discreto con acción integral desde $h = 0.31$ m.....	38
Fig. 21: Acción de control del controlador discreto con acción integral desde $h = 0.31$ m .....	38
Fig. 22: Maqueta virtual del levitador magnético programada mediante Easy Java Simulations .....	40
Fig. 23: Interfaz gráfica de la maqueta virtual del levitador .....	40
Fig. 24: Código para la implementación de los controladores discretos .....	43
Fig. 25: Distancia al imán en la maqueta virtual con el controlador discreto .....	44
Fig. 26: Acción de control en la maqueta virtual con el controlador discreto.....	45
Fig. 27: Distancia al imán en la maqueta virtual con el controlador discreto con acción integral .....	46
Fig. 28: Acción de control en la maqueta virtual con el controlador discreto con acción integral .....	47
Fig. 29: Distancia máxima al imán en función de la posición de los pares de polos en el controlador con acción integral .....	48





## Índice de tablas

Tabla 1: Constantes del levitador magnético.....	18
Tabla 2: Bloques de Simulink utilizados.....	20
Tabla 3: Resultados de los experimentos de los controladores.....	39
Tabla 4: Variables de las ecuaciones en diferencias de los controladores.....	43





UNIVERSITAT  
POLITÈCNICA  
DE VALÈNCIA



Escuela Técnica Superior de Ingeniería del Diseño

# MEMORIA

**CURSO ACADÉMICO: 2020/2021**



# Índice de la memoria

<b>1. Objetivos .....</b>	<b>15</b>
<b>2. Introducción .....</b>	<b>16</b>
2.1. Motivación .....	16
2.2. Descripción de un levitador magnético.....	16
2.3. Aplicaciones de un levitador magnético .....	17
<b>3. Modelado del levitador.....</b>	<b>18</b>
3.1. Ecuaciones del levitador .....	18
3.2. Modelo no lineal en <i>Simulink</i> .....	19
3.3. Linealización y modelo en espacio de estados .....	22
3.3.1. Cálculo del punto de funcionamiento .....	22
3.3.2. Obtención de la función de transferencia.....	23
3.3.3. Validación del modelo linealizado.....	26
<b>4. Diseño y validación de estrategias de control .....</b>	<b>28</b>
4.1. Diseño de controladores.....	28
4.2. Validación de controladores .....	33
<b>5. Implementación en una maqueta virtual mediante Easy Java Simulations .....</b>	<b>40</b>
5.1. Introducción a Easy Java Simulations .....	40
5.2. Implementación del control del levitador en la maqueta virtual.....	41
<b>6. Validación de la maqueta virtual y resultados obtenidos .....</b>	<b>44</b>
6.1. Validación de la maqueta virtual .....	44
6.2. Efecto de los polos no dominantes sobre el dominio de atracción .....	48
<b>7. Conclusiones .....</b>	<b>50</b>
<b>8. Bibliografía .....</b>	<b>51</b>
<b>9. Anexo I: Pliego de condiciones .....</b>	<b>53</b>
<b>10. Anexo II: Presupuesto .....</b>	<b>57</b>
<b>11. Anexo III: Planos .....</b>	<b>61</b>
<b>12. Anexo IV: Código de Matlab .....</b>	<b>70</b>



# 1. Objetivos

Los objetivos del presente Trabajo Fin de Grado se pueden desglosar en los siguientes ítems:

- Diseñar una función de transferencia para el modelo linealizado de un levitador magnético que funcione de forma similar al modelo no lineal en torno al punto de funcionamiento. Esta función de transferencia ha de ser validada observando una salida similar a la del modelo no lineal, pues se va a utilizar como base para el diseño de los controladores.
- Diseñar controladores discretos para estabilizar y controlar el levitador magnético. Mediante el uso de métodos algebraicos se van a diseñar controladores que permitan estabilizar y controlar el levitador magnético, estos controladores se van a validar tanto en el modelo no lineal en espacio de estados, como en la maqueta virtual.
- Validar el correcto funcionamiento de la maqueta virtual aportada por el tutor, realizada mediante la herramienta Easy Java Simulations. Esta maqueta virtual ha de simular el proceso y permitir el uso de controladores discretos para el control del levitador magnético, además de mostrar la distancia de la bola al imán y la acción de control aplicada al proceso. Mediante la implementación de los controladores discretos, se podrá determinar la validez de la maqueta para ser utilizada en futuros proyectos.

Estos objetivos tienen como finalidad obtener una maqueta virtual que emule el comportamiento de un levitador magnético real. Además, la maqueta obtenida permitirá validar distintos esquemas de control diseñados para este proceso.

Como objetivos secundarios, se ha planteado analizar las diferencias entre la salida simulada en Matlab/Simulink y la salida obtenida en la maqueta virtual del levitador magnético. Y también analizar el dominio de atracción del levitador en función de la posición de los polos no dominantes en los controladores.

Estos objetivos aportan datos difíciles de analizar de forma matemática, y que resultan de interés a la hora de diseñar los controladores que ofrecen mejores especificaciones.

## 2. Introducción

En este Trabajo Fin de Grado se va a detallar el proceso de diseño del control y validación de una maqueta virtual de un levitador magnético. Este tipo de proceso resulta interesante pues es muy utilizado en la educación debido a que ayuda a entender el funcionamiento de los campos magnéticos y tiene una gran variedad de aplicaciones. Además, se trata de un proceso inestable que supone un reto a la hora de diseñar e implementar un control.

### 2.1. Motivación

La motivación para la decisión de realizar este Trabajo Fin de Grado es la de profundizar en el desarrollo de controladores mediante métodos algebraicos. Junto con la oportunidad de aplicarlos a un proceso inestable como es el levitador magnético [1], y analizar la respuesta de este ante diferentes controladores.

En añadido, se va a validar el correcto funcionamiento de una maqueta virtual de un levitador magnético realizada por el tutor, que podrá ser utilizada en un futuro en los campos de la docencia y la investigación científica.

### 2.2. Descripción de un levitador magnético

Un levitador magnético utiliza el método de levitación magnética para hacer flotar a un objeto compuesto de material ferromagnético sin ningún contacto, mediante el uso de campos magnéticos. Éstos generan una fuerza de mismo valor y sentido contrario a la gravedad en el objeto, consiguiendo así, que el objeto levite en el aire.

Para la creación y manipulación de este campo magnético se utiliza un electroimán, este está compuesto de una bobina de hilo de cobre enrollada entorno a un núcleo ferroso. Al inducir una corriente por la bobina, se genera un campo magnético que afecta a los dominios magnéticos del núcleo. Los dominios magnéticos son como pequeños imanes dentro del material ferromagnético ordenados de forma arbitraria, de forma que se cancelan unos a otros y el campo magnético resultante del objeto es reducido. El campo magnético de la bobina penetra en mayor o menor medida en el material según su intensidad, y afecta a los dominios magnéticos, re direccionándolos en la dirección del campo magnético. De esta forma genera un campo magnético variable con suficiente fuerza como para atraer al objeto que se quiera hacer levitar. A mayor sea la corriente por la bobina, mayor será el campo magnético y mayor atracción generará el electroimán en el objeto.

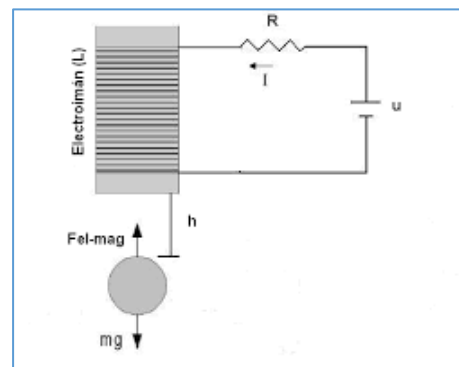


Fig. 1: Esquema eléctrico de un levitador magnético [2]



### 2.3. Aplicaciones de un levitador magnético

Dadas las características nombradas anteriormente, los levitadores magnéticos, también conocidos como Maglev, presentan claras ventajas respecto a otros sistemas. Al no requerir de contacto, no requieren de la aplicación regular de lubricantes y el deterioro del sistema es mínimo. Esto resulta en un coste de mantenimiento muy bajo, que puede ser determinante a la hora de elegir los levitadores magnéticos.

Actualmente tienen una gran variedad de aplicaciones, la más conocida es el tren MAGLEV [3] (En referencia a la tecnología de levitación magnética que utiliza) en Japón (Fig. 2). Este tren utiliza levitación magnética para suspender los vagones del tren en el aire, lo que le permite alcanzar altas velocidades. Siendo la velocidad máxima alcanza superior a los 500 kilómetros por hora.



Fig. 2: Tren magnético MAGLEV (Japón) [4]

Existen otras muchas aplicaciones no tan conocidas, pero con una gran utilidad, ventiladores que se mueven mediante levitación magnética, plataformas de levitación magnética (Fig. 3) que mantienen objetos a flote, focos, etc [5]. Incluso en el campo de la medicina, se utilizan dispositivos de asistencia ventricular mediante levitación magnética (Fig.4) para casos de fallo cardiaco.

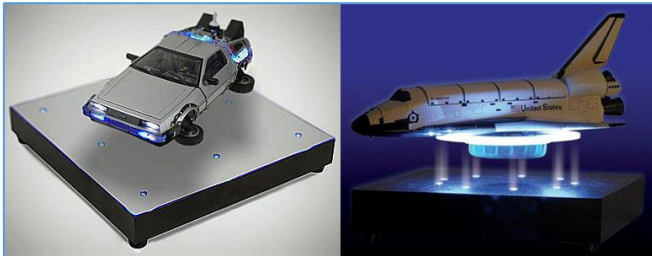


Fig. 3: Ejemplo de plataformas de levitación magnética



Fig. 4: Dispositivo de asistencia ventricular [6]

En añadido, la levitación magnética puede resultar muy importante para grandes avances científicos. Si en el futuro se llega a comprender completamente e implementar la fusión atómica para la generación de energía, las altas temperaturas requeridas para este proceso harían imposible que el material estuviera en contacto con cualquier superficie que lo contenga. Por tanto, la levitación magnética de este material sería clave para poder implementarlo. Como este, existen muchos otros futuros avances como pueden ser los coches voladores y el lanzamiento de cohetes al espacio sin necesidad de combustible fósil.

Como se puede observar, la levitación magnética y los levitadores magnéticos tienen un amplio rango de aplicaciones, y es muy probable que se amplíe aún más en el futuro. Esto sumado al bajo coste de mantenimiento, hace que este sistema resulte de gran interés científico y se quiera ampliar el conocimiento sobre este.

### 3. Modelado del levitador

Para diseñar los controladores que permitirán controlar y estabilizar el levitador, se requiere de un modelo matemático del sistema. Este modelo ha de relacionar la entrada de voltaje que se le proporciona a la bobina con la altura del objeto que se quiere hacer levitar.

El modelo se obtiene de la ecuaciones eléctricas y físicas del funcionamiento del levitador. Una vez se obtenga, se podrá expresar el modelo en espacio de estados y se obtendrá la función de transferencia continua del modelo en el dominio de Laplace.

#### 3.1. Ecuaciones del levitador

Como se ha nombrado anteriormente, para realizar el modelo es necesario conocer las ecuaciones que relacionan la entrada, en este caso el voltaje aplicado a la bobina, con la salida, la altura de la bola que se quiere hacer levitar.

Para obtener las ecuaciones, es necesario conocer las características de los materiales y los componentes que forman el levitador magnético. Puesto que este Trabajo Fin de Grado se realiza mediante un laboratorio virtual, los valores de los componentes del proceso se han obtenido de un levitador magnético real. Este levitador magnético presenta la bobina en la parte superior, por tanto, la fuerza a generar para hacer levitar el objeto ha de ser de atracción, en este caso, el objeto se trata de una pequeña bola metálica. Los valores medidos y que se van a utilizar para este trabajo son los siguientes (Tabla 1):

Variables	Valor	Unidades	Descripción
M	0.004	kg	Masa de la bola
K0	0.7185	-	Constante medida en el laboratorio
K1	0.005023	-	Constante medida en el laboratorio
L	4.8	H	Resistencia de la bobina
R	18	$\Omega$	Inductancia de la bobina

Tabla 1: Constantes del levitador magnético

Las constante  $K_1$  y  $K_2$  definidas en la Tabla 1 han sido medidas experimentalmente en el laboratorio. Se determinan a partir de las características de los materiales que forman el levitador.

Con las constantes ya determinadas, se puede pasar a las ecuaciones. Empezando por la ecuación que contiene la variable de entrada  $u$ , el voltaje de entrada a la bobina, limitado en el rango de  $\pm 10$  V. Esta variable se encuentra en la ecuación para el cálculo de la intensidad en la bobina, que está determinada por el voltaje instantáneo aplicado a la bobina y la derivada de la intensidad en el tiempo:

$$I = \frac{u}{R} - \frac{\dot{I} \cdot L}{R}$$

Con esta ecuación se puede calcular el valor de la intensidad en la bobina. Esta intensidad generada en la bobina es la que genera el campo magnético variable y, por tanto, la fuerza en la bola, que se puede expresar con la siguiente ecuación:

$$f_{em} = K_1 \cdot \left( \frac{K_0 + I}{h} \right)^2$$

Siendo  $h$  la distancia de la bola al imán.

Esta ecuación expresa la fuerza generada por la bobina y el núcleo ferroso sobre la bola, para poder relacionar esta fuerza con la gravedad es necesario expresar esta como una aceleración. Esto se calcula fácilmente puesto que una fuerza se puede expresar como una masa por una aceleración, y se conoce la masa de la bola, por lo que se obtiene una ecuación que calcula la aceleración de la bola:

$$f_{em} = M \cdot a_{em} \rightarrow a_{em} = \frac{f_{em}}{M}$$

$$a = g - a_{em} \rightarrow a = g - \frac{f_{em}}{M}$$

$$a = g - \frac{K_1}{M} \cdot \left( \frac{K_0 + I}{h} \right)^2$$

La última ecuación relaciona la aceleración de la bola con su altura, pues la aceleración de un objeto es la segunda derivada de su posición:

$$a = \ddot{h}$$

Con esta ecuación se han obtenido todas las ecuaciones necesarias para relacionar la entrada de voltaje  $u$ , con la salida  $h$ , que representa la distancia de la bola al imán.

### 3.2. Modelo no lineal en *Simulink*

Una vez se han obtenido todas las ecuaciones necesarias para definir el proceso, se puede crear en *Simulink* un modelo del levitador que funciona de forma aproximada.

Para poder calcular la intensidad, se va a utilizar la ecuación de la intensidad en la bobina según el voltaje de entrada, pero es necesario despejar la derivada de la intensidad. Desde esta se calculará la intensidad mediante la integral, la ecuación despejada es la siguiente:

$$\dot{I} = \frac{u}{L} - \frac{R \cdot I}{L}$$

La variable  $h$  se va a calcular desde la ecuación que calcula la aceleración de la bola, sustituyendo la variable de la aceleración como la segunda derivada de  $h$ :

$$\ddot{h} = g - \frac{K_1}{M} \cdot \left( \frac{K_0 + I}{h} \right)^2$$

La distancia de la bola al imán se podrá obtener realizando la doble integral del resultado calculado mediante esta ecuación.

Una vez se han despejado las derivadas, se puede construir el modelo en *Simulink* utilizando estas ecuaciones. Estos modelos se construyen mediante los bloques que proporciona *Simulink*, que permiten realizar operaciones con diferentes señales, y obtener señales tanto constantes como variables. Los bloques que se van a utilizar son los siguientes (Tabla 2):









Nombre	Apariencia	Funcionamiento
Step		Genera una señal en escalón con valor y tiempo del escalón configurable
Constant		Genera una señal constante del valor especificado
Gain		Multiplca la señal de entrada por el valor especificado
Sum		Suma o resta las señales entrantes, se pueden configurar los signos para realizar la operación deseada
Product		Multiplca las señales de entrada, se puede configurar el número de entradas deseadas
Divide		Realiza la división entre las dos entradas, siendo el numerador la señal en la entrada marcada con X y el denominador la señal en la entrada marcada con ÷
Square		Eleva al cuadrado la señal de entrada
Integrator		Integra la señal de entrada

Tabla 2: Bloques de Simulink utilizados

Se va a comenzar por el diseño del modelo de la ecuación de la intensidad en la bobina (Fig. 5). Se representa la entrada  $u$  mediante un bloque *step*, este bloque simula la entrada de voltaje al sistema. Mediante el bloque *gain* se multiplica la intensidad, obtenida de integrar el resultado de la ecuación, por la constante  $R$ , y utilizando el bloque *constant* se obtiene una señal de valor  $L$ . Esta señal se utiliza para dividir, mediante el bloque *divide*, la diferencia entre  $u$  y la resistencia por la intensidad. Estas señales se restan utilizando el bloque *sum* configurado con un signo positivo para la señal  $u$  y un signo negativo para la multiplicación de  $R$  e  $I$ . A continuación, se utiliza el bloque *integrator* para integrar la señal y se obtiene la variable  $I$ .

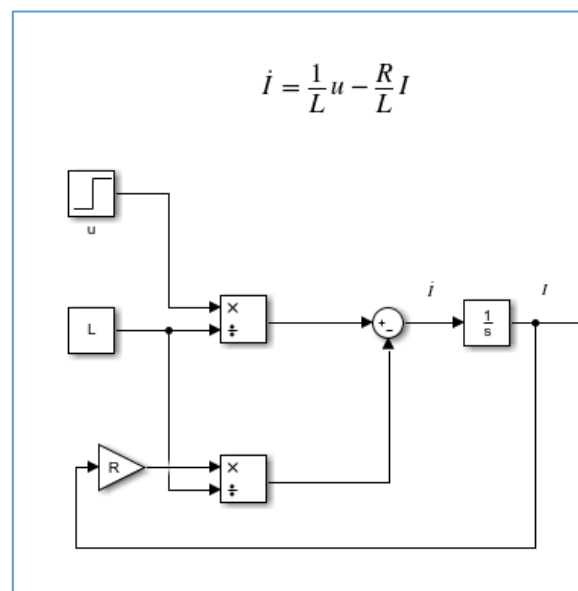


Fig. 5: Modelo de Simulink para la ecuación de la derivada de  $I$

Con el modelo para el cálculo de  $I$  ya creado, se puede pasar a diseñar un modelo para la ecuación restante (Fig. 6). La ecuación requiere de  $I$  como una de sus entradas variables y proviene directamente de la integral del resultado del modelo anterior.

Se comienza el modelo sumando la variable  $I$  obtenida anteriormente a la constante  $K_0$  mediante un bloque *sum* configurado para sumar ambas señales. La señal obtenida se divide mediante un bloque *divide* entre la distancia al imán, calculada al final del modelo. Esta señal ya dividida se introduce a un bloque *square* que eleva al cuadrado la señal, a continuación, se calcula la división entre  $K_1$  y  $M$ , obteniendo la señales de estas mediante un bloque *constant*. La división se realiza mediante un bloque *divide* y posteriormente, se multiplican las dos señales utilizando un bloque *product*. Finalmente se resta la señal obtenida a una señal con el valor de la gravedad mediante un bloque *sum* y se integra dos veces la señal resultante. Obteniendo así, la distancia de la bola al imán, la cual se dirige a un *scope* que permite visualizar-la.

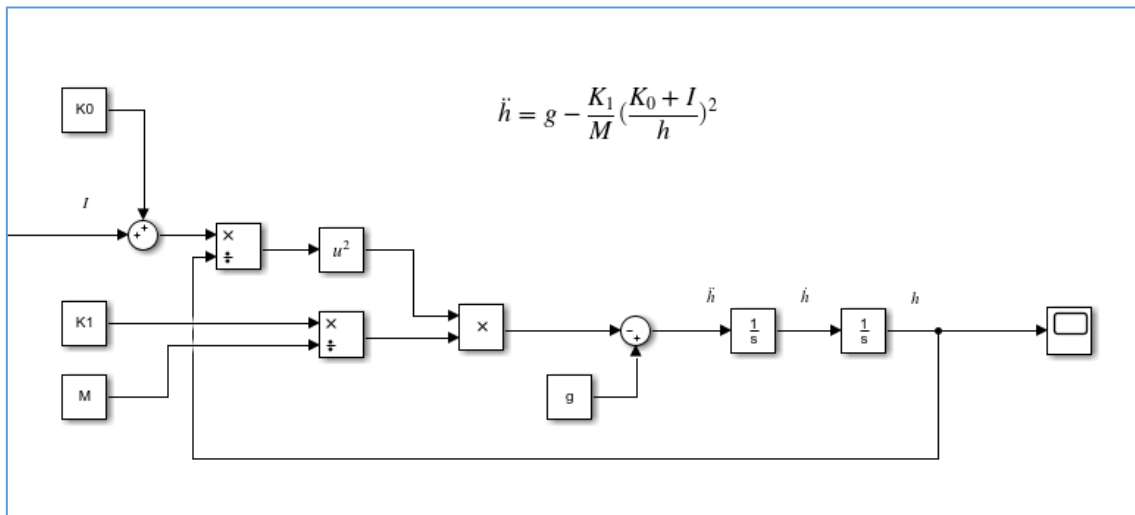


Fig. 6: Modelo de Simulink para la ecuación de la aceleración

Para que el modelo funcione correctamente con los controladores que se van a diseñar, es necesario que la distancia al imán inicial de la bola sea la calculada para el punto de funcionamiento. Esto se consigue modificando el parámetro *Initial Condition* del bloque *Integrator* de la derecha de la figura 6. Para configurar este parámetro, se accede a los parámetros del bloque y se sustituye con el valor deseado, como se puede ver en la figura 7.

Con este modelo diseñado, se puede simular el funcionamiento del levitador magnético. Este se va a utilizar para validar los controladores diseñados, y analizar las salidas obtenidas para determinar el correcto funcionamiento de los controladores previamente a su implementación en la maqueta.

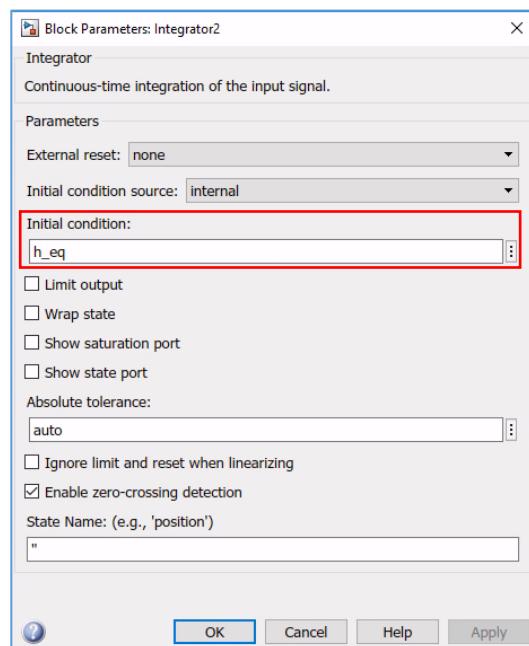


Fig. 7: Parámetros del bloque Integrator

### 3.3. Linealización y modelo en espacio de estados

Para poder diseñar el control del levitador magnético, se ha de obtener un modelo lineal que permita expresar el modelo como una función de transferencia.

Analizando las ecuaciones del levitador magnético, se puede observar que la ecuación de la aceleración de la bola no es lineal. Esto se debe a que presenta división entre variables y estas mismas elevadas al cuadrado, operaciones matemáticas no lineales. Para poder diseñar un control para este proceso, es necesario obtener un modelo linealizado de este. Para ello se emplea la linealización, que consiste en obtener una ecuación lineal que represente una relación similar entre las variables. Esta ecuación lineal se consigue calculando la derivada de la ecuación en un punto de funcionamiento. Sin embargo, el modelo calculado solo será válido cuando las variables tengan un valor cercano al del punto de funcionamiento.

Los puntos de funcionamiento, también llamados puntos de equilibrio, son mínimos locales de energía del sistema. Esto significa que son configuraciones a las que el modelo tiende a estabilizarse, y que la variación temporal de las variables es nula, pues el proceso quedaría estable en este punto, con valores fijos para las variables.

#### 3.3.1. Cálculo del punto de funcionamiento

Como se ha explicado anteriormente, para realizar la linealización de las ecuaciones es necesario linealizarlas en un punto de funcionamiento. Este se ha de calcular con la condición de que las derivadas son igual a cero, puesto que la variación temporal de las variables es nula.

Las ecuaciones del levitador magnético presentan tres variables no nulas en el punto de funcionamiento, la distancia al imán  $h$ , la intensidad por la bobina  $I$  y la tensión aplicada a la bobina  $u$ . Puesto que el sistema está formado por dos ecuaciones, existe un grado de libertad, y este se fija en la distancia al imán, eligiendo como punto de funcionamiento una distancia al imán de 0.3 metros.

Con el grado de libertad fijado, queda un sistema de ecuaciones que se puede resolver y obtener los valores del resto de variables para el punto de funcionamiento. Se despeja la intensidad desde la ecuación de la aceleración y posteriormente el voltaje con la función de la intensidad en la bobina:

$$h_{eq} = 0.3 \text{ m}$$

$$I_{eq} = \sqrt{g \cdot \frac{M}{K_1} \cdot h} - K_0 = 0.12 \text{ A}$$

$$u_{eq} = R \cdot I_{eq} = 18 \cdot 0.12 = 2.16 \text{ V}$$

Estos tres valores de las variables del sistema forman el punto de funcionamiento sobre el que se va a linealizar el modelo y se van a diseñar los controladores.

### 3.3.2. Obtención de la función de transferencia

Con el punto de funcionamiento calculado, el siguiente paso es linealizar las ecuaciones y obtener la función de transferencia. Para realizar la linealización de la ecuación, se sigue un proceso que se ha de realizar para todas las variables presentes. Se ha de calcular la derivada parcial de la ecuación respecto de la variable, y una vez calculada, se han de sustituir los valores en el punto de funcionamiento calculado anteriormente y se obtendrá una constante K.

Se va a calcular primero la constante de la intensidad:

$$K_I = \left. \frac{\partial \ddot{h}(I, h)}{\partial I} \right|_{eq} = - \left( \frac{K_1 \cdot (2I + 2K_0)}{M \cdot h^2} \right) \Big|_{eq} = - \left( \frac{K_1 \cdot (2I_{eq} + 2K_0)}{M \cdot h_{eq}^2} \right) = -23.3989$$

Y seguidamente la constante de la distancia al imán:

$$K_h = \left. \frac{\partial \ddot{h}(I, h)}{\partial h} \right|_{eq} = \frac{2K_1 \cdot (I + K_0)^2}{(M \cdot h^3)} \Big|_{eq} = \frac{2K_1 \cdot (I_{eq} + K_0)^2}{(M \cdot h_{eq}^3)} = 65.4$$

Una vez calculadas las dos constantes, se puede expresar la ecuación linealizada como la suma de las constantes por las variaciones de las variables correspondiente. Puesto que esta ecuación representa la variación respecto del punto de funcionamiento, el resultado obtenido será la variación de la aceleración respecto del punto de funcionamiento:

$$\begin{aligned} \Delta \ddot{h} &= K_I \cdot \Delta I + K_h \cdot \Delta h \\ \Delta \ddot{h} &= -23.3989 \cdot \Delta I + 65.4 \cdot \Delta h \end{aligned}$$

Para comprobar que la linealización de la ecuación se ha realizado de forma correcta, se va a mostrar la función linealizada respecto a la original. El resultado esperado es que las ecuaciones coincidan para el punto de funcionamiento, y que la ecuación linealizada represente la derivada de la ecuación, tangente a esta únicamente en el punto de funcionamiento.

Dado que la ecuación linealizada está expresada con las variaciones de las variables, para graficarla con el mismo resultado que la ecuación no lineal se van a sumar los valores en el punto de funcionamiento a la ecuación. Quedando de la siguiente forma:

$$\ddot{h} = K_I \cdot (\Delta I + I_{eq}) + K_h \cdot (\Delta h + h_{eq})$$

Puesto que las ecuaciones presentan dos variables, se van a graficar en dos partes (Fig. 8). Para la primera gráfica, se va a mantener la variable h en el punto de funcionamiento y se va a variar la variable I. Para la segunda gráfica, se va a realizar al contrario, de esta forma se puede comprobar que la ecuación es correcta para ambas variables.

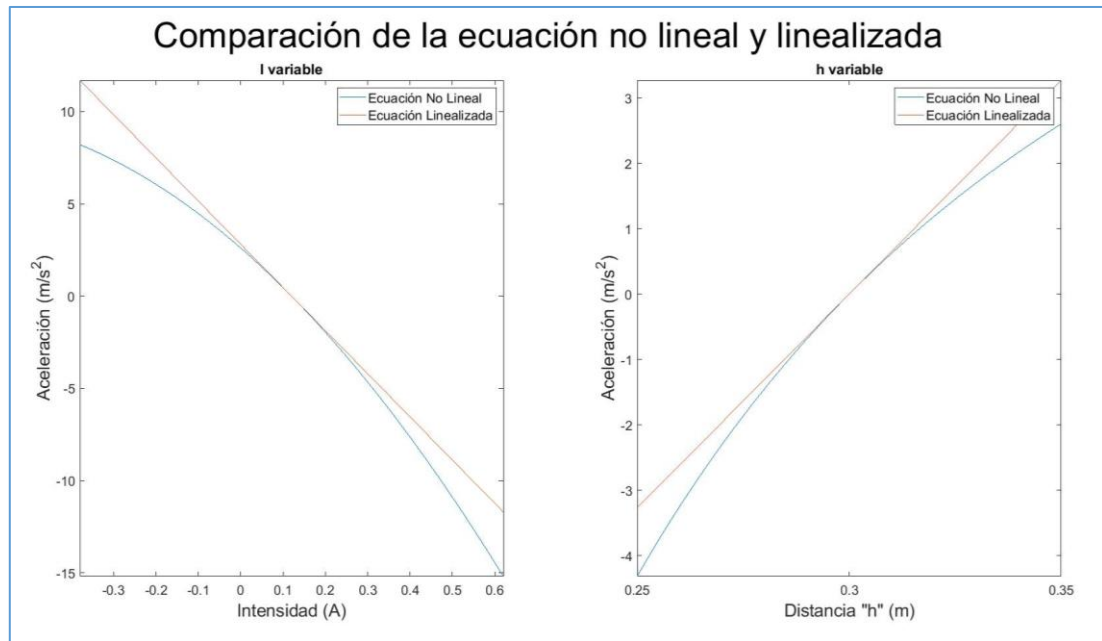


Fig. 8: Comparación de la ecuación no lineal y linealizada

Como se puede observar en la figura 8, el resultado obtenido es el que se esperaba. La ecuación linealizada es tangente a la no linealizada y se comporta de forma similar en las cercanías del punto de funcionamiento (centro de la gráfica). Por tanto, se puede concluir que el proceso de linealización se ha realizado correctamente y las ecuaciones obtenidas se puede utilizar para obtener el modelo linealizado del sistema.

Para obtener este modelo, se va a utilizar la representación en espacio de estados. Esta es un tipo de representación interna que representa la información interna del proceso. Se construye mediante las ecuaciones del modelo y sigue esta estructura:

$$\begin{aligned}\dot{x} &= A \cdot x + B \cdot u \\ y &= C \cdot x + D \cdot u\end{aligned}$$

Donde  $x$  son los estados del sistema,  $\dot{x}$  las derivadas de dichos estados,  $u$  las entradas al sistema y la variable y las salidas del sistema.

Para seleccionar los estados, se han de buscar variables que tengan derivadas en el sistema y se puedan calcular. Puesto que se han despejado las derivadas en las ecuaciones para realizar el modelo no lineal en *Simulink*, no hace falta volver a modificarlas. Atendiendo a las ecuaciones, se seleccionan tres estados,  $h$ ,  $\dot{h}$  e  $I$ , puesto que existe derivada de los tres y se pueden calcular.

Como entradas este modelo presenta el voltaje aplicado a la bobina  $u$ , y la única salida es la distancia al imán  $h$ .

El diseño de la representación en espacio de estados comienza con las variables  $A$  y  $B$ . Estas son matrices que representan las ecuaciones en función de los estados y las entradas respectivamente para el cálculo de las derivadas de los estados.

$$A = \begin{bmatrix} 0 & 1 & 0 \\ K_h & 0 & K_I \\ 0 & 0 & -\frac{R}{L} \end{bmatrix} \quad B = \begin{bmatrix} 0 \\ 0 \\ \frac{1}{L} \end{bmatrix}$$



Con las matrices A y B diseñadas, se pueden sustituir y comprobar que corresponden con las ecuaciones del modelo.

$$\begin{bmatrix} \dot{h} \\ \dot{h} \\ \dot{j} \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 \\ K_h & 0 & K_I \\ 0 & 0 & -\frac{R}{L} \end{bmatrix} \cdot \begin{bmatrix} h \\ \dot{h} \\ I \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ \frac{1}{L} \end{bmatrix} \cdot [u]$$

$$\begin{bmatrix} \dot{h} \\ \dot{h} \\ \dot{j} \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 \\ 65.4 & 0 & -23.3989 \\ 0 & 0 & -3.75 \end{bmatrix} \cdot \begin{bmatrix} h \\ \dot{h} \\ I \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ 0.2083 \end{bmatrix} \cdot [u]$$

La derivada de h coincide con uno de los estados seleccionados, por lo que no tiene ecuación ya que es simplemente igual a otra variable.

Las variables C y D también son matrices, pero en este caso son más sencillas de diseñar puesto que representan las salidas. La variable C representa las salidas respecto a los estados y la D las salidas respecto a las entradas. Puesto que la salida de este proceso es h, la distancia de la bola al imán, la salida es directamente una de las variables de estado seleccionadas.

$$C = [1 \quad 0 \quad 0] \quad D = [0]$$

Sustituidas en el sistema quedaría:

$$[h] = [1 \quad 0 \quad 0] \cdot \begin{bmatrix} h \\ \dot{h} \\ I \end{bmatrix} + [0] \cdot [u]$$

Con todas las matrices diseñadas, se ha concluido el diseño de la representación en espacio de estados de este modelo:

$$\begin{bmatrix} \dot{h} \\ \dot{h} \\ \dot{j} \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 \\ 65.4 & 0 & -23.3989 \\ 0 & 0 & -3.75 \end{bmatrix} \cdot \begin{bmatrix} h \\ \dot{h} \\ I \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ 0.2083 \end{bmatrix} \cdot [u]$$

$$[h] = [1 \quad 0 \quad 0] \cdot \begin{bmatrix} h \\ \dot{h} \\ I \end{bmatrix} + [0] \cdot [u]$$

Finalmente, se va a obtener la función de transferencia, una representación externa del modelo que proporciona la salida del modelo en función de una entrada dada. Esta se calcula mediante las variables A, B, C y D diseñadas anteriormente.

$$G(s) = C \cdot (sI - A)^{-1} \cdot B + D$$

$$G(s) = \frac{-4.8748}{(s - 8.087)(s + 8.087)(s + 3.75)}$$

La función de transferencia obtenida, está expresada en el dominio de Laplace que es una transformada que permite expresar la evolución temporal de un modelo según una entrada dada. Además, la función de transferencia proporciona información dinámica y estática del proceso.

Las raíces del denominador, conocidas como polos, son los valores de  $s$  que hacen cero al denominador, para este proceso, se puede observar que existen tres polos, en 8.087, -8.087 y -3.75. El hecho de que uno de estos polos exista en el plano positivo, indica que el proceso será inestable, esto significa que tiende al infinito en vez de a un valor final si se introduce una mínima perturbación al punto de equilibrio. Por tanto, resulta necesario implementar un control para hacer levitar la bola.

En añadido, se puede observar que la función de transferencia no presenta ningún polo en cero, también conocidos como integradores. Lo que indica que el proceso no alcanzará necesariamente la referencia cuando se implemente un control si no se diseña para que la alcance a la perfección. Este análisis previo resulta interesante para conocer las características del modelo con el que se va a trabajar.

### 3.3.3. Validación del modelo linealizado

Previamente a comenzar el diseño de controladores para el control del levitador magnético, es necesario validar la función de transferencia que se ha obtenido del modelo linealizado. Esto se realiza comprobando la evolución temporal de la función de transferencia y del modelo no lineal diseñado anteriormente en *Simulink*. El modelo es correcto si su salida es muy similar a la del modelo no lineal en las cercanías del punto de funcionamiento.

Para realizar este experimento, se ha diseñado un modelo en *Simulink* que contiene ambos modelos y se pueden visualizar a la vez. Para comprobar que el modelo linealizado funciona correctamente, se va a aplicar un voltaje equivalente a la entrada de ambos modelos.

Como se ha explicado anteriormente, el modelo linealizado funciona con las variaciones respecto del punto de funcionamiento de las variables. Por tanto, el voltaje que se le ha de introducir ha de ser únicamente esa variación, en este caso, se va a aplicar 1 mV por encima del valor en el equilibrio. Esto se debe a que el sistema es inestable y evoluciona de forma muy rápida ante perturbaciones. Para poder comparar las salidas, se le ha de sumar la distancia al imán en el punto de funcionamiento a la salida del modelo linealizado, puesto que este genera una salida correspondiente a la variación de la distancia.

El modelo de *Simulink* (Fig. 9) se ha creado utilizando los bloques *step* para generar las señales de entrada. El modelo no lineal se ha simulado con el modelo creado anteriormente y el modelo linealizado se ha simulado mediante la función de transferencia implementada en un bloque *LTI System*.

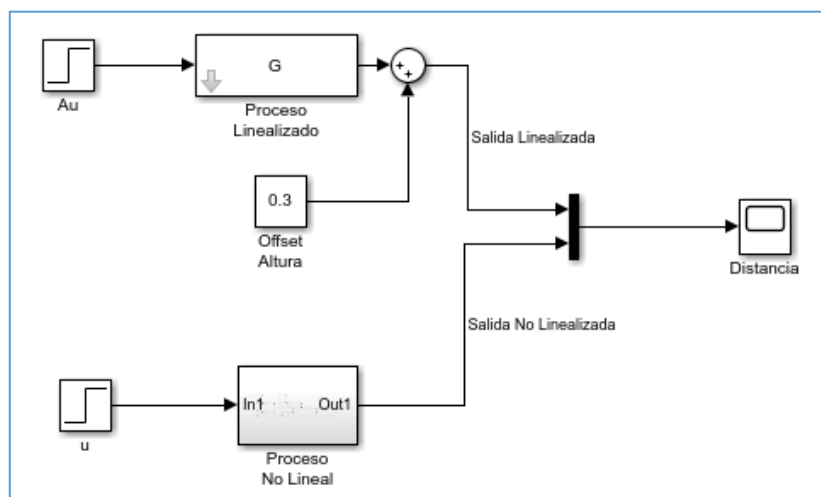


Fig. 9: Modelo de Simulink para la comparación de los modelos

La señal nombrada  $\Delta u$  tiene un valor de 1 mV y la señal nombrada  $u$  tiene un valor de 2.161 V, es decir, el voltaje en el punto de funcionamiento añadiéndole la perturbación de 1 mV.

Con el modelo creado, se puede realizar la simulación y analizar los resultados obtenidos (Fig.10):

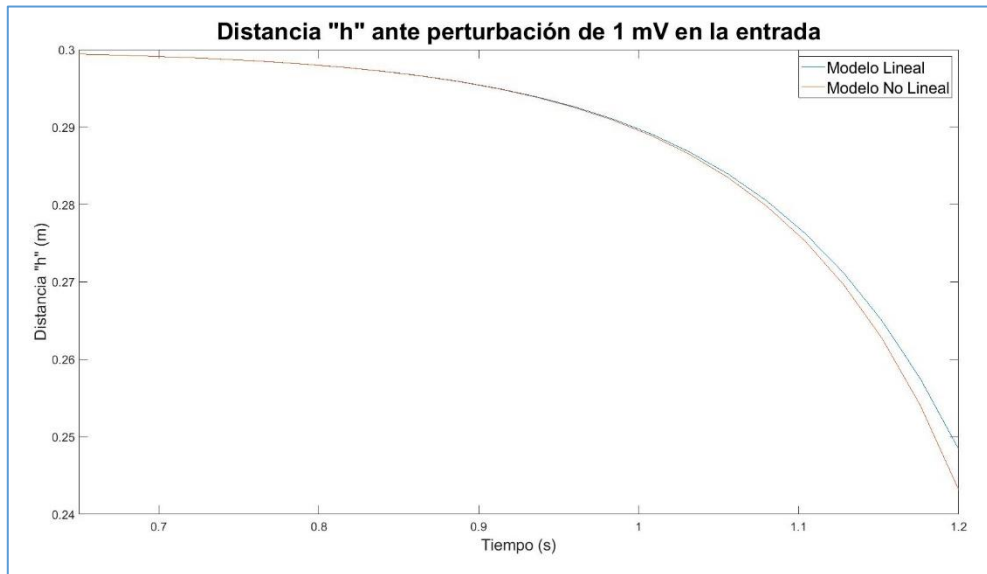


Fig. 10: Salida de los dos modelos ante una perturbación de 1 mV

Como se puede observar en la figura 10, los modelos se comportan de forma idéntica mientras la distancia al imán es cercana a 0.3 m. Una vez comienza a alejarse de este punto, se puede observar como los modelos presentan resultados ligeramente diferentes.

Dados los resultados obtenidos y analizados, se puede concluir que la función de transferencia obtenida mediante el modelo linealizado es correcta. Por tanto, se puede utilizar como base para diseñar los controladores, por lo que se ha cumplido el primero objetivo marcado para este Trabajo Fin de Grado.

## 4. Diseño y validación de estrategias de control

El siguiente objetivo a completar, es el diseño y validación de las estrategias de control. Para realizar el diseño de los controladores se ha optado por el control discreto en vez del control continuo mediante PID. Esta decisión se ha tomado por las características del control discreto, este tipo de control permite generar la función de transferencia exacta deseada, en cambio, los controladores discretos tipo PID generan funciones de transferencia aproximadas. Esto se debe a que las estructuras de los controladores continuos están muy limitadas. Puesto que realizar un control continuo complejo genera muchas complicaciones a la hora de construir físicamente el controlador. Sin embargo, el control discreto se puede implementar en un microcontrolador sin demasiadas complicaciones. Estas características hacen que el control discreto sea el idóneo para esta aplicación.

El control discreto se caracteriza por tomar datos dado un intervalo de tiempo fijado, llamado tiempo de muestreo, para este proyecto se va a utilizar un tiempo de muestreo de 50 ms. Se ha seleccionado este tiempo puesto que es el menor que permite la maqueta virtual.

La implementación de las estrategias de control se hará mediante un control en bucle cerrado (Fig.11). Esto significa que la salida del sistema se realimenta a la entrada, calculando la diferencia entre la referencia de distancia al imán a alcanzar y la distancia actual. El esquema es el siguiente:

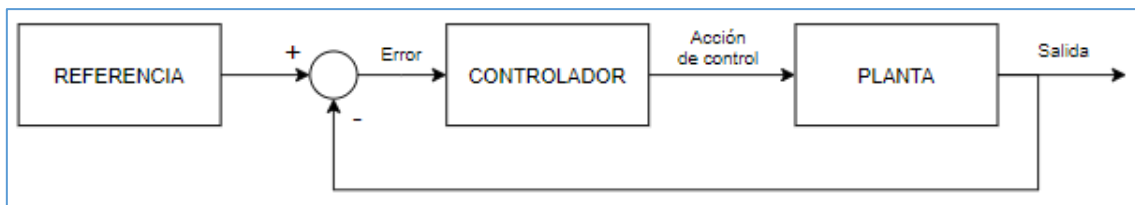


Fig. 11: Esquema de control en bucle cerrado

### 4.1. Diseño de controladores

Existen varios métodos de diseño de controladores, en este Trabajo Fin de Grado se va a utilizar el método de síntesis directa. Este método permite diseñar directamente la función de transferencia que se desea para el proceso y calcular el controlador a partir de esta, además, permite compensar las oscilaciones del sistema.

Atendiendo al esquema de control presentado en la figura 10, siendo GR el controlador discreto y G(z) la función de transferencia de la planta, la función de transferencia del proceso completo en bucle cerrado sería la siguiente:

$$G_{BC} = \frac{GR \cdot Gz}{1 + GR \cdot Gz}$$

Para realizar el método de síntesis directa, el diseño de esta función de transferencia es el primer paso, se diseña la función de transferencia del proceso en bucle cerrado para tener la dinámica deseada. Una vez se ha diseñado, se puede calcular el controlador despejando de la ecuación anterior:

$$G_{BC} = \frac{GR \cdot Gz}{1 + GR \cdot Gz} \rightarrow G_{BC} + G_{BC} \cdot GR \cdot Gz = GR \cdot Gz \rightarrow G_{BC} = GR \cdot Gz \cdot (1 - G_{BC})$$

$$GR = \left(\frac{1}{G_z}\right) \cdot \left(\frac{G_{BC}}{1 - G_{BC}}\right)$$

Con el método explicado y aclarado, se va a proceder al diseño de los controladores. Para controlar el levitador magnético, se han planteado dos controladores discretos, ambos diseñados para minimizar las oscilaciones del sistema y el tiempo de establecimiento. Este es el tiempo mínimo que tarda el sistema en alcanzar un 98% de la referencia y estabilizarse.

El primer controlador únicamente incluirá las especificaciones dinámicas de la oscilación y el tiempo de establecimiento. El segundo controlador atenderá también la especificación estática de error de posición cero, añadiéndole acción integral. Esto significa que este controlador incluirá un integrador y que la salida final del proceso controlado siempre será igual a la referencia. Además, podrá rechazar perturbaciones constantes siempre que no salga del dominio de atracción.

Como paso previo al diseño de los controladores, se ha de obtener la función de transferencia de la planta en tiempo discreto. La discretización se ha realizado mediante el método del Retenedor de Orden Zero (Zero Order Hold o ZoH). Este método genera una discretización exacta del proceso para entradas en tipo escalón.

El método de discretización mediante ZoH proporciona una función de transferencia discreta equivalente a mantener el valor de entrada medido cada tiempo de muestreo (50 ms). Obteniendo una señal como la que se puede ver en la figura 12. De esta forma, no se introduce ningún retardo temporal a la función de transferencia.

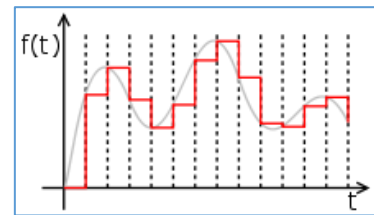


Fig. 12: Señal discretizada mediante ZoH

Utilizando el comando *c2d* se calcula la función de transferencia discretizada del levitador magnético:

$$G_{(z)} = \frac{-9.7778 \cdot 10^{-5} \cdot (z + 3.597) \cdot (z + 0.2531)}{(z - 1.498) \cdot (z - 0.829) \cdot (z - 0.6674)}$$

A diferencia de la función de transferencia obtenida en tiempo continuo, esta función de transferencia también presenta ceros, éstos son las raíces del numerador, es decir, los valores de *z* que hacen cero el numerador. Se puede observar que todos estos ceros se encuentran en el plano negativo, lo que significa que generarán oscilaciones en la salida y hay que tenerlos en cuenta a la hora de diseñar los controladores.

Respecto a los polos, en dominio discreto el criterio que determina la estabilidad de un sistema es que todos los polos tengan un módulo menor que uno. Como se puede observar en la función de transferencia, se obtienen tres polos localizados en 1.498, 0.829 y 0.6674. El primer polo con módulo mayor que uno es el responsable de que el proceso sea inestable y también se ha tener en cuenta a la hora de diseñar los controladores.

Con la función de transferencia discretizada y analizada, se va a comenzar el diseño del primer controlador. Como se ha explicado anteriormente, el primer paso para el diseño de controladores mediante síntesis directa es diseñar la función de transferencia en bucle cerrado deseada. El diseño de esta está sujeto a ciertos criterios, la relación entre polos y ceros ha de ser la misma que la de la función de transferencia del proceso, por tanto, obligatoriamente la función de transferencia tendrá que presentar un polo más que ceros. Además, la función de transferencia ha de mantener los ceros negativos del proceso, porque lo que se tendrán que mantener los dos ceros originales, y, por tanto, tendrá que presentar 3 polos.

El último criterio va en referencia a los polos inestables del proceso, éstos han de estar presentes como polos de la función  $1 - G_{BC}$ . Con todos estos criterios atendidos se puede diseñar la función de transferencia en bucle cerrado:

$$G_{BC} = \frac{K \cdot (z + 3.597) \cdot (z + 0.2531)}{(z - 0.85)^2 \cdot z}$$

Se han mantenido los ceros del proceso original, y se han diseñado los polos de forma que el proceso sea estable, con la máxima velocidad posible y la mínima sobreoscilación. Se puede observar que existe una variable no resuelta, esta se va a resolver con las ecuaciones obtenidas resolviendo el criterio de los polos inestables.

$$1 - G_{BC} = \frac{(z - 1.498) \cdot (z - p_1) \cdot (z - p_2)}{(z - 0.85)^2 \cdot z}$$

Para poder igualar ambas partes, han de tener el mismo orden de  $z$  en el numerador, por tanto, existen otras dos variables que no se conocen. Esto genera un sistema con 3 variables y 3 ecuaciones que se obtendrán del término independiente y de los coeficientes de  $z$  al desarrollar la ecuación.

$$1 - \frac{K \cdot (z + 3.597) \cdot (z + 0.2531)}{(z - 0.85)^2 \cdot z} = \frac{(z - 1.498) \cdot (z - p_1) \cdot (z - p_2)}{(z - 0.85)^2 \cdot z}$$

$$\frac{(z - 0.85)^2 \cdot z - K \cdot (z + 3.597) \cdot (z + 0.2531)}{(z - 0.85)^2 \cdot z} = \frac{(z - 1.498) \cdot (z - p_1) \cdot (z - p_2)}{(z - 0.85)^2 \cdot z}$$

En este momento de la ecuación se pueden igualar los numeradores:

$$(z - 0.85)^2 \cdot z - K \cdot (z + 3.597) \cdot (z + 0.2531) = (z - 1.498) \cdot (z - p_1) \cdot (z - p_2)$$

$$p_1 z^2 + p_2 z^2 - z^2 K - 0.202 z^2 - 1.498 p_1 z - p_1 p_2 z - 1.498 p_2 z - 3.8501 z K + 0.7225 z + 1.498 p_1 p_2 - 0.9104007 K = 0$$

Desde esta ecuación, se pueden obtener 3 ecuaciones para completar el sistema, dos ecuaciones agrupando los coeficientes de las  $z$ , y la última del término independiente. El sistema de ecuaciones resultante es el siguiente:

$$\begin{cases} p_1 + p_2 - K - 0.202 \\ -1.498 \cdot p_1 - p_1 \cdot p_2 - 1.498 \cdot p_2 - 3.8501 \cdot K + 0.7225 \\ 1.498 \cdot p_1 \cdot p_2 - 0.9104007 \cdot K \end{cases}$$

Resolviendo el sistema se obtienen los siguientes valores:

$$K = 0.0705$$

$$p_1 = 0.1363 + 0.1558i$$

$$p_2 = 0.1363 - 0.1558i$$

Aunque se obtienen resultados imaginarios para los polos, esto no presenta ningún problema, puesto que la transformada  $z$  utilizada acepta estos resultados. Con la  $K$  calculada, se puede finalizar el diseño de función de transferencia en bucle cerrado y posteriormente el diseño del controlador.

$$G_{BC} = \frac{0.0705 \cdot (z + 3.597) \cdot (z + 0.2531)}{(z - 0.85)^2 \cdot z}$$

Para el cálculo del controlador, se utiliza la ecuación despejada al principio de esta apartado:

$$GR_1 = \left(\frac{1}{G_z}\right) \cdot \left(\frac{G_{BC}}{1 - G_{BC}}\right)$$

$$GR_1 = \frac{-721 \cdot z^2 + 1079 \cdot z - 399}{z^2 - 0.2725 \cdot z + 0.04285}$$

Tras este cálculo, se ha obtenido la función de transferencia del controlador de forma correcta para el control del levitador magnético. En el siguiente apartado se implementará en un modelo de *Simulink* para comprobar su funcionamiento y validarlo.

A continuación, se va a diseñar el controlador con acción integral. Este controlador incluye la especificación de error de posición cero, lo que significa que se ha añadido una condición más, que se traducirá en el diseño como una ecuación más. Para poder resolver este controlador, se ha de aumentar el orden del mismo, por lo que la función de transferencia en bucle cerrado a diseñar presentará tres ceros y cuatro polos.

Conociendo la forma que ha de tener la función de transferencia, se puede comenzar el diseño de la misma siguiendo los mismos pasos realizados anteriormente.

$$G_{BC} = \frac{K \cdot (z + 3.597) \cdot (z + 0.2531) \cdot (z - z_1)}{(z - 0.925)^2 \cdot (z - 0.5)^2}$$

En este caso, la necesidad de aumentar el orden de la función de transferencia genera una nueva incógnita, esta representa la posición del nuevo cero. Al igual que en el controlador anterior, los polos se han diseñado para minimizar el tiempo de establecimiento y la sobreoscilación.

Para el siguiente paso, uno de los ceros de la función  $1 - G_{BC}$  será el polo inestable, pero dada la condición de error de posición cero, otro de los ceros tendrá que ser un integrador. En tiempo discreto, los integradores son polos en la posición uno, por lo que se ha de diseñar el controlador para que contenga estos dos valores.

$$1 - G_{BC} = \frac{(z - 1.498) \cdot (z - 1) \cdot (z - p_1) \cdot (z - p_2)}{(z - 0.925)^2 \cdot (z - 0.5)^2}$$

Junto con las dos variables, K y la posición del cero, de la función de transferencia en bucle cerrado diseñada, las dos variables que presenta esta ecuación completan el sistema de cuatro ecuaciones y cuatro variables para resolver el controlador.

$$1 - \frac{K \cdot (z + 3.597) \cdot (z + 0.2531) \cdot (z - z_1)}{(z - 0.925)^2 \cdot (z - 0.5)^2} = \frac{(z - 1.498) \cdot (z - 1) \cdot (z - p_1) \cdot (z - p_2)}{(z - 0.925)^2 \cdot (z - 0.5)^2}$$

$$\frac{(z - 0.925)^2 \cdot (z - 0.5)^2 - K \cdot (z + 3.597) \cdot (z + 0.2531) \cdot (z - z_1)}{(z - 0.925)^2 \cdot (z - 0.5)^2} = \frac{(z - 1.498) \cdot (z - 1) \cdot (z - p_1) \cdot (z - p_2)}{(z - 0.925)^2 \cdot (z - 0.5)^2}$$

Eliminando los numeradores y desarrollando las ecuaciones quedaría:

$$p_1 z^3 + p_2 z^3 - z^3 K - 0.352 z^3 - 2.498 p_1 z^2 - p_1 p_2 z^2 - 2.498 p_2 z^2 - 3.8501 z^2 K + z^2 K z_1$$

$$+ 1.457625 z^2 + 1.498 p_1 z + 2.498 p_1 p_2 z + 1.498 p_2 z - 0.9104007 z K + 3.8501 z K z_1$$

$$- 1.318125 z - 1.498 p_1 p_2 + 0.9104007 K z_1 + 0.21390625$$

De esta ecuación, se pueden obtener las cuatro necesarias para resolver el controlador, tres obtenidas de los coeficientes de  $z$ , y la última obtenida del término independiente. El sistema de ecuaciones resultante sería el siguiente:

$$\begin{cases} p_1 + p_2 - K - 0.352 \\ -2.498 \cdot p_1 - p_1 \cdot p_2 - 2.498 \cdot p_2 - 3.8501 \cdot K + K \cdot z_1 + 1.457625 \\ 1.498 \cdot p_1 + 2.498 \cdot p_1 \cdot p_2 + 1.498 \cdot p_2 - 0.9104007 \cdot K + 3.8501 \cdot K \cdot z_1 - 1.318125 \\ -1.498 \cdot p_1 \cdot p_2 + 0.9104007 \cdot K \cdot z_1 + 0.21390625 \end{cases}$$

Resolviendo el sistema se obtienen los siguientes valores:

$$\begin{aligned} K &= 0.0731 \\ z_1 &= 0.9967 \\ p_1 &= 0.2126 + 0.3767i \\ p_2 &= 0.2126 - 0.3767i \end{aligned}$$

Con todas las variables resueltas, se puede completar el diseño de la función de transferencia en bucle cerrado.

$$G_{BC} = \frac{0.0731 \cdot (z + 3.597) \cdot (z + 0.2531) \cdot (z - 0.9967)}{(z - 0.925)^2 \cdot (z - 0.5)^2}$$

Y se calcula finalmente el controlador:

$$GR_2 = \left(\frac{1}{G_z}\right) \cdot \left(\frac{G_{BC}}{1 - G_{BC}}\right)$$

$$GR_2 = \frac{-747.7 \cdot z^3 + 1864 \cdot z^2 - 1529 \cdot z + 412.3}{z^3 - 1.42 \cdot z^2 + 0.6122 \cdot z - 0.1871}$$

Tras este último cálculo, se ha obtenido el controlador que permite controlar el levitador magnético, mejorando sus características dinámicas y con error de posición nulo.

Seguidamente, en el siguiente apartado, se van a implementar ambos controladores diseñados en un modelo en *Simulink* en los dos modelos diseñados. Esto va a permitir analizar el funcionamiento de los controladores y determinar si son válidos para el control del levitador magnético.



## 4.2. Validación de controladores

Para realizar la validación de los controladores, se van a implementar en ambos modelos del levitador magnético obtenidos, tanto en el lineal como en el no lineal. De esta forma se podrá observar el funcionamiento de los controladores y analizar las características de la salida obtenida por cada uno de ellos.

La implementación se ha realizado mediante un modelo en *Simulink* (Fig.13) siguiendo el esquema de control presentado en este apartado (Fig. 11). Los controladores se van a implementar utilizando un bloque *Discrete Transfer Fcn* utilizando las funciones de transferencia diseñadas. Para simular la saturación de el voltaje de entrada, se va a utilizar el bloque *Saturation*, este limita la acción de control, que en este caso es el voltaje, al rango de  $\pm 10$  V. Para hacer funcionar correctamente el modelo no lineal con el controlador linealizado, se ha de sumar el voltaje en el punto de funcionamiento a la acción de control diseñada. Esto se debe a que el controlador únicamente proporciona la variación de voltaje respecto del funcionamiento. De igual manera, para poder comparar las salidas, se le va a añadir la distancia al imán en el equilibrio al modelo linealizado.

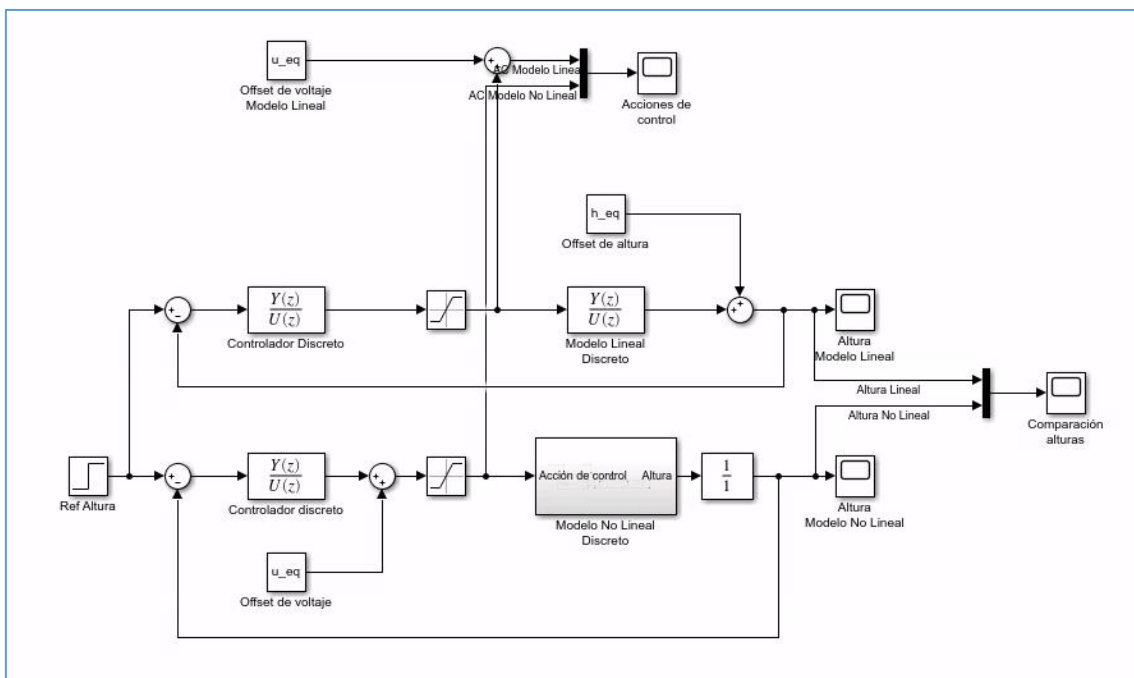


Fig. 13: Modelo de Simulink para la implementación de controladores discretos

Para simular el comportamiento de estos controladores en ambos modelos, se va a utilizar una referencia de 0.301 m, implementada en el bloque *Step*. De esta forma se podrá utilizar el modelo linealizado, puesto que requiere de una variación respecto del punto de funcionamiento. Además, sirve para simular posible ruido en la señal de referencia, que permite comprobar si el sistema alcanzaría la referencia satisfactoriamente.

Comenzando por el primer controlador, la simulación proporciona la siguiente salida (Fig.14):

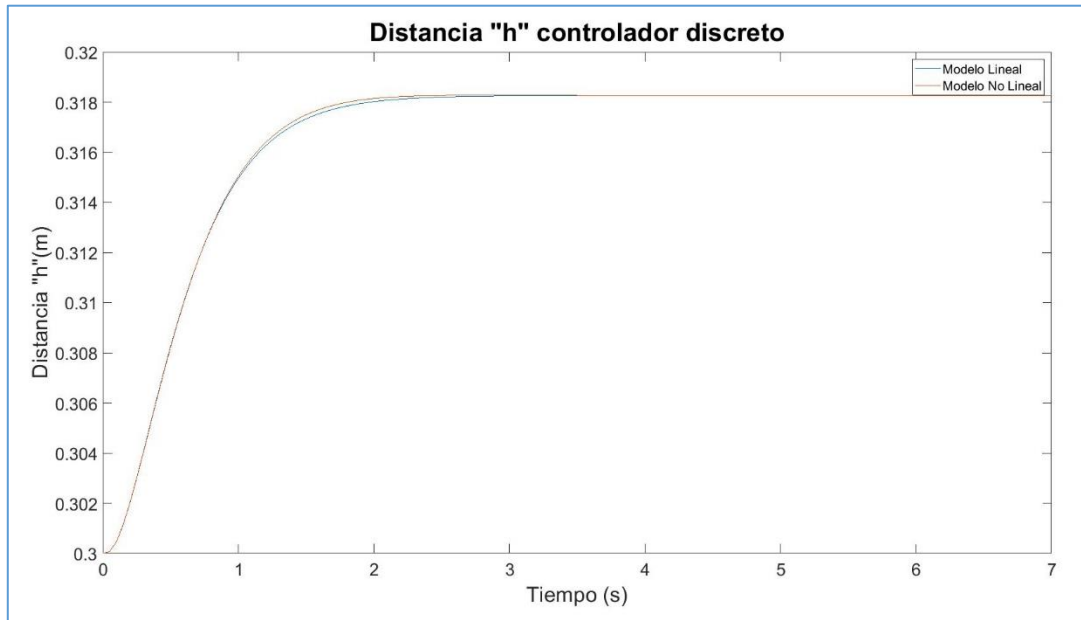


Fig. 14: Gráfica de la salida del modelo con el controlador discreto

Como se puede observar en la figura 14, el controlador consigue estabilizar el sistema partiendo del equilibrio y no presenta grandes diferencias entre los dos modelos. Atendiendo a las características dinámicas, presenta sobreoscilación nula y un tiempo de establecimiento de en torno a 1.6s, unos resultados muy buenos que demuestran la eficacia del controlador. Sin embargo, el valor al que se estabiliza la salida no es el de la referencia, pues esta tiene un valor de 0.301, se observa un error de posición de 0.018 m.

Para poder determinar el correcto funcionamiento del controlador, se ha de analizar también la señal de control. Puesto que, si se encontrara saturada durante muchos instantes, podría presentar un problema a la hora de la implementación en la maqueta virtual o en un levitador magnético real. La acción de control de este controlador es la siguiente (Fig.15):

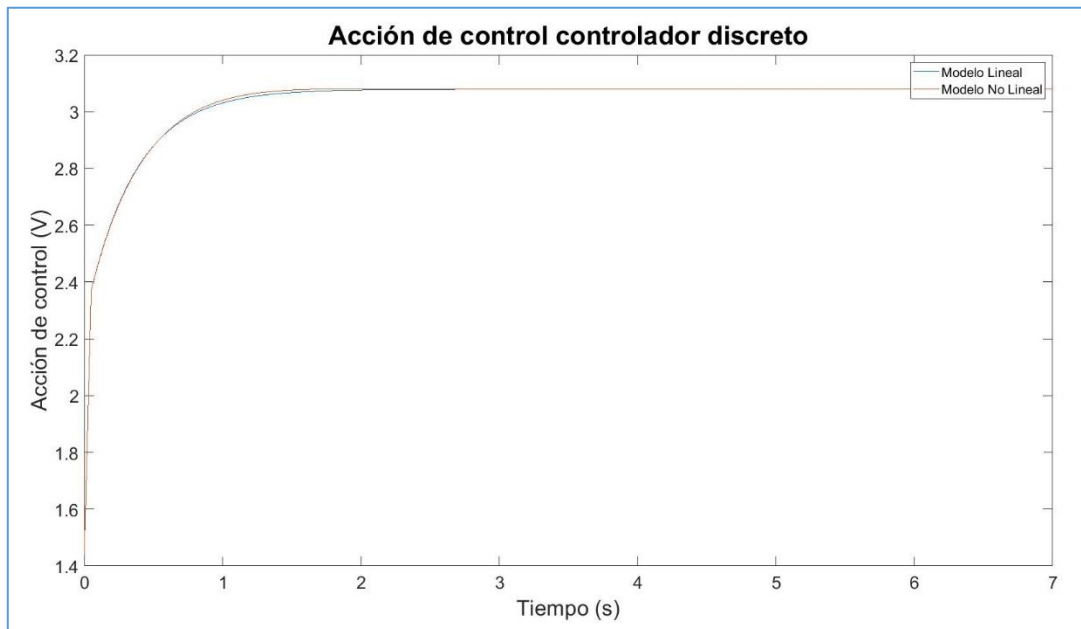


Fig. 15: Acción de control del controlador discreto

Según la gráfica obtenida, la acción de control no satura en ningún momento. Se puede observar que se estabiliza en un valor significativamente superior al del punto de equilibrio calculado. Esto se debe al error de posición que presenta el sistema, y no a un funcionamiento incorrecto del controlador.

Con los experimentos realizados, se puede concluir que el controlador funciona correctamente y consigue estabilizar el sistema cuando parte del punto de funcionamiento. Pero partir desde el punto de funcionamiento exacto en un levitador magnético real es poco probable. Para determinar si el controlador tiene un funcionamiento satisfactorio y es válido para implementarlo en la maqueta, se va a realizar una simulación del sistema partiendo de una distancia diferente a la del punto de funcionamiento. Esto mostrará si el controlador es capaz de estabilizar el sistema, aunque la bola no se coloque en el lugar exacto, y, por tanto, resulte un controlador más robusto y con una mayor utilidad.

Esta prueba solo se puede realizar en el modelo no lineal, puesto que permite configurar la distancia inicial de la bola con el parámetro *Initial Condition* del bloque *Integrator* que calcula esta variable. La salida y la acción del control resultantes partiendo de una distancia al imán inicial de 0.31 m es la siguiente (Fig.16 y Fig.17):

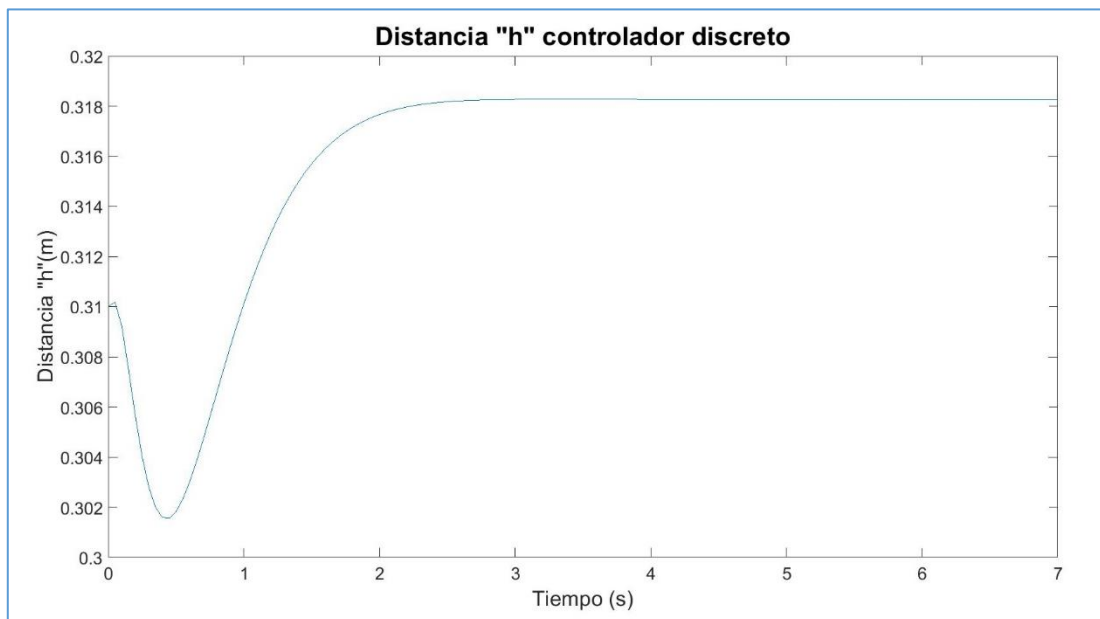
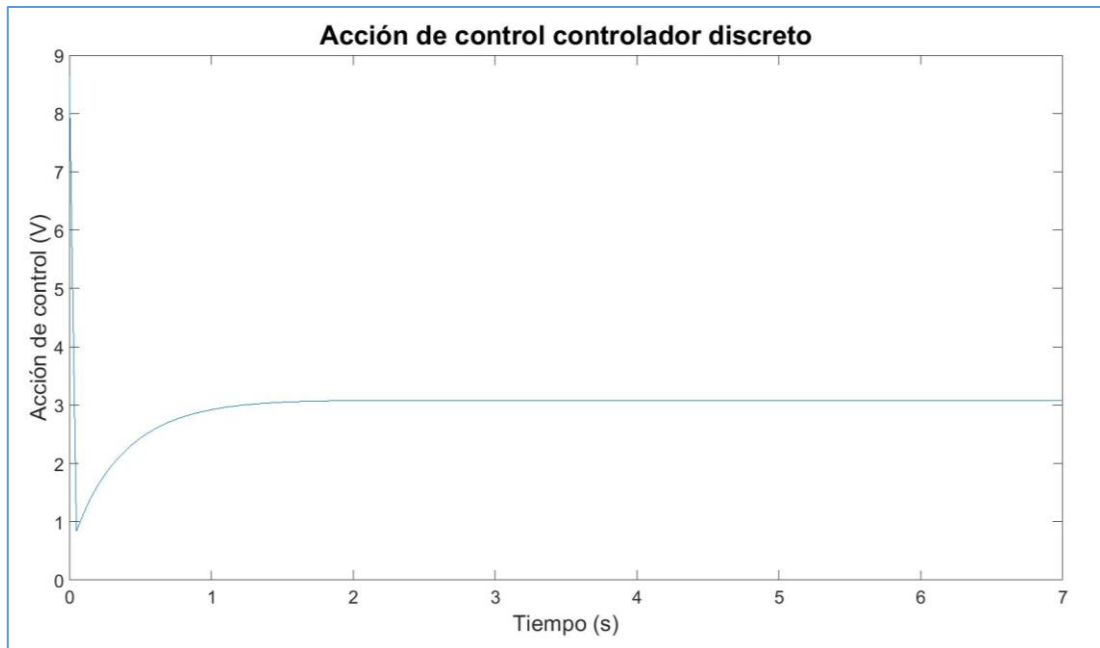


Fig. 16: Gráfica de la salida del modelo con el controlador discreto desde  $h = 0.31$  m



*Fig. 17: Acción de control del controlador discreto desde  $h = 0.31\text{ m}$*

En las gráficas anteriores se puede observar como el controlador consigue estabilizar el sistema, aunque la bola parte de un punto diferente al funcionamiento. La distancia al imán presenta una pequeña oscilación al principio, se consigue estabilizar en el mismo punto que las anteriores y presenta un tiempo de establecimiento de aproximadamente 2.5 s, un valor superior al del experimento anterior. Respecto a la acción de control, presenta un gran pico de voltaje al principio, generado por la diferencia entre la distancia de la bola y la referencia en el primer instante. En los siguientes instantes se reduce considerablemente y sigue una curva suave hasta estabilizarse en torno a 3 V.

Habiendo realizado ambos experimentos, y observando que consigue estabilizar el sistema satisfactoriamente, se puede concluir que el primer controlador diseñado funciona de forma correcta y es válido para implementar en la maqueta virtual.

A continuación, se van a realizar los mismos experimentos con el controlador con acción integral. Se va comenzar intentando estabilizar el sistema partiendo del punto de funcionamiento, con una referencia de 0.301 m.

Las gráficas obtenidas son las siguientes (Fig.18 y Fig.19):

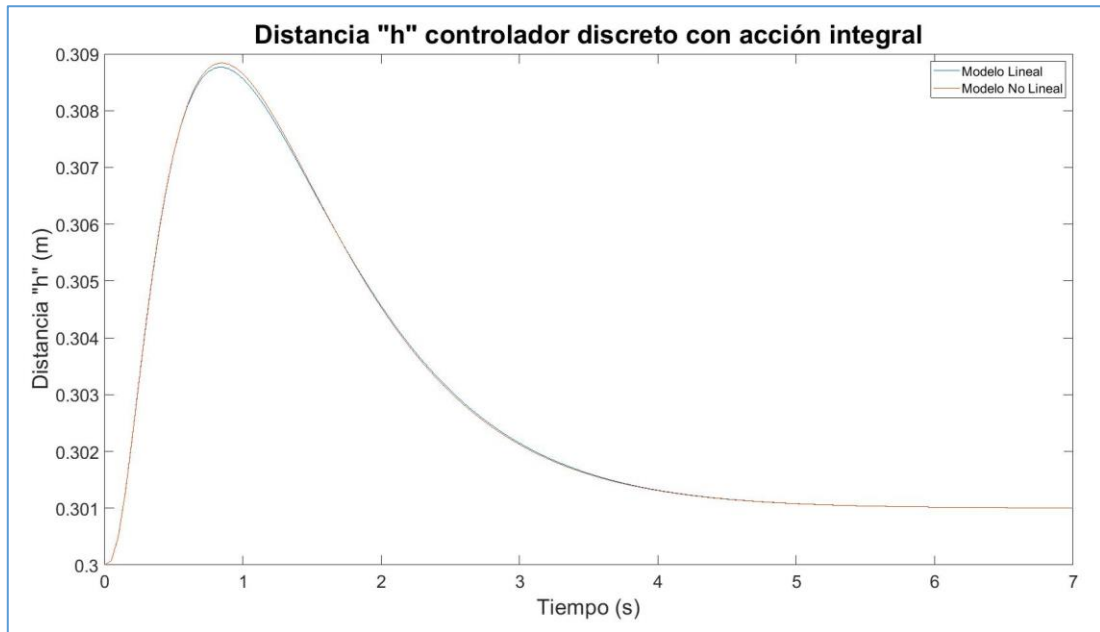


Fig. 18: Gráfica de la salida del modelo con el controlador discreto con acción integral

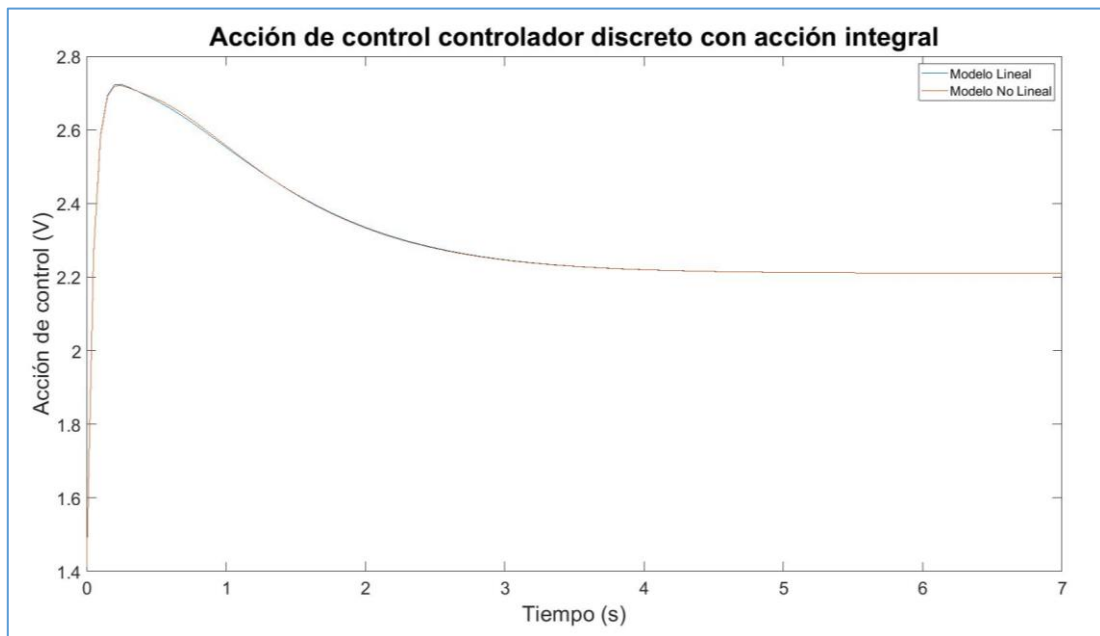


Fig. 19: Acción de control del controlador discreto con acción integral

Como se puede observar en la figura 18, el controlador consigue estabilizar el sistema y llevar la bola a la referencia de 0.301 m y no presenta diferencias significativas entre los dos modelos. Respecto a las características dinámicas, presenta una mayor sobreoscilación puesto que el valor final es menor, teniendo como pico un valor de 0.309 m. El tiempo de establecimiento también es notablemente superior, estabilizándose la señal entorno a los 4.5 segundos.

La acción de control (Fig.19) también presenta diferencias notables, esta vez presenta una pequeña sobreoscilación. Esta es generada por la necesidad de aumentar el voltaje para atraer la bola al imán y posicionarla a la distancia determinada. Además, se puede observar como esta vez se estabiliza en un valor cercano al del voltaje en el punto de funcionamiento, debido a que el valor final es muy cercano a los 0.3 m del punto de funcionamiento.

Con los resultados obtenidos, se ha comprobado que el controlador es capaz de estabilizar el sistema aún con la condición de presentar un error de posición nulo. Para determinar la robustez del controlador y su validez para ser implementado en la maqueta, se va a realizar el segundo experimento. Se va a configurar la posición inicial de la bola a 0.31 m para poder comprar el funcionamiento de los dos controladores correctamente.

Los gráficos obtenidos de este experimento son los siguientes (Fig.20 y Fig.21):

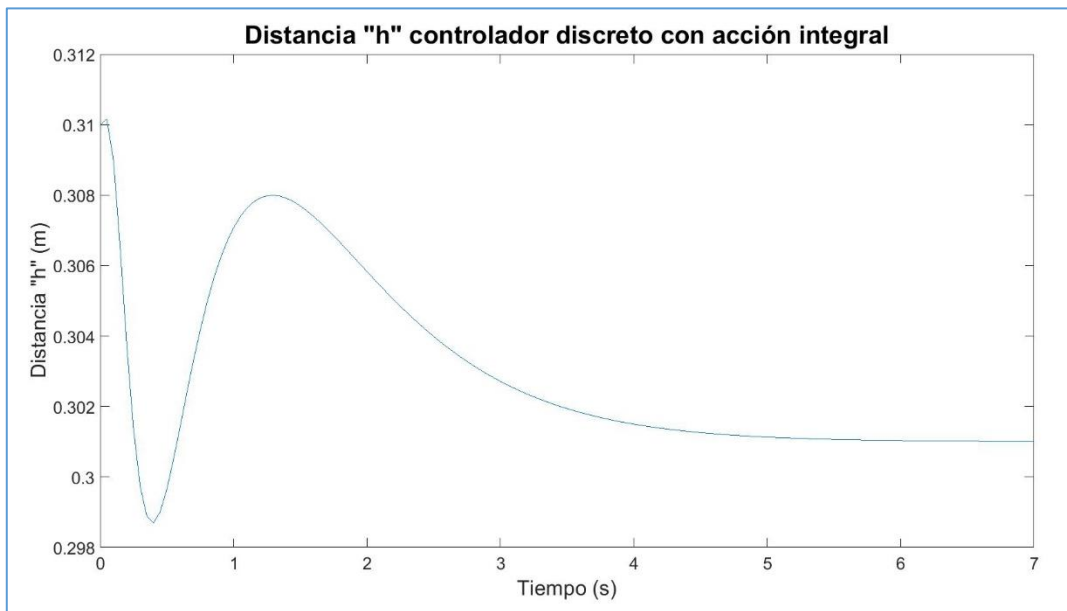


Fig. 20: Gráfica de la salida del modelo con el controlador discreto con acción integral desde  $h = 0.31$  m

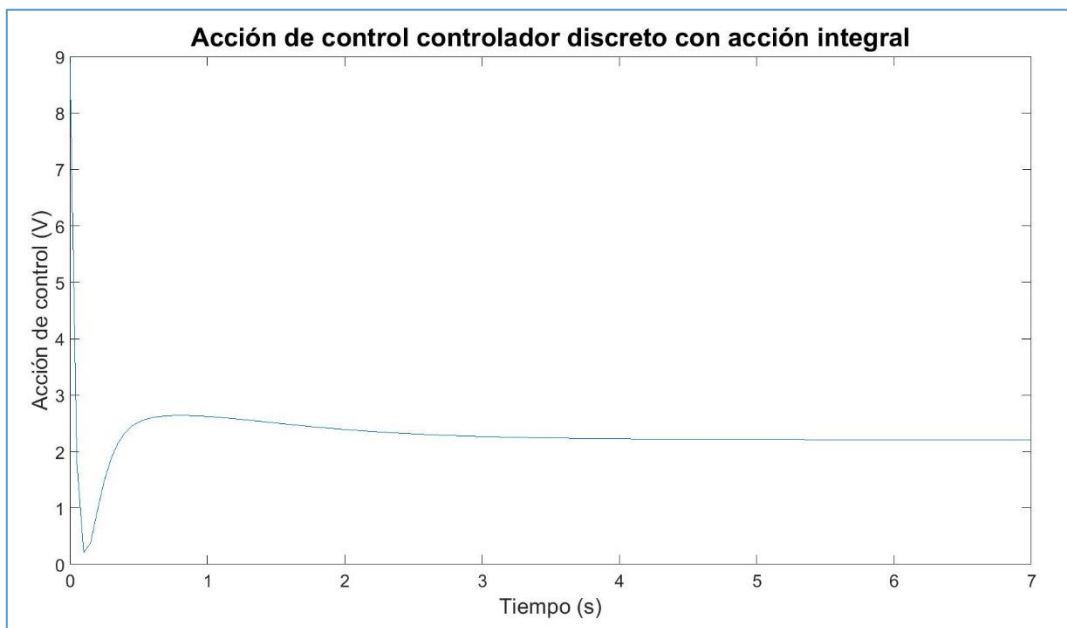


Fig. 21: Acción de control del controlador discreto con acción integral desde  $h = 0.31$  m

Analizando los datos obtenidos en este último experimento, se puede observar como el controlador con acción integral consigue estabilizar el levitador magnético aun comenzando desde un punto diferente al funcionamiento. Analizando las gráficas, se puede ver que la señal de salida contiene una oscilación inicial, debida a la posición inicial diferente. La sobreoscilación que presenta es la misma que en el experimento anterior y el tiempo de establecimiento es algo superior, entorno a los 5 segundos.

La acción de control presenta un gran pico en el primer instante, debido a la diferencia instantánea entre la distancia de la bola y la referencia. Esta baja rápidamente y a continuación describe una curva suave, hasta que se estabiliza en un valor cercano al del voltaje en el punto de funcionamiento.

Habiendo realizado los experimentos y analizado los resultados obtenidos, se puede concluir que este controlador también funciona correctamente y es válido para ser implementado en la maqueta virtual.

Para finalizar el apartado, se van a recopilar las características de los dos controladores y comprar ligeramente.

Controlador	Experimento	Sobreoscilación	T. establecimiento	Error de posición
Sin acción integral	$h = 0.3$	0	1.6 s	18 mm
	$h = 0.31$	0	2.5 s	18 mm
Con acción integral	$h = 0.3$	8 mm	4.5 s	0
	$h = 0.31$	8 mm	5 s	0

*Tabla 3: Resultados de los experimentos de los controladores*

Como se puede ver en los resultados de la Tabla 3, el primer controlador, proporciona un menor tiempo de establecimiento con oscilación nula, pero tiene el problema de presentar un error de posición de 18 mm. Lo que se traduce en un 18% de error de posición.

Por la otra parte, el controlador con acción integral, presenta un tiempo de establecimiento mayor con algo de sobreoscilación, entorno al 8%, pero tiene el beneficio de proporcionar un error de posición cero. Esto asegura que se sigue la referencia perfectamente.

Ambos controladores son válidos y van a ser implementados en la maqueta. Dependiendo del uso que se le quiera dar, se decidiría por uno de los controladores. Siempre se han de tener en cuenta las especificaciones necesarias en el levitador para la aplicación seleccionada.

Con la validación de los controladores realizada, se ha completado el segundo objetivo marcado para este Trabajo Fin de Grado. A continuación, se va a proceder a la validación de la maqueta mediante la implementación de estos controladores.

## 5. Implementación en una maqueta virtual mediante Easy Java Simulations

Dada la situación de pandemia en la que se ha desarrollado este Trabajo Fin de Grado, se ha decidido realizar una maqueta virtual en vez de implementarlo en un levitador real, que requeriría de acudir a un laboratorio. Para realizar la maqueta virtual se ha seleccionado el software Easy Java Simulations.

Esta maqueta ha sido proporcionada por el tutor, está diseñada con la finalidad de asemejarse lo máximo posible a la realidad. La maqueta virtual comienza la simulación al soltar la barra que varía la distancia de la bola al imán. La interfaz completa de la maqueta se puede ver la figura 22.

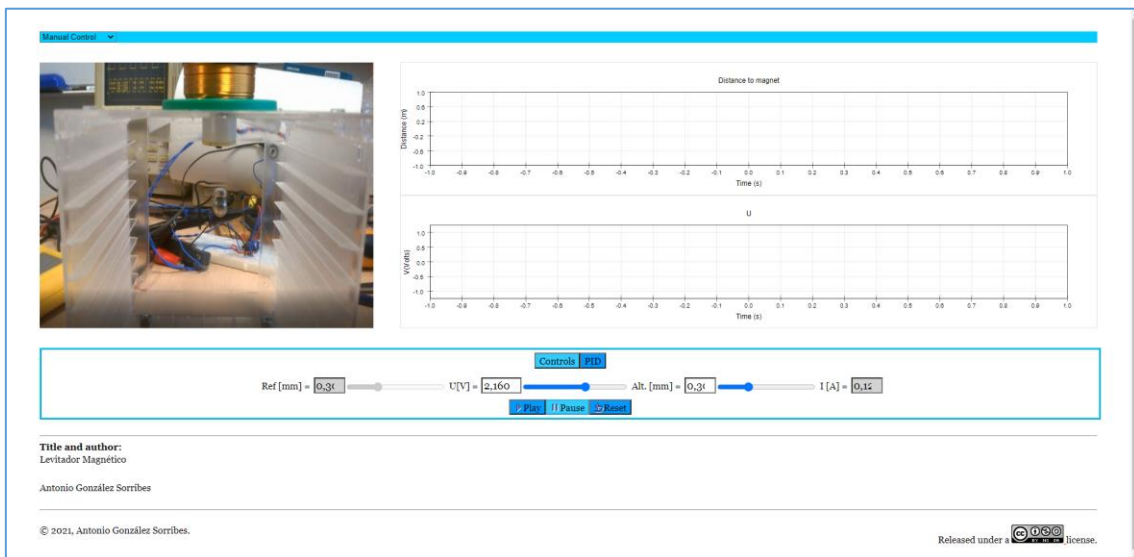


Fig. 22: Maqueta virtual del levitador magnético programada mediante Easy Java Simulations

### 5.1. Introducción a Easy Java Simulations

Easy Java Simulations (EJS) es una herramienta software diseñada específicamente para realizar simulaciones discretas mediante computador [7].

Esta herramienta se utiliza principalmente en el campo de la docencia y de la investigación científica [8][9]. Debido a que software permite simular procesos mediante el uso de un navegador y sin la necesidad de construir una maqueta real para realizar los experimentos.

Los procesos se simulan mediante el cálculo de las ecuaciones diferenciales que describen la evolución del proceso. Estas se asocian a las variables que se pueden visualizar tanto en gráficas, como en objetos móviles que se pueden configurar para simular el funcionamiento real del proceso. Por ejemplo, la interfaz de la maqueta del levitador magnético que se puede ver en la figura 23.

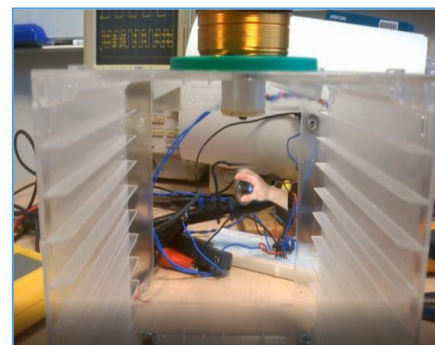


Fig. 23: Interfaz gráfica de la maqueta virtual del levitador



El programa EJS proporciona una plataforma sencilla para realizar simulaciones mediante ordenador. Permitiendo así, que el usuario dedique la mayoría del tiempo a los aspectos específicos de la simulación. Ya sean las ecuaciones diferenciales a simular, o las interfaces, presentando de forma automática toda la programación necesaria para realizar la simulación en navegador web.

Cabe destacar que la motivación original de los laboratorios virtuales no fue la pandemia, sino ofrecer la posibilidad de enriquecer el baremo de prácticas de laboratorio con fines académicos. No obstante, esta herramienta ha resultado de gran utilidad para facilitar la adaptación a docencia no presencial en asignaturas de control en tiempos de pandemia [10].

## 5.2. Implementación del control del levitador en la maqueta virtual

La maqueta que se va a validar, ha sido proporcionada por el tutor. Esta maqueta está completada a falta de la validación del modelo. Esto se va a realizar mediante la implementación de los controladores diseñador previamente.

Para la implementación de los controladores, se ha de calcular la ecuación en diferencias de éstos. Esto se calcula partiendo de las funciones de transferencia, realizando la transformada para expresar esta función de transferencia en instantes. La ecuación en diferencias es una expresión recursiva que se calcula mediante las entradas y salidas del controlador pasadas, transformando las  $z$  en instantes pasados.

El cálculo de la ecuación en diferencias del primer controlador diseñado es el siguiente:

$$GR_1 = \frac{-721 \cdot z^2 + 1079 \cdot z - 399}{z^2 - 0.2725 \cdot z + 0.04285}$$

La función de transferencia discreta representa la relación entre la salida del controlador y la entrada. La salida siendo la acción de control aplicada al modelo y la entrada la diferencia entre la distancia al imán actual y la referencia, que se llamará error.

$$\frac{U_{(z)}}{E_{(z)}} = \frac{-721 \cdot z^2 + 1079 \cdot z - 399}{z^2 - 0.2725 \cdot z + 0.04285}$$

$$U_{(z)} \cdot (z^2 - 0.2725 \cdot z + 0.04285) = E_{(z)} \cdot (-721 \cdot z^2 + 1079 \cdot z - 399)$$

$$U_{(z)} \cdot z^2 - U_{(z)} \cdot 0.2725 \cdot z + U_{(z)} \cdot 0.04285 = -E_{(z)} \cdot 721 \cdot z^2 + E_{(z)} \cdot 1079 \cdot z - E_{(z)} \cdot 399$$

Con las ecuaciones desarrolladas se puede realizar la transformación a  $z$  con exponente negativo, se multiplica ambos lados de la ecuación por  $z^{-2}$ , de esta forma se cancela el mayor orden de  $z$ .

$$\begin{aligned} U_{(z)} - U_{(z)} \cdot 0.2725 \cdot z^{-1} + U_{(z)} \cdot 0.04285 \cdot z^{-2} \\ = -E_{(z)} \cdot 721 \cdot z^{-2} + E_{(z)} \cdot 1079 \cdot z^{-1} - E_{(z)} \cdot 399 \cdot z^{-2} \end{aligned}$$

Para representarlo de forma temporal, se transforma las  $z$  con orden negativo en instantes pasados, esto significa que  $z^{-1}$  se representa como el valor pasado de la variable, en caso de ser  $z^{-2}$  el valor de dos instante anteriores y así hasta el exponente mínimo de  $z$ .

$$U_{(k)} - U_{(k-1)} \cdot 0.2725 + U_{(k-2)} \cdot 0.04285 = -E_{(k)} \cdot 721 \cdot z^{-2} + E_{(k-1)} \cdot 1079 - E_{(k-2)} \cdot 399$$

Finalmente, se despeja la variable  $U_{(k)}$ , esta es la acción de control en el instante actual, y es la que se ha de calcular para aplicar esta acción de control al modelo y controlarlo.

$$U_{1(k)} = U_{(k-1)} \cdot 0.2725 - U_{(k-2)} \cdot 0.04285 - E_{(k)} \cdot 721 + E_{(k-1)} \cdot 1079 - E_{(k-2)} \cdot 39$$

El cálculo de la ecuación en diferencias del controlador con acción integral se calcula del mismo modo.

$$GR_2 = \frac{-747.7 \cdot z^3 + 1864 \cdot z^2 - 1529 \cdot z + 412.3}{z^3 - 1.42 \cdot z^2 + 0.6122 \cdot z - 0.1871}$$

$$\frac{U_{(z)}}{E_{(z)}} = \frac{-747.7 \cdot z^3 + 1864 \cdot z^2 - 1529 \cdot z + 412.3}{z^3 - 1.42 \cdot z^2 + 0.6122 \cdot z - 0.1871}$$

$$U_{(z)} \cdot (z^3 - 1.42 \cdot z^2 + 0.6122 \cdot z - 0.1871) = E_{(z)} \cdot (-747.7 \cdot z^3 + 1864 \cdot z^2 - 1529 \cdot z + 412.3)$$

Desarrollando esta ecuación se obtiene:

$$U_{(z)} \cdot z^3 - U_{(z)} \cdot 1.42 \cdot z^2 + U_{(z)} \cdot 0.6122 \cdot z - U_{(z)} \cdot 0.1871 = -E_{(z)} \cdot 747.7 \cdot z^3 + E_{(z)} \cdot 1864 \cdot z^2 - E_{(z)} \cdot 1529 \cdot z + E_{(z)} \cdot 412.3$$

Se multiplica en ambas partes por  $z^{-3}$  para expresar la ecuación en exponentes de  $z$  negativos.

$$U_{(z)} - U_{(z)} \cdot 1.42 \cdot z^{-1} + U_{(z)} \cdot 0.6122 \cdot z^{-2} - U_{(z)} \cdot 0.1871 \cdot z^{-3} = -E_{(z)} \cdot 747.7 + E_{(z)} \cdot 1864 \cdot z^{-1} - E_{(z)} \cdot 1529 \cdot z^{-2} + E_{(z)} \cdot 412.3 \cdot z^{-3}$$

Se transforman los exponentes negativos de  $z$  en retardos temporales en las variables.

$$U_{(k)} - U_{(k-1)} \cdot 1.42 + U_{(k-2)} \cdot 0.6122 - U_{(k-3)} \cdot 0.1871 = -E_{(k)} \cdot 747.7 + E_{(k-1)} \cdot 1864 - E_{(k-2)} \cdot 1529 + E_{(k-3)} \cdot 412.3$$

Finalmente se puede despejar la variable  $U_{(k)}$  para obtener la ecuación en diferencias de la acción de control del controlador con acción integral.

$$U_{2(k)} = U_{(k-1)} \cdot 1.42 - U_{(k-2)} \cdot 0.6122 + U_{(k-3)} \cdot 0.1871 - E_{(k)} \cdot 747.7 + E_{(k-1)} \cdot 1864 - E_{(k-2)} \cdot 1529 + E_{(k-3)} \cdot 412.3$$

Para implementar estas ecuaciones en diferencia, se va a utilizar la forma genérica de expresar la ecuación. Esta está expresada con variables  $a$  y  $b$ , estas multiplican los valores de la acción de control y el error respectivamente. La función genérica utilizada tiene la siguiente forma:

$$U_{(k)} = b_0 \cdot E_{(k)} + b_1 \cdot E_{(k-1)} + b_2 \cdot E_{(k-2)} + b_3 \cdot E_{(k-3)} - a_1 \cdot U_{(k-1)} - a_2 \cdot U_{(k-2)} - a_3 \cdot U_{(k-3)} - a_4 \cdot U_{(k-4)}$$

Tal y como está programado, para implementar los controladores, únicamente se han de sustituir estas variables en el código del control discreto. Este código calcula la acción de control cada tiempo de muestreo y la aplica al levitador magnético. Las variables a modificar se pueden observar en la figura 24.

```

//Coeficientes de G(z)
a0 = 1; // Siempre a 1 (no se utiliza)
a1 = -0.2725;
a2 = 0.04285;
a3 = 0;
a4 = 0;

b0 = -721;
b1 = 1079;
b2 = -399;
b3 = 0;

// Calculo de la accion de control
var uk = b0*ek + b1*ek1 + b2*ek2 + b3*ek3 - a1*uk1 - a2*uk2 - a3*uk3 - a4*uk4;

```

Fig. 24: Código para la implementación de los controladores discretos

Para finalizar este apartado, se van a mostrar las variables correspondientes para cada controlador:

Variable	Controlador sin acción integral	Controlador con acción integral
a1	-0.2725	-1.42
a2	0.04285	0.6122
a3	0	-0.1871
a4	0	0
b0	-721	-747.7
b1	1079	1864
b2	-399	1529
b3	0	412.3

Tabla 4: Variables de las ecuaciones en diferencias de los controladores

Con las variables definidas en la Tabla 4, se pueden implementar los controladores y validar la maqueta.

## 6. Validación de la maqueta virtual y resultados obtenidos

En este apartado, se van a presentar los resultados obtenidos de la implementación de los controladores en la maqueta. Éstos se van a comparar con los obtenidos en *Simulink* y a analizar sus diferencias.

Para mostrar los resultados, se va a simular la maqueta virtual con el controlador y se van a guardar los datos obtenidos. Los datos se cargarán en *Matlab* para poder crear las gráficas.

Además de estos resultados, durante la implementación y diseño de controladores, se ha observado que la distancia máxima inicial de la bola a la que el controlador puede estabilizar el sistema varía en función de los polos del controlador. Esto resulta interesante, pues es muy complicado de calcular matemáticamente, y se ha decidido analizar mediante *Simulink*.

### 6.1. Validación de la maqueta virtual

Para comprobar que la maqueta funciona correctamente, se van a implementar los dos controladores diseñados. El correcto funcionamiento de éstos determinará que la maqueta simula correctamente el funcionamiento de un levitador magnético.

Los experimentos de los que se van a obtener los resultados, se van a realizar iniciando la bola desde una posición diferente a la del punto de funcionamiento diseñado. Puesto que colocar la bola en el punto exacto en un modelo real no es posible. Por tanto, se va a partir de 0.31 m, con una referencia de 0.3 m.

El primer experimento se va a realizar con el primer controlador diseñado, se analizará y comparará tanto la variación de la distancia al imán como la acción de control obtenidas.

Realizando la simulación, se obtiene la siguiente distancia al imán en función del tiempo (Fig.25):

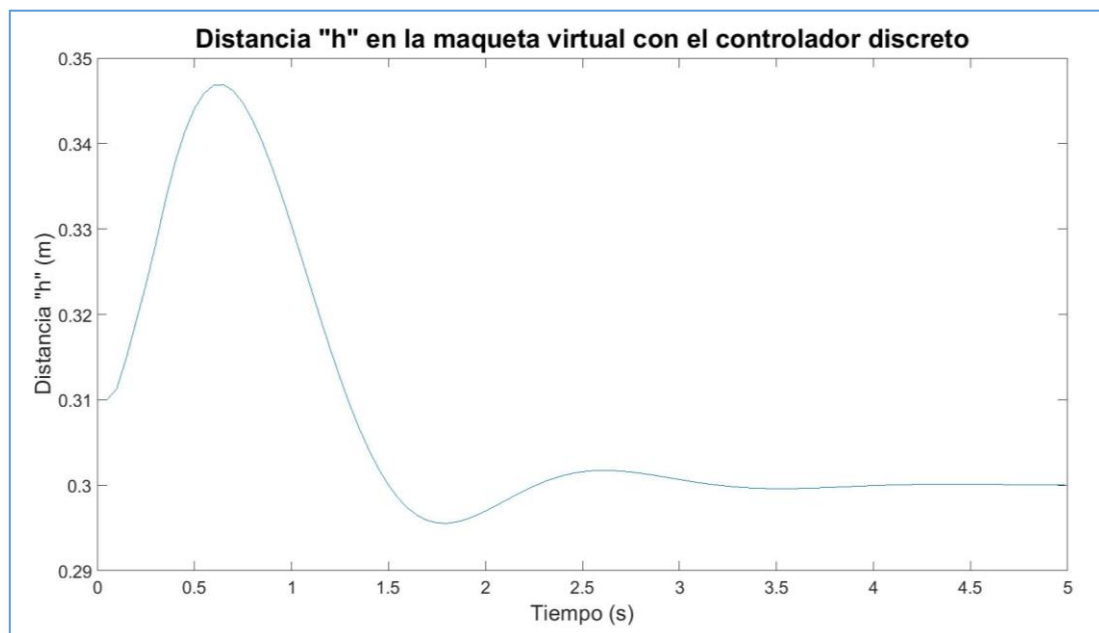


Fig. 25: Distancia al imán en la maqueta virtual con el controlador discreto

Como se puede observar en la gráfica obtenida, se consigue estabilizar la bola aun partiendo de una distancia inicial al imán diferente a la del punto de funcionamiento. La gráfica presenta una gran oscilación, alcanzando su máximo entorno a un valor de 0.345 m. Esto significa que en los primeros instantes la bola comienza a caer, pero el levitador magnético consigue atraerla hasta la referencia. El tiempo de establecimiento es de en torno a 3 s. Además, en la implementación en la maqueta virtual, este controlador presenta un error de posición mínimo, que no se puede apreciar en la gráfica. Esto puede ser debido a que la simulación genera una referencia exacta a 0.3, el punto donde está linealizado el modelo en el que se basan estos controladores.

Para determinar el correcto funcionamiento del controlador, se va a analizar la acción de control obtenida (Fig.26).

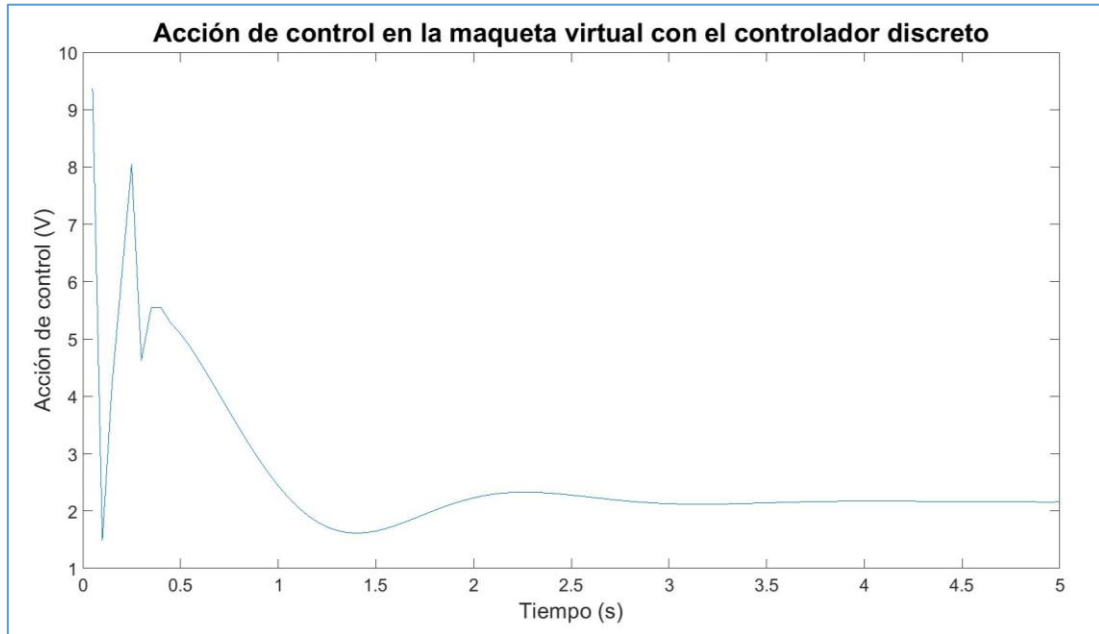


Fig. 26: Acción de control en la maqueta virtual con el controlador discreto

En la acción de control obtenida de la maqueta virtual, se pueden observar picos de voltaje en la bobina en los primeros instantes. Esto se debe a la diferencia de la distancia con la referencia en los primeros intentos, que resulta en grandes acciones de control que se suavizan con el tiempo. La señal no satura durante grandes periodos de tiempo, lo que confirma que el controlador funciona correctamente. Incluso se podría aumentar la distancia inicial al imán de la bola, hasta que la acción de control alcanzara valores más cercanos a la saturación.

Comparando estos resultados con los obtenidos en *Simulink*, el control en la maqueta virtual resulta más lento y con mayores oscilaciones. Esto es razonable puesto que el modelo de *Simulink* es una aproximación al proceso, y la maqueta simula de forma más realista el levitador magnético.

A continuación, se va a implementar el controlador con acción integral y se van a analizar las salidas obtenidas, tanto la distancia al imán como la acción de control.

La distancia al imán en función del tiempo obtenida implementando el controlador en la maqueta virtual (Fig. 27) es la siguiente:

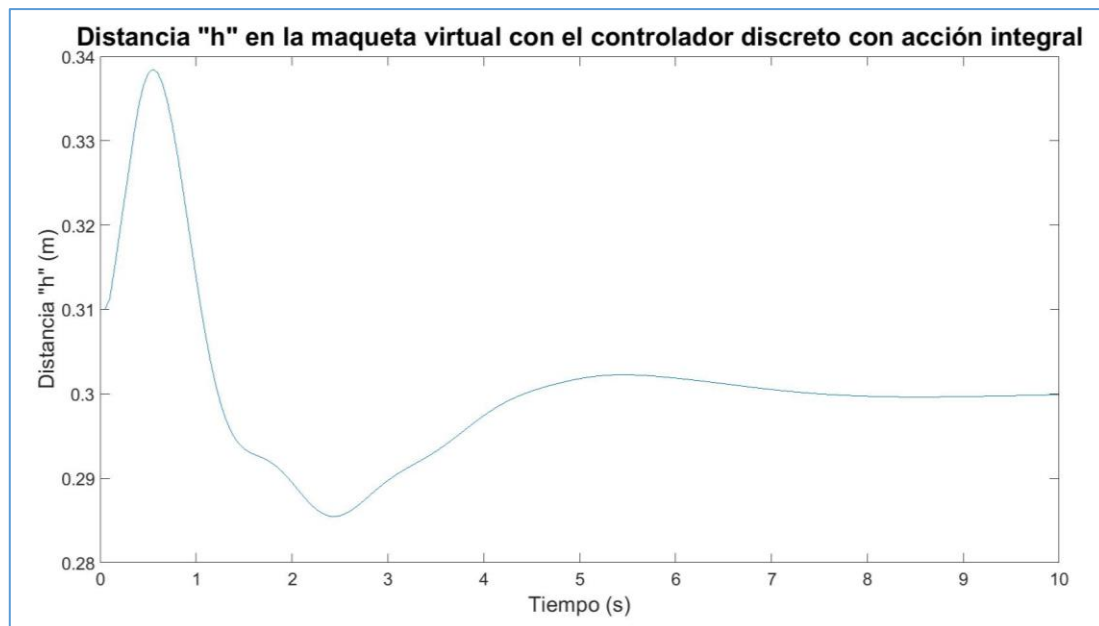


Fig. 27: Distancia al imán en la maqueta virtual con el controlador discreto con acción integral

Para este controlador se vuelve a obtener una gran oscilación en los primeros instantes, aunque esta vez es algo menor, alcanzando como máximo un valor cercano a 0.338 m. Esto implica que la bola cae menos en los primeros instantes. En cambio, presenta una mayor oscilación en los siguientes instantes, atrayendo la bola hasta 0.285 m. Posteriormente baja lentamente, hasta que alcanza el establecimiento en aproximadamente 8 segundos.

Este controlador consigue controlar correctamente el levitador magnético, aunque presenta peores especificaciones dinámicas. Principalmente, existe una gran diferencia en el tiempo de establecimiento. Este tiempo de establecimiento superior, está generado por la especificación de error de posición cero, que ralentiza el sistema. Sin embargo, este controlador presentará siempre error de posición nulo a la hora de la implementación.

Para finalizar el análisis de la implementación de este controlador, se va a analizar la acción de control (Fig. 28).

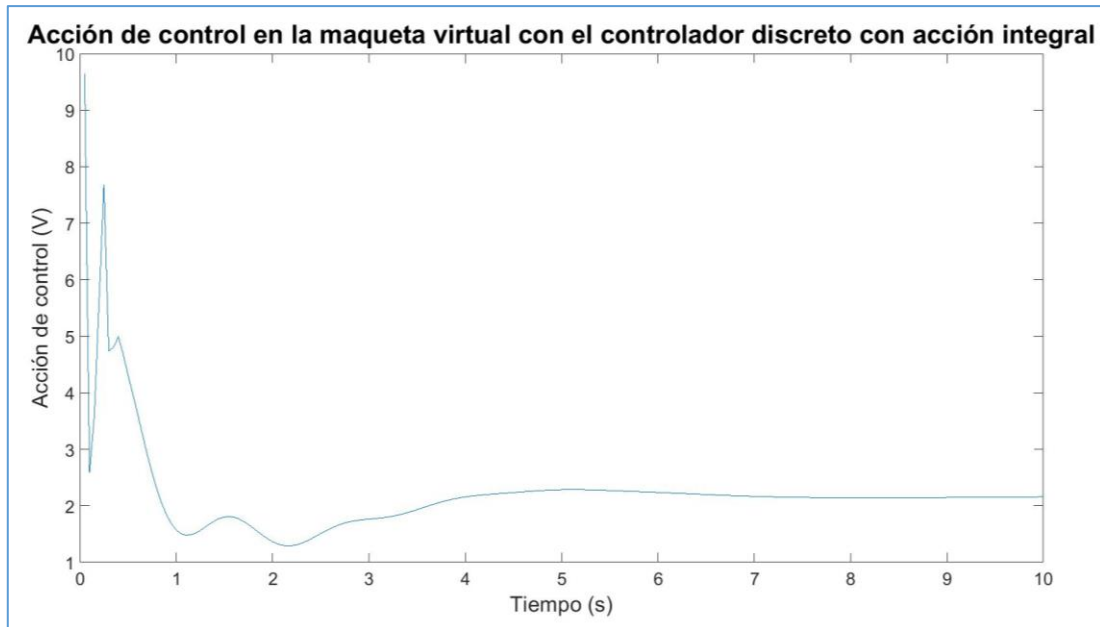


Fig. 28: Acción de control en la maqueta virtual con el controlador discreto con acción integral

Al igual que la acción de control del controlador anterior, los primeros instantes presentan grandes picos, que se suavizan hasta estabilizarse en el voltaje final.

El controlador no resulta saturado en ningún instante del control, y, exceptuando el primer instante, no se aproxima demasiado a los valores de saturación. Esto indica que el controlador funciona de forma correcta en la maqueta.

Comparando las gráficas obtenidas en la maqueta virtual con las gráficas obtenidas en el modelo de *Simulink*, se puede observar que la sobreoscilación aumenta. Especialmente en los primeros instantes donde la bola cae unos centímetros. El tiempo de establecimiento también presenta un aumento considerable. Aun así, el controlador consigue estabilizar el sistema partiendo de un punto diferente al de funcionamiento.

Habiendo analizando el funcionamiento de ambos controladores en la maqueta, se puede concluir que la maqueta funciona correctamente y simula el proceso de un levitador magnético de forma correcta. Por tanto, la maqueta resulta válida para ser utilizada tanto en docencia como en investigación científica.

Con esto, se habrá completado satisfactoriamente el tercer objetivo de este Trabajo Fin de Grado.

## 6.2. Efecto de los polos no dominantes sobre el dominio de atracción

Durante el testeado de controladores, se detectó que la distancia al imán máxima a la que se podía iniciar la simulación en la maqueta para que estabilizara el proceso variaba entre controladores. Esta distancia es un dato importante puesto que delimita el rango de funcionamiento del levitador magnético.

Habiendo detectado este fenómeno, se decidió estudiar el efecto en esta distancia según los controladores. Para aumentar la consistencia del experimento, se ha decidido realizar este estudio con el controlador que presenta acción integral. Dado que el valor final de los controladores será el mismo. Para analizar esta variable, se utiliza el control discreto implementado en *Simulink*, mediante un bucle automático se calcula la distancia máxima, aumentado la misma en 0.001 m cada prueba, hasta que el sistema no se consigue estabilizar.

Para comprobar el efecto de los controladores, y simplificar el experimento, se van a agrupar los cuatro polos a diseñar en el controlador en dos parejas. La pareja de polos dominantes (de mayor valor) y la pareja de polos no dominantes (de menor valor). Los rangos de posición de los polos que se van a utilizar en el estudio son de 0.8 hasta 0.95 con intervalos de 0.01 para la pareja de polos dominantes. Y de 0 hasta 0.8 con intervalos de 0.05 para la pareja de polos no dominantes.

La gráfica que se obtiene con todas las distancias máximas calculadas es la siguiente (Fig.29):

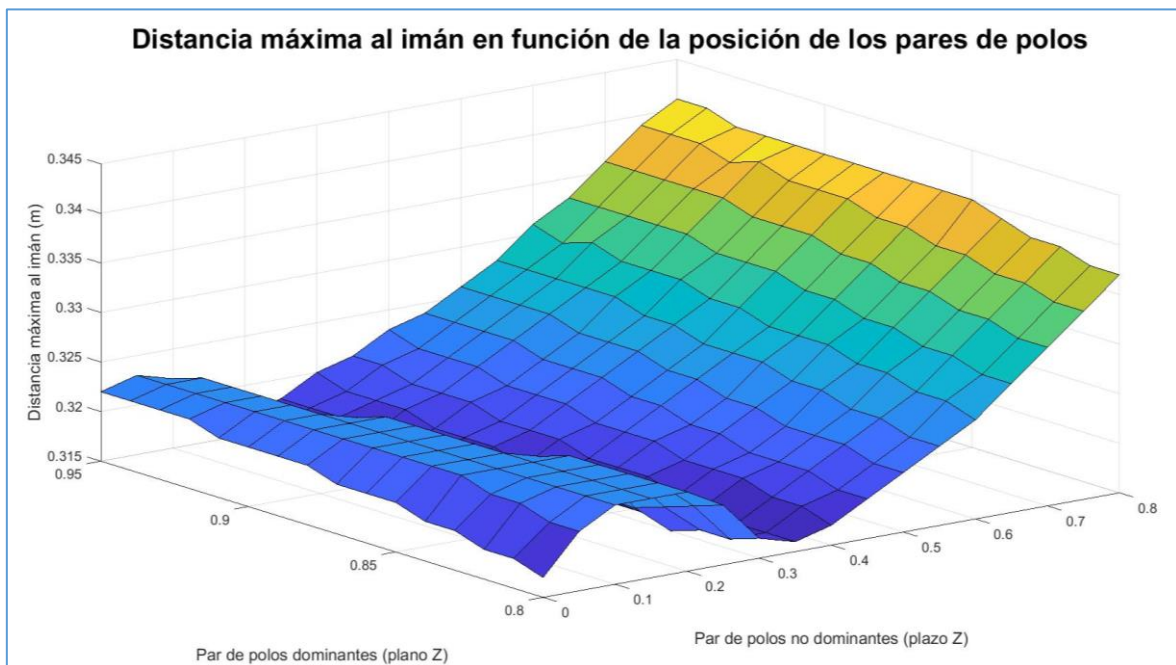


Fig. 29: Distancia máxima al imán en función de la posición de los pares de polos en el controlador con acción integral

Analizando el gráfico 3D obtenido, se puede determinar que el par de polos con mayor influencia es el par no dominante. Se observa una tendencia a aumentar en la distancia máxima con el aumento del módulo de estos polos. Para valores bajos, se observa un pequeño mínimo local en torno a 0.322 m, que desciende rápidamente a un mínimo local cercano a 0.315 m.

Por el otro lado, el par de polos no dominante tiene una influencia muy pequeña, aumentando ligeramente la distancia máxima conforme aumenta el módulo, pero no es muy significativo.

Hay que tener en cuenta a la hora de diseñar, que a mayor sea el módulo de los polos mayor será el tiempo de establecimiento del proceso.



Con el análisis de la gráfica realizado, se puede concluir que la variable presenta una tendencia a aumentar para valores superiores a 0.2 en los polos no dominantes. A cambio, se aumenta el tiempo de establecimiento del sistema.

Este estudio ha resultado de gran interés, puesto que estos datos son complicados de calcular matemáticamente y pueden servir de ayuda a la hora de diseñar controladores para un levitador magnético.

## 7. Conclusiones

Durante la extensión de este Trabajo Fin de Grado, se ha explicado detalladamente el proceso de diseño y validación del control discreto para un levitador magnético.

Se han cumplido de forma satisfactoria todos los objetivos planteados. Obteniendo en primer lugar un modelo linealizado correcto del proceso, que se comporta de forma similar al modelo no lineal en los puntos cercanos al punto de funcionamiento diseñado. Se han diseñado y validado dos controladores con distintas especificaciones. Y se ha conseguido estabilizar el sistema con éstos tanto en el modelo creado en *Simulink*, como en la maqueta virtual aportada por el tutor. Finalmente, se ha validado el correcto funcionamiento de la maqueta, la cual podrá ser usada tanto en docencia como en investigación en el campo de la levitación magnética.

En añadido, se han aportado datos experimentales de mucho valor sobre los levitadores. Tanto la diferencia entre la simulación del proceso y la implementación en un entorno más real, como el análisis de la distancia máxima de la bola para estabilizar el sistema.

Teniendo en cuenta esto, se puede concluir que se han obtenido resultados satisfactorios de este Trabajo Fin de Grado, y que se ha realizado de forma correcta.

Para finalizar el trabajo, destacar que ha resultado muy interesante trabajar con el levitador magnético. Además de haber tenido la oportunidad de aplicar las técnicas de control discreto aprendidas durante el grado a un proceso inestable y no lineal como es el levitador magnético.

## 8. Bibliografía

- [1] Yang, Jun, et al. "Robust control of nonlinear MAGLEV suspension system with mismatched uncertainties via DOBC approach." *ISA transactions* 50.3 (2011): 389-396.
- [2] Roman Gonzalez, Avid, Brian Meneses Claudio, and Zeila Torres Santos. "Study and design of a Magnetic Levitator System." (2019).
- [3] Lee, Hyung-Woo, Ki-Chan Kim, and Ju Lee. "Review of maglev train technologies." *IEEE transactions on magnetics* 42.7 (2006): 1917-1925.
- [4] Japan Rail Pass. *El Maglev japonés, el tren bala más rápido del mundo*. 2018
- [5] Yaghoubi, Hamid. "The most important maglev applications." *Journal of Engineering* 2013 (2013).
- [6] Greatrex, Nicholas, et al. "The Maglev Heart." *IEEE Spectrum* 56.09 (2019): 22-29.
- [7] Easy Java Simulations Wiki. <https://www.um.es/fem/EjsWiki/Main/HomePage>
- [8] Sánchez, J., et al. "Easy java simulations: an open-source tool to develop interactive virtual laboratories using matlab/simulink." *International Journal of Engineering Education* 21.5 (2005): 798.
- [9] Dormido, Sebastian, et al. "Adding interactivity to existing Simulink models using Easy Java Simulations." *Proceedings of the 44th IEEE Conference on Decision and Control*. IEEE, 2005.
- [10] González Sorribes, Antonio. "Laboratorios virtuales web como herramienta de apoyo para prácticas de ingeniería no presenciales." *IN-RED 2020: VI Congreso de Innovación Educativa y Docencia en Red*. Editorial Universitat Politècnica de València, 2021.





UNIVERSITAT  
POLITÈCNICA  
DE VALÈNCIA



Escuela Técnica Superior de Ingeniería del Diseño

# **ANEXO I: PLIEGO DE CONDICIONES**

**CURSO ACADÉMICO: 2020/2021**



## Pliego de condiciones

Durante la extensión del presente documento, se muestran las extensiones a existir en el contrato entre contratista y propietario en la ejecución del Trabajo Fin de Grado.

### Condiciones de los materiales

En este documento, se ha validado el correcto funcionamiento de una maqueta virtual que simula el funcionamiento de un levitador magnético. Para comprobar el funcionamiento de esta se han diseñado dos reguladores en tiempo discreto con diferentes especificaciones. Todo el contenido de esta Trabajo Fin de Grado será accesible a cualquier persona para su uso y la posible ampliación del tema tratado en el trabajo. La regulación utilizada en el presente Trabajo Fin de Grado se corresponde a las condiciones de uso de la plataforma Riunet donde estará disponible.

El contenido disponible que se ha generado durante el proyecto es el siguiente:

- Modelo no lineal implementado mediante diagrama de bloques en *Simulink* que simula el funcionamiento de un levitador magnético real.
- Función de transferencias, tanto continua como discreta, del modelo linealizado en el punto de funcionamiento seleccionado.
- Función de transferencia y ecuación en diferencias de los controladores discretos diseñados para el control entorno al punto de funcionamiento del levitador magnético.

A continuación, se detallan las condiciones de utilización de los softwares utilizados:

- Matlab y Simulink. La licencia utilizada para la realización del Trabajo Fin de Grado se corresponde con una licencia individual perpetua.
- Easy Java Simulations. La licencia utilizada para la realización del Trabajo Fin de Grado se corresponde con una licencia gratuita para usos no comerciales.

### Condiciones de utilización

El contenido del presente Trabajo Fin de Grado se ha utilizado y se ha de utilizar con fines didácticos, nunca con la intención de obtener una ventaja comercial o una ganancia financiera.







UNIVERSITAT  
POLITÈCNICA  
DE VALÈNCIA



Escuela Técnica Superior de Ingeniería del Diseño

# **ANEXO II: PRESUPUESTO**

**CURSO ACADÉMICO: 2020/2021**



## Presupuesto

En este documento se va a detallar el presupuesto necesario para la realización del Trabajo Fin de Grado. Se van a diferenciar los costes según su naturaleza, dividiendo los costes entre el coste de materiales, de software y de mano de obra.

Posteriormente se va a presentar un desglose global de los costes para el cálculo del presupuesto.

## Desglose de costes en función de la naturaleza

### Coste de materiales

Dada la naturaleza virtual del presente Trabajo Fin de Grado, únicamente ha sido necesario el uso de un ordenador portátil. Este se trata de un ordenador portátil marca *ACER*, con un procesador *Intel Core i5-7200U 2.5GHz*, con una tarjeta gráfica *NVIDIA GeForce 940MX*, 8 GB de memoria RAM tipo *DDR4* y 1TB de memoria interna.

Para calcular el coste proporcional del ordenador portátil, se ha considerado su periodo de utilización es de 5 años. También se va a calcular el coste aproximado del consumo energético. Tomando como precio medio 0.09 €/kWh, se obtiene un coste de 3.24 €.

Material	Coste (€)	Coste proporcional (€)
Ordenador portátil	600	10.76
<b>Total</b>	600	10.76

### Coste de software

Para la realización de este trabajo, ha sido necesario la utilización del software de pago Matlab R2021a, cuya licencia individual perpetua asciende a un total de 2000 €. Para realizar el cálculo del coste proporcional, se va a asumir un periodo de utilización de la licencia de 5 años.

También ha sido necesario la utilización del software Easy Java Simulations, el cual presenta una licencia perpetua gratuita.

Software	Coste (€)	Coste proporcional (€)
Matlab R2021a	2000 (Perpetua)	35.86
Easy Java Simulations	0 (Perpetua)	0
<b>Total</b>	2000	35.86

## Coste de mano de obra

Para la realización del trabajo, se ha requerido de un total de 180 h de trabajo por parte del autor y 15 h por parte del tutor, la mano de obra necesaria para la realización de este proyecto se ha dividido en los siguientes ítems:

- **Diseño:** Horas dedicadas al desarrollo matemático del modelo linealizado y de los controladores, diseño de modelos de *Simulink* y modificaciones realizadas al funcionamiento de la maqueta.
- **Programación:** Horas dedicadas a la programación de los scripts de Matlab y modificaciones del código de la maqueta.
- **Pruebas:** Horas dedicadas a la prueba de controladores y análisis de resultados tanto en la maqueta virtual como en los modelos de *Simulink*.
- **Documentación:** Horas dedicadas a la búsqueda de bibliografía y a la redacción de la documentación del Trabajo Fin de Grado.

El coste de las horas, se ha calculado con una remuneración de 20 €/h para el autor y 40 €/h para el tutor del trabajo.

<b>Personal</b>	<b>Tareas</b>	<b>Coste temporal (h)</b>	<b>Coste (€)</b>
Autor	Diseño	50	1000
	Programación	35	700
	Pruebas	25	500
	Documentación	70	1400
Tutor	Revisión y asesoramiento	15	600
<b>Total</b>		195	4200

## Desglose de costes totales

Agrupando todos los costes necesarios para el desarrollo del trabajo, el coste total del trabajo sería el siguiente:

<b>Desglose de costes totales</b>	
Material	10.76 €
Consumo energético	3.24 €
Software	35.86 €
Mano de obra	4200 €
<b>Total</b>	4249.86 €

El presente Trabajo Fin de Grado, ha resultado en un coste total de Cuatro Mil Doscientos Cuarenta y Nueve Euros con Ochenta y Seis Céntimos (4249.86 €).



UNIVERSITAT  
POLITÈCNICA  
DE VALÈNCIA



Escuela Técnica Superior de Ingeniería del Diseño

## **ANEXO III: PLANOS**

**CURSO ACADÉMICO: 2020/2021**

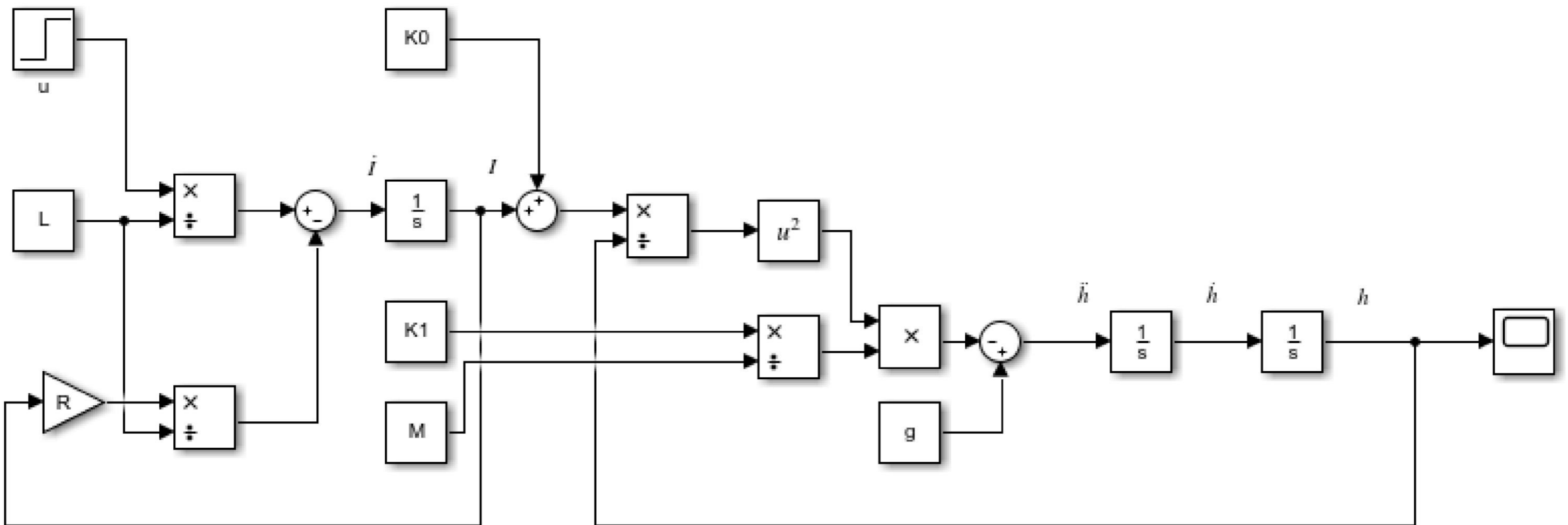


## **Índice de planos**

<b>1. Modelo no lineal.....</b>	<b>65</b>
<b>2. Comparación de modelos.....</b>	<b>66</b>
<b>3. Control continuo.....</b>	<b>67</b>
<b>4. Control discreto.....</b>	<b>68</b>
<b>5. Cálculo de distancia máxima.....</b>	<b>69</b>







PROYECTO: DISEÑO Y VALIDACIÓN DEL CONTROL DE UN LEVITADOR  
MAGNÉTICO EN UN LABORATORIO VIRTUAL MEDIANTE  
EASY JAVA SIMULATIONS

Fecha: 17/06/2021

Escala  
**S/E**

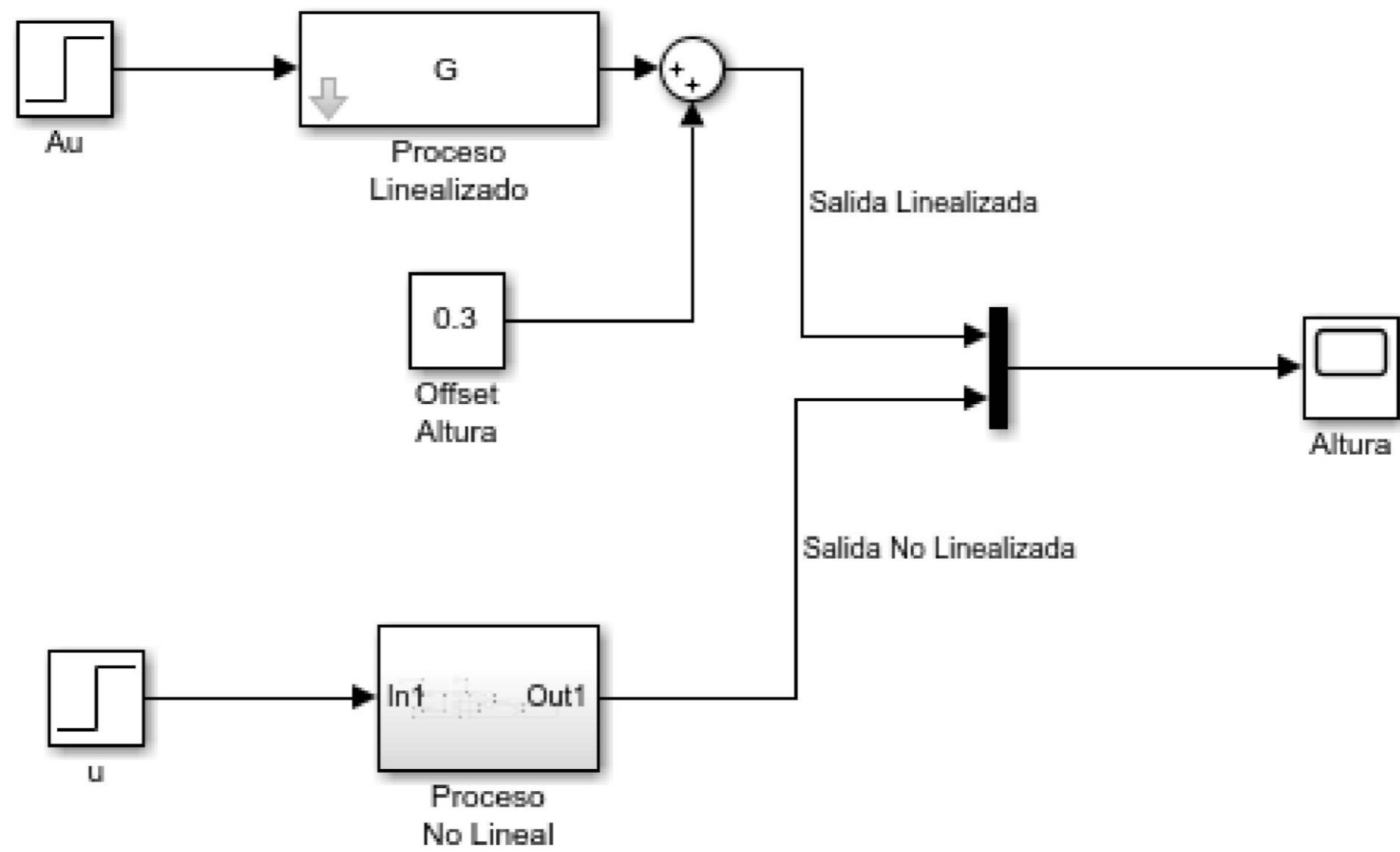
Autor: Pau Mora

Plano:

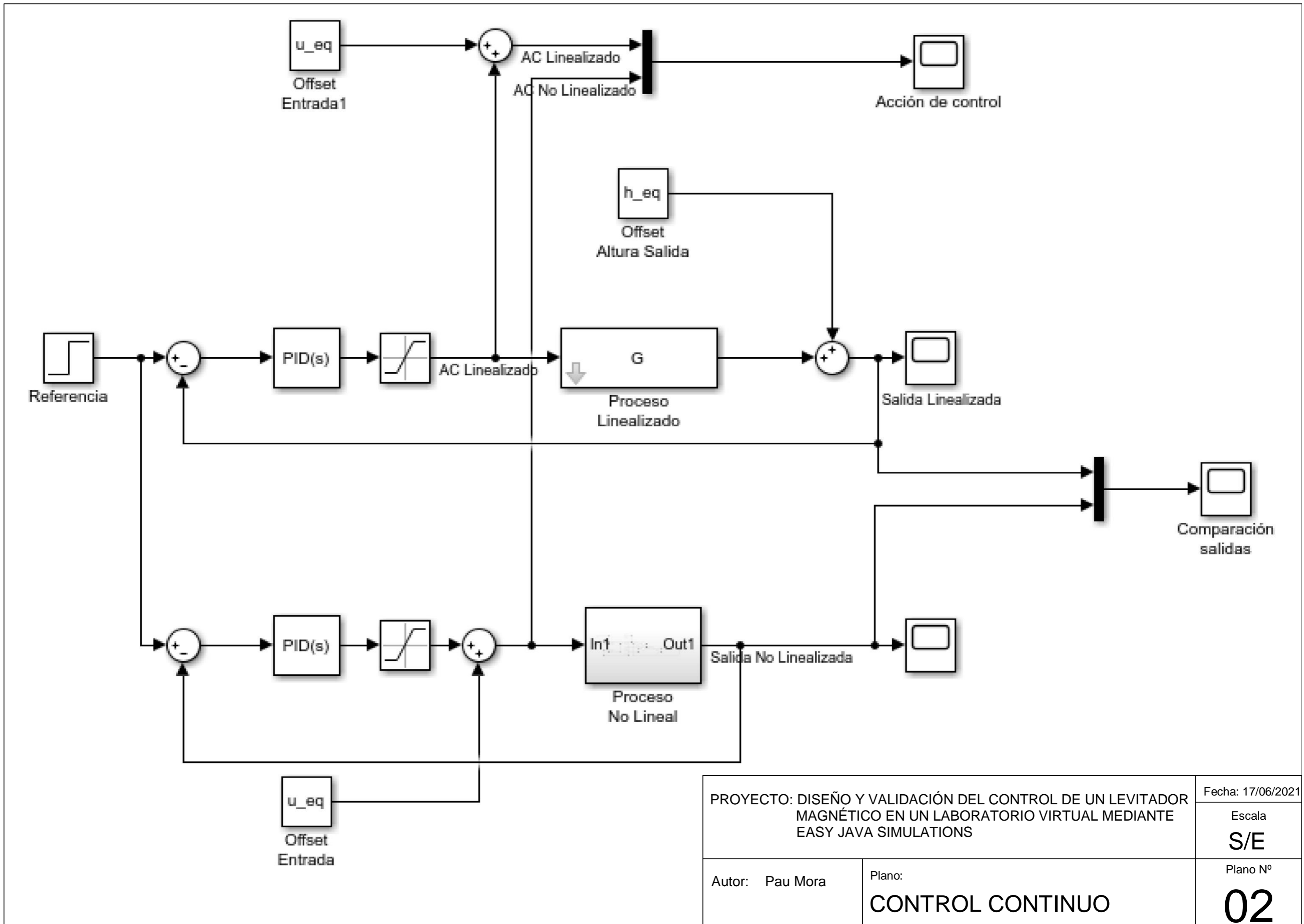
**MODELO NO LINEAL**

Plano Nº

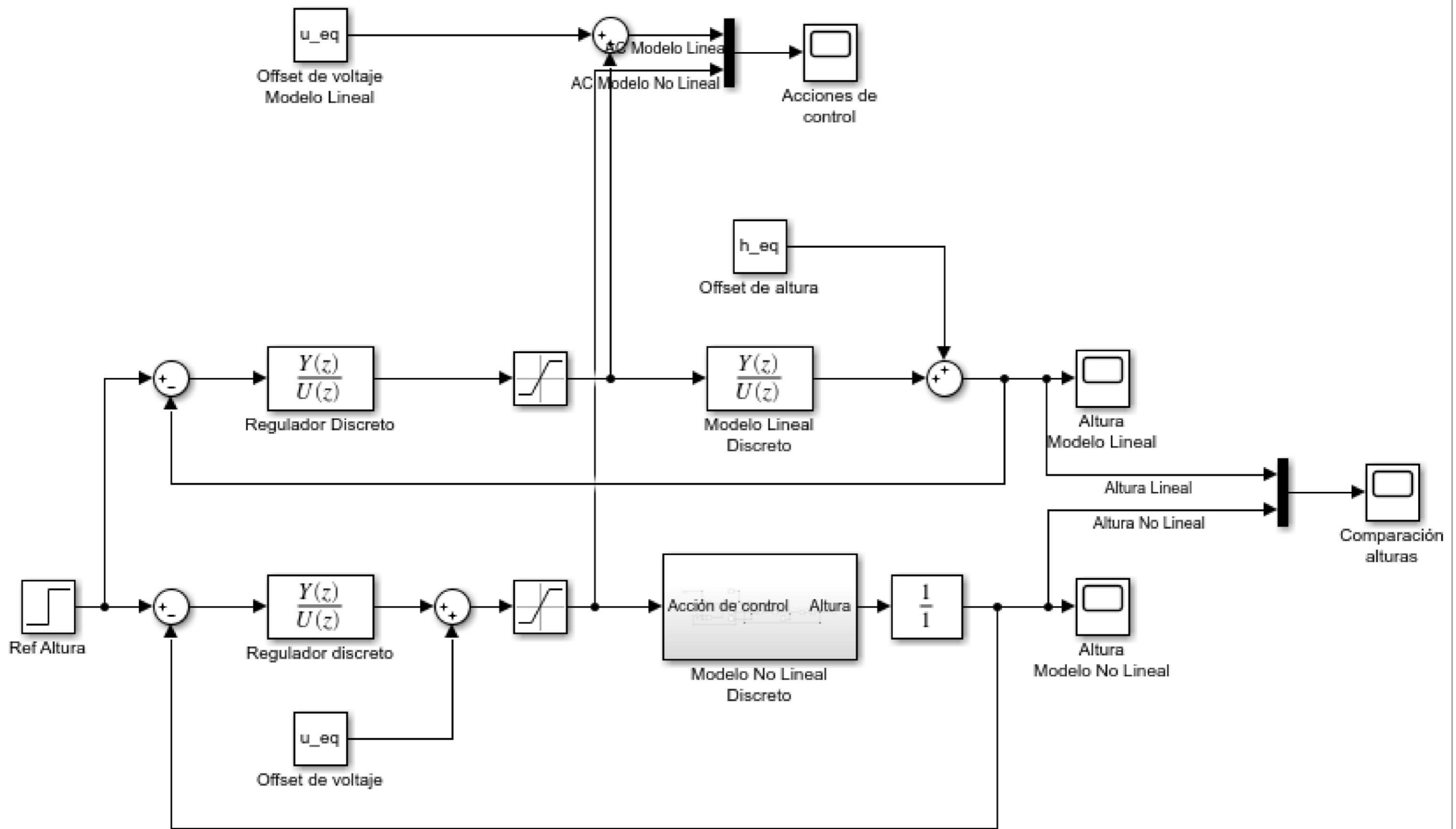
**00**



PROYECTO: DISEÑO Y VALIDACIÓN DEL CONTROL DE UN LEVITADOR MAGNÉTICO EN UN LABORATORIO VIRTUAL MEDIANTE EASY JAVA SIMULATIONS		Fecha: 17/06/2021
		Escala <b>S/E</b>
Autor: Pau Mora	Plano: <b>COMPARACIÓN DE MODELOS</b>	Plano Nº <b>01</b>



PROYECTO: DISEÑO Y VALIDACIÓN DEL CONTROL DE UN LEVITADOR MAGNÉTICO EN UN LABORATORIO VIRTUAL MEDIANTE EASY JAVA SIMULATIONS		Fecha: 17/06/2021
		Escala <b>S/E</b>
Autor: Pau Mora	Plano: <b>CONTROL CONTINUO</b>	Plano Nº <b>02</b>



PROYECTO: DISEÑO Y VALIDACIÓN DEL CONTROL DE UN LEVITADOR  
MAGNÉTICO EN UN LABORATORIO VIRTUAL MEDIANTE  
EASY JAVA SIMULATIONS

Fecha: 17/06/2021

Escala  
**S/E**

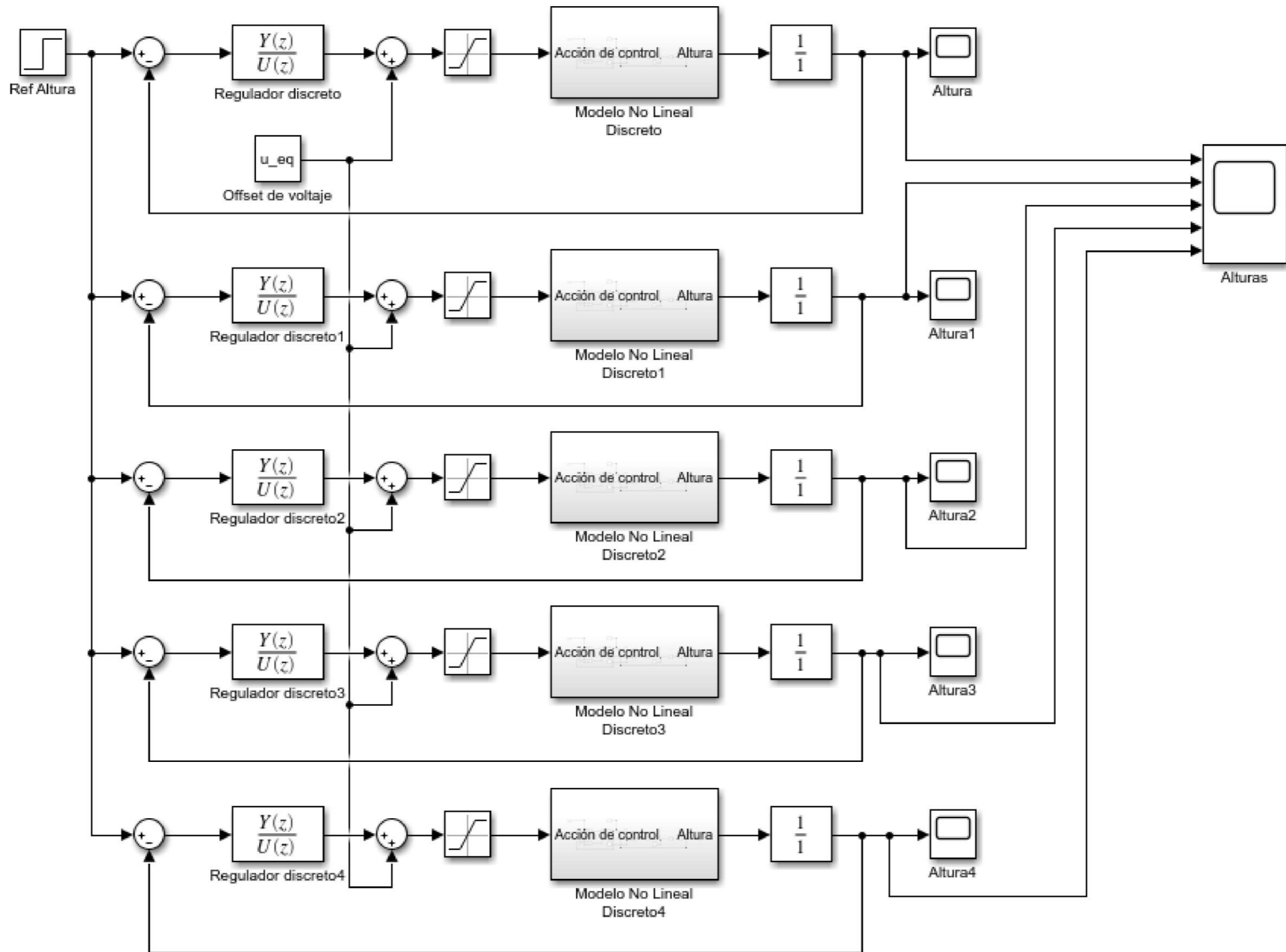
Autor: Pau Mora

Plano:

**CONTROL DISCRETO**

Plano Nº

**03**



PROYECTO: DISEÑO Y VALIDACIÓN DEL CONTROL DE UN LEVITADOR MAGNÉTICO EN UN LABORATORIO VIRTUAL MEDIANTE EASY JAVA SIMULATIONS

Fecha: 17/06/2021

Escala  
**S/E**

Autor: Pau Mora

Plano:  
**CÁLCULO DISTANCIA MAX**

Plano Nº  
**04**



UNIVERSITAT  
POLITÈCNICA  
DE VALÈNCIA



Escuela Técnica Superior de Ingeniería del Diseño

# **ANEXO IV: CÓDIGO DE MATLAB**

**CURSO ACADÉMICO: 2020/2021**



## **Índice de anexos**

<b>1. Script de Matlab ‘Linealizacion.m’ .....</b>	<b>74</b>
<b>2. Script de Matlab ‘Control_Continuo.m’ .....</b>	<b>76</b>
<b>3. Script de Matlab ‘Control_Discreto.m’ .....</b>	<b>78</b>
<b>4. Script de Matlab ‘Estudio_Distancia.m’ .....</b>	<b>81</b>
<b>5. Script de Matlab ‘Captura_Datos.m’ .....</b>	<b>83</b>





## 1. Script de Matlab 'Linealizacion.m'

```
clear
clc

% Parámetros del levitador
M = 0.004;
K0 = 0.7185;
K1 = 5.023*10^-3;
L = 4.8;
R = 18;
g = 9.81;

% Cálculo del punto de equilibrio
h_eq = 0.3;

syms I u

I_eq = solve(0==g-(K1/M)*((K0+I)/h_eq)^2,I); % Cálculo de la
intensidad en el equilibrio
I_eq = double(I_eq);
I_eq = I_eq(2); % Se selecciona la intensidad de 0.12 A

u_eq = solve(0==(u/L)-(R*I_eq)/L,u); % Cálculo del voltaje en el
equilibrio
u_eq = double(u_eq);

clear I u

% Linealización de las ecuaciones
syms h I

ec_nl = g - (K1/M)*((K0 + I)/h)^2; % Ecuacion no linealizada

ec_lI = diff(ec_nl,I); % Derivada parcial respecto a I
ec_lI = double(subs(ec_lI, [I, h], [I_eq, h_eq])); % Sustitución de
los valores en el equilibrio

ec_lh = diff(ec_nl,h); % Derivada parcial respecto a h
ec_lh = double(subs(ec_lh, [I, h], [I_eq, h_eq])); % Sustitución de
los valores en el equilibrio

clear I u

% Comprobación visual de la linealización

syms I

figure;sgtitle('Comparación de la ecuación no lineal y
linealizada','FontSize',40);hold;

% Figura de las ecuaciones con h_eq y I variable
subplot(121);fplot(g-(K1/M)*((K0+I)/h_eq)^2,[I_eq - 0.5, I_eq +
0.5]);
hold; fplot(ec_lI*(I-I_eq) + ec_lh*0,[I_eq - 0.5, I_eq + 0.5]);
title('I variable','FontSize',25);legend('Ecuación No
Lineal','Ecuación Linealizada','FontSize',15);
```

```

set(gca,'FontSize',15);xlabel('Intensidad
(A)', 'FontSize',20);ylabel('Aceleración (m/s^2)', 'FontSize',20);

clear I
syms h
% Figura de las ecuaciones con I_eq y h variable
subplot(122); fplot(g-(K1/M)*((K0+I_eq)/h)^2,[h_eq - 0.05, h_eq +
0.05]);
hold; fplot(ec_lI*0 + ec_lh*(h-h_eq),[h_eq - 0.05, h_eq + 0.05]);
title('h variable', 'FontSize',25);legend('Ecuación No
Lineal', 'Ecuación Linealizada', 'FontSize',15);
set(gca,'FontSize',15);xlabel('Distancia "h"
(m)', 'FontSize',20);ylabel('Aceleración (m/s^2)', 'FontSize',20);

clear h

% Creación del espacio de estados, se seleccionan tres estados,
derivada de
% h, h e I

A = [0 1 0; ec_lh 0 ec_lI; 0 0 -R/L]; % Derivadas de los estados en
función de los estados
B = [0; 0; 1/L]; % Derivadas de los estados en función de las
entradas
C = [1 0 0]; % Salidas en función de los estados
D = [0];

sys_M = ss(A,B,C,D); % Creación del sistema en espacio de estados

G = tf(sys_M); % Obtención de la función de transferencia en dominio
continuo

Ts = 0.05;
Gz = c2d(G,Ts,'zoh'); % Obtención de la función de transferencia en
dominio discreto

% Validación del modelo lineal

sim('Comparacion_Modelos');

figure;

plot(ans.Dist_Comp(34:end,1),ans.Dist_Comp(34:end,2)); % Gráfica de
la salida del modelo lineal
hold;
plot(ans.Dist_Comp(34:end,1),ans.Dist_Comp(34:end,3)); % Gráfica de
la salida del modelo no lineal

set(gca,'FontSize',15);title('Distancia "h" ante perturbación de 1
mV en la entrada', 'FontSize',30);
legend('Modelo Lineal', 'Modelo No Lineal', 'FontSize',20);
ylabel('Distancia "h" (m)', 'FontSize',20);xlabel('Tiempo
(s)', 'FontSize',20);xlim([0.65,1]);

```

## 2. Script de Matlab 'Control\_Continuo.m'

```
% Diseño de controladores continuos

run('Linealizacion'); % Ejecución del script de linealización
close all;

rltool(G); % Lugar de las raíces del proceso para diseño del
regulador

% Cálculo del regulador PD

K = -34.5;
zd = 2;
Td = 1/zd;

Kp = K*zd;
Kd = K*zd*Td;
Ki = 0;

% Simulación y resultados del regulador PD

sim('Regulador_Continuo'); % Simulación de la implementación del
regulador discreto

figure;
plot(ans.Dist_Continue(1:end,1),ans.Dist_Continue(1:end,2)); %
Gráfica de la salida del modelo lineal
hold;
plot(ans.Dist_Continue(1:end,1),ans.Dist_Continue(1:end,3)); %
Gráfica de la salida del modelo no lineal

title('Altura control PD');legend('Modelo Lineal','Modelo No
Lineal');
ylabel('Altura (m)');xlabel('Tiempo (s)');

% Cálculo del regulador PID

K = -25;
zd = 2;
zi = 0.4;
Td = 1/(zd+zi);
Ti = (zd+zi)/(zd*zi);

Kp = K*(zd+zi);
Kd = Kp*Td;
Ki = Kp/Ti;

% Simulación y resultados del regulador PID

sim('Regulador_Continuo'); % Simulación de la implementación del
regulador discreto

figure;
plot(ans.Dist_Continue(1:end,1),ans.Dist_Continue(1:end,2)); %
Gráfica de la salida del modelo lineal
hold;
```

```
plot(ans.Dist_Continue(1:end,1),ans.Dist_Continue(1:end,3)); %  
Gráfica de la salida del modelo no lineal  
  
title('Altura control PID');legend('Modelo Lineal','Modelo No  
Lineal');  
ylabel('Altura (m)');xlabel('Tiempo (s)');
```

### 3. Script de Matlab 'Control\_Discreto.m'

```
% Diseño de reguladores discretos

run('Linealizacion'); % Ejecución del script de linealización
close all;

% Diseño de un regulador algebraico mediante síntesis directa

syms K z p1 p2

pd = 0.85; % Posición del polo dominante
pr = 0; % Posición del polo restante

Gbc = K*((z + 3.597)*(z + 0.2531))/((z - pd)^2*(z - pr)); % Diseño
de Gbc con estos polos

T_eq = (z - pd)^2*(z - pr) - K*(z + 3.597)*(z + 0.2531) - (z -
1.498)*(z - p1)*(z - p2); % Ecuación 1 -Gbc
[v t] = coeffs(T_eq, z); % Obtención de las ecuaciones con los
coeficientes de z
sol = solve(v); % Solución de las ecuaciones

% Obtención de las variables

K = double(sol.K(1));
p1 = double(sol.p1(1));
p2 = double(sol.p2(1));

clear z

% Cálculo del regulador

z = tf('z',Ts);

Gbc = K*((z + 3.597)*(z + 0.2531))/((z - pd)^2*(z - pr));
GR = minreal((1/Gz)*(Gbc/(1-Gbc)),0.001);

% Simulación y resultados

sim('Regulador_Discreto'); % Simulación de la implementación del
regulador discreto

figure;
plot(ans.Dist_Discrete(1:end,1),ans.Dist_Discrete(1:end,2)); %
Gráfica de la salida del modelo lineal
hold;
plot(ans.Dist_Discrete(1:end,1),ans.Dist_Discrete(1:end,3)); %
Gráfica de la salida del modelo no lineal

set(gca,'FontSize',20);
title('Distancia "h" controlador
discreto','FontSize',30);legend('Modelo Lineal','Modelo No
Lineal','FontSize',15);
ylabel('Distancia "h"(m)','FontSize',25);xlabel('Tiempo
(s)','FontSize',25);ylim([0.3,0.32]);

figure;
```

```

plot(ans.u_modelos(1:end,1),ans.u_modelos(1:end,2)); % Gráfica de la
acción de control del modelo lineal
hold;
plot(ans.u_modelos(1:end,1),ans.u_modelos(1:end,3)); % Gráfica de la
acción de control del modelo no lineal

set(gca,'FontSize',20);
title('Acción de control controlador
discreto','FontSize',30);legend('Modelo Lineal','Modelo No
Lineal','FontSize',15);
ylabel('Acción de control (V)','FontSize',25);xlabel('Tiempo
(s)','FontSize',25);

% Diseño de un regulador algebraico mediante síntesis directa ep = 0

syms K z p1 p2 z1

pd = 0.925; % Posición del polo dominante
pr = 0.5; % Posición del polo restante

Gbc = K*((z + 3.597)*(z + 0.2531)*(z - z1))/((z - pd)^2*(z - pr)^2);
% Diseño de Gbc con estos polos

T_eq = (z - pd)^2*(z - pr)^2 - K*(z + 3.597)*(z + 0.2531)*(z - z1) -
(z - 1.498)*(z - 1)*(z - p1)*(z - p2); % Ecuación 1 -Gbc
[v t] = coeffs(T_eq, z); % Obtención de las ecuaciones con los
coeficientes de z
sol = solve(v); % Solución de las ecuaciones

% Obtención de las variables

K = double(sol.K(1));
z1 = double(sol.z1(1));
p1 = double(sol.p1(1));
p2 = double(sol.p2(1));

clear z

% Cálculo del regulador

z = tf('z',Ts);

Gbc = K*((z + 3.597)*(z + 0.2531)*(z - z1))/((z - pd)^2*(z - pr)^2);
GR = minreal((1/Gz)*(Gbc/(1-Gbc)),0.001);

% Simulación y resultados

sim('Regulador_Discreto'); % Simulación de la implementación del
regulador discreto

figure;
plot(ans.Dist_Discrete(1:end,1),ans.Dist_Discrete(1:end,2)); %
Gráfica de la salida del modelo lineal
hold;
plot(ans.Dist_Discrete(1:end,1),ans.Dist_Discrete(1:end,3)); %
Gráfica de la salida del modelo no lineal

set(gca,'FontSize',20);

```

```

title('Distancia "h" controlador discreto con acción
integral','FontSize',30);legend('Modelo Lineal','Modelo No
Lineal','FontSize',15);
ylabel('Distancia "h" (m)','FontSize',25);xlabel('Tiempo
(s)','FontSize',25);

figure;
plot(ans.u_models(1:end,1),ans.u_models(1:end,2)); % Gráfica de la
acción de control del modelo lineal
hold;
plot(ans.u_models(1:end,1),ans.u_models(1:end,3)); % Gráfica de la
acción de control del modelo no lineal

set(gca,'FontSize',20);
title('Acción de control controlador discreto con acción
integral','FontSize',30);legend('Modelo Lineal','Modelo No
Lineal','FontSize',15);
ylabel('Acción de control (V)','FontSize',25);xlabel('Tiempo
(s)','FontSize',25);

```



#### 4. Script de Matlab 'Estudio\_Distancia.m'

```
% Script para la obtención de la distancia máxima inicial de la bola
al imán
% para que se estabilize el sistema

run('Linealizacion'); % Ejecución del script de linealización
close all;

pd = 0.8; % Posición inicial de los polos dominantes

alturas = []; % Variable de resultados

for pd = 0.8:0.01:0.95 % Bucle para recorrer las posiciones de los
polos dominantes

pr = 0; % Posición inicial de los polos no dominantes

max_values = []; % Variables de alturas máximas para este polo
dominante

for pr = 0:0.05:0.8 % Bucle para recorrer las posiciones de los
polos no dominantes

syms K z p1 p2 z1

Gbc = K*((z + 3.597)*(z + 0.2531)*(z - z1))/((z - pd)^2*(z - pr)^2);
% Diseño de Gbc con estos polos

T_eq = (z - pd)^2*(z - pr)^2 - K*(z + 3.597)*(z + 0.2531)*(z - z1) -
(z - 1.498)*(z - 1)*(z - p1)*(z - p2); % Ecuación 1 -Gbc
[v t] = coeffs(T_eq, z); % Obtención de las ecuaciones con los
coeficientes de z
sol = solve(v); % Solución de las ecuaciones

% Obtención de las variables

K = double(sol.K(1));
z1 = double(sol.z1(1));
p1 = double(sol.p1(1));
p2 = double(sol.p2(1));

clear z

% Cálculo del regulador

z = tf('z',Ts);

Gbc = K*((z + 3.597)*(z + 0.2531)*(z - z1))/((z - pd)^2*(z - pr)^2);
GR = minreal((1/Gz)*(Gbc/(1-Gbc)),0.001);

clear K z p1 p2 z1

% Altura inicial

h_set = 0.300;
```

```

% Bucle para detectar la distancia máxima

maxed = 0; % Variable para detectar si se ha alcanzado la distancia
máxima
max_h = 0; % Variable de distancia máxima para esta iteración

while maxed == 0
    sim('Comprob_Distancia'); % Simula el modelo con 5 procesos
    Sim_Out = ans;

    % Comprobación del valor final de las 5 simulaciones
    if(round(Sim_Out.Salidas.signals(1).values(end),3) > 0.300)
        maxed = 1;
        max_h = h_set - 0.001;

    elseif(round(Sim_Out.Salidas.signals(2).values(end),3) > 0.300)
        maxed = 1;
        max_h = h_set;

    elseif(round(Sim_Out.Salidas.signals(3).values(end),3) > 0.300)
        maxed = 1;
        max_h = h_set + 0.001;

    elseif(round(Sim_Out.Salidas.signals(4).values(end),3) > 0.300)
        maxed = 1;
        max_h = h_set + 0.002;

    elseif(round(Sim_Out.Salidas.signals(5).values(end),3) > 0.300)
        maxed = 1;
        max_h = h_set + 0.003;
    end

    % Aumento de la altura inicial para la siguiente iteración
    h_set = h_set + 0.005;
end

% Guardado de la altura máxima de esta iteración
max_values = [max_values max_h];

end

% Guardado de las alturas máximas
alturas = [alturas; max_values];

end

% Creación del gráfico de superficie para representar los datos
calculados
surf(0:0.05:0.8,0.8:0.01:0.95,alturas);
set(gca,'FontSize',15);
xlabel('Par de polos no dominantes (plazo Z)','FontSize',18);
ylabel('Par de polos dominantes (plano Z)','FontSize',18);
zlabel('Distancia máxima al imán (m)','FontSize',18);

```

## 5. Script de Matlab 'Captura\_Datos.m'

```
% Script para la captura de datos de la maqueta virtual del
levitador magnético

% Fichero entrada datos (EDITAR AQUI EL NOMBRE DEL FICHERO DE DATOS
IMPORTADO)
fichero_entrada_datos = 'datos_controlador_N1.log';

% Lectura del fichero con los datos de lectura capturados
fid = fopen(fichero_entrada_datos); % Apertura del fichero
tline = fgetl(fid); % Obtención de la primera línea

% Parámetros para dividir la línea
DELIM1 = '****';
DELIM2 = '=';

% Variables para guardar los datos
time = [];
dist = [];
u = [];

while ischar(tline) % Bucle que recorre las líneas

    % Procesado de las líneas
    [TOKEN,REMAIN] = strtok(tline,DELIM1); % División de la línea en
'****'
    [parametro,valor2] = strtok(REMAIN, DELIM2); % División de la
línea en '=' separando la referencia y el valor de la línea
    valor = strtok(valor2, DELIM2); % División de la línea en '='
para obtener el valor sin el '='

    % Comprobación del parámetro y escritura en la variable
correspondiente
    if strcmp(parametro,'****t')
        time = [time str2num(valor)];
    end
    if strcmp(parametro,'****h_lev')
        dist = [dist str2num(valor)];
    end
    if strcmp(parametro,'****u_local')
        u = [u str2num(valor)];
    end

    % Actualización de la variable a la siguiente línea
    tline = fgetl(fid);
end

fclose(fid);

figure;
plot(time(1:99),dist(1:99)); set(gca,'FontSize',20); % Gráfica de la
distancia al imán
title('Distancia "h" en la maqueta virtual con el controlador
discreto','FontSize',30);
ylabel('Distancia "h" (m)','FontSize',25); xlabel('Tiempo
(s)','FontSize',25); xlim([0,5]);

figure;
```

```
plot(time(1:99),u(1:99)); set(gca,'FontSize',20); % Gráfica de la
acción de control
title('Acción de control en la maqueta virtual con el controlador
discreto','FontSize',30);
ylabel('Acción de control (V)','FontSize',25); xlabel('Tiempo
(s)','FontSize',25); xlim([0,5]);
```