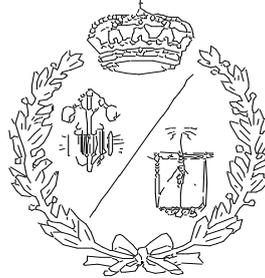


ESCUELA TÉCNICA SUPERIOR DE INGENIEROS
INDUSTRIALES Y DE TELECOMUNICACIÓN

UNIVERSIDAD DE CANTABRIA



Proyecto Fin de Grado

**CONTROL DE UN MANIPULADOR FLEXIBLE
DE UN ÚNICO SEGMENTO**

(Control of a single-link flexible manipulator)

Para acceder al Título de

**GRADUADO EN INGENIERÍA ELECTRÓNICA
INDUSTRIAL Y AUTOMÁTICA**

Autor: Jaime Capa López

Tutor académico: José Ramón Llata García

Mayo - 2022

INDICE

1. INTRODUCCIÓN	9
1.1. <i>Motivación</i>	10
1.2. <i>Objetivo</i>	13
2. ECUACIONES DINÁMICAS DEL SISTEMA	14
2.1. <i>Modelo dinámico del elemento flexible</i>	18
2.2. <i>Modelo matemático del robot</i>	26
3. CONTROL PID DE SIMPLE LAZO.....	30
4. CONTROL PID DE DOBLE LAZO.....	36
5. CONTROL GPI	39
5.1. <i>Modelo simplificado de parámetros concentrados</i>	41
5.2. <i>Modelo del conjunto Motor-Reductor</i>	43
5.3. <i>Modelo del Conjunto Motor-Reductor-Segmento Flexible y su Simulación</i>	45
5.4. <i>Diseño del Lazo Externo del Control GPI</i>	52
5.5. <i>Diseño del Lazo Interno del Control GPI</i>	55
5.6. <i>Simulación del conjunto con el Control GPI de Doble Lazo</i>	59
6. CONTROL FUZZY	64
6.1. <i>Diseño</i>	67
6.2. <i>Comprobación de resultados por simulación</i>	74
7. CONTROL CON REDES NEURONALES	78
7.1. <i>Diseño y simulación de un controlador de NN</i>	81
7.2. <i>Diseño y simulación de dos controladores NN</i>	85
8. REFERENCIAS BIBLIOGRÁFICAS	87

INDICE DE GRÁFICOS

Gráfico 1. Instalaciones anuales de robots industriales en los 15 mercados más grandes	11
Gráfico 2. Stock operativo de robots industriales en el mundo (miles de unidades).....	12

INDICE DE TABLAS

Tabla 1. Tabla FAM para 3 variables lingüísticas en cada bloque	70
Tabla 2. Tabla FAM para 5 variables lingüísticas en cada bloque	70
Tabla 3. Tabla FAM para 7 variables lingüísticas en cada bloque	70

INDICE DE FIGURAS

Figura 1. Vista esquemática de un manipulador de eslabones flexibles	15
Figura 2. Representación de la ecuación general de la flexión (Euler- Bernouilli)	21
Figura 3. Movimiento idealizado de un cuerpo elástico según D. Bernoulli	22
Figura 4. Movimiento del cuerpo elástico en caso de presencia de la fuerza de gravedad.	23
Figura 5. Clasificación de robots flexibles por el grado de deflexión.....	24
Figura 6. Mecanismo de un robot	25
Figura 7. Robot flexible con un grado de libertad	26
Figura 8. Ejemplo de controlador de temperatura en un CSTR	31
Figura 9. Diagrama de bloques de un sistema de control PID	32
Figura 10. Control PID de lazo abierto y en lazo cerrado.....	33
Figura 11. Diagrama general de resultados del proceso.....	34
Figura 12. Control de doble lazo en el posicionamiento preciso de los ejes lineales.....	37
Figura 13. Esquema típico de control de doble lazo.....	38
Figura 14. Diagrama de un brazo flexible de un único segmento	42
Figura 15. Diagrama de bloques en simulink de un eslabón flexible.	43
Figura 16. Primera parte del diagrama de bloques	44
Figura 17. Segunda parte del diagrama de bloques.....	44
Figura 18. Tercera parte del diagrama de bloques.....	44

Figura 19. Cuarta parte del diagrama de bloques	45
Figura 20. Diagrama de bloques de un eslabón flexible con servomotor.....	45
Figura 21. Parámetros introducidos en Matlab	47
Figura 22. Funciones de transferencia del motor y el eslabón introducidas en Matlab.....	48
Figura 23. Bloques y gráfica de una entrada impulso	48
Figura 24. Salidas del ángulo del motor y del ángulo del eslabón frente a una entrada impulso	49
Figura 25. Bloque y gráfica de una entrada escalón	50
Figura 26. Salidas del ángulo del motor y del ángulo del eslabón frente a una entrada escalón	50
Figura 27. Bloque y gráfica de una entrada rampa	51
Figura 28. Salidas del ángulo del motor y del ángulo del eslabón frente a una entrada rampa	51
Figura 29. Controlador lazo externo introducido en Matlab.....	55
Figura 30. Controlador lazo interno introducido en Matlab.....	58
Figura 31. Variables deseadas introducidas en Matlab.....	59
Figura 32. Diagrama de bloques completo en Simulink.	60
Figura 33. Valor de U_c deseada.	60
Figura 34. Valor de θ_m^* y θ_t^*	60
Figura 35. Valor de θ_t^* y de θ_t con un solo controlador	61
Figura 36. Valor de θ_t^* y de θ_t	61
Figura 37. Valor de θ_t^* frente a θ_t	62
Figura 38. Valor de θ_m^* frente a θ_m	63
Figura 39. Alineación mecánica del robot AIR-2 para el movimiento en superficies esféricas ...	66
Figura 40. Diagrama básico con un controlador de lógica difusa.....	67
Figura 41. Diagrama de un controlador de lógica difusa.	67
Figura 42. Diagrama de bloques con controlador de lógica difusa en simulink.	68
Figura 43. Entradas y salida del controlador Fuzzy en Matlab.....	68
Figura 44. Distintas funciones de pertenencia para las variables fuzzy.....	69
Figura 45. Diagrama de funciones para la variable Error.....	71
Figura 46. Diagrama de funciones para la variable dError.....	72
Figura 47. Diagrama de funciones para la variable de salida.....	73

Figura 48. Reglas de inferencia en Matlab.	74
Figura 49. Valor de la salida deseada frente a la salida obtenida.	75
Figura 50. Valor de la salida deseada frente a la salida obtenida de forma aumentada.	76
Figura 51. Valor del error a lo largo de la simulación.	77
Figura 52. Arquitectura simplificada de una RNA.	78
Figura 53. Datos obtenidos del controlador GPI.	81
Figura 54. Introducción de las variables en nuestra red neuronal.	82
Figura 55. Opciones de entrenamiento para nuestra red neuronal.	82
Figura 56. Gráficas de regresión de nuestra red neuronal.	83
Figura 57. Diagrama de bloques con nuestro controlador de redes neuronales.	83
Figura 58. Valor de la salida deseada frente a la salida obtenida.	84
Figura 59. Nuevos datos obtenidos del controlador GPI.	85
Figura 60. Diagrama de bloques con dos controladores de redes neuronales.	85
Figura 61. Valor de la salida deseada frente a la salida obtenida.	86

CONTROL DE UN MANIPULADOR FLEXIBLE DE UN ÚNICO SEGMENTO

RESUMEN

En aplicaciones de robótica es común utilizar elementos mecánicos y eslabones rígidos. Esto se realiza así especialmente porque simplifica enormemente el modelado matemático, así como la obtención de controladores dinámicos y cinemáticos. Todo esto conlleva el poder obtener manipuladores que permiten una elevada precisión en el movimiento y en el posicionamiento.

Sin embargo, cada día es más frecuente que los robots interactúen con los operadores humanos en diferentes tareas. Ejemplos de esto pueden encontrarse en las aplicaciones industriales donde los robots colaborativos tienen mucho éxito, pero también en aplicaciones médicas y de servicio a personas discapacitadas, donde un robot puede hacer tareas de atención que conlleven una interacción con la persona. Es en estos campos de interacción con las personas donde un robot que incorpore segmentos mecánicos flexibles, tales que el contacto con las personas sea totalmente inocuo, presenta un futuro de interés (además de las aplicaciones espaciales).

En el presente trabajo se analizarán y diseñarán distintos controladores basados en redes neuronales, lógica difusa y control GPI con el objetivo de evaluar su funcionamiento en un sistema que incluya eslabones mecánicos flexibles.

CONTROL OF A SINGLE-LINK FLEXIBLE MANIPULATOR

SUMMARY

In robotics applications it is common to use mechanical elements and rigid links. This is done especially because it greatly simplifies mathematical modeling, as well as obtaining dynamic and kinematic controllers. All this leads to manipulators that allow high precision in movement and positioning.

However, it is becoming increasingly common for robots to interact with human operators in different tasks. Examples of this can be found in industrial applications where collaborative robots are very successful, but also in medical and service applications for disabled people, where a robot can perform care tasks that involve interaction with the person. It is in these fields of interaction with people that a robot incorporating flexible mechanical segments, such that contact with people is completely harmless, presents a future of interest (in addition to space applications).

In this work, different controllers based on neural networks, fuzzy logic and GPI control will be analyzed and designed in order to evaluate their performance in a system including flexible mechanical links.

1. INTRODUCCIÓN

La demanda generada por la automatización industrial ha motivado el incremento en el uso de manipuladores robóticos en múltiples aplicaciones (Tokhi y Azad, 2008). Dentro del ámbito industrial existen multitud de ejemplos de aplicaciones de la ingeniería robótica, que van desde la soldadura, el corte, la manipulación, la pintura. etc. (Jaramillo, 2015).

Dentro de este campo, los robots flexibles han generado desde hace varias décadas un campo de investigación en el que se han producido importantes aportaciones, que han desembocado en el desarrollo de la industria aeroespacial, entre otras (Feliu, 2006).

La robótica industrial ha permitido mejorar la capacidad de adaptación de las líneas de fabricación en la industria, logrando mejorar las líneas de producción mediante cambios simples, rápidos, minimizando los retrasos o los costes que después son trasladados al producto final (*International Federation of Robotics Frankfurt*, 2018). Es en este caso, donde autores como Spöng Hutchinson y Vidyasagar (2005), resaltan la ventaja que supone la robótica industrial sobre otras formas de automatización, puesto que los robots tienen una capacidad de adaptación mayor, simplemente con la implementación de cambios en la programación del robot.

En los últimos años, la adaptación de los sensores a la robótica, ha permitido mejorar la flexibilidad de los mismos al dotarles de cierta inteligencia, de manera que se adaptan a su campo de trabajo, tanto de manera parcial como total. El uso de robots industriales con diseño de sistemas asistidos por ordenador (CAD) y los sistemas de fabricación asistidos por ordenador (CAM) son una tendencia en automatización de los procesos de fabricación que han contribuido a mejorar la automatización industrial y que en el momento actual, presentan un grado de desarrollo con un alcance aún desconocido (Barrientos, 2007).

La robótica es sinónimo de progreso y desarrollo tecnológico, y aquellos países y organizaciones que cuentan con una presencia importante de robots en sus métodos de producción, alcanzan niveles de competitividad y productividad más elevados. Por este motivo, las inversiones en robótica se encuentran en aumento. Dentro del sector industrial, la robótica ha sido implementada con éxito desde hace varias décadas, y entre sus beneficios se encuentran la mejora de la productividad, de la flexibilidad, la calidad y la seguridad, pues se minimiza la presencia de personas en aquellos procesos productivos que son más peligrosos (Comité Español de Automática, 2015).

Los robots manipuladores industriales se han convertido en máquinas de uso general que permiten una mejora de la productividad, la flexibilidad y la calidad de los productos, el ahorro de los costes de producción y la eliminación de trabajos peligrosos y desagradables. Es por ello por lo que el control del movimiento de dichos robots se convierte en una competencia clave para los fabricantes, y la principal razón para mejorar el rendimiento de los mismos a través de la mejora de los modelos matemáticos y de los medios de control (Moberg, 2010).

Argumentos como el comentado, que se basa en la importancia del desarrollo de la robótica, junto a los expuestos a continuación, son los que justifican la realización de un trabajo con el que se pretende profundizar en el conocimiento del desarrollo de la robótica flexible. Más concretamente, en el presente trabajo se abordan diferentes aspectos relacionados con el modelado y control de manipuladores flexibles.

1.1. Motivación

De acuerdo con los datos publicados por la *International Federation of Robotics* (en adelante IFR) en el Informe Mundial de Robótica 2020, en el año 2019 se alcanzó la cifra de 2,7 millones de robots trabajando en fábricas de todo el mundo, y se alcanzó el nivel más alto de la historia en el “stock de robots industriales que operan en fábricas de todo el mundo” (IFR, 2020, p.1).

Además, las “ventas de nuevos robots se mantienen en un nivel alto con 373.000 unidades enviadas a nivel mundial en 2019. Esto es un 12% menos en comparación con 2018, pero sigue siendo el tercer volumen de ventas más alto jamás registrado” (IFR, 2020). Si se tienen en cuenta los datos por áreas geográficas, Asia sigue siendo el mercado más importante en lo referente a los robots industriales, de manera que, las existencias operativas en China han aumentado un 21%, llegando a las 783.000 unidades en el año 2019. En segundo lugar se encuentra Japón, con 355.000 unidades, equivalentes al 12%, como se muestra en el siguiente gráfico:

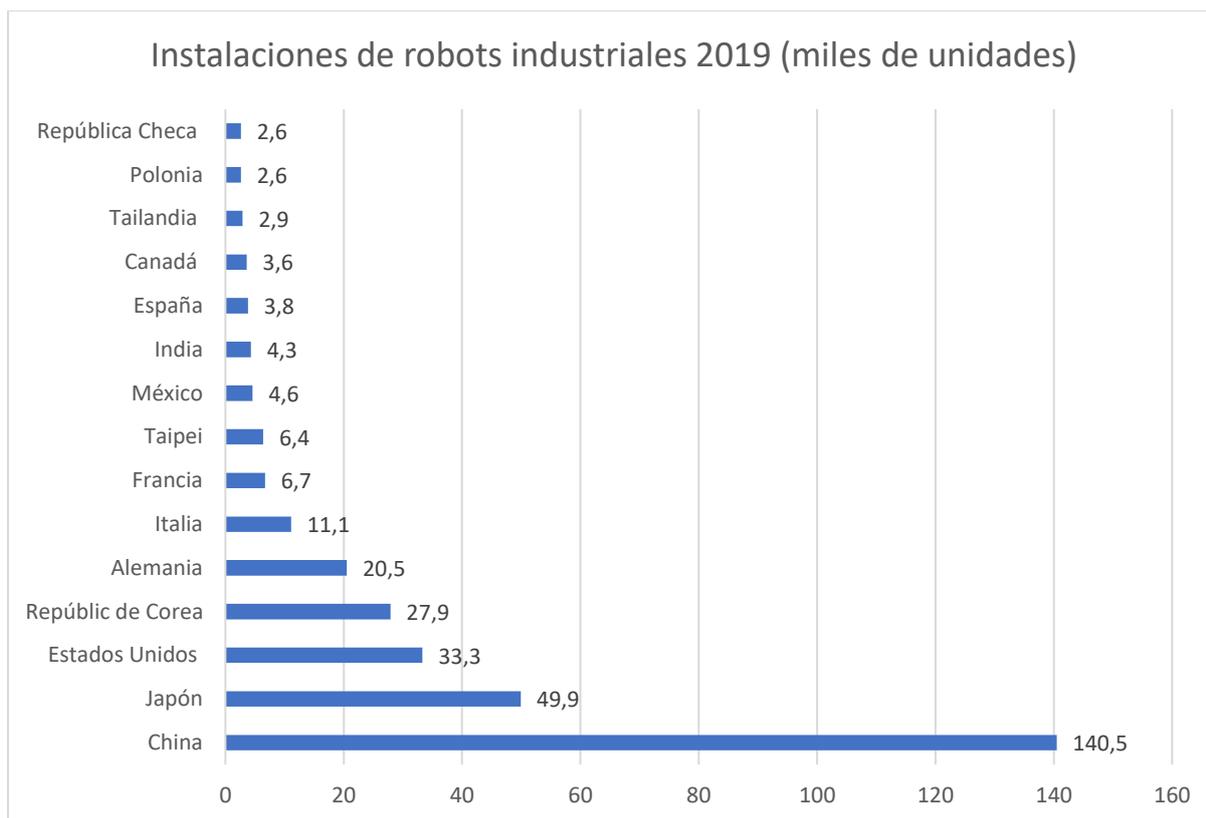


Gráfico 1. Instalaciones anuales de robots industriales en los 15 mercados más grandes

Fuente: IFR (2020)

En el caso de Europa, el stock de robots operativos en el año 2019 era de 580.000 unidades, lo que supone un aumento del 7% con respecto a los datos del año anterior. A la cabeza se encuentra Alemania, con 221.500 unidades instaladas, equivalentes al triple de las unidades de Italia (74.400 unidades), el quíntuple de las de Francia (42.000 unidades) y diez veces más que en el Reino Unido (21.700 unidades).

Como se observa en el siguiente gráfico, donde se comprueba el crecimiento de la robótica a partir de 2009 ha sido notable, tanto la robótica, como las tecnologías de vanguardia que están relacionadas con el campo de la robótica, el aprendizaje automático, la Inteligencia Artificial (en adelante IA), la colaboración hombre-máquina o los sistemas móviles autónomos. De hecho, los robots inteligentes son considerados como una parte crucial de la digitalización de la industria manufacturera (IFR, 2020).

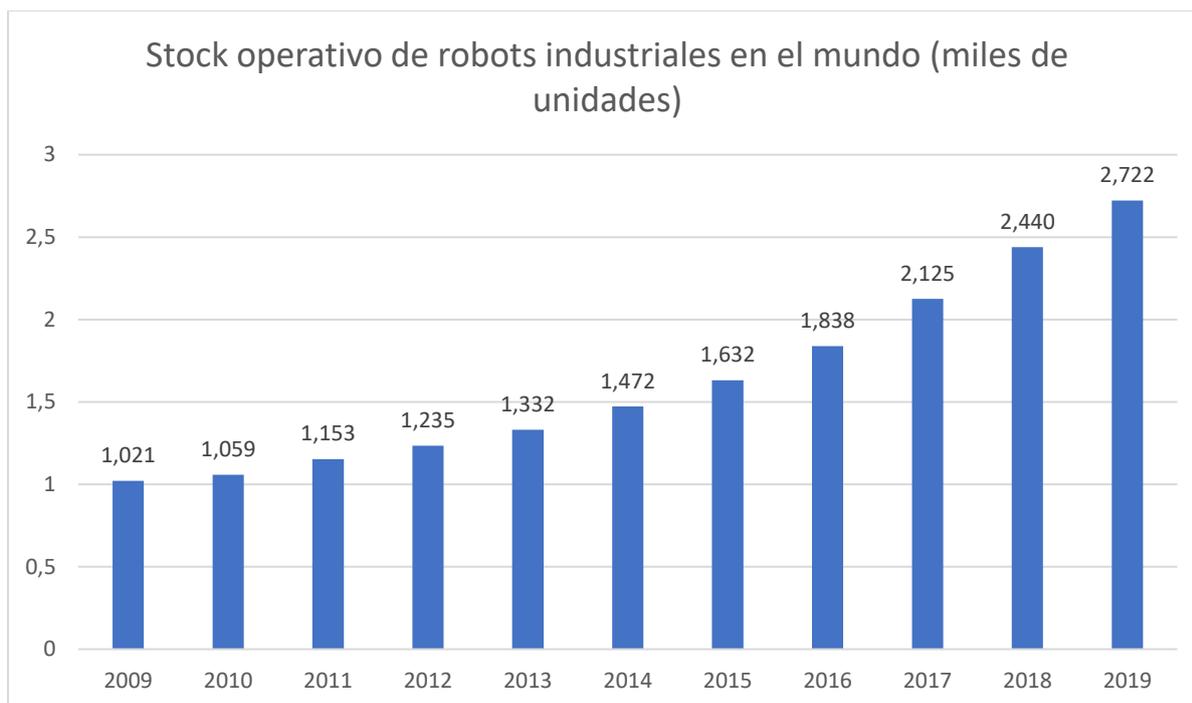


Gráfico 2. Stock operativo de robots industriales en el mundo (miles de unidades)

Fuente: IFR (2020)

El uso cada vez mayor de manipuladores robóticos en los últimos años ha estado dictado por la propia automatización de la producción industrial, y por las ventajas de éstos respecto de los manipuladores rígidos, dando lugar al mismo tiempo a una nueva línea de investigación en torno a la dinámica de flexión. De hecho, otro de los argumentos que justifican la realización de este trabajo reside en la importancia alcanzada dentro del espacio de investigación y académico, de esta línea de trabajo.

Entre la producción científica más reciente se encuentran trabajos como los realizados por Ramalingam y Mohideen (2019), en el que realizan una revisión de la literatura sobre los modelos dinámicos y de simulación de manipuladores robóticos flexibles en el periodo comprendido entre los años 2006 y 2015, donde concluyen que una de las líneas de investigación más destacadas es la centrada en la investigación de diferentes materiales compuestos, nanomateriales y aleaciones con memoria de forma.

En la misma línea de revisión de la literatura en torno a la flexibilidad de la robótica, se encuentra el trabajo publicado por Rahimi y Nazemizadehb (2014), en el que profundizan en el análisis dinámico y en las técnicas de control inteligente de los manipuladores robóticos flexibles, en total 115 artículos correspondientes al periodo comprendido entre los años 1970 y 2013, en el que concluyen que el uso del modelo de elementos finitos para abordar la flexibilidad, suele generar una matriz de rigidez, en la que como resultado del gran número

de ecuaciones, el tiempo de simulación numérica es elevado para los modelos de elementos finitos. Además, los controles heurísticos que incorporan la lógica difusa y la red neuronal que no necesitan un conocimiento a priori de la dinámica del sistema, son áreas emergentes de investigación.

El interés de la investigación por los manipuladores flexibles se observa también en la amplia gama de aplicaciones, desde las aplicaciones en la industria, hasta las aplicaciones microquirúrgicas, mantenimiento de centrales nucleares o la robótica espacial (Dwivedy y Eberhard, 2006). Para aprovechar al máximo el potencial que ofrecen los manipuladores flexibles se ha desarrollado una abundante producción bibliográfica que también exponen Valizadeh y Shariatee (2021), para quienes además de la flexibilidad de los robots, otro de los temas de investigación en auge es el de los materiales flexibles y las posibles limitaciones derivadas de la deformación de los eslabones de los manipuladores.

Las ventajas de esos manipuladores flexibles han sido analizadas por Sayahkaraj, Mohamed y Ahmad Athif Mohd Faudzi (2016), para quienes realizar una revisión de una “literatura tan vasta y variada” es imprescindible y de gran utilidad para los investigadores, en tanto en cuanto, éstos pueden adaptar sus intereses al área concreta de análisis.

Como se desprende de lo indicado a lo largo de este apartado, existen motivos que justifican la realización de un trabajo como el que se plantea, tanto por cuestiones académicas, como de desarrollo y crecimiento industrial.

1.2. Objetivo

El objetivo del presente trabajo está totalmente relacionado con este tipo de manipuladores, y consiste en diseñar un sistema de control dinámico para un manipulador flexible de un único elemento. Se realizará un estudio, análisis y diseño de elementos de control para un manipulador de un único segmento, siendo este flexible, actuado por un motor de corriente continua, y se atenderá especialmente el efecto de las no-linealidades en el motor por fricción de Coulomb. Así, será uno de los principales objetivos del sistema de control el amortiguar la vibración del manipulador flexible de forma que se posicione adecuadamente en la posición requerida sin las oscilaciones propias de estos elementos flexibles, atendiendo así mismo a la no linealidad de la fricción del rotor. Con este objetivo final, se analizarán varias técnicas de control.

2. ECUACIONES DINÁMICAS DEL SISTEMA

Un robot flexible se caracteriza por ser ligero, presentar una elevada flexibilidad y ser capaz de soportar una gran carga (Flores et al., 2014). Mediante el diseño de la estructura, y más concretamente de la flexibilidad, es posible optimizar el robot (Patane et al., 2017).

No obstante, y teniendo en cuenta que el robot debe cumplir con una serie de condiciones para ser aplicado en trabajos reales, el diseño e implementación de ese método de control está sometido a diferentes dificultades (Kaouk, Haber y Autorino, 2014).

En el entorno real, o dicho de otra forma, en la práctica, el cambio de rigidez o la deformación elástica, son absolutos, lo que hace que la consideración de la dinámica de los robots flexibles se realice en función de factores integrales de deformación y control (Xu et al., 2020). De esta forma, tanto el modelado dinámico, como el análisis de la flexibilidad, se convierten en cuestiones esenciales (Zhang y Zhaohui, 2021).

El manipulado flexible se emplea de manera casi generalizada en la industria aeroespacial, entre otros campos. En este aspecto, la precisión de los robots se ve afectada por la flexibilidad, la fricción de las articulaciones y la carga terminal (Zhang y Zhaohui, 2021), motivo por el cual, el modelado dinámico resulta ser una de las piezas fundamentales en el diseño de robots flexibles.

Además, la importancia de lo anterior se hace evidente en el modelado de los manipuladores, pues los manipuladores flexibles presentan una serie de ventajas sobre los manipuladores rígidos (Cao, et al., 2020). Al mismo tiempo, los manipuladores flexibles presentan una serie de limitaciones, como la complejidad de control, la incertidumbre, los sistemas MIMO y los sistemas no lineales, el desbordamiento de control, o el desbordamiento observado entre otros (Xin, 2011).

La selección de los modelos dinámicos se convierte, por tanto, en la decisión más compleja (Spector y Flashner, 1990) y muestra de ello es la abundante bibliografía de carácter académico que se ha publicado en los últimos años, centrada en el diseño de métodos discretos para robots flexibles, que incluyen el método de elementos finitos (FEM), el método de modo asumido (AMM) y el método de parámetros agrupados (Zhang, Feng y Xu, 2004).

Los métodos de modelado dinámico generalmente se establecen en base a métodos AMM y Lagrangianos. Estos métodos tienen las ventajas de una alta eficiencia computacional y una selección flexible de las condiciones de contorno (Wang y Gao, 2003)

Autores como Fukuda y Arakawa (1987) fijaron las características dinámicas del manipulador flexible de dos enlaces bajo la influencia de la carga útil y la gravedad, aunque solo analizaron el efecto de la vibración de flexión en el manipulador. Años más tarde, Bufinton (1992) analizó el miembro de traslación del brazo estándar como la viga elástica y estableció la ecuación de movimiento que contiene el movimiento de traslación del miembro elástico del brazo estándar. Usando la amplitud de modo variable finita y la función de modo de característica espacial, estableció además la ecuación dinámica.

En el año 2013 Rahimi y Nazemizadeh crearon una nueva metodología para seleccionar las condiciones de control contorno de enlace y sus funciones propias modales. Además, utilizaron un modelo dinámico para discutir su desviación en términos de modos de vibración de configuración.

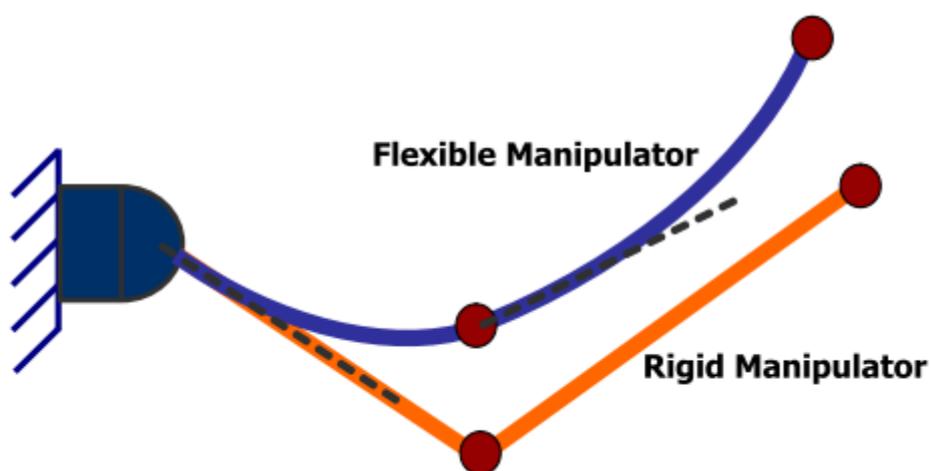


Figura 1. Vista esquemática de un manipulador de eslabones flexibles

Fuente: Rahimi y Nazemizadeh (2013)

En fechas similares Zhang y Bai (2012) formularon un método o modelo dinámico de la carga útil combinada utilizando el método de modo asumido (AMM), e incorporando verificaciones numéricas y experimentales.

Alguno de los estudios posteriores sobre el modelado dinámico, son los de Scaglioni y Ferretti (2015), en el que elaboran un modelo dinámico de tipo cerrado de un robot flexible basándose en Newtons- Euler, y mediante Matlab- Simulink, Modelica- Dymola, y MSC-Adams, entre los software empleados.

Khairudin et al. (2010) modelaron y llegaron a construir un robot flexible de dos enlaces como base para el desarrollo del modelo dinámico a partir de la influencia de la inercia y la carga

en el eje amortiguador de la estructura. Pradhan y Subudhi (2011) aplicaron mínimos cuadrados extendidos recursivos (RELS) con el objetivo de conseguir identificar en tiempo real el modelo NARMA, para lo cual, aplicaron diferentes cargas.

Khairudin et al., (2014) elaboraron un modelo dinámico y un manipulador flexible plano de dos enlaces TLFM con carga útil, aplicando el modelo de Euler-Lagrange y el modelo AMM para definir el modelo dinámico del manipulador, que fue verificado mediante simulación numérica y experimentación.

La fuente de energía del movimiento en un robot depende principalmente del momento de conducción de la articulación, en el que se debe considerar la pérdida de energía del robot. Muchos estudios se centran principalmente en el estudio de la pérdida por fricción articular. En la actualidad, los modelos de fricción más comunes son el modelo de fricción Stribeck, el modelo de fricción Coulomb-Viscous, el modelo mane y el modelo de fricción Coulomb (Li, Liu y Li, 2008).

Autores como Alinalbú et al (2001), aplicaron los modelos de fricción de Coulomb y también de fricción estática, con el objetivo de describir el fenómeno de fricción del movimiento articular, y establecer un modelo dinámico teniendo en cuenta tanto la flexibilidad como la fricción de las articulaciones. Por su parte, Moreno y Kelly (2005), utilizaron el modelo de manipulador con dos grados de libertad y, a partir de éste, configuraron un modelo dinámico de manipulador viscoso friccional. También analizaron la influencia de la fricción en las articulaciones sobre las características dinámicas y los errores de posición final del manipulador.

El modelo de manipulador de LuGre fue aplicado por He, Mei y Fang (2017), para establecer un modelo de dinámica articular con fricción y comprobar cuál era el consumo de energía a partir de esa fricción en las articulaciones del robot. Más tarde, Wu, Zhu y Wang, (2013), ampliaron el diseño a un modelo extendido de fricción articular.

Para resolver el problema que plantea la fuerza de fricción, Liu, Liu y Feng, (2014) empearon el modelo de fricción de Coulomb para analizar el efecto de la fricción articular en la fuerza motora de la articulación manipuladora espacial, alcanzando la conclusión de que la gravedad ejerce una importante influencia en las características dinámicas del manipulador.

También partiendo de la teoría de la fricción de Coulomb y el algoritmo multiplicador de Lagrangian, los investigadores Dou y Yue (2018), desarrollaron un modelo de fricción con

estados deslizantes y viscosos y estudiaron la pérdida de energía provocada por la fricción en el sistema de colisión.

Los estudios e investigaciones mencionados, permiten concluir que en general, el modelo de fricción de Coulomb permite resolver el problema de la articulación del manipulador, siendo la fuerza de fricción proporcional a la carga normal, opuesta a la dirección del movimiento relativo, e independiente de la teoría de la fricción del área de contacto. Además, este modelo de fricción de Coulomb resulta relativamente intuitivo y fácil de implementar (Mahamood y Pedro, 2011).

Tras esta breve revisión se puede afirmar que son muchas las investigaciones que han abordado el análisis de los métodos de control de robots flexibles a partir de métodos de control clásicos de robots rígidos tradicionales. Posteriormente, se realizaron otras pruebas, como por ejemplo el análisis del control para diferentes cargas útiles. Mohamed et al., (2014), emplearon un controlador de desigualdad de matriz lineal (PD LMI), para controlar la posición y la supresión de vibraciones finales del manipulador flexible de dos enlaces bajo la condición de cambiar el efecto de carga, donde se constatan la robustez de la tecnología de control a través de la comparación entre la simulación numérica y los resultados del experimento.

Walsh y Richard (2015), elaboraron un modelo de control de un robot escalable y flexible en el que el control se ejercía a través de un controlador PD sobre la posición terminal del manipulador. Una metodología diferente fue la empleada por Matsumo y Yamamoto (2007). pues aplicaron tecnología de perturbación singular para separar el sistema dinámico en dos partes: por un lado, el subsistema lento y por otro lado, el subsistema rápido. Y adaptaron el control de retroalimentación de estado, para controlar la vibración al final del enlace flexible.

Sawada e Itamiya (2012), utilizaron el método de par calculado, o la extensión linealizada de retroalimentación, para controlar la posición del robot flexible, mientras que Vandini et al., (2014), trataron de controlar el movimiento flexible de dos enlaces, aplicando un sensor visual que genera señales de retroalimentación, mientras que en este caso, los controladores PID son usados para controlar el movimiento de los robots flexibles. Chu y Cui (2015), trataron de suprimir la vibración interna del manipulador flexible de dos enlaces para adaptarlo a la variación de los parámetros del modelo, que estaban formados por un modelador de entrada y una retroalimentación de posición positiva adaptativa multimodo.

2.1. Modelo dinámico del elemento flexible

El modelado de los sistemas robóticos flexibles ha sido un reto para los investigadores a lo largo de las últimas décadas (Filipovic, 2008). Autores como Book, Maizza y Whitney (1976) consideran que el control de robots con articulaciones elásticas en contacto con un entorno dinámico, se forma a través de eslabones elásticos, esto es, sistemas de dos vigas y dos articulaciones.

En Hughes (1977) se empleó un sistema de control de la retroalimentación del robot, con dos eslabones elásticos (sistemas de dos vigas y dos articulaciones), con flexibilidad distribuida. Mientras que Moalhem, Khorasani y Patel (1996), presentaron una estrategia de control no lineal para el seguimiento de la trayectoria de la posición de la punta de unos manipuladores multienlace flexibles.

Matsuno, Wakashiro e Ikeda (1994), y Matsumo y Kanzawa (1996), derivaron ecuaciones dinámicas del ángulo de la articulación, la vibración del brazo flexible y la fuerza de contacto; y Yim (1996), presentó una aproximación al control del punto final de los manipuladores elásticos basándose en la Teoría del control predictivo no lineal. Autores como Surdilovic y Vukobratovic (1996a, 1996b) presentaron un método para la generación de modelos cinemáticos y dinámicos eficientes, de robots flexibles.

Spong (1989) discutió el problema del control de fuerza de los manipuladores, y el propio Spong, junto con Ghorbel (1992), extendieron el problema de la matriz integral para el control de manipuladores de robots flexibles en el caso de parámetros conocidos al caso adaptativo. En Krishnan (1995), el autor establece una ley de control para la regulación local de los vectores de fuerza de contacto y posición a vectores constantes deseados.

Frente a los enfoques convencionales Cheong, Chung y Youm (2000) propusieron un diseño del control del movimiento de la pieza rígida y la selección del ancho de banda del subsistema rígido. Una derivación de las ecuaciones de movimiento para manipuladores mecánicos de aplicación con eslabones elásticos, fue la propuesta realizada por Low (1988). Mientras que el propio Low y Vidyasagar (1988) establecieron un sistema en el que las ecuaciones se derivan empleando el principio de Hamilton, por lo que son ecuaciones diferenciales integradas no lineales, en las que la formulación se basa en la expresión de las energías cinética y potencial del sistema manipulador.

El método de separación de variables y el enfoque de Galarkin fueron aplicados por Low (1989) para resolver el problema de valor límite con condición de contorno dependiente del tiempo.

Otros autores, como De Luka y Siciliano (1991), Jang, Krishnan y Ang, (1997), De Luca (2000), Cheong, Chung y Youm (2002), Khadem y Pirmohammadi (2003), y Meirovitch (1967) han estudiado la solución general del movimiento de un sistema robótico elástico, considerando las deformaciones elásticas como oscilaciones transversales que pueden ser determinadas por el método de las integrales particulares de Bernoulli. De acuerdo con esto, la deformación elástica de un cuerpo, puede ser provocada por:

- Fuerzas perturbadoras que provocan el comportamiento oscilatorio del movimiento.
- Fuerzas estacionarias causantes de la naturaleza estacionaria del movimiento.

De esta forma, cualquier perturbación elástica se puede presentar superponiendo las soluciones de carácter oscilatorio de Bernoulli y soluciones estacionarias de carácter forzado.

En la mayor parte de las propuestas, se entiende que todas las características de elasticidad en el sistema, tanto la rigidez, como la amortiguación, son "conocidas", o al menos lo son en parte, y que por tanto, pueden incluirse en el proceso de definición del movimiento de referencia (Filipovic, 2009, p. 5691).

La ecuación de Euler-Bernoulli (basada en las leyes conocidas de la dinámica) debe complementarse con todas las fuerzas que participan en la formación del momento de flexión del modo considerado. La matriz de rigidez es una matriz completa. El amortiguamiento es una característica de elasticidad omnipresente de los sistemas reales, por lo que se incluye naturalmente en la ecuación de Euler-Bernoulli. Sin embargo, la integral particular de Daniel Bernoulli es solo una componente de la deformación elástica total de la punta, a la que es necesario agregar una componente de la deformación elástica de un régimen estacionario de acuerdo con los requisitos de complejidad del movimiento de un sistema de robot elástico (Filipovic, 2009).

El modo de enlace de la ecuación de la línea elástica de un sistema de robot elástico complejo se define de acuerdo con el conocido como "Enfoque de Euler-Bernoulli" (en adelante ABE), en el que se muestra que la ecuación de equilibrio de todas las fuerzas presentes en el punto de punta del modo ("enfoque de masa concentrada" o LMA), se deriva directamente de la ecuación de la línea elástica para condiciones de contorno especificadas (Filipovic, 2009).

La ecuación de Euler Bernouilli y la solución planteada por ésta, han sido aplicadas a la literatura científica (Kim, Siuzuki y Konno, 1996; Khadem y Pirmohammadi, 2003) y definida por Meirovitch (1967). Desde que en 1750 se publicara la ecuación de Euler Bernouilli hasta el momento actual, ha sido utilizada como base del conocimiento, especialmente en el campo de la robótica, la teoría de la oscilación y la teoría de la elasticidad, dando lugar a extensiones de la ecuación de Euler Bernouilli.

Cualquier deformación elástica puede presentarse superponiendo las soluciones particulares de Bernouilli, como se puede comprobar en los trabajos de Filipovic y Vukobratovic (2005); Filipovic y Vukobratovic (2006); Filipovic y Vukobratovic (2008a) y Filipovic y Vukobratovic (2008b).

Meirovitch (1967) definió la “técnica de los modos asumidos”, que ha sido utilizada a lo largo de las últimas décadas para formar la ecuación de Euler Bernouilli, si bien, en el caso del trabajo de Filipovic (2009), no es aplicada dado que la autora entiende que dicha técnica, aunque puede ser útil en otras áreas de investigación, se utiliza de manera errónea en la robótica, la teoría de las oscilaciones y la teoría de la elasticidad. En este trabajo, se parte de la premisa de que la deformación elástica y la frecuencia circular de cada modo de elemento elástico son consecuencia del movimiento dinámico global del sistema robótico.

Filipovic (2009) propone una solución en la que el modelo matemático incluye en su raíz la posibilidad de analizar simultáneamente los dos fenómenos presentes, esto es, la elasticidad de los engranajes y la flexibilidad de eslabones, además de la idea propuesta por Book y Majette (1983), teniendo en cuenta que el objetivo que se plantea es el de plantear una investigación que se dirija a la aplicación de la elasticidad de los engranajes y la flexibilidad de los eslabones en cualquier modelo de robot rígido y también en el modelo de robot rígido reconfigurable que planteaban Djuric y ElMaraghy (2007).

Como paso previo a la exposición práctica del trabajo, es importante reflejar, al menos de manera sintética, la ecuación general de la flexión de Euler- Bernouilli:

$$\frac{d\theta}{ds} = - \frac{M(s)}{EI}$$

donde,

θ es la pendiente

s es la longitud del arco

M el momento flector

E el módulo de young

I es el momento de inercia.

De manera gráfica, la representación es la siguiente, la cual se refiere a la deflexión estática de una segmento empotrado por un extremo y sometido a una fuerza en el otro:

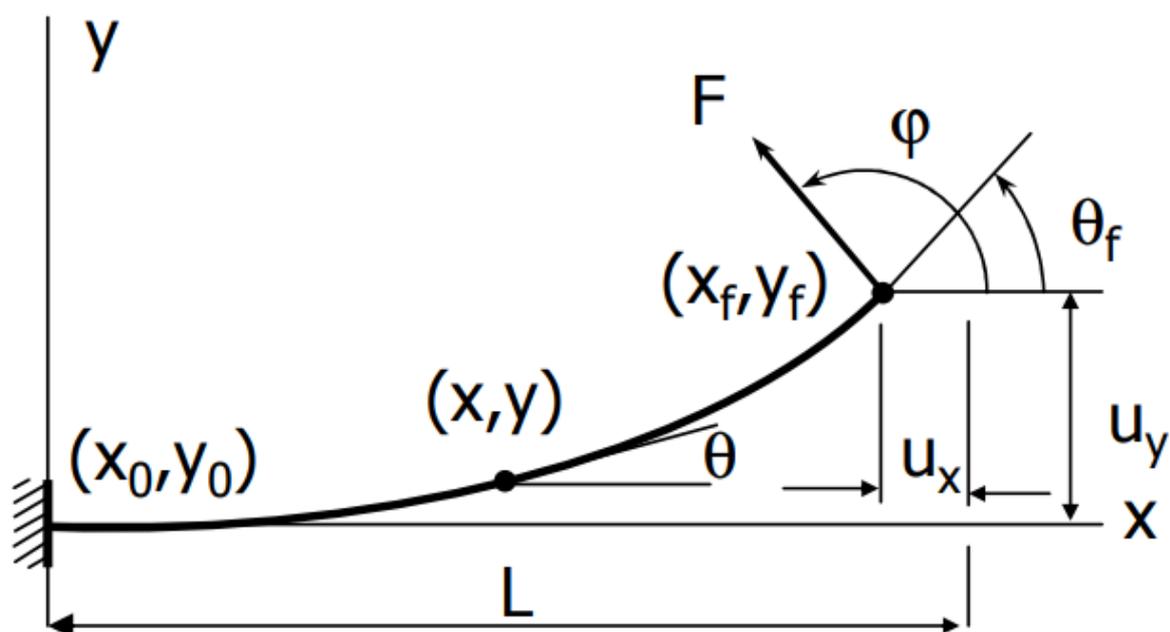


Figura 2. Representación de la ecuación general de la flexión (Euler- Bernouilli)

Fuente: Feliu (2007)

No obstante, esta ecuación como indica Feliú (2007, p. 14), “no puede ser resuelta de una manera exacta y necesita del cálculo de algunas integrales elípticas que pueden ser evaluadas utilizando métodos de integración numérica”.

Entre los trabajos que han evaluado la deflexión de las barras o eslabones a través de métodos de integración numérica, se encuentran los de Wang (1981) y de Beléndez et al., (2002) para eslabones de sección transversal fija y el trabajo de Lee (1993) para eslabones con sección transversal variable (Feliú, 2009).

En aquellos casos en los que las deformaciones son pequeñas, esto es, cuando $dy/dx = 0$, la ecuación general de la flexión puede ser calculada mediante la expresión

$$\frac{d^2y}{dx^2} = -\frac{M(s)}{EI}$$

donde x e y son las coordenadas de un punto de la viga expresadas en un sistema cartesiano que rota con la articulación del motor.

Bernoulli supuso la posición horizontal del cuerpo observado como su estado estacionario, que en la siguiente figura puede observarse en el eje de las abscisas:

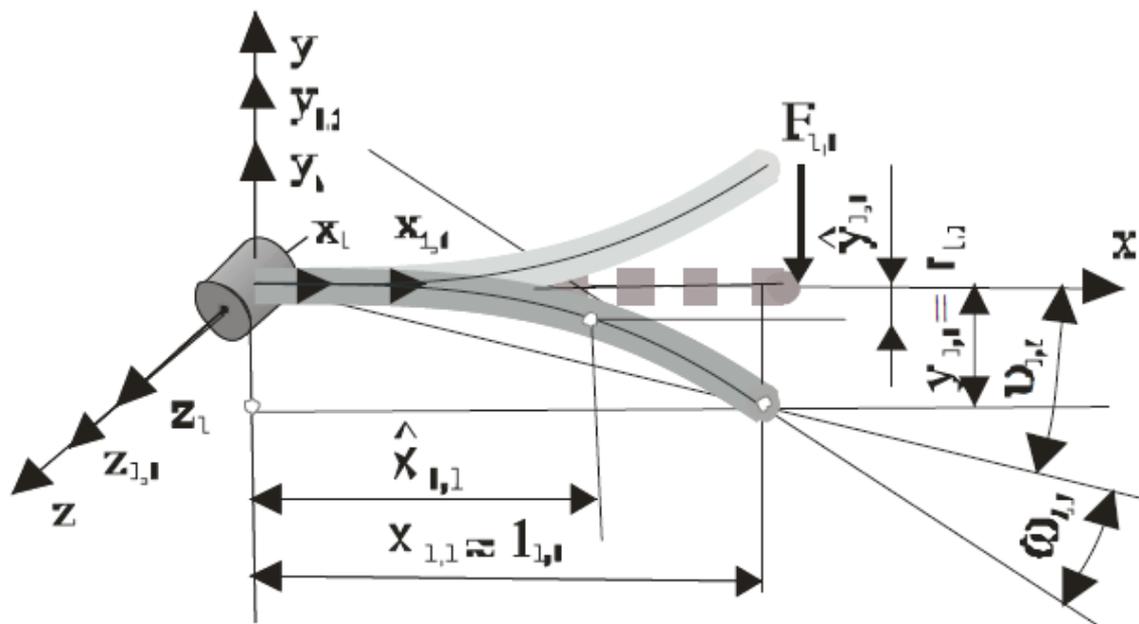


Figura 3. Movimiento idealizado de un cuerpo elástico según D. Bernoulli

Fuente: Filipovic (2009)

Bajo la suposición anterior, las oscilaciones ocurren justo alrededor del eje de abscisas. En el caso de que Bernoulli hubiera incluido la fuerza de la gravedad (G) en el modelo, la situación habría sido más real, pues en ese caso, la posición estacionaria del cuerpo inmóvil no habría coincidido con la posición del eje de abscisas, sino que la posición del cuerpo habría sido un poco más baja y las oscilaciones se habrían producido en torno a la nueva posición estacionaria, como se muestra en la siguiente figura (Filipovic, 2009):

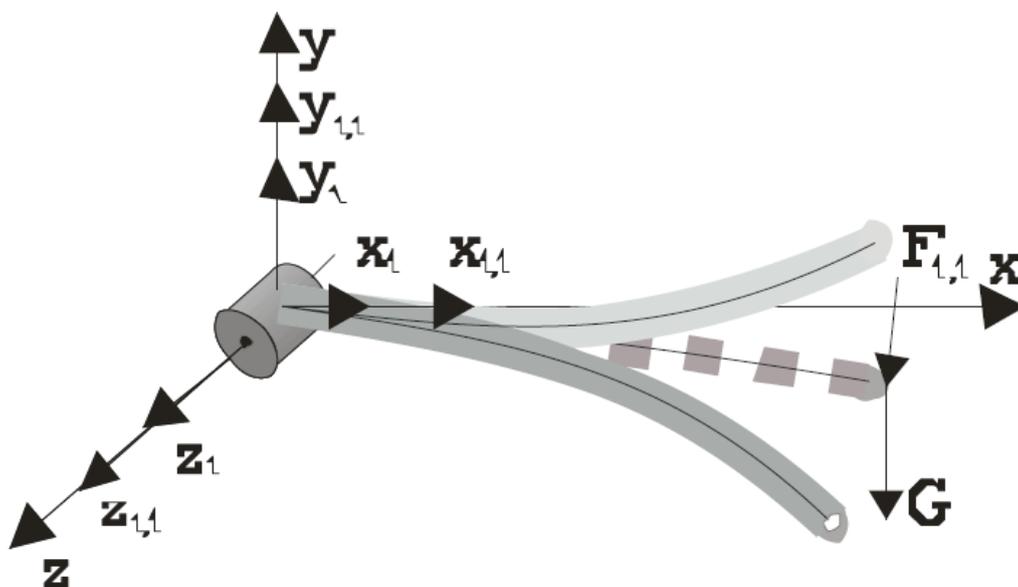


Figura 4. Movimiento del cuerpo elástico en caso de presencia de la fuerza de gravedad.

Fuente: Filipovic (2009)

De esta forma, en la siguiente figura se muestra la clasificación de los robots flexibles de acuerdo con las fuerzas normalizadas que sufren en el movimiento habitual, de acuerdo con las estimaciones publicadas en diferentes investigaciones, a partir de una clasificación en tres categorías, esto es, los valores de $F_n > 0.5$ (robots construidos de materiales metálicos en los que la aproximación lineal de la deflexión es válida); un segundo grupo cuyos valores pertenecen al rango $0,5 < F_n < 3$ (robots construidos con materiales compuestos en los que la aproximación no es válida); y un tercer grupo, en el que $3 < F_n < 12$, correspondiente robots extremadamente sensibles (Feliú, 2009, p. 31):

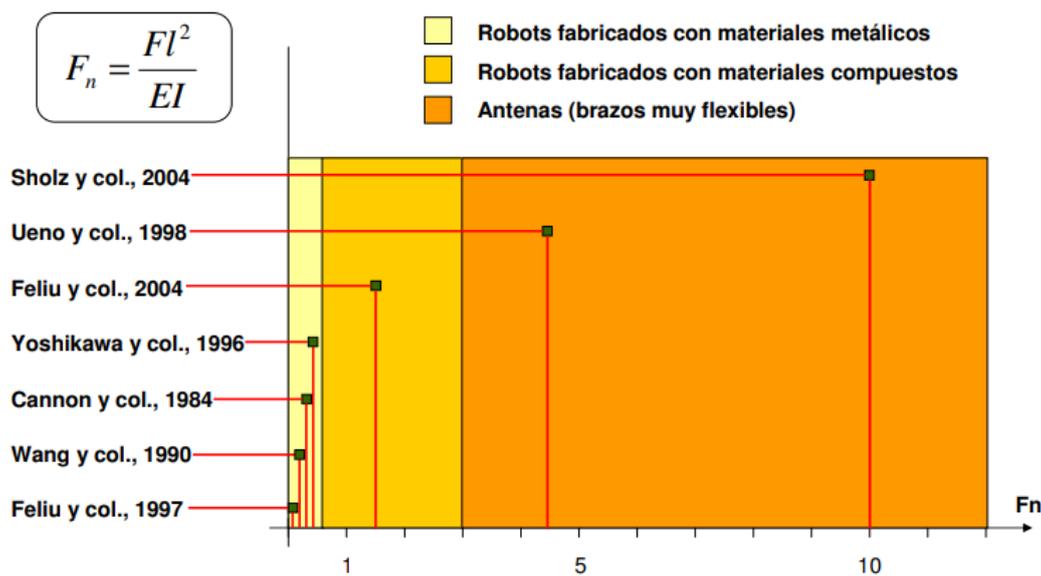


Figura 5. Clasificación de robots flexibles por el grado de deflexión

Fuente: Feliú (2009)

Respecto del planteamiento de Bernouilli, Filipovic (2009) introduce una serie de consideraciones. En robótica la trayectoria de referencia se define de forma puramente cinemática, es decir, geométrica, y ahora en la presencia de los elementos de elasticidad se pueden incluir también los valores de deformación elástica en el nivel de referencia, es decir, en el nivel de conocer las características de la elasticidad durante la trayectoria de referencia. Hay dos aspectos en la definición de la trayectoria de referencia del ángulo del motor, que son la deformación elástica, considerada por una parte, como una cantidad que no está incluida en la trayectoria de referencia; y por otra parte, la deformación elástica considerada como una cantidad que está al menos parcialmente incluida en la trayectoria de referencia. Así, en la siguiente figura, en la que se representa el mecanismo de un robot, parte del punto A, y se desplaza hacia un punto B, en un tiempo previsto $T = 2$ (s). La dinámica de la fuerza del entorno se incluye en la dinámica del movimiento del sistema, y el perfil de velocidad adoptado es trapezoidal con un periodo de aceleración/desaceleración de $0,2T$ (Potkonjak y Vukobratovic, 1999).

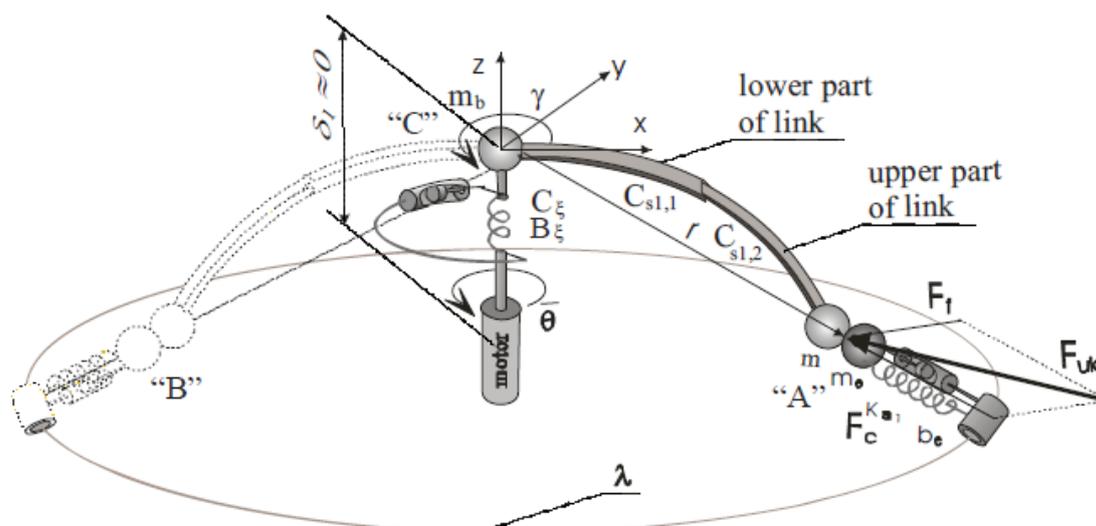


Figura 6. Mecanismo de un robot

Fuente: Filipovic (2009)

Por tanto, en el modelo planteado por Filipovic (2009), a partir de la ecuación de Euler-Bernouilli, definen la ecuación de la línea elástica de un sistema robótico complejo y demuestran que la ecuación del movimiento de todas las fuerzas que intervienen en cualquier punto se deduce directamente de la ecuación de Euler- Bernouilli. Al definir las condiciones de contorno para la punta, se obtiene la ecuación del movimiento en ese punto, lo que es la forma clásica del modelo matemático del sistema robótico elástico (Filipovic, 2009).

Entre las conclusiones o hallazgos planteados en la investigación de Filipovic (2009), se encuentran los siguientes:

- La trayectoria de referencia depende del nivel de conocimiento de las características de la elasticidad. Dichas características de elasticidad pueden estimarse e incluirse en la trayectoria de referencia y, por tanto, en la ley de control.
- La ampliación de la ecuación de Euler- Bernouilli a través de las fuerzas que participan en la formación del momento flector del modo considerado, dando lugar a diferencias en la estructura de las ecuaciones para cada modo. También la necesidad de que la estructura de la matriz de rigidez (y amortiguación) tenga los elementos fuera de la diagonal, debido a la existencia de un fuerte acoplamiento entre las fuerzas de elasticidad implicadas. A lo anterior se une el hecho de que el amortiguamiento es una característica de elasticidad omnipresente en sistemas reales, por lo que se incluye naturalmente en la ecuación de Euler- Bernouilli.

2.2. Modelo matemático del robot

Usaremos el siguiente modelo matemático desarrollado bajo el criterio de masas concentradas. Se considera la masa de la carga útil m_t , la inercia rotacional de la carga útil J_t y la masa del segmento m_b .

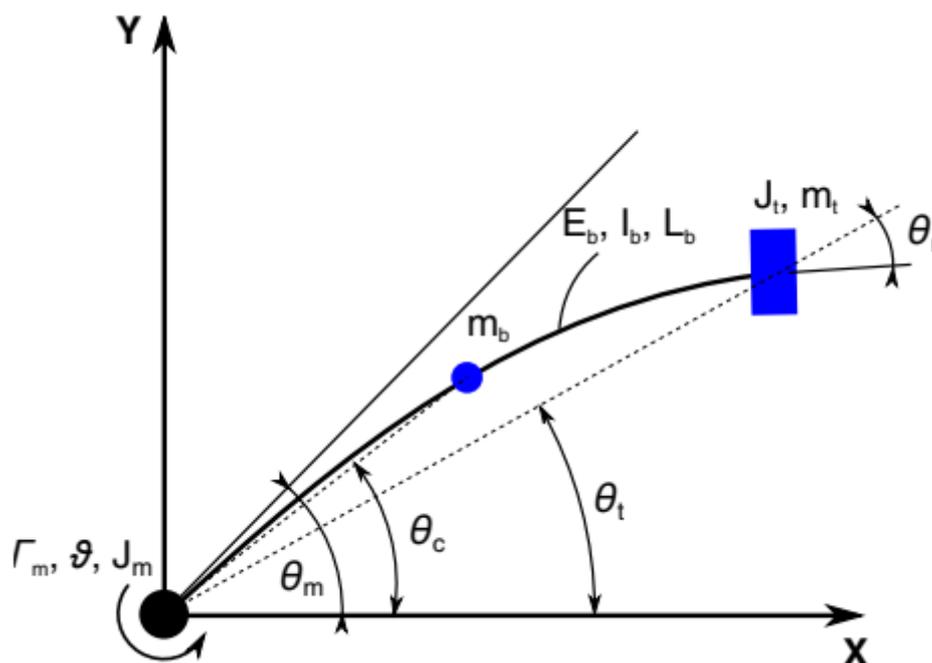


Figura 7. Robot flexible con un grado de libertad

Fuente: Jaramillo (2015)

Esta figura representa el sistema completo donde L_b representa la longitud del segmento, E_b el módulo de Young del material de dicho segmento, I_b la inercia del área de la sección transversal del segmento, J_m la inercia de rotación del eje del motor, v el coeficiente de fricción viscosa del motor y Γ_m el par aplicado por el motor.

También nos encontramos gran cantidad de ángulos como θ_m el ángulo del motor, θ_t el ángulo real del extremo del robot, θ_c el ángulo que describe el punto central del segmento y θ_r que es el ángulo que describe la carga útil.

Lo primero que debemos hacer para obtener el modelo del robot es definir el Lagrangiano L como:

$$L = E_c - U$$

Donde E_c representa la energía cinética y U la energía potencial elástica únicamente, ya que la energía potencial gravitatoria no tiene un efecto significativo

Para obtener la energía cinética del lagrangiano se necesitan las velocidades de la carga útil y del segmento, para ello derivaremos la posición del extremo del segmento y del centro del segmento respecto al tiempo.

$$P_t = L_b \cos(\theta_t) \hat{i} + L_b \sin(\theta_t) \hat{j}$$

$$\theta = (\theta_t - \theta_r) \hat{k}$$

$$P_c = \frac{L_b}{2} \cos(\theta_c) \hat{i} + \frac{L_b}{2} \sin(\theta_c) \hat{j}$$

De forma que la velocidad se define como:

$$\dot{P}_t = -L_b \sin(\theta_t) \dot{\theta}_t \hat{i} + L_b \cos(\theta_t) \dot{\theta}_t \hat{j}$$

$$\dot{\theta} = (\dot{\theta}_t - \dot{\theta}_r) \hat{k}$$

$$\dot{P}_c = -\frac{L_b}{2} \sin(\theta_c) \dot{\theta}_c \hat{i} + \frac{L_b}{2} \cos(\theta_c) \dot{\theta}_c \hat{j}$$

En términos generales, para un sistema con n cuerpos rígidos, la energía cinética E_c se puede calcular de la siguiente manera:

$$E_c = \frac{1}{2} \sum_{i=1}^n (m_i \|\dot{P}_i\|^2 + I_i \omega_i^2)$$

En nuestro caso, $n = 2$, donde los cuerpos tenidos en cuenta son la carga útil y el eslabón. Es de anotar que la inercia de rotación del eslabón se desprecia.

$$E_c = \frac{1}{2} \left[m_b \frac{L_b^2}{4} \dot{\theta}_c^2 + m_t L_b^2 \dot{\theta}_t^2 + J_t (\dot{\theta}_t^2 - 2\dot{\theta}_t \dot{\theta}_r + \dot{\theta}_r^2) \right]$$

Por otro lado, la energía potencial elástica se define como:

$$U = \frac{1}{2}k_b(\theta_m - \theta_t)^2 + \frac{1}{2}\left(\frac{16}{5}k_b\right)(\theta_m - \theta_c)^2 + \frac{1}{2}\left(\frac{L_b^2}{3}k_b\right)(\theta_t - \theta_r)^2$$

Donde,

$$k_b = \frac{3E_b I_b}{L_b}$$

Ahora que ya conocemos ambos términos del Lagrangiano, podemos definir las coordenadas generalizadas que usaremos para definir la dinámica. En nuestro caso las coordenadas generalizadas son θ_m , θ_t , θ_c y θ_r .

$$\Gamma_m = \frac{d}{dt} \left(\frac{\partial L}{\partial \dot{\theta}_m} \right) - \frac{\partial L}{\partial \theta_m}$$

$$0 = \frac{d}{dt} \left(\frac{\partial L}{\partial \dot{\theta}_c} \right) - \frac{\partial L}{\partial \theta_c}$$

$$0 = \frac{d}{dt} \left(\frac{\partial L}{\partial \dot{\theta}_r} \right) - \frac{\partial L}{\partial \theta_r}$$

$$0 = \frac{d}{dt} \left(\frac{\partial L}{\partial \dot{\theta}_t} \right) - \frac{\partial L}{\partial \theta_t}$$

Sustituyendo en estas ecuaciones el valor del lagrangiano obtenemos:

$$\Gamma_m = k_b(\theta_m - \theta_t) + \frac{16}{5}k_b(\theta_m - \theta_c)$$

$$0 = m_b \frac{L_b^2}{4} \ddot{\theta}_c + \frac{16}{5}k_b(\theta_c - \theta_m)$$

$$0 = J_t \ddot{\theta}_r - J_t \ddot{\theta}_m + \frac{L_b^2}{3}k_b(\theta_r - \theta_t)$$

$$0 = m_t L_b^2 \ddot{\theta}_t + J_t \ddot{\theta}_t - J_t \ddot{\theta}_r + k_b(\theta_t - \theta_m) + \frac{L_b^2}{3} k_b(\theta_r - \theta_t)$$

De estas ecuaciones podemos conseguimos el ángulo del motor:

-Si $J_t=0$; y la masa de $m_t \gg m_b$ por tanto $m_b = 0$.

$$\theta_m = \frac{m_t L_b^2}{k_b} \ddot{\theta}_t + \theta_t$$

-Si $J_t=0$; y la masa de $m_t \gg m_b$ pero $m_b > 0$

$$m_b \frac{L_b^2}{4} \ddot{\theta}_c = -\frac{16}{5} k_b(\theta_c - \theta_m)$$

$$-m_b \frac{5L_b^2}{64(\theta_c - \theta_m)} \ddot{\theta}_c = k_b$$

$$0 = m_t L_b^2 \ddot{\theta}_t + k_b(\theta_t - \theta_m) + \frac{L_b^2}{3} k_b(\theta_r - \theta_t)$$

$$m_t L_b^2 \ddot{\theta}_t = m_b \frac{5L_b^2(\theta_t - \theta_m)}{64(\theta_c - \theta_m)} \ddot{\theta}_c + \frac{5}{3} * \frac{m_b L_b^4(\theta_r - \theta_t)}{64(\theta_c - \theta_m)} \ddot{\theta}_c$$

Esta ecuación es demasiado compleja, es por ello que la masa del eslabón se considera insignificante, y por tanto nula, respecto a la masa de la carga útil.

3. CONTROL PID DE SIMPLE LAZO

Los manipuladores de articulación flexible (FJM) ofrecen varias ventajas con respecto a sus homólogos rígidos, como el peso ligero, el menor coste, actuadores más pequeños, mayor volumen de trabajo, mejor maniobrabilidad y transportabilidad, mayor velocidad de funcionamiento, eficiencia energética y mayor número de aplicaciones (Ibrahim y Sharkawy, 2018).

Dado que en muchas ocasiones es necesario operar a una alta velocidad para obtener una elevada productividad, se presenta un obstáculo con relación al control de la articulación, siendo uno de los problemas más habituales de la robótica el hecho de que la torsión relativa de la articulación no se compensa y provoca errores en la posición del eslabón y de la articulación con cargas pesadas (Readman, 1994; Siciliano et al., 2009).

En algunos casos la flexibilidad de la articulación puede conducir a la inestabilidad cuando se descuida en el diseño del control, como señalan Talole et al., (2010). Por lo tanto, los requisitos contradictorios entre la alta velocidad y alta precisión hacen que la tarea de control sea un problema de investigación difícil.

La investigación sobre el modelado dinámico y el control de los robots flexibles ha recibido una mayor atención en las últimas décadas. Un primer paso hacia el diseño de una estrategia de control eficiente para los manipuladores con articulaciones flexibles se ha orientado al desarrollo de modelos dinámicos que puedan caracterizar la flexibilidad de las articulaciones con precisión. El diseño de un controlador que minimice los efectos de los desplazamientos flexibles en robots ligeros es muy demandado en muchas aplicaciones industriales y espaciales que requieren un control preciso de la trayectoria, bien para lograr una posición objetivo, o una trayectoria prescrita. En el caso del primero, se busca que el tiempo de asentamiento sea corto mientras que en el segundo caso, se planifica el desplazamiento del brazo del robot para seguir esa trayectoria. No obstante, en ambos casos es necesario aplicar acciones de control para lograr la precisión en el movimiento (Ibrahim y Sharkawy, 2018).

Una revisión de la literatura académica permite comprobar que la mayor parte de las estrategias de control adaptativo de las articulaciones flexibles que aparecen en la literatura son técnicas basadas en modelos, como es el caso de Spong, Hutchinson y Vidyasagar (2006), Ozgoli y Taghirad (2006), o Ulrich y Sasiadek (2014).

Estos algoritmos de control tienen un rendimiento de seguimiento razonablemente bueno sólo cuando disponemos del conocimiento sustancial del modelo matemático de la planta y sus

parámetros de la planta. En consecuencia, si surgen variaciones significativas o impredecibles de los parámetros de la planta como resultado de la degradación de los mecanismos conjuntos, o si hay errores de modelado debido a comportamientos complejos de la dinámica flexible, los enfoques de control basados en el modelo no son adecuados (Ibrahim y Sharkawy, 2018).

Los controles de proceso, en general, son instrumentos que se emplean para controlar un parámetro. Los controladores PID son un tipo de controlador continuo dado que ajustan continuamente la salida en comparación con un controlador de encendido/ apagado, cuando se observan las condiciones de alimentación hacia delante o hacia atrás, como se puede comprobar en la siguiente figura:

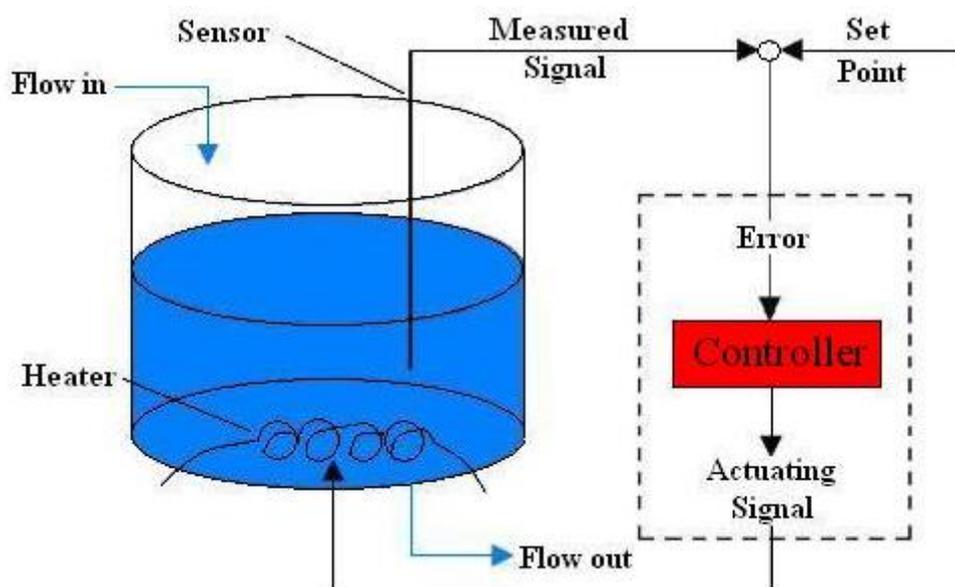


Figura 8. Ejemplo de controlador de temperatura en un CSTR

Fuente: Wolf (2021)

En este sentido, los controladores PID convencionales para máquinas automatizadas, son ampliamente aceptados por la industria, dado que resultan muy fáciles de entender al tener significados físicos claros, esto es, pasado, presente y predictivo; además son fáciles de implementar y de explicar. Se estima que el 95% de las operaciones de circuito cerrado del sector de la automatización industrial utilizan controladores PID (Elprocus, 2022).

PID significa *Proporcional Integral Derivativo* y son estos tres controladores los que combinados producen una señal de control. Se trata de un tipo de dispositivo que permite controlar diferentes variables de un proceso, como el flujo, la presión, la temperatura o la

velocidad de las aplicaciones industriales. La siguiente figura muestra el diagrama de bloques de un sistema de control PID para un robot de articulación flexible, donde los símbolos G , $Y_R(t)$, $Y(t)$, $V(t)$ y $\theta(t)$ se definen como planta del robot de articulación flexible, referencia, ángulo de giro, entrada de control y ángulo de oscilación, respectivamente.

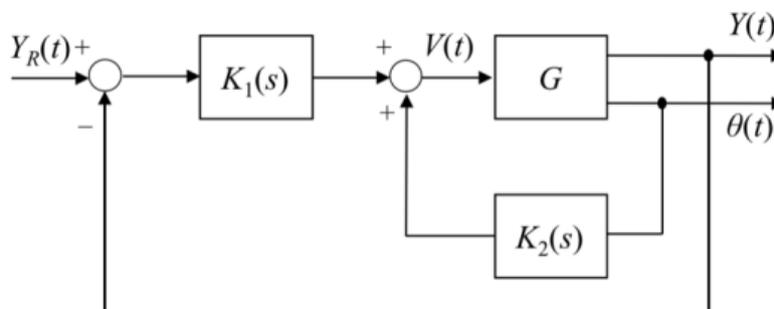


Figura 9. Diagrama de bloques de un sistema de control PID

Fuente: Ashraf (2020)

El PID o controlador proporcional integral derivativo, se puede definir como “uno de los algoritmos más comunes utilizados para los sistemas de control” (Woolf, 2021, p.1). De acuerdo con lo indicado por el autor, el PID es usado habitualmente gracias a que el citado algoritmo no involucra matemáticas de orden superior, a pesar de que contiene muchas variables, que permiten ajustar fácilmente el sistema a la configuración deseada. El algoritmo para el PID usa un circuito de retroalimentación para corregir la diferencia entre un determinado valor medido, y el punto de referencia. Para ello se calcula y emite una acción que corrige este error (Wollf, 2021).

Para realizar un control de una variable de proceso, se pueden utilizar los métodos de lazo abierto y métodos de lazo cerrado. En el caso del primero, la información sobre las características estáticas y dinámicas de la planta derivan en lazo abierto en respuesta a un escalón. Mientras que en el caso del segundo, la información de las características del lazo se obtienen a partir de un test en lazo cerrado, usualmente con un controlador con acción proporcional:

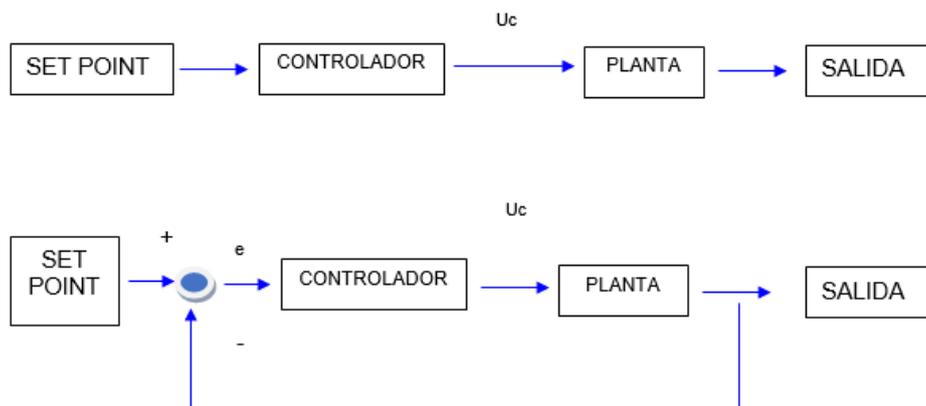


Figura 10. Control PID de lazo abierto y en lazo cerrado

Fuente: Bennet (1996)

El control PID fue propuesto por Nikolai Fyodorovich al desarrollar la idea de que era posible leer la variable del proceso a la salida y de esta forma, obtener en la salida del actuador, un término de compensación para la entrada, sumando la parte proporcional, integral y derivativa (Bennet, 1996). Un controlador PID tiene un control proporcional, integral y derivado que maneja el futuro actual, pasado y previsto del error de la señal. Las limitaciones más comunes que ocurren dentro del control PID involucran específicamente al control integral. El controlador P generalmente tiene errores de estado estable (la diferencia en el punto de ajuste y el resultado real) a menos que la ganancia de control sea grande. A medida que aumenta la ganancia de control, surgen problemas con la estabilidad del circuito de retroalimentación. Por ejemplo, reducir el tiempo de subida implica una ganancia proporcional alta, y reducir el sobreimpulso y las oscilaciones implica una ganancia proporcional pequeña. Esto no es posible de lograr en todos los sistemas, tal y como se muestra en la siguiente figura:

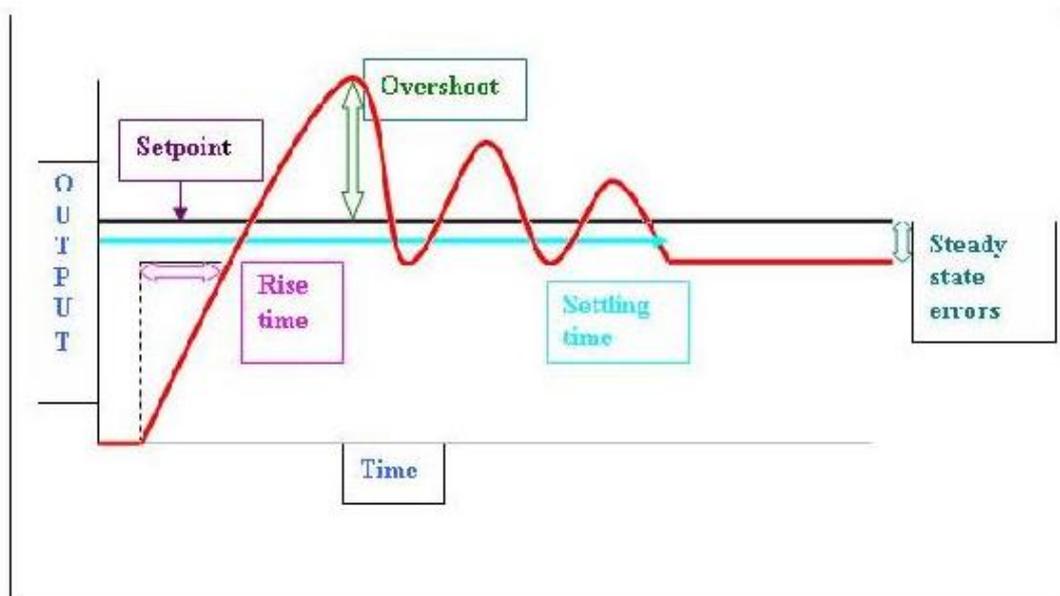


Figura 11. Diagrama general de resultados del proceso

Fuente: Wolf (2021)

En el proceso de implementación de un controlador PID es preciso definir los valores que deberán tener los parámetros de la ganancia, el tiempo integral y el tiempo derivativo, con la finalidad de que el sistema ofrezca la respuesta que se pretende.

Dicho proceso de sintonización del PID requiere de la aplicación de la información estática y dinámica del lazo, para lo cual, existen diferentes metodologías, como el Método Ziegler-Nichols (1942), el método basado en la curva de reacción o el método de Tyreus y Luyben en lazo cerrado.

La sintonía de controladores PID a través del método de Ziegler- Nichols se basa en el diseño de controladores para garantizar la relación de un cuarto de amplitud, o una amortiguación de un cuarto de amplitud. Tal y como exponen Žáková y Huba (1997), para los tipos básicos de controladores lineales continuos definieron:

P- controller $C(s) = K_c$

PI- controller $C(s) = K_c \left(1 + \frac{1}{T_{is}}\right)$

PID- controller $C(s) = K_c \left(1 + \frac{1}{T_{is}} + T_{Ds}\right)$

Además, recomendaron los siguientes conjuntos de parámetros:

	K_c	$1/T_i$	T_D
P	1/ RL	-	-
PI	0,9/ RL	0,3 / RL	-
PID	1,2 /RL	0,5 / RL	0,5 L

4. CONTROL PID DE DOBLE LAZO

Las oscilaciones pueden estar presentes en muchos procesos reales, bien de naturaleza biológica (Cardon e Iberall, 1970), en turbulencias químicas (Kim et al., 2001), en reacciones químicas interactivas (Cohen y Neu, 1979), en procesos de fermentación extractiva (Nandong, Samiyudia y Tade, 2006), control de patrones en sistemas oscilatorios (Huang et al., 2011), sistemas robóticos bioinspirados (Porras y Llinas, 2014), etc.

En la mayor parte de los casos, una oscilación excesiva en los sistemas puede dar lugar a varios problemas. Por ejemplo, en un sistema oscilante mal controlado, puede ser el origen de fatiga mecánica, motivo por el cual, en la mayoría de las industrias se aplica el controlador proporcional integral derivativo (PID). Cabe señalar que la mayoría de las reglas de sintonización o diseño existentes para los controladores PID están pensadas o son adecuadas para sistemas estables sobreamortiguados o ligeramente subamortiguados (O'Dwyer, 2006).

Sorprendentemente, a pesar de la naturaleza oscilatoria común en los sistemas reales, las reglas de sintonía PID para sistemas oscilatorios son todavía muy limitadas (Nandong, 2015). En los sistemas servoaccionados, la retroalimentación para el control de la posición, la velocidad y/o el par normalmente la proporciona un codificador rotatorio. Pero los componentes mecánicos y las conexiones en el tren motriz, como las cajas de engranajes, los husillos de bolas y los acoplamientos, no son perfectamente rígidos e introducen juego, flexibilidad y torsión en el sistema. Esta falta de rigidez provoca una discrepancia entre dónde “debería” estar la carga en función de la posición del eje del motor y dónde se encuentra realmente la carga. Si el sistema de servocontrol se basa únicamente en la retroalimentación del motor, es posible que nunca se alcance la posición deseada y el sistema puede experimentar inestabilidades como oscilaciones (Collins, 2021).

En los sistemas que muestran contragolpe, cumplimiento y giro (es decir, prácticamente todos los sistemas de movimiento) pero que también necesitan un control de posición muy preciso, se requiere un segundo codificador para medir directamente la posición de la carga. Este método de usar dos dispositivos de retroalimentación, uno en el motor y otro ubicado en la carga, generalmente se denomina control de doble lazo o retroalimentación de doble lazo (Collins, 2021), como se representa en la siguiente figura:

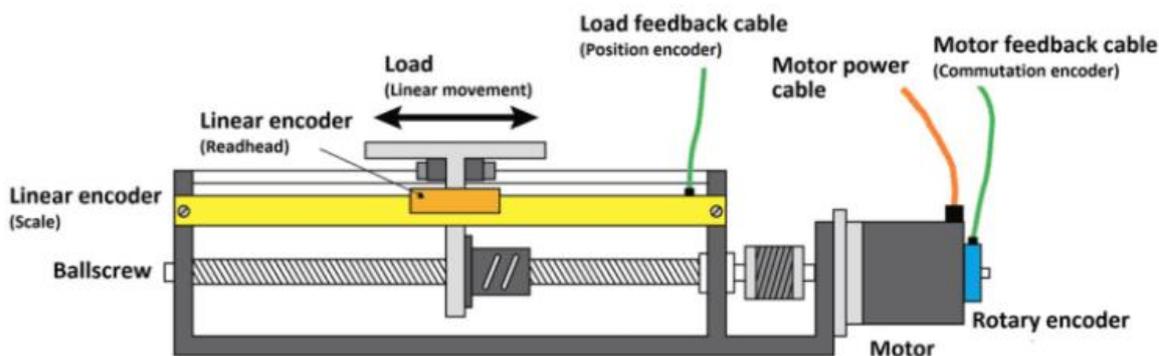


Figura 12. Control de doble lazo en el posicionamiento preciso de los ejes lineales

Fuente: Collins (2021)

Hay varios métodos de operación con control de doble lazo, pero en el método más básico, el controlador primero cierra el lazo interno, que es el lazo de control de velocidad. El lazo de control de velocidad recibe retroalimentación del codificador del motor, y esta retroalimentación determina la ganancia de retroalimentación de velocidad adecuada (K_v), que imparte un efecto de amortiguación en el sistema para reducir las oscilaciones (collins, 2021).

La ganancia de retroalimentación de velocidad (K_v) es similar a la ganancia derivada (K_d), excepto que la retroalimentación de velocidad escala solo la velocidad medida por el codificador del motor, mientras que la ganancia derivada escala la derivada del error de posición. La ganancia de retroalimentación de velocidad se usa a menudo en el control de doble bucle. Después, el controlador cierra el bucle exterior o bucle de posición, y la retroalimentación del codificador de cara se usa para determinar las ganancias integrales y proporcionales apropiadas (K_p y K_i), donde la ganancia proporcional es directamente proporcional al error de posicionamiento e influye en la rigidez del sistema; mientras que la ganancia integral acumula el error de posición a lo largo del tiempo y "empuja" el sistema a un error de posicionamiento cero (Collins, 2021).

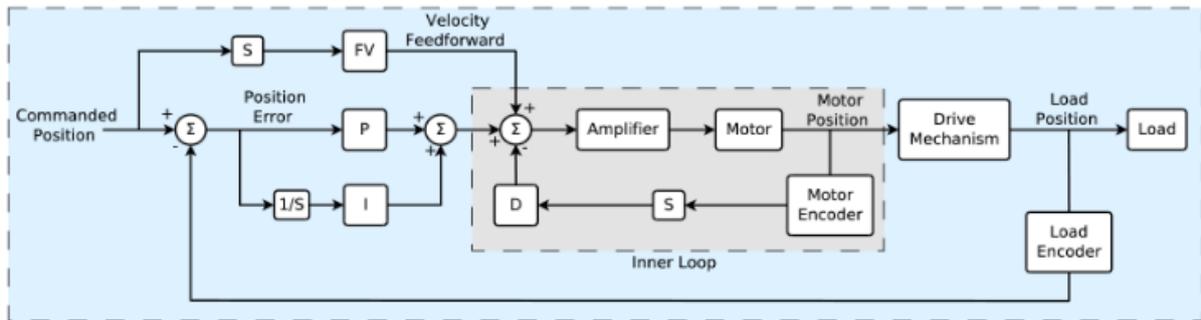


Figura 13. Esquema típico de control de doble lazo

Fuente: Collins (2021)

El sistema de servocontrol también puede incluir un bucle de control de corriente, situado dentro del bucle de velocidad. Aunque en este escenario habría tres lazos de control, el término "control de doble lazo" simplemente se refiere a los dos lazos de control (posición y velocidad) que reciben retroalimentación directa de los codificadores. A diferencia de los lazos de control en cascada tradicionales, en el control de doble lazo, la respuesta del lazo de velocidad es independiente de la retroalimentación (y por tanto de la resolución) del codificador de carga. Asimismo, la respuesta del lazo de posición es independiente de la retroalimentación del codificador del motor. Los sistemas de retroalimentación de doble bucle pueden usar dos codificadores rotatorios o un codificador rotatorio y uno lineal. Las combinaciones lineales rotativas son más comunes en los sistemas de movimiento lineal, donde hay un codificador rotatorio ubicado en el motor y un codificador lineal montado en el eje lineal (Collins, 2021).

5. CONTROL GPI

Desde los comienzos de la robótica, el ser humano ha tratado de dotar a los robots de capacidad para manipular cualquier tipo de objeto a través de manos automáticas. El control de fuerzas es un aspecto importante de los robots en los que un objeto debe ser manipulado sin sufrir daños, motivo por el cual, resulta de gran importancia realizar un análisis de las fuerzas de contacto entre el manipulador y dicho objeto. De esta forma, autores como Gilardi y Sharf (2002), consideran que el uso de dedos robóticos deformables o flexibles en tareas en las que hay contacto con el entorno, mejora las capacidades limitadas de los dedos rígidos robóticos, como también se muestra en el estudio de Shimoga (1996). La elasticidad de los dedos flexibles permite una mayor adaptabilidad entre el manipulador y el objeto y, además, evita daños en las superficies de contacto (Becedas, Payo, Feliu y Sira, 2008).

En las últimas décadas se han publicado un voluminoso número de trabajos con relación a la cuestión del control de la fuerza de los manipuladores flexibles, utilizando diferentes técnicas de control, como el control óptimo (Matsuno y Kasai, 1998), control híbrido de posición/fuerza (Chiou y Shahinpoor, 1988), técnicas de control en el dominio de la frecuencia (Chapnik et al., 1993), las redes neuronales (Tian et al., 2004) o la lógica difusa (Shi y Trabia, 2005).

Choi y Cho (2002) utilizaron una aleación con memoria de forma (SMA) para actuar sobre una pinza flexible implementando un controlador H^∞ pero la fuerza de respuesta del actuador SMA es relativamente lenta y sus aplicaciones son limitadas.

Fukuda et al. (1986) diseñaron un control de fuerza adaptativo para un par de puntas de agarre hechas de placas de aluminio con bandas extensométricas y accionadas por un motor de corriente continua. Gorce y Fontaine (1996) llevaron a cabo una metodología para el diseño de pinzas flexibles. Tanaka et al. (1996) se ocuparon del control de la fuerza de un dedo flexible utilizando un piezoeléctrico como actuador y fuerza distribuidos a lo largo del dedo flexible.

En la teoría de control son de gran relevancia las investigaciones centradas en el seguimiento o rechazo de las señales periódicas, puesto que afecta a una gran variedad de aplicaciones en el campo de la ingeniería, donde este tipo de señales se está aplicando. Entre estas aplicaciones se incluyen las aportaciones de Wu y Tomizuka (2010) en los controladores de discos duros, en robótica (Tayebi et al., 2008), en inversores electrónicos (Zhang et al., 2010) o turbinas eólicas (Houtzager et al., 2013).

La introducción de las ideas de inyección de energía y amortiguación se introdujeron en el control de los robots gracias al artículo de Takegaki y Arimoto (1981). En él se emplea la retroalimentación de la velocidad para inyectar la amortiguación necesaria. Autores como Tomei (1991), utilizaron esta estrategia para la regulación de la posición en robots de articulación flexible.

Basándose en estas ideas, se han publicado varios trabajos, sobre el control de robots de articulación flexible (Ailon y Ortega, 1993; Berghnis y Nijmeijer, 1993; Kelly et al., 1994; Ortega et al., 1994). En estas investigaciones el elemento común ha sido el uso de la inyección de amortiguación mediante una señal similar a la de la velocidad obtenida mediante el filtrado de posición.

El GPI ha sido aplicado por Fliess, Márquez, Delaleau, y Sira (2002), Fliess (2000), Márquez (2001), Márquez, Fliess y Mounier (2001), Márquez, Delaleau y Fliess (2000) o Márquez (2001) en sistemas lineales. Una de las principales características de este esquema de control es la introducción de reconstructores integrales para estimar las variables de estado, siempre que se disponga de medidas de entrada y salida. Sin embargo, la mayoría de los trabajos sobre el control GPI se han realizado para sistemas lineales y muy pocos de ellos han sido realizados para sistemas de control de estado. Es por ello por lo que se han propuesto esquemas de control GPI para varios sistemas no lineales (Hernández y Sira, 2001, 2002a, 2002b, 2002c).

En el caso de los manipuladores robóticos, el problema de control de seguimiento de trayectoria en los robots rígidos, se han obtenido resultados de estabilidad asintótica local (Hernández y Sira, 2002b). El principal obstáculo para obtener resultados globales en este caso es la dependencia cuadrática de la velocidad del término de Coriolis cuyos efectos se amplían aún más por el reconstructor de velocidad. Otro obstáculo importante es la dependencia de la matriz de inercia de la configuración.

Al respecto, autores como Hernández y Sira (2003), han utilizado un esquema de control GPI para robots de articulación flexible y demostrado la estabilidad asintótica global gracias a la estructura lineal de la dinámica del actuador y la integración de un término integral adicional destinado a compensar el valor inicial desconocido de las velocidades del actuador. En este caso, el principal problema con respecto a los controladores basados en la conformación de energía y la inyección de amortiguación es la necesidad de medir, adicionalmente, las posiciones de los eslabones. En el esquema de control que presentan los autores se combinan las ideas de control GPI y de inyección de amortiguación de energía. Del control GPI toman los reconstructores de velocidad y la retroalimentación integral de entrada. A lo

anterior, incorporan la estrategia de inyección de amortiguación de energía de Tomei (1991) para robots con articulaciones flexibles.

El control GPI o control integral proporcional generalizado se configura en la actualidad como un elemento con un buen rendimiento en el control de los sistemas no lineales. El control GPI proporciona una mejor respuesta dinámica que otros como el PID, especialmente en cuestiones referentes a la estabilización, y también en cuanto a la robustez en el rechazo de perturbaciones (Fernández, Belmonte, Morales y Somolinos, 2015).

Ello es debido a tratar de evitar la necesidad de los tradicionales observadores de estado asintóticos y de estado asintóticos tradicionales y porque procede directamente a utilizar, en una ley de control de retroalimentación de estado previamente diseñada, estimaciones de estado estructural en lugar de las variables de estado reales. Estas estimaciones estructurales están basadas en reconstruidores integrales y sólo requieren entradas y salidas integrales iteradas de dichas señales disponibles. El efecto de los estados iniciales despreciados se compensa adecuadamente por medio de un número suficientemente grande de integrales iteradas de salida, de integrales de entrada y de acciones de control. Este efecto se demuestra desde una perspectiva teórica en trabajos como el de Fliess, Márquez, Delaleau y Sira (2002), y en estudios empíricos como los publicados por Sira (2003) y Morales, Sira y Feliu (2014).

En este apartado se presenta el diseño de un controlador GPI (Proporcional Integral Generalizado) de doble lazo para controlar la posición de un segmento flexible acoplado a un motor de corriente continua a través de un reductor. Se ha seleccionado esta técnica debido a que permite desarrollar controladores lineales robustos que trabajan adecuadamente en situaciones en las que existen perturbaciones no lineales.

Para realizar este diseño se tomará como referencia el trabajo realizado por Becedas et al. (2007).

5.1. Modelo simplificado de parámetros concentrados

A continuación, se obtiene un modelo simplificado del eslabón flexible. Para ello se considera un modelo de parámetros concentrados en el que se supone que toda la masa del eslabón flexible se encuentra en la punta del mismo.

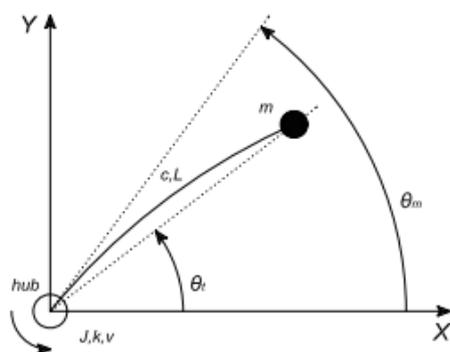


Figura 14. Diagrama de un brazo flexible de un único segmento

Basado en este diagrama y anotaciones anteriores, se propone la siguiente ecuación para el segmento flexible:

$$m * L^2 * \ddot{\theta}_t = c * (\theta_m - \theta_t) \quad [1]$$

Donde m [kg] es la masa desconocida en la posición de la punta, L [m] es la longitud del brazo flexible y c [N*m] es la rigidez del segmento, esta última depende de la rigidez a la flexión $E * I$ [N*m²] y de la longitud del segmento de la siguiente manera:

$$c = \frac{3 * E * I}{L}$$

Por último, θ_m [rad] es la posición angular del motor, θ_t [rad] es la posición angular no medida y $\ddot{\theta}_t$ [rad/s²] es la aceleración angular de la punta.

Aplicando Laplace en esta ecuación obtenemos:

$$m * L^2 * s^2 * \theta_t = c * (\theta_m - \theta_t)$$

Seguidamente procedemos a mover la c al otro lado de la ecuación:

$$\frac{m * L^2 * s^2}{c} * \theta_t = \theta_m - \theta_t$$

Por último, despejando la θ_t obtenemos:

$$\theta_t = \frac{c}{m * L^2 * s^2} * (\theta_m - \theta_t)$$

Ahora, para que resulte más sencillo transformar esta ecuación en bloques de simulink vamos a realizar el siguiente cambio:

$$\theta_t = \frac{c}{m * L^2} * \frac{1}{s} * \frac{1}{s} * (\theta_m - \theta_t)$$

De tal forma el diagrama de bloques en simulink para esta parte flexible quedaría de la siguiente manera:

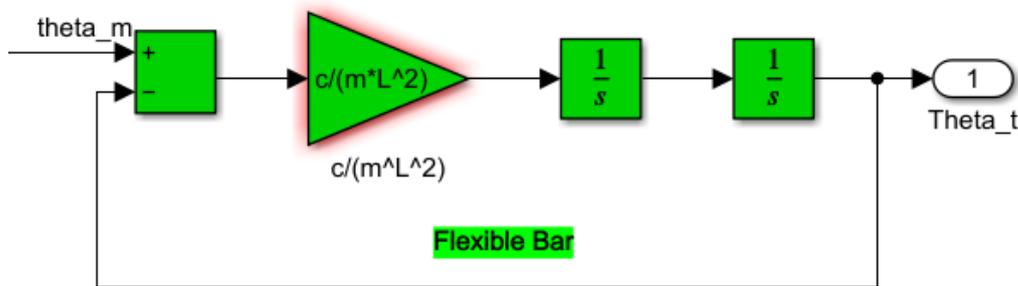


Figura 15. Diagrama de bloques en simulink de un eslabón flexible.

5.2. Modelo del conjunto Motor-Reductor

A continuación procedemos con la parte rígida del sistema:

Un actuador electromecánico habitual en muchos sistemas de control es el motor de corriente continua. El motor de CC utilizado aquí es alimentado por un servoamplificador con un control de bucle interno de corriente. Podemos escribir la ecuación dinámica del sistema utilizando la segunda ley de Newton:

$$k * u = J * \ddot{\theta}_m + v * \dot{\theta}_m + \hat{\Gamma}_c * \dot{\theta}_m + \frac{\Gamma}{n} \quad [2]$$

Donde J [$\text{kg} \cdot \text{m}^2$] es la inercia del motor, v [$\text{N} \cdot \text{m} \cdot \text{s}$] es el coeficiente de fricción viscosa, $\ddot{\theta}_m$ [rad/s^2] y $\dot{\theta}_m$ [rad/s] son la aceleración angular del motor y la velocidad angular del motor respectivamente.

Γ_c [$\text{N} \cdot \text{m}$] es el par de fricción de Coulomb desconocido que afecta a la dinámica del motor. Este término de fricción no lineal se considera como una perturbación que depende sólo del signo de la velocidad angular del motor. Como consecuencia, la fricción de Coulomb cuando $\dot{\theta}_m \neq 0$ sigue el siguiente modelo:

$$\hat{\Gamma}_c * \text{sign}(\dot{\theta}_m) \begin{cases} \hat{\Gamma}_c (\dot{\theta}_m > 0) \\ -\hat{\Gamma}_c (\dot{\theta}_m < 0) \end{cases} \quad [3]$$

Continuando con los parámetros de la ecuación [2], el parámetro k [N*m/V] es la constante electromecánica conocida del sistema servoamplificador del motor. Γ [N*m] es el par de acoplamiento medido en el eje y n es la relación de reducción del engranaje del motor. Por último u [V] es la tensión de entrada del motor.

En esta segunda parte se usará $\hat{\theta}_m$ en lugar de θ_m , la diferencia entre ambas es que $\hat{\theta}_m$ es la posición angular del motor a la salida del rotor, mientras que θ_m es la posición angular a la salida de la reductora.

Al igual que hicimos antes en la ecuación [1], aplicamos Laplace en la ecuación [2] y obtenemos:

$$k * u = J * s^2 * \hat{\theta}_m + v * s * \hat{\theta}_m + \Gamma c * s * \hat{\theta}_m + \frac{\Gamma}{n}$$

En esta ecuación despejamos la primera θ_m para obtener lo siguiente:

$$\hat{\theta}_m = \frac{1}{J * s^2} * [k * u - v * s * \hat{\theta}_m - \Gamma c * s * \hat{\theta}_m - \frac{\Gamma}{n}]$$

A continuación realizaremos por partes la transformación en un diagrama de bloques para simulink.

$\frac{1}{J * s^2}$ se podría dibujar como

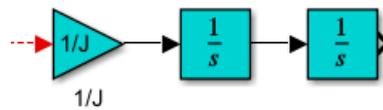


Figura 16. Primera parte del diagrama de bloques

$k * u$ de la siguiente manera

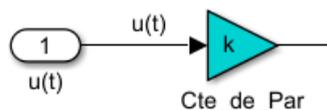


Figura 17. Segunda parte del diagrama de bloques

Agrupando $v * s * \hat{\theta}_m$ y $\Gamma c * s * \hat{\theta}_m$ obtenemos un diagrama de este estilo:

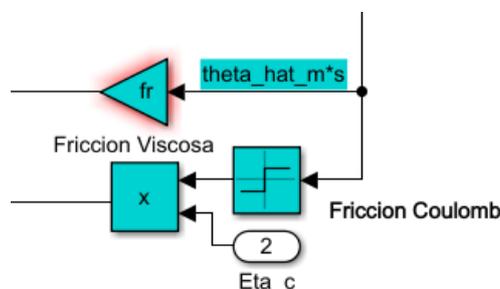


Figura 18. Tercera parte del diagrama de bloques

Por último gracias a una ecuación de apoyo:

$$\Gamma = c * (\theta_m - \theta_t) \quad [4]$$

Podemos transformar $\frac{\Gamma}{n}$ en $\frac{c}{n} * (\theta_m - \theta_t)$, y asimismo transformarlo en un bloque de simulink de la siguiente manera:

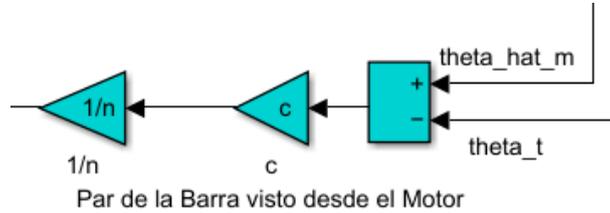


Figura 19. Cuarta parte del diagrama de bloques

5.3. Modelo del Conjunto Motor-Reductor-Segmento Flexible y su Simulación

Finalmente si juntamos todos los bloques que hemos creado anteriormente, tanto para la parte flexible como para la parte rígida obtenemos un modelo en simulink como este:

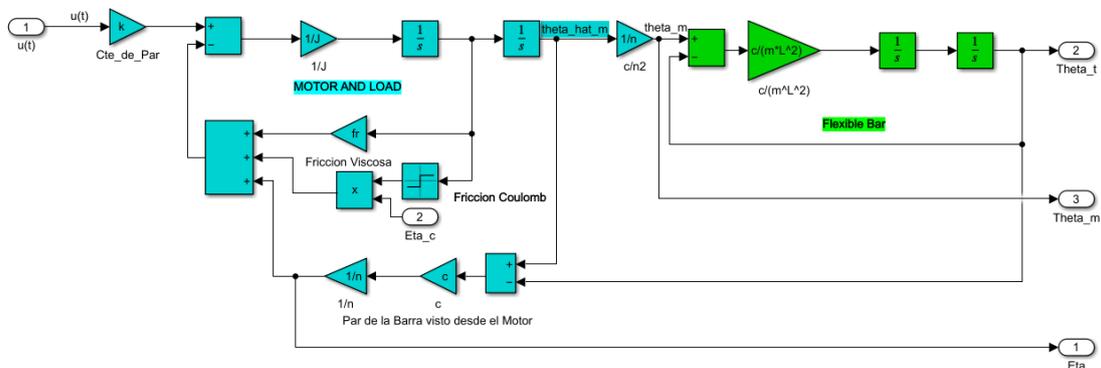


Figura 20. Diagrama de bloques de un eslabón flexible con servomotor.

Una vez conseguido este diagrama, y antes de proceder al desarrollo de los controladores, añadiremos ciertas ecuaciones que nos serán de ayuda posteriormente.

Primero obtendremos la función de transferencia (F.T. en adelante) de la siguiente forma:

$$\frac{\theta_t(s)}{\theta_m(s)} = \frac{c}{m * L^2 * s^2 + c} = \frac{\frac{c}{m * L^2}}{s^2 + \frac{c}{m * L^2}}$$

Por lo que la F.T. quedaría: $G_b(s) = \frac{W_0^2}{s^2 + W_0^2}$

Donde $W_0 = \sqrt{\frac{c}{m * L^2}}$ es la frecuencia natural de oscilación del eslabón

Por otro lado, recuperando la ecuación [2]:

$$k * u = J * \ddot{\theta}_m + v * \dot{\theta}_m + \hat{\Gamma}_c * \dot{\theta}_m + \frac{\Gamma}{n}$$

En el caso de que se pretenda una aceleración mayor, se puede incrementar el par que da el motor con la cantidad necesaria para acelerar el eslabón, en cuyo caso, el nuevo término de la señal al motor deberá ser el siguiente:

$$U = U_d + \frac{\Gamma}{K * n}$$

Donde,

U es la nueva tensión aplicada al motor.

U_d es la tensión anterior, es decir, antes de incorporar este término.

Por lo tanto,

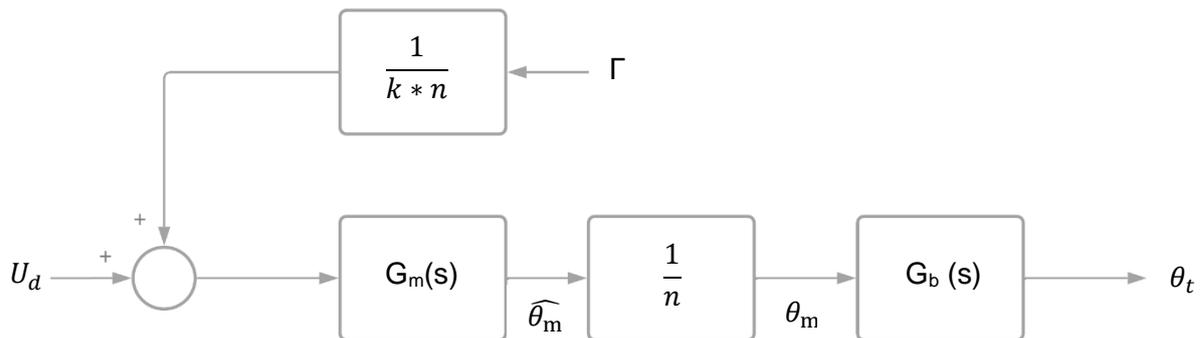
$$K * \left[U_d + \frac{\Gamma}{K * n} \right] = J * \ddot{\theta}_m + V * \dot{\theta}_m + \hat{\Gamma}_c * (\dot{\theta}_m) + \frac{\Gamma}{n}$$

$$K * U_d + \frac{\Gamma}{n} = J * \ddot{\theta}_m + V * \dot{\theta}_m + \hat{\Gamma}_c * (\dot{\theta}_m) + \frac{\Gamma}{n}$$

$$K * U_d = J * \ddot{\theta}_m + V * \dot{\theta}_m + \hat{\Gamma}_c * (\dot{\theta}_m)$$

Donde $\hat{\Gamma}_c * (\dot{\theta}_m)$ se considera como una perturbación que afecta al motor y que hay que reducir todo lo posible, por lo que no la tendremos en cuenta

Luego, el nuevo esquema es el siguiente:



Aplicando Laplace en la última ecuación obtenida conseguimos:

$$K * U_d = J * \ddot{\theta}_m + V * \dot{\theta}_m$$

$$K * U_d(s) = [J * s^2 + V * s] * \hat{\theta}_m$$

Ahora, para conseguir la F.T. del motor relacionamos $\hat{\theta}_m(s)$ con $U_d(s)$ consiguiendo lo siguiente:

$$\frac{\hat{\theta}_m(s)}{U_d(s)} = \frac{K}{J * s^2 + V * s} = \frac{\frac{K}{J}}{s * (s + \frac{V}{J})}$$

Podemos simplificar dicha F.T. hasta el punto que tenemos:

$$G_m(s) = \frac{\hat{\theta}_m(s)}{U_d(s)} = \frac{A}{s * (s + B)}$$

Donde

$A = K/J$, sabiendo que $K = k/n$

$B = V/J$

Una vez establecidas todas estas ecuaciones, procedemos a introducir los valores de todos los parámetros en Matlab, de nuevo usaremos los datos conseguidos por Becedas et Al., (2007) de manera experimental en su investigación de 2007, dichos parámetros, ya introducidos en Matlab, son los siguientes:

```
n = 50; %Relación de reducción
J = 6.87*10^(-5); %Inercia del motor [kg · m2]
fr = 1.04*10^(-3); %Coeficiente fricción viscosa [N · m · s]
k = 0.21; %constante electromecánica [N · m/V ]
L = 0.5; % Longitud de la barra [m]
d = 3*10^-3; %Diametro de la barra [m].
EI = 0.260; %Rigidez a la flexión de la barra [N · m2]
c = 1.585; %Rigidez de la barra [N · m]
m = 3*10^-2; %Masa [kg].
w0 = 14.5373; %Frecuencia natural de oscilación de la barra

%Parametros F.T. del motor
A = 61.14;% [N/(V · kg · s)]
B = 15.15;% [N · s/(kg · m)].
```

Figura 21. Parámetros introducidos en Matlab

```
%Función de transferencia del Motor:  
K=k/n;A=K/J;B=fr/J;  
Gm=tf(A,[1 B 0])  
  
%Función de Transferencia de la barra  
w0=sqrt(c/(m*L^2))  
Gb=tf(w0^2,[1 0 w0^2])
```

Figura 22. Funciones de transferencia del motor y el eslabón introducidas en Matlab

Establecidos todos los parámetros, podemos proceder a comprobar el comportamiento de nuestro diagrama de bloques y conseguir las respuestas del sistema ante diferentes entradas $u(t)$, usando siempre una entrada Γ_c nula.

-Entrada Impulso:

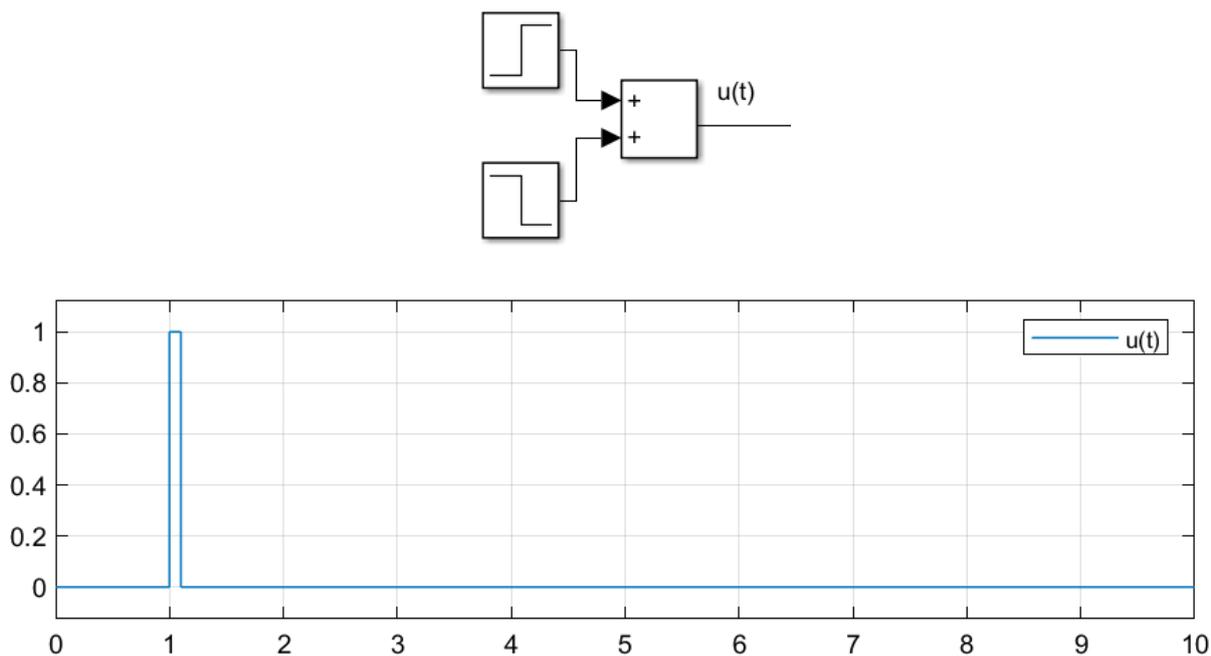


Figura 23. Bloques y gráfica de una entrada impulso

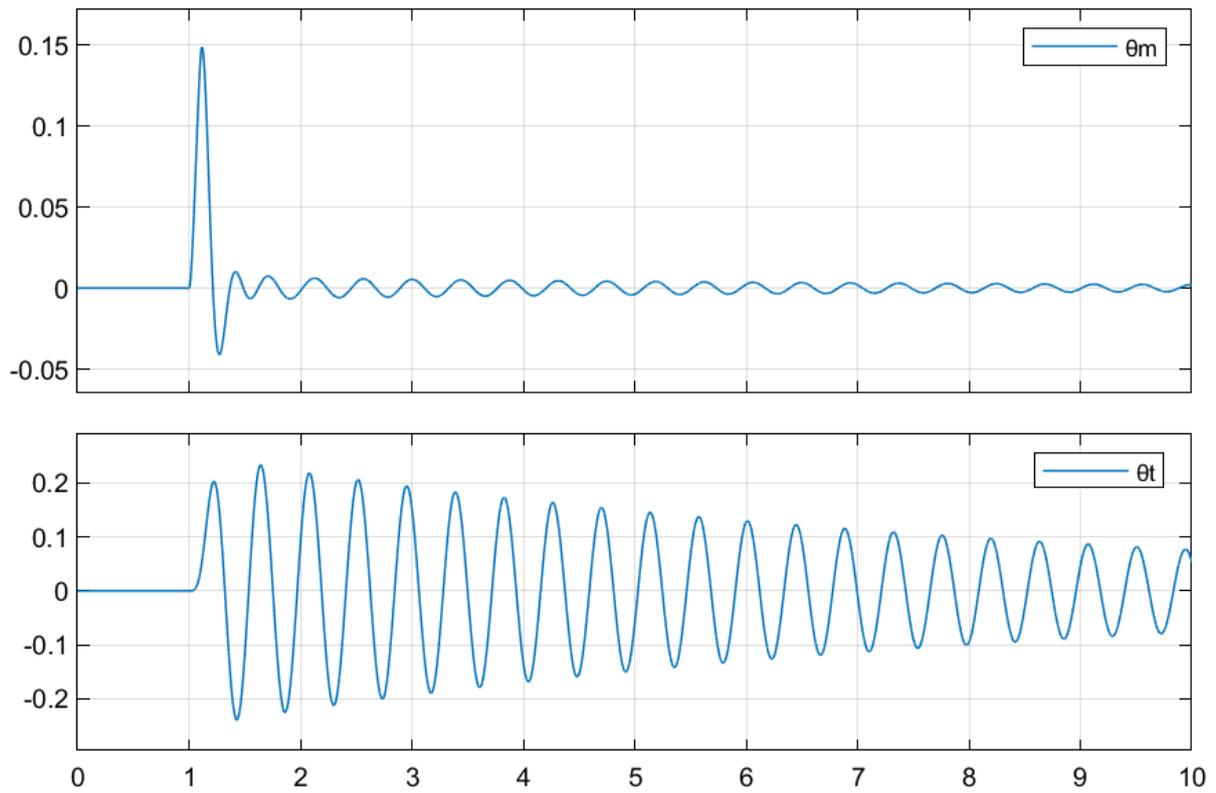


Figura 24. Salidas del ángulo del motor y del ángulo del eslabón frente a una entrada impulso

-Entrada Escalón:

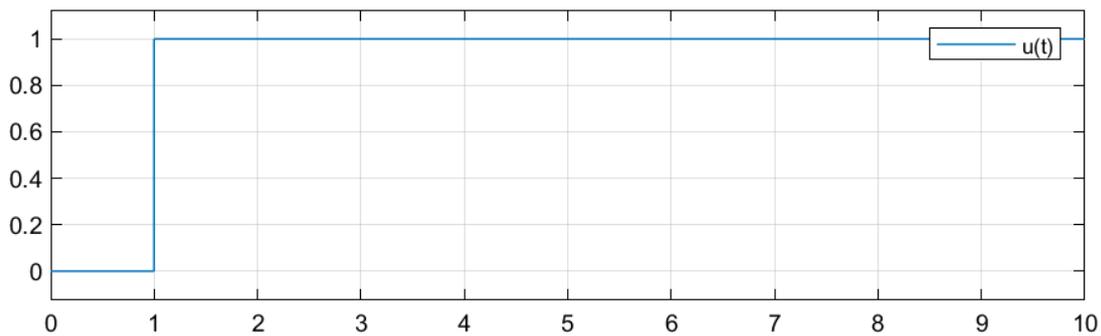
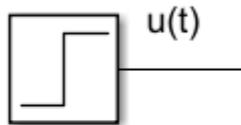


Figura 25. Bloque y gráfica de una entrada escalón

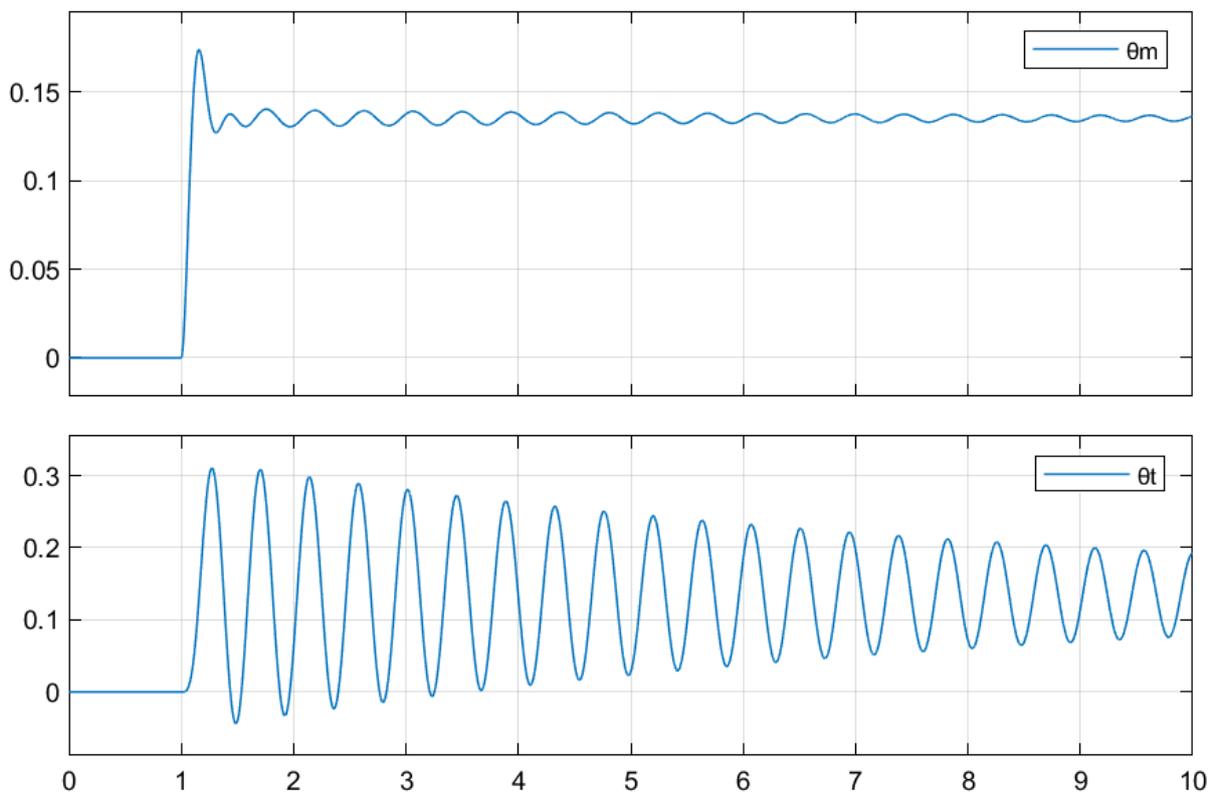


Figura 26. Salidas del ángulo del motor y del ángulo del eslabón frente a una entrada escalón

-Entrada Rampa:

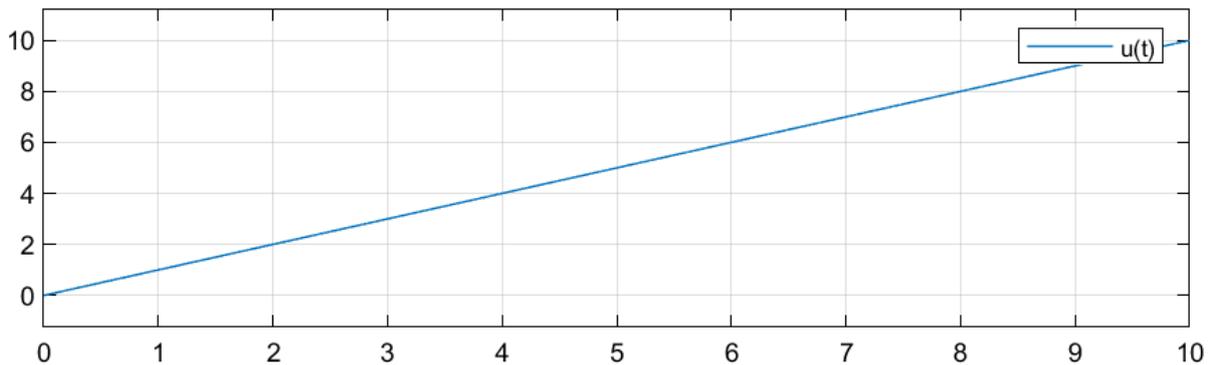


Figura 27. Bloque y gráfica de una entrada rampa

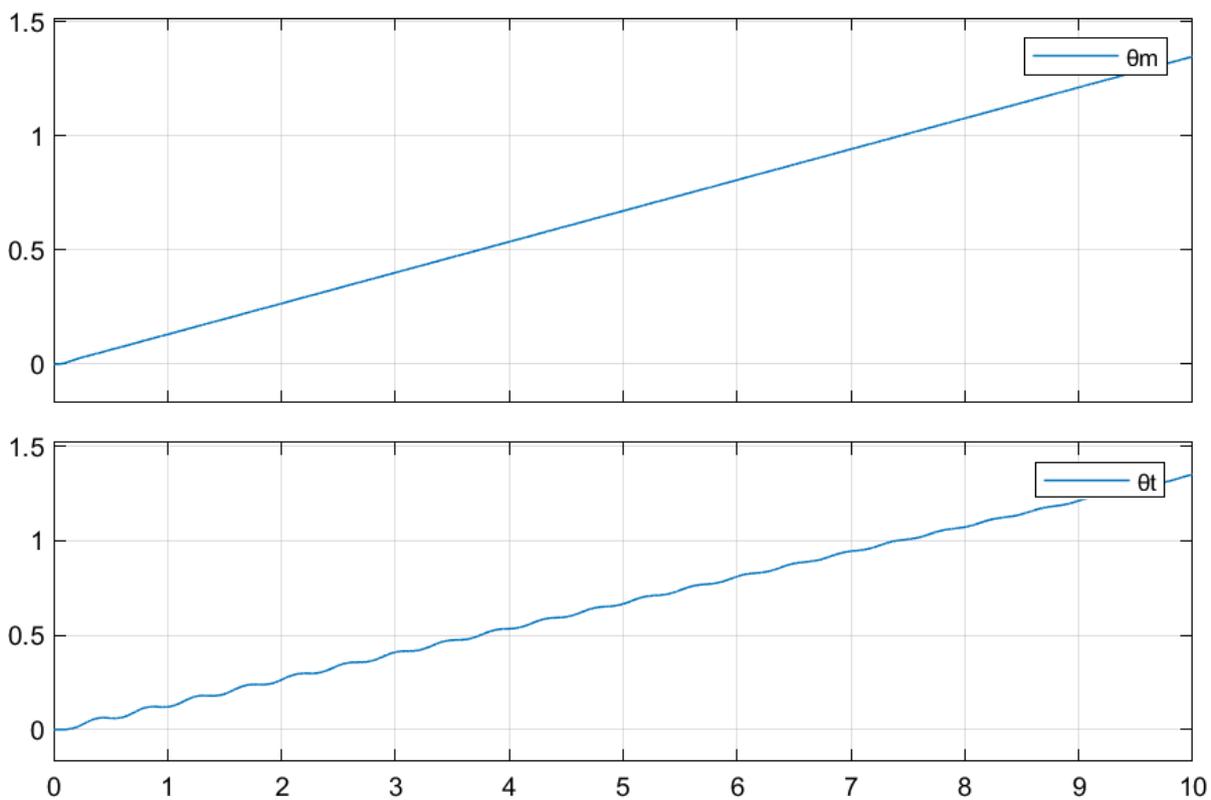


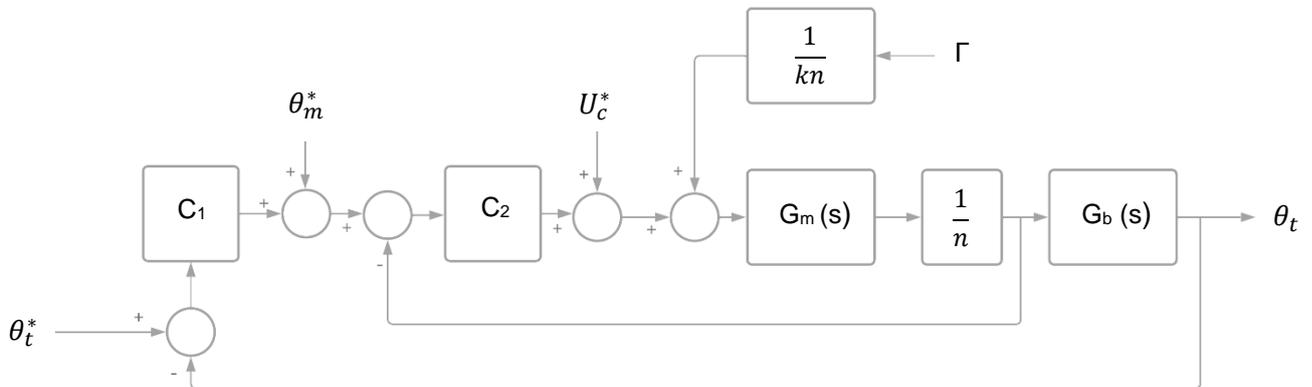
Figura 28. Salidas del ángulo del motor y del ángulo del eslabón frente a una entrada rampa

Como podemos comprobar la salida de θ_t en los dos primeros casos (impulso y escalón), es prácticamente idéntica salvo la posición final, en el caso del impulso el eslabón tiende a volver a la posición inicial, mientras que con la entrada escalón el ángulo del eslabón tiende a alcanzar el ángulo del motor, que en este caso posee un valor fijo.

En el último caso vemos que como la entrada sigue aumentando constantemente, tanto el motor como el eslabón dan vueltas sin detenerse.

5.4. Diseño del Lazo Externo del Control GPI

El sistema de control que se pretende desarrollar es de la siguiente forma:



Despejando la ecuación [1] se obtiene:

$$m * L^2 * \ddot{\theta}_t = c * (\theta_m - \theta_t)$$

$$\theta_m = \frac{m * L^2}{c} * \ddot{\theta}_t + \theta_t$$

$$\theta_m = \frac{1}{\omega_0^2} * \ddot{\theta}_t + \theta_t \quad [5]$$

Ahora suponemos que tenemos un valor deseado de θ_t al que llamaremos θ_t^* , este valor a su vez debería generar un valor de θ_m deseado al que se le llamará θ_m^* .

Entonces, dada esta θ_t^* , se puede obtener una θ_m^* (deseada) que actúe como una señal de entrada auxiliar que debe cumplir la ecuación [5]:

$$\theta_m^* = \frac{1}{\omega_0^2} * \ddot{\theta}_t^* + \theta_t^* \quad [6]$$

Por otra parte, si restamos [5] - [6], nos queda:

$$(\theta_m - \theta_m^*) = \frac{1}{\omega_0^2} * (\ddot{\theta}_t - \ddot{\theta}_t^*) + (\theta_t - \theta_t^*)$$

Si a esta diferencia entre θ_m y θ_m^* la llamamos e_{θ_m} (error), conseguimos la siguiente ecuación:

$$e_{\theta_m} = \frac{1}{W_0^2} * \ddot{e}_{\theta_t} + e_{\theta_t}$$

$$\ddot{e}_{\theta_t} = W_0^2 * (e_{\theta_m} - e_{\theta_t}) \quad [7]$$

Suponiendo que somos capaces de medir el error de seguimiento de la posición angular, e_{θ_t} , entonces el controlador incremental de retroalimentación del lazo exterior C1 podría proponerse como el siguiente PID:

$$e_{\theta_m} = e_{\theta_t} - \frac{1}{W_0^2} * (k_2 * \dot{e}_{\theta_t} + k_1 * e_{\theta_t} + k_o * \int e_{\theta_t} dt) \quad [8]$$

Introducimos esta última ecuación en la ecuación [7]

$$\ddot{e}_{\theta_t} = W_0^2 * (e_{\theta_t} - \frac{1}{W_0^2} * (k_2 * \dot{e}_{\theta_t} + k_1 * e_{\theta_t} + k_o * \int e_{\theta_t} dt - e_{\theta_t}))$$

$$\ddot{e}_{\theta_t} = W_0^2 * (-\frac{1}{W_0^2} * (k_2 * \dot{e}_{\theta_t} + k_1 * e_{\theta_t} + k_o * \int e_{\theta_t} dt))$$

$$\ddot{e}_{\theta_t} = -(k_2 * \dot{e}_{\theta_t} + k_1 * e_{\theta_t} + k_o * \int e_{\theta_t} dt)$$

$$\ddot{e}_{\theta_t} = -(k_2 * \ddot{e}_{\theta_t} + k_1 * \dot{e}_{\theta_t} + k_o * e_{\theta_t})$$

$$\ddot{e}_{\theta_t} + k_2 * \ddot{e}_{\theta_t} + k_1 * \dot{e}_{\theta_t} + k_o * e_{\theta_t} = 0 \quad [9]$$

Esta ecuación [9] que hemos conseguido es la ecuación de la dinámica del error e_{θ_t} en lazo cerrado.

Los parámetros de diseño $\{k_2, k_1, k_o\}$, se eligen entonces para convertir el polinomio característico del lazo cerrado, en un polinomio de Hurwitz con raíces deseables. Sin embargo, en lugar de incorporar un sensor, utilizaremos un estimador.

Este estimador lo conseguimos integrando en una ocasión la ecuación [7]:

$$\dot{e}_{\theta_t} = W_0^2 * (e_{\theta_m} - e_{\theta_t})$$

$$[\dot{e}_{\theta_t}]_e = W_0^2 * \int_0^t (e_{\theta_m} - e_{\theta_t}) dt \quad [10]$$

Sustituyendo [10] en [8] obtenemos:

$$e_{\theta_m} = e_{\theta_t} - \frac{1}{W_0^2} * (k_1 * e_{\theta_t} + k_o * \int e_{\theta_t} dt + k_2 * W_0^2 \int e_{\theta_m} dt - k_2 * W_0^2 \int e_{\theta_t} dt)$$

A continuación tomamos la transferencia de Laplace y desarrollamos:

$$E\theta_m(s) * W_0^2 = W_0^2 * E\theta_t(s) - \frac{k_o}{s} E\theta_t(s) - k_1 E\theta_t(s) - \frac{k_2 * W_0^2}{s} E\theta_m(s) + \frac{k_2 * W_0^2}{s} E\theta_t(s)$$

$$E\theta_m(s) \left[W_0^2 + \frac{k_2 * W_0^2}{s} \right] = E\theta_t(s) \left[(W_0^2 - k_1) + \left(\frac{k_2 * W_0^2 - k_0}{s} \right) \right]$$

$$E\theta_m(s) \left[\frac{s * W_0^2 + k_2 * W_0^2}{s} \right] = E\theta_t(s) \left[\frac{s(W_0^2 - k_1) + (k_2 * W_0^2 - k_0)}{s} \right]$$

$$\frac{E\theta_m(s)}{E\theta_t(s)} = \frac{s(W_0^2 - k_1) + (k_2 * W_0^2 - k_0)}{s * W_0^2 + k_2 * W_0^2} = \frac{s \left(1 - \frac{k_1}{W_0^2} \right) + \left(k_2 - \frac{k_0}{W_0^2} \right)}{s + k_2}$$

Por tanto, el controlador queda formulado como sigue:

$$C_1(s) = \frac{E\theta_m(s)}{E\theta_t(s)} = \frac{s Y_1 + Y_0}{s + Y_2}$$

Donde,

$$Y_1 = 1 - \frac{k_1}{W_0^2}$$

$$Y_0 = k_2 - \frac{k_0}{W_0^2}$$

$$Y_2 = k_2$$

Despejando θ_m obtenemos el valor final del ángulo del motor :

$$(\theta_m - \theta_m^*) = C_1(s) * (\theta_t - \theta_t^*)$$

$$(\theta_m^* - \theta_m) = C_1(s) * (\theta_t^* - \theta_t)$$

$$\theta_m = \theta_m^* - C_1(s) * (\theta_t^* - \theta_t)$$

Para obtener los valores de Y_0 , Y_1 y Y_2 , tenemos en cuenta solo el lazo exterior y obtenemos la siguiente ecuación característica:

$$1 + C_1(s) * G_b(s) = 0$$

Sustituyendo y operando obtenemos:

$$1 + \frac{s Y_1 + Y_0}{s + Y_2} * \frac{W_0^2}{s^2 + W_0^2} = 0$$

$$(s + Y_2) * (s^2 * W_0^2) + (s * Y_1 + Y_0) * W_0^2 = 0$$

$$s^3 + Y_2 * s^2 + W_0^2 (1 + Y_1) * s + W_0^2 (Y_0 + Y_2) = 0$$

Escogemos un polinomio e identificamos, en este caso:

$$(s + a)^3 = s^3 + 3 * a * s^2 + 3 * a^2 * s + a^3 = 0$$

$$Y_2 = 3 * a$$

$$Y_1 = \frac{3 * a^2}{\omega_0^2} - 1$$

$$Y_0 = \frac{a^2}{\omega_0^2} - Y_2$$

Una vez conseguidos todos los datos, procedemos a introducirlos en Matlab dando un valor a, que representará la localización deseada de los polos, por ejemplo, a=12.

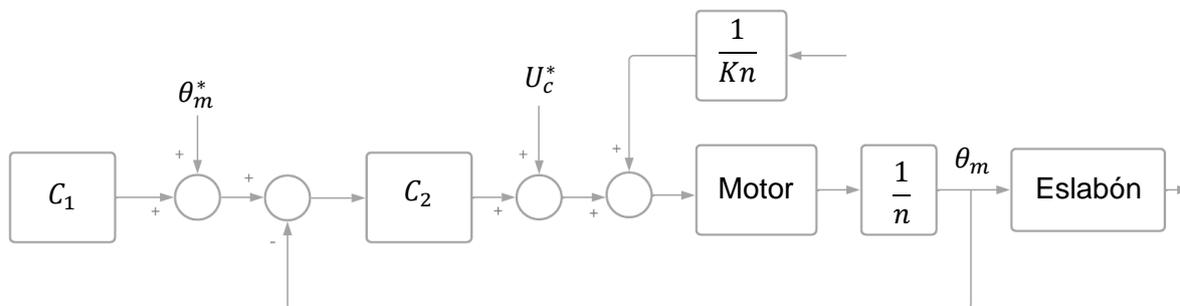
```

%CALCULO DEL CONTROLADOR EXTERNO-->C1(s)
a=12;
Gamm2=3*a;
Gamm1=((3*a^2)/(w0^2))-1;
Gamm0=(((a)^3)/(w0^2))-Gamm2;
%Controlador G1(s)
C1=tf([Gamm1 Gamm0],[1 Gamm2])
    
```

Figura 29. Controlador lazo externo introducido en Matlab.

5.5. Diseño del Lazo Interno del Control GPI

La situación de partida es la siguiente:



Para comenzar se propone este controlador de lazo interno C_2 , que incorpora parte derivada, proporcional, integral e integral doble:

$$e_v = \frac{v}{K} * [\dot{e}_{\theta_m}]_e + \frac{J}{K} * [-k_3 * [\dot{e}_{\theta_m}]_e - k_2 * e_{\theta_m} - k_1 \int_0^t e_{\theta_m} dt - k_0 \iint [e_{\theta_m} dt dt] \quad [11]$$

Al igual que en el anterior controlador de lazo externo C_1 que diseñamos, en lugar de medir la señal de la velocidad del error \dot{e}_{θ_m} , usaremos un reconstructor integral de la siguiente manera:

$$[\dot{e}_{\theta_m}]_e = \frac{K}{J} \int_0^t e_v dt - \frac{v}{J} * e_{\theta_m} \quad [12]$$

Esta ecuación se ha obtenido integrando la ecuación del motor, cambiando a las señales definidas actualmente.

$$J * \ddot{\theta}_m + v * \dot{\theta}_m = K * e_v$$

$$\ddot{\theta}_m = \frac{K}{J} * e_v - \frac{v}{J} * \dot{\theta}_m$$

Sabiendo esto, seguimos con el procedimiento e introducimos la ecuación [12] en la ecuación [11].

$$e_v(s) = \frac{v}{k} * \left[\frac{K}{J} * \frac{e_v}{s} - \frac{v}{J} * e_{\theta_m} \right] + \frac{J}{K} * \left[-K_3 * \frac{K}{J} * \frac{e_v}{s} + K_3 * \frac{v}{J} * e_{\theta_m} - K_2 * e_{\theta_m} - K_1 * \frac{e_{\theta_m}}{s} - K_0 * \frac{e_{\theta_m}}{s^2} \right]$$

$$e_v(s) = \frac{e_v}{s} * \left[\frac{v}{J} - K_3 \right] + e_{\theta_m} * \left[\frac{-v^2}{J * K} + K_3 * \frac{v}{K} - K_2 * \frac{J}{K} \right] + \frac{e_{\theta_m}}{s} * \left[\frac{-J}{K} * K_1 \right] + \frac{e_{\theta_m}}{s^2} * \left[-K_0 * \frac{J}{K} \right]$$

$$e_v * \left[1 - \frac{v - K_3 * J}{J * s} \right] = e_{\theta_m} * \left[a + \frac{b}{s} + \frac{c}{s^2} \right]$$

$$e_v * \left[\frac{s + d}{s} \right] = e_{\theta_m} * \left[\frac{a * s^2 + b * s + c}{s^2} \right]$$

$$e_v = \frac{a * s^2 + b * s + c}{s * (s + d)} * e_{\theta_m}$$

Donde a, b y c ya han sido señalizadas antes y $d = -\frac{v}{J} - k_3$

La última ecuación se podría reescribir de esta manera:

$$U_C - U_c^* = \frac{a * s^2 + b * s + c}{s * (s + d)} * (\theta_{mr}^* - \theta_m)$$

Y obtenemos la U_c como habíamos visto en el diagrama de bloques:

$$U_c = C_2(s) * (\theta_{mr}^* - \theta_m) + U_c^*$$

Para obtener U_c^* usamos la ecuación del motor que definimos al principio del proceso:

$$G_m(s) = \frac{\hat{\theta}_m(s)}{U_d(s)} = \frac{A}{s * (s + B)}$$

$$U_c^*(t) = A^{-1} * \ddot{\theta}_m^* + B * A^{-1} * \dot{\theta}_m^*$$

Para obtener los valores de a, b, c y d usamos la siguiente ecuación característica del lazo cerrado:

$$1 + C_2(s) * G_m(s) = 0$$

$$1 + \frac{a * s^2 + b * s + c}{s * (s + d)} * \frac{A}{s * (s + B)} = 0$$

$$s^2 * (s + d) * (s + B) + A * (a * s^2 + b * s + c) = 0$$

$$s^4 + (B + d) * s^3 + (B * d + a * A) * s^2 + A * b * s + c * A = 0$$

Ahora generamos un comportamiento deseado con un polinomio $(s + p)^4 = 0$ e igualamos términos:

$$s^4 + 4 * p * s^3 + 6 * p^2 * s^2 + 4 * p^3 * s + p^4 = 0$$

Donde obtenemos que

$$d = 4 * p - B$$

$$a = \frac{6 * p^2 - B * d}{A}$$

$$b = \frac{4 * p^3}{A}$$

$$c = \frac{p^4}{A}$$

Una vez conseguidos todos los datos, los introducimos en Matlab dando un valor a, que representará la localización deseada de los polos, por ejemplo, $p=110$.

```
%CALCULO DEL CONTROLADOR INTERNO-->C2(s)
p=110;

cr2=(p^4)/A;
br2=(4*p^3)/A;
dr2=4*p-B;
ar2=(6*p^2-B*dr2)/A;
%Controlador C2(s)
C2=tf([ar2 br2 cr2],[1 dr2 0])
```

Figura 30. Controlador lazo interno introducido en Matlab.

5.6. Simulación del conjunto con el Control GPI de Doble Lazo

Una vez tenemos definidos los dos controladores, procedemos a definir el resto de variables (deseadas) que nos faltan por definir en Matlab, en este caso son U_c^* , θ_m^* y θ_t^* .

```

%Muestreo
h=0.05;
t=0:h:5;
l=length(t);
ui=ones(1,l);
%Filtrado:
pp=4;
den=conv(conv([1 pp],[1 pp]),[1 pp]);
Gf=tf(den(end),den);
theta_t=lsim(Gf,ui,t);

%GENERACION DE Theta_m*
theta_t_d=diff(theta_t);
theta_t_dd=diff(theta_t_d);
aux1=length(theta_t_dd);
theta_t=theta_t(1:aux1); %Ajusto longitud vector.
theta_m_ast=(1/w0^2)*theta_t_dd+theta_t;

%GENERACION DE Uc*
theta_m_ast_d=diff(theta_m_ast);
theta_m_ast_dd=diff(theta_m_ast_d);
aux2=length(theta_m_ast_dd);
theta_m_ast_d=theta_m_ast_d(1:aux2);%Ajusto longitud vector.
Uc_ast=(1/A)*theta_m_ast_dd+(B/A)*theta_m_ast_d;

%Ajusto definitivamente Longitudes de vectores e incorporo el tiempo:
theta_t_ast=[t(1:aux2)',theta_t(1:aux2)];
theta_m_ast=[t(1:aux2)',theta_m_ast(1:aux2)];
Uc_ast=[t(1:aux2)',Uc_ast(1:aux2)];

```

Figura 31. Variables deseadas introducidas en Matlab.

Con todas las variables definidas creamos el diagrama completo en simulink y lo acoplamos al bloque “Motor and bar” que definimos en el apartado 7.1.

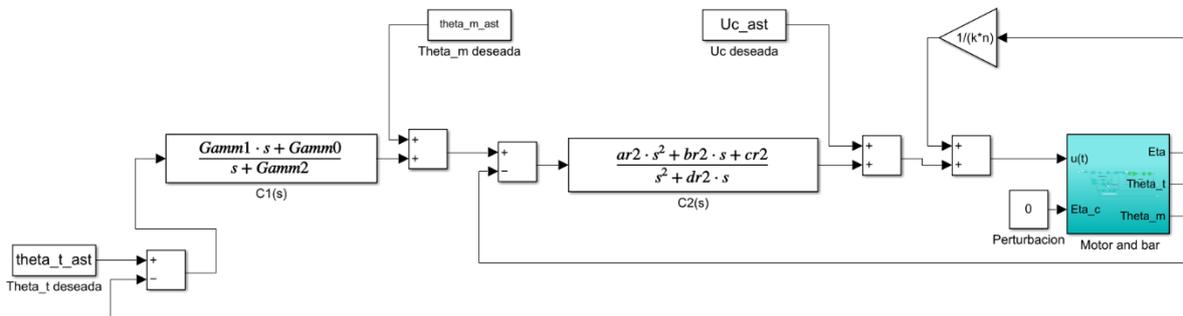


Figura 32. Diagrama de bloques completo en Simulink.

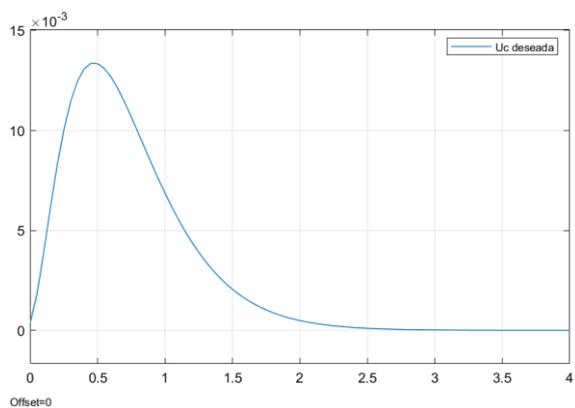


Figura 33. Valor de Uc deseada.

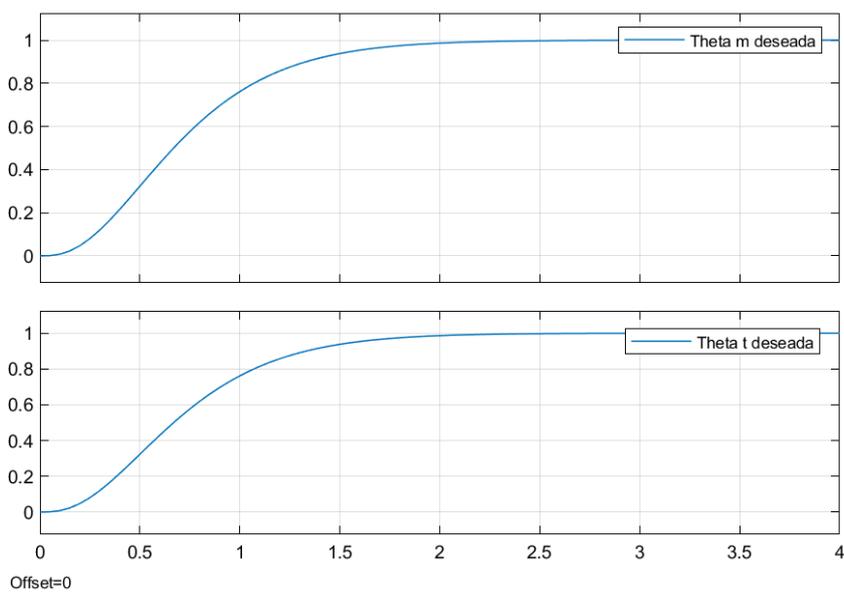


Figura 34. Valor de θ_m^* y θ_t^* .

Como se puede apreciar ambos valores son idénticos, ya que buscamos que el robot se comporte como si el eslabón flexible fuese rígido y, en ese caso, el ángulo a la salida del reductor será idéntico al del extremo del segmento

A continuación realizaremos la simulación únicamente con el controlador de lazo interno C_2 .

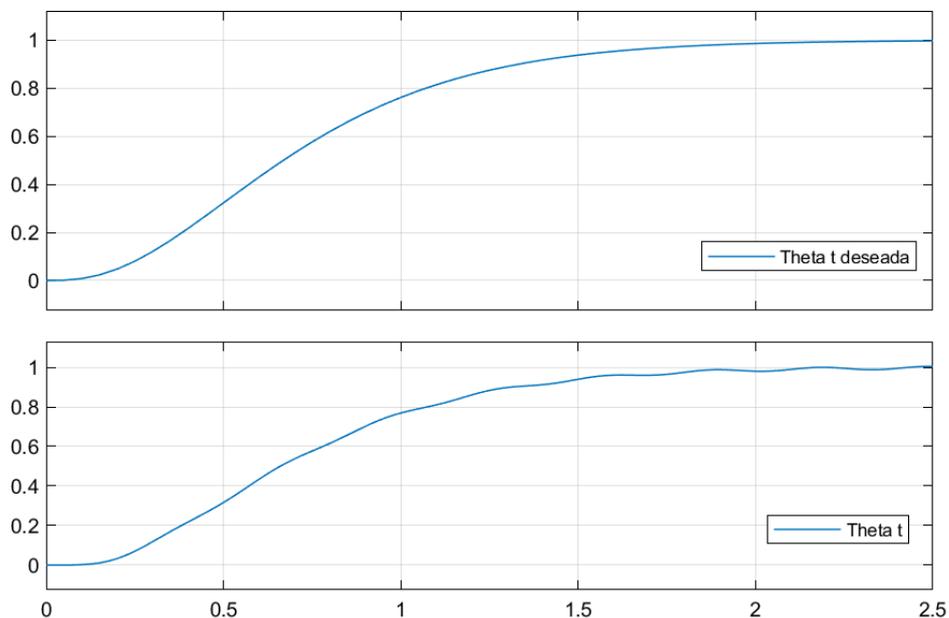


Figura 35. Valor de θ_t^* y de θ_t con un solo controlador

Como podemos observar, la respuesta final del sistema se asemeja en gran porcentaje a la respuesta deseada, aunque termina con una continua perturbación que no sería lo ideal, para ello añadimos el controlador de lazo externo C_1 y simulamos con el circuito completo.

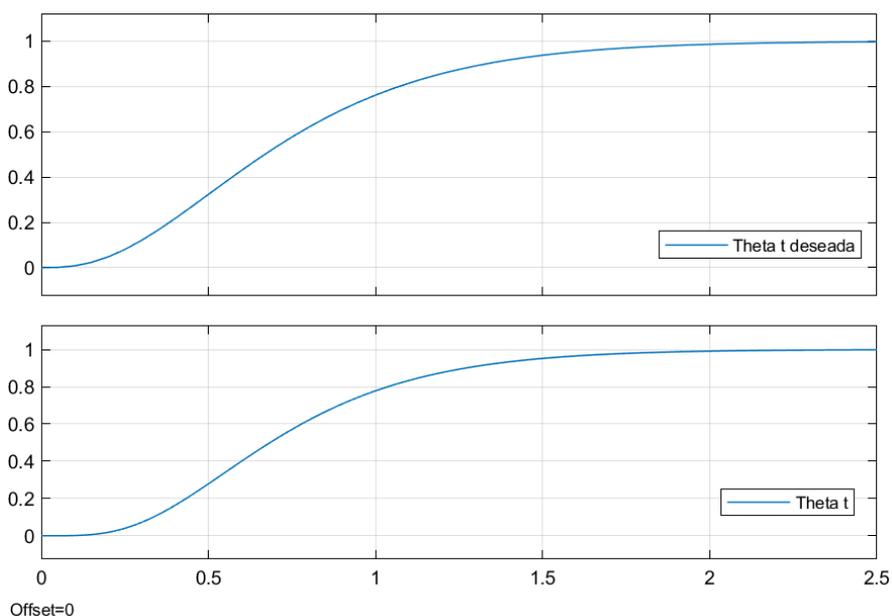


Figura 36. Valor de θ_t^* y de θ_t

Como podemos comprobar, ahora la salida es mucho más precisa, prácticamente idéntica, para ver la diferencia superpondremos ambas gráficas.

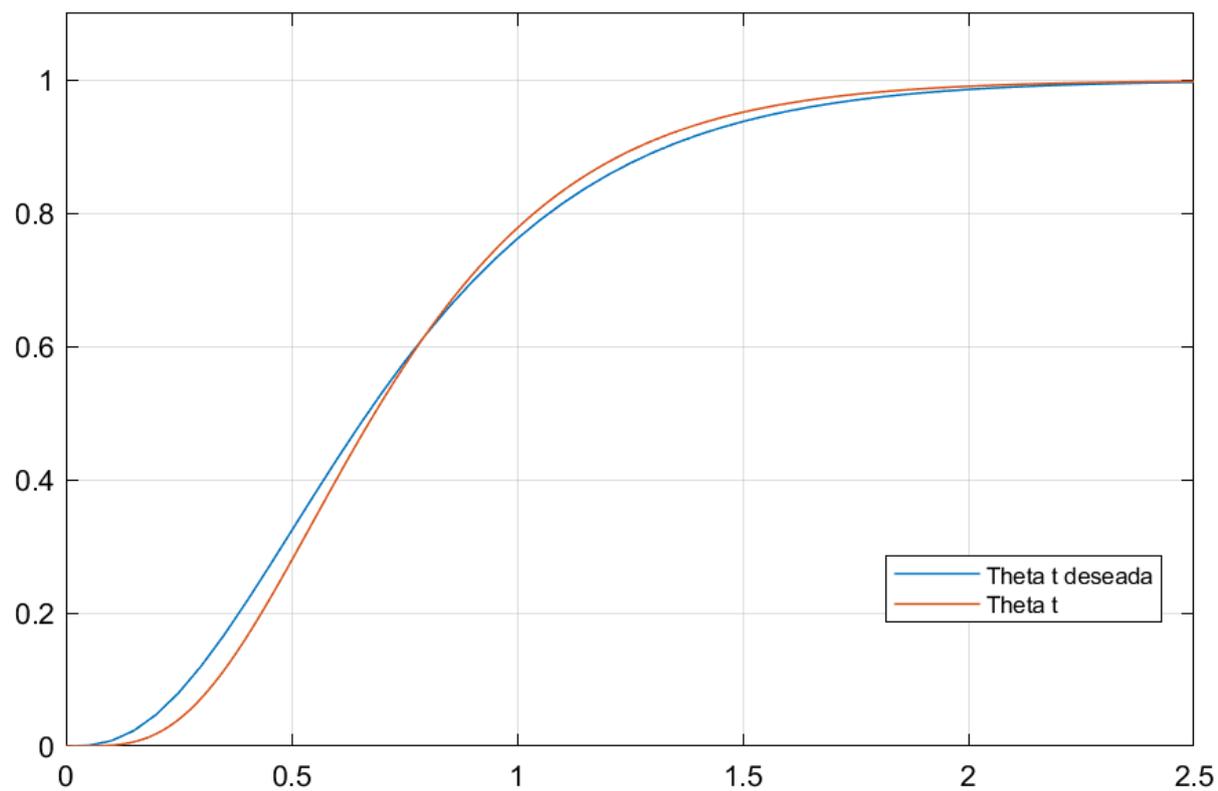


Figura 37. Valor de θ_t^* frente a θ_t

Aquí podemos apreciar mejor como el recorrido es algo distinto aunque la posición final es la deseada, por lo que el control GPI ha resultado ser un éxito

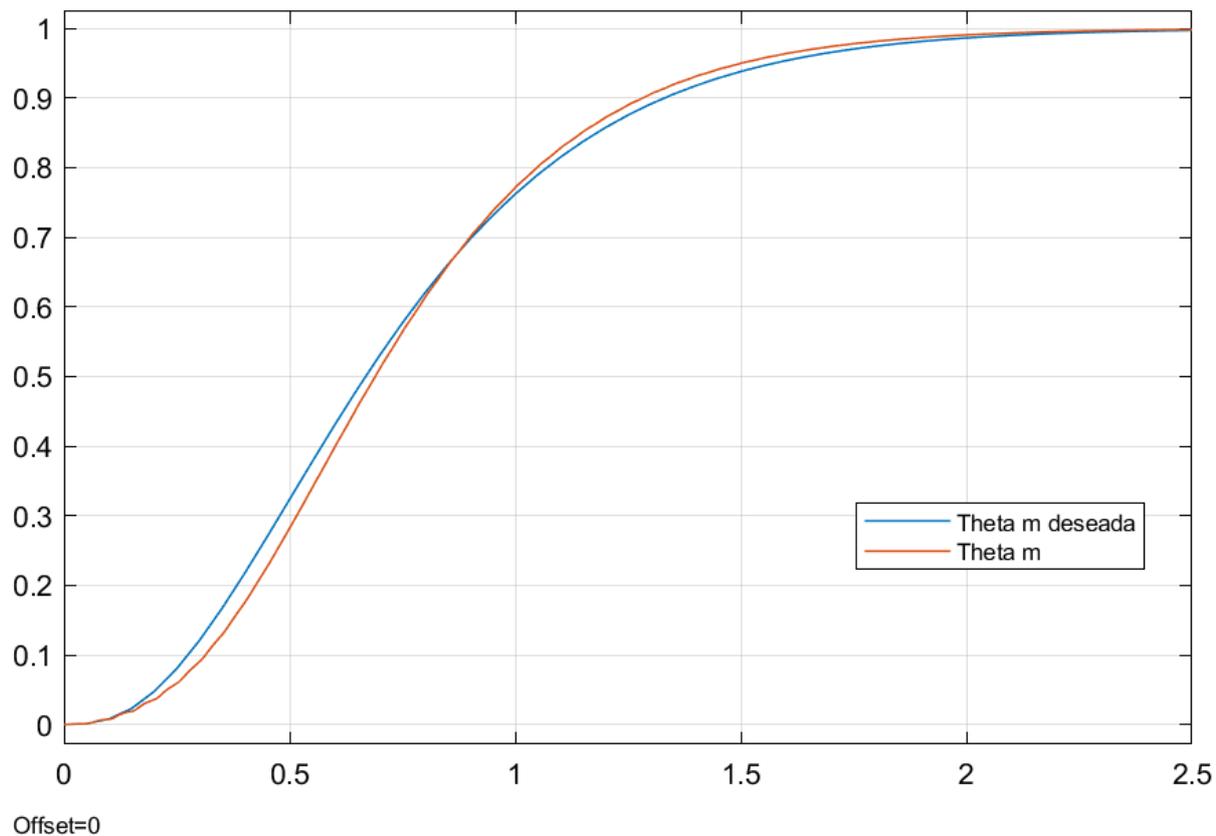


Figura 38. Valor de θ_m^* frente a θ_m

Como podemos comprobar el ángulo del motor también es muy preciso respecto ángulo deseado desde el principio.

6. CONTROL FUZZY

Los avances surgidos en el campo de la robótica en los últimos años han permitido el desarrollo de nuevos robots, a la par que aparecen nuevos entornos de programación que incorporan las sinergias procedentes de nuevos modelos de robots, una nueva sensorica asociada y entornos de trabajo que permiten la interacción de los robots con entornos dinámicos (Tibaduiza, et al., 2011).

Desde los primeros robots han surgido otros más actuales y adaptados a diferentes medios que incorporan nuevos modelos en el desarrollo de controladores a partir de los cuales probar diferentes estrategias de control, tanto desde una perspectiva cinemática como dinámica. En este contexto, técnicas como el fuzzy permiten disponer de una potente herramienta para obtener una solución a problemas impredecibles (Alvandar y Nigam, 2008).

El control difuso es una técnica agrupada en la temática de los sistemas expertos, y que debe su amplio conocimiento a la dificultad o imposibilidad para la modelización a través de sistemas de ecuaciones matemáticas. No obstante, al mismo tiempo, presenta una importante utilidad y es el aprovechamiento del conocimiento de los expertos (Looney, 1988), sin que se tenga que modelizar matemáticamente el proceso, haciéndolo si cabe más útil en el caso de los procesos altamente complejos.

La lógica fuzzy o lógica difusa es una “extensión de la lógica clásica, diseñada para permitir el razonamiento sobre conceptos imprecisos” (Escolano et al., 2001, p. 184), y forman parte de las aplicaciones el “desarrollo de sistemas expertos difusos para el reconocimiento de patrones y clasificación, fuzzy clustering, control fuzzy, monitoreo, diagnóstico, optimización” (Tibaduiza y Daza, 2008, p. 138).

A lo largo de varias décadas se han venido produciendo importantes avances en la lógica fuzzy o lógica difusa, entre los que se pueden citar los siguientes:

- Híbrido de control difuso y red neuronal. Desarrollado por Daxwanger y Schmidt (1995). Su intención es la de imitar el conocimiento de un experto.
- Bentalba, Hajjaji y Rachid (1999). proponen un sistema fuzzy para controlar un robot móvil al mismo tiempo que se logra estabilizar la respuesta de dos sistemas lineales.
- Yang (2004). Sus estudios se centran en un robot no-holonómico. Entre las aportaciones realizadas por este estudio se encuentra la definición de una serie de

reglas basadas en el método directo de Lyapunov, que dan como resultado el ángulo de giro que se pretendía y un seguimiento satisfactorio de la trayectoria.

- Raimondi (2005). A partir del método Lyapunov y del lema de Barbalat, estudian el problema de seguimiento de trayectoria.
- Mazhari y Kumar (2008) logran sintonizar un controlador PID-Fuzzy para un robot PUMA 560 a través de la comparación de técnicas heurísticas de búsqueda global, algoritmos genéticos, optimización por enjambre de partículas, algoritmos genéticos híbridos con Nelder-Mead o búsqueda generalizada de patrones, entre otros elementos.
- Piltan et al., (2011), aplicaron el control fuzzy deslizante a un robot PUMA con tres grados de libertad, logrando un control fuzzy con una estructura en paralelo de una superficie deslizante $P + D$ y otra $P + I + D$.
- Zheng-jie (2010). Diseña un modelo de control combinado, compuesto por un modo de control aerodinámico y un modo de control de aceleración lateral directa, mejorando la velocidad de respuesta y reduciendo el nivel de sobrecarga en los cambios.
- Lin et al., (2010) aplicaron la lógica fuzzy a la estabilización de un robot bípedo. Entre sus logros se encuentra el de conseguir un desplazamiento en plano horizontal y en el ascenso de los escalones, al mismo tiempo que lograban reducir la inestabilidad y la duración de las oscilaciones transitorias.

Estudios más recientes como el publicado por Barbosa et al., (2017) sobre la utilización de robots de inspección en la limpieza de tanques esféricos de grandes dimensiones, en los que dichos robots trepadores deben navegar dentro de una esfera con una perturbación gravitatoria variable, muestran que existen diferentes enfoques sobre los sistemas de adherencia flexible.

Entre estos diferentes enfoques, se encuentra el propuesto por Zhu et al. (2015), quienes presentaron un método para la detección autónoma de la postura y la alineación del módulo de succión para la escalada de paredes con un robot bípedo. Lee et al., (2015), mostraron una plataforma robótica con un sistema de adhesión basado en la succión y una gran capacidad de carga útil para la limpieza de edificios. Koh et al., (2016), desarrollaron un robot trepador con un sistema de adhesión híbrido basado en la electrostática y el elastómero. Schmidt y Berns (2013) también presentaron un estudio de robots trepadores para el

mantenimiento y la realización de inspecciones. Los trabajos mencionados, en general, se basan en robots cuyo principal interés se basa en la adherencia a las superficies, pero suelen ignorar las características o restricciones propias de las tareas de inspección que realizan, como por ejemplo, las perturbaciones en las ruedas magnéticas, la fricción con las superficies, etc. Ante esto, la aportación de Barbosa et al (2017), presenta la segunda generación del robot de inspección autónomo (AIR-2), basado en cuatro ruedas magnéticas dirigibles independientes diseñadas para un movimiento cuasi-omnidireccional sobre tanques de almacenamiento.

Dicho robot está desarrollado con una topología de diamante bipartita que permite la superación de los pequeños obstáculos, como los cordones de soldadura, al mismo tiempo que mantiene una óptima adherencia magnética, gracias al uso de controladores fuzzy programados para permitir el movimiento de inspección del robot con un comportamiento casi omnidireccional sin desplazamiento lateral, de manera que se garantiza la máxima adherencia en cualquier configuración (Barbosa, et al., 2017).

Concretamente, el control difuso logra reducir la complejidad del controlador a través de un sencillo mecanismo que permite incrustar el conocimiento sobre el comportamiento del sistema en el controlador, al mismo tiempo que permite minimizar las influencias de las perturbaciones en el rendimiento del control (Karim y Ardestani, 2016; Liu, 2009).

Los controles de posición y velocidad propuestos para el AIR-2 de Barbosa et al., (2017) se basan en el enfoque Fuzzy, de manera que el controlador de posición se implementa en cascada con el controlador de velocidad, tal y como se muestra en la siguiente figura, que reproduce la alineación mecánica del movimiento del robot AIR-2 para la orientación en el movimiento en superficies esféricas:

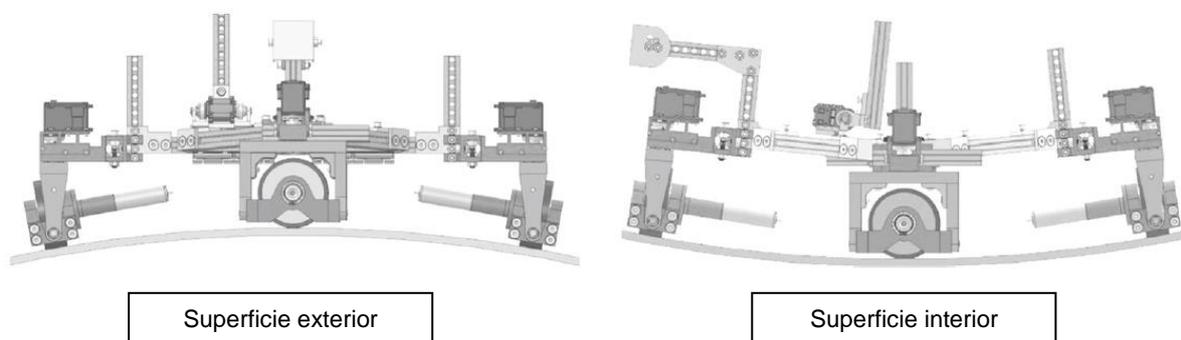


Figura 39. Alineación mecánica del robot AIR-2 para el movimiento en superficies esféricas

Fuente: Barbosa et al., (2017)

6.1. Diseño

Iniciamos el proceso creando el diagrama de lazo cerrado genérico de un controlador de lógica difusa, en este diagrama se encuentra, una entrada de referencia $T_d(t)$, el controlador tipo fuzzy, la planta que vamos a modelar, en este caso, el eslabón flexible con el motor, y por último la salida del diagrama $T(t)$

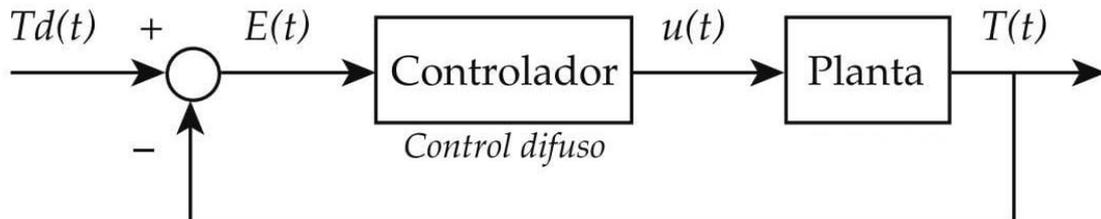


Figura 40. Diagrama básico con un controlador de lógica difusa.

Fuente: Castaño (2006)

Ampliando el controlador fuzzy obtenemos la siguiente distribución de bloques:

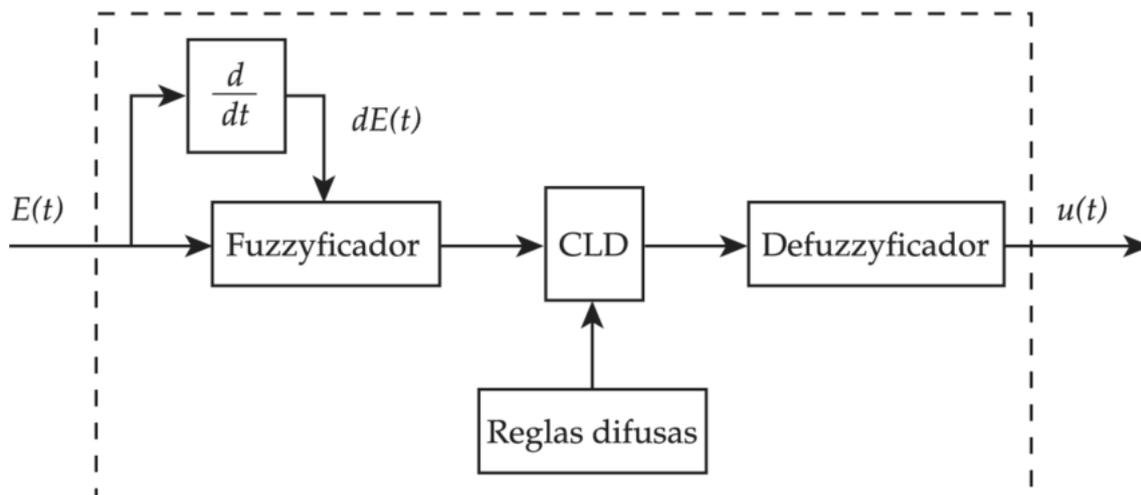


Figura 41. Diagrama de un controlador de lógica difusa.

Fuente: Castaño (2006)

Donde $E(t)$ es el error entre la entrada de referencia y la salida, este error a su vez se deriva para conseguir $dE(t)$, ya que estamos interesados en conocer el cambio del error que está ocurriendo. La derivada deberá pasar por un bloque de saturación para que se mantenga en un límites preestablecidos. Estas dos señales, serán las dos entradas del bloque controlador Fuzzy

A la salida del controlador nos encontramos con un bloque integrador, parte del PI, al que también se le puede llamar bloque defuzzyficador, después de esto también añadiremos una saturación para que la entrada $u(t)$ no sobrepase unos valores determinados.

A continuación montaremos nuestro diagrama de bloques en simulink, los valores del bloque del motor y el eslabón serán los anteriormente usados en la figura 21 a su vez la entrada de referencia que usaremos en este caso será la θ_t^* (theta_t_ast) que ya conseguimos en la figura 31.

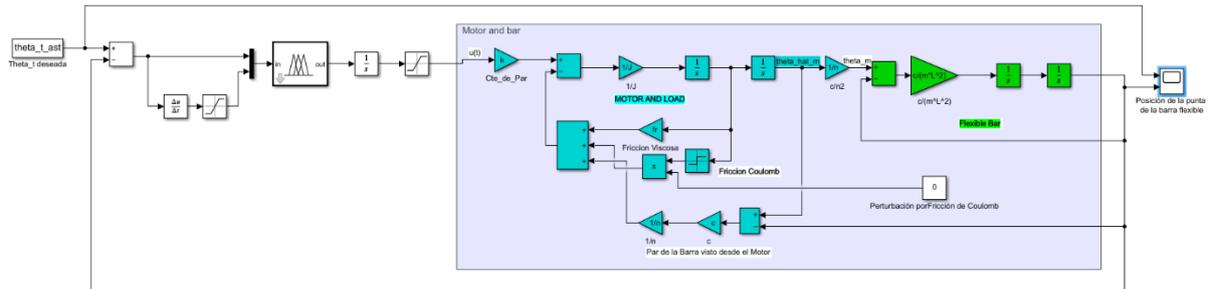


Figura 42. Diagrama de bloques con controlador de lógica difusa en simulink.

Seguidamente comenzaremos a diseñar nuestro controlador de lógica difusa gracias al comando “fuzzy” en Matlab, que nos permitirá diseñar mediante una toolbox, dicho controlador.

Lo primero sería establecer las dos entradas, y una única salida:

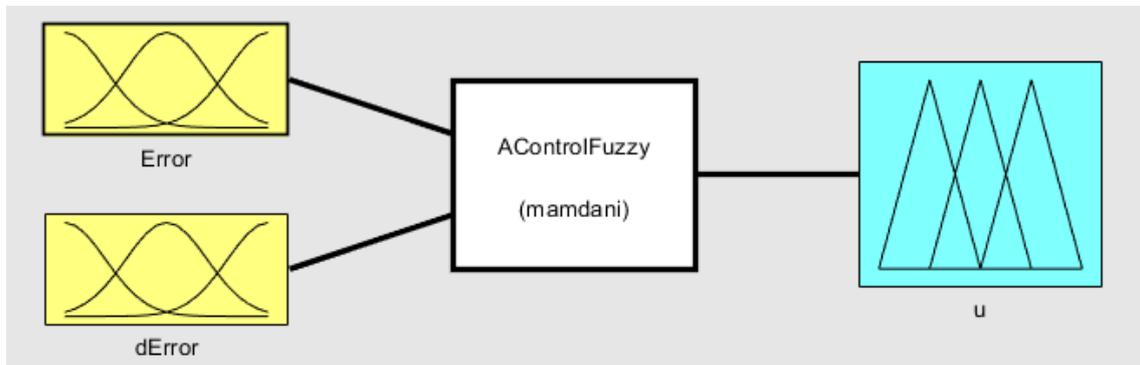


Figura 43. Entradas y salida del controlador Fuzzy en Matlab.

Lo siguiente es definir las funciones de pertenencia de los tres bloques, estas funciones son unas curvas que definen el grado (entre 0 y 1) en que las variables pertenecen a los conjuntos difusos propuestos.

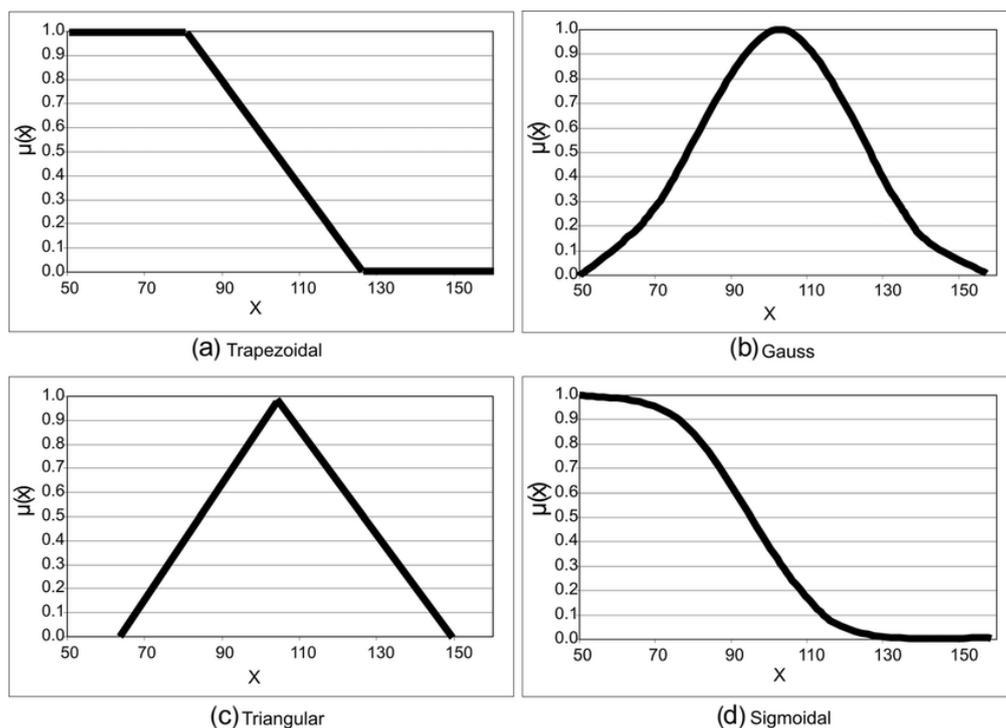


Figura 44. Distintas funciones de pertenencia para las variables fuzzy.

Fuente: researchgate.net

Como podemos observar, existen distintas opciones a la hora de elegir dichas funciones de pertenencia. En este caso vamos a utilizar la forma gaussiana para los tres bloques, ya que en principio esa forma continua se ajusta muy bien a nuestras necesidades.

Lo siguiente que deberemos elegir, es el número de variables lingüísticas por bloque. Aquí tenemos muchas más opciones, pero si estipulamos que todos los bloques posean el mismo número de variables, podremos usar 3,5,7 o 9.

El número de variables es correlativo, es decir, si usamos 3 variables en cada bloque habrá $3 \times 3 = 9$ reglas, tanto que si usamos 7 variables, el número de reglas será de $7 \times 7 = 49$.

La base de reglas es la manera que tiene el sistema Fuzzy de guardar el conocimiento lingüístico que le permita resolver el problema para el cual ha sido diseñado. Estas reglas son del tipo If y Then.

La elaboración de las reglas del sistema difuso se realiza mediante la tabla FAM, la cual involucra todas las combinaciones posibles que puede presentarse en el proceso.

$\frac{e}{de}$	Negativo	Cero	Positivo
Negativo	Negativo	Negativo	Cero
Cero	Negativo	Cero	Positivo
Positivo	Cero	Positivo	Positivo

Tabla 1. Tabla FAM para 3 variables lingüísticas en cada bloque

$\frac{e}{de}$	Negativo Grande	Negativo Pequeño	Cero	Positivo Pequeño	Positivo Grande
Neg. G	NG	NG	NG	NP	C
Neg. P	NG	NG	NP	C	PP
Cero	NG	NP	C	PP	PG
Pos. P	NP	C	PP	PG	PG
Pos. G	C	PP	PG	PG	PG

Tabla 2. Tabla FAM para 5 variables lingüísticas en cada bloque

$\frac{e}{de}$	Negativo Grande	Negativo Mediano	Negativo Pequeño	Cero	Positivo Pequeño	Positivo Mediano	Positivo Grande
Neg. G	NG	NG	NG	NG	NM	NP	C
Neg. M	NG	NG	NG	NM	NP	C	PP
Neg. P	NG	NG	NM	NP	C	PP	PM
Cero	NG	NM	NP	C	PP	PM	PG
Pos. P	NM	NP	C	PP	PM	PG	PG
Pos. M	NP	C	PP	PM	PG	PG	PG
Pos. G	C	PP	PM	PG	PG	PG	PG

Tabla 3. Tabla FAM para 7 variables lingüísticas en cada bloque

En esta ocasión usaremos la última tabla, para que el controlador sea lo más preciso posible, las variables de dicha tabla son las siguientes:

La primera variable de entrada es el error, como hemos dicho anteriormente este es la diferencia que hay entre el valor de referencia, y los datos de la variable del sistema.

- ErrorNG (Error negativo grande): Indica que la salida del sistema está muy por encima del valor de referencia.

- ErrorNM (Error negativo medio): Indica que la salida del sistema está medianamente por encima del valor de referencia.
- ErrorNP (Error negativo pequeño): Indica que la salida del sistema está por encima pero con muy poca diferencia del valor de referencia.
- ErrorCero (Error cero) indica cuando la salida del sistema es igual al valor de referencia.
- ErrorPP (Error positivo pequeño): Indica que la salida del sistema está por debajo pero con muy poca diferencia del valor de referencia.
- ErrorPM (Error positivo medio): Indica que la salida del sistema está medianamente por debajo del valor de referencia.
- ErrorPG (Error positivo grande): Indica que la salida del sistema está muy por debajo del valor de referencia.

Este error puede ser hasta de media vuelta (π radianes), por lo que el rango de dicha variable aproximadamente variará desde -3.2 hasta 3.2

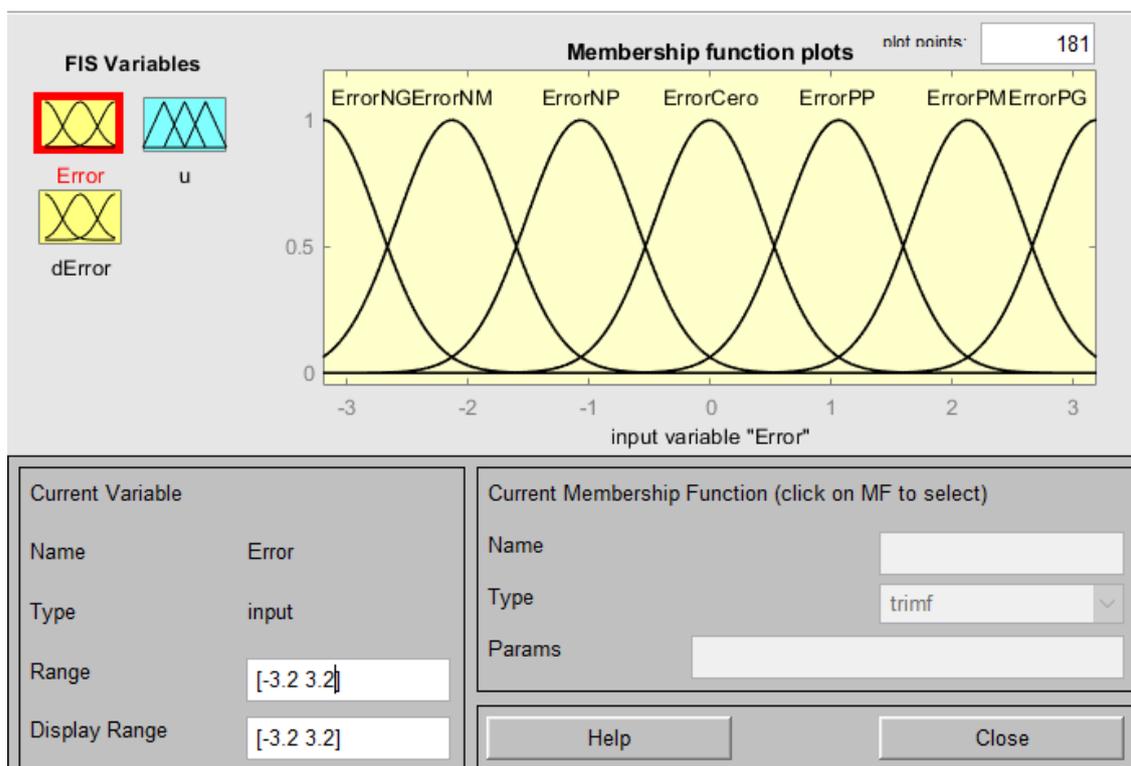


Figura 45. Diagrama de funciones para la variable Error.

La segunda variable de entrada para el sistema de control difuso es la derivada del error, esta entrada podría ser también una integral de dicho error, pero esto implicaría tener un universo del discurso demasiado amplio, por lo que se opta por usar el diferencial del error y después integrar la salida del sistema difuso.

- dErrorNG (Derivada error negativo grande): Señala que la salida posee una pendiente positiva.
- dErrorNM (Derivada error negativo mediano): Señala que la salida posee una pendiente medianamente positiva.
- dErrorNP (Derivada error negativo pequeño): Señala que la salida posee una pendiente positiva casi constante.
- dErrorCero (Derivada error cero): Señala que la salida permanece constante.
- dErrorPP (Derivada error positivo pequeño): Señala que la salida posee una pendiente negativa casi constante.
- dErrorPM (Derivada error positivo mediano): Señala que la salida posee una pendiente medianamente negativa.
- dErrorPG (Derivada error positivo grande): Señala que la salida posee una pendiente negativa.

Para hallar este rango, hay que realizar unas simulaciones de comprobación, en las que hemos conseguido hallar que un valor de 4 sería un error muy grande, por lo que este rango ocurrirá entre -4 y 4.

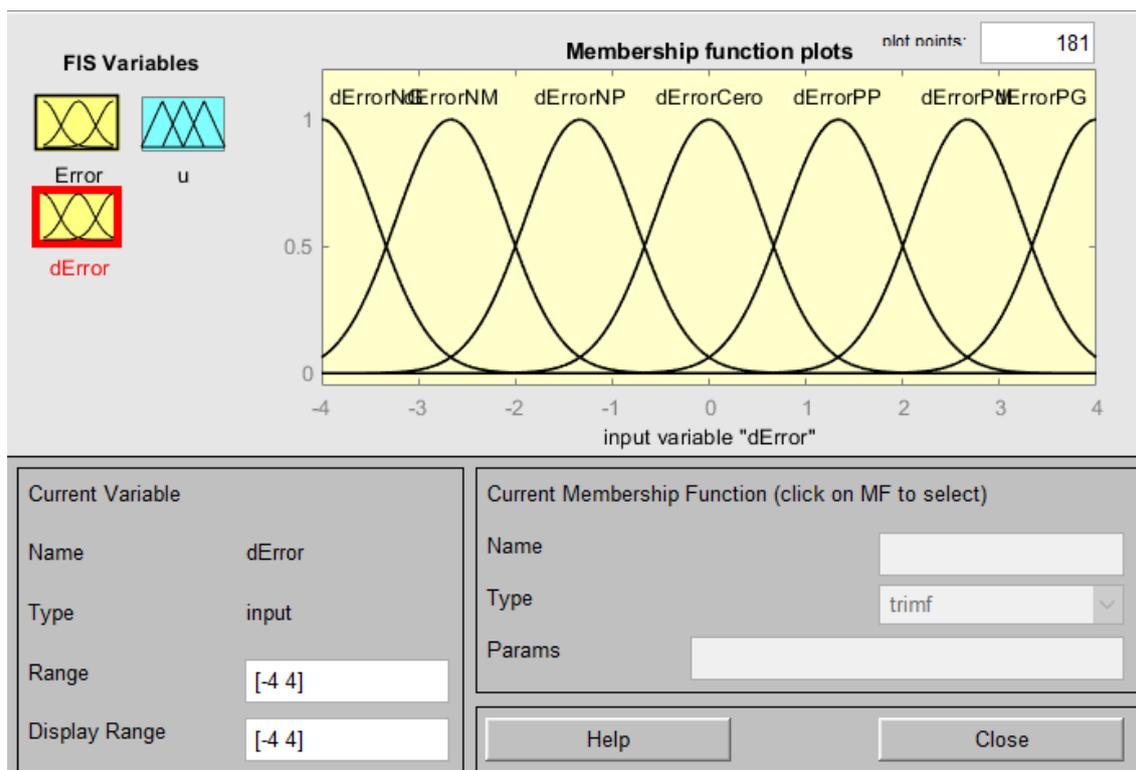


Figura 46. Diagrama de funciones para la variable dError.

Por último tenemos la variable de salida, que en este caso es el voltaje $u(t)$ que se suministrará a nuestro motor para que mueva el eslabón flexible.

- u_{NG} (Salida negativa grande): Cuando se necesita disminuir de manera inmediata el voltaje del motor.
- u_{NM} (Salida negativa mediana): Cuando se necesita disminuir de manera rápida el voltaje del motor.
- u_{NP} (Salida negativa pequeña): Cuando se necesita disminuir de manera lenta el voltaje del motor.
- u_{Cero} (Salida cero): Cuando no se necesita variar el voltaje del motor
- u_P (Salida positiva pequeña): Cuando se necesita aumentar de manera lenta el voltaje del motor.
- u_{PM} (Salida positiva mediana): Cuando se necesita aumentar de manera rápida el voltaje del motor.
- u_{PG} (Salida positiva grande): Cuando se necesita aumentar de manera inmediata el voltaje del motor.

El rango de la salida irá desde 0 voltios, con el motor apagado, hasta 7.5 voltios, que son suficientes para que el eslabón flexible consiga la posición deseada.

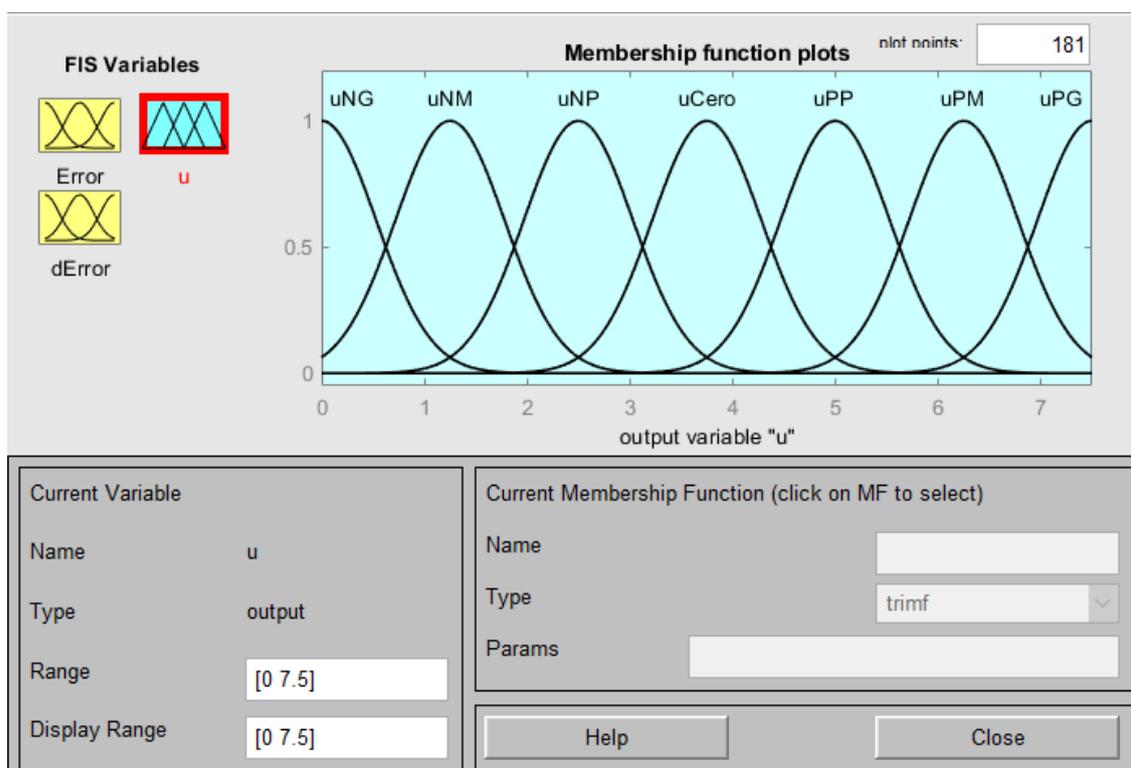


Figura 47. Diagrama de funciones para la variable de salida.

Matlab nos facilita la introducción de las 49 reglas gracias a la ventana 'Rule editor'

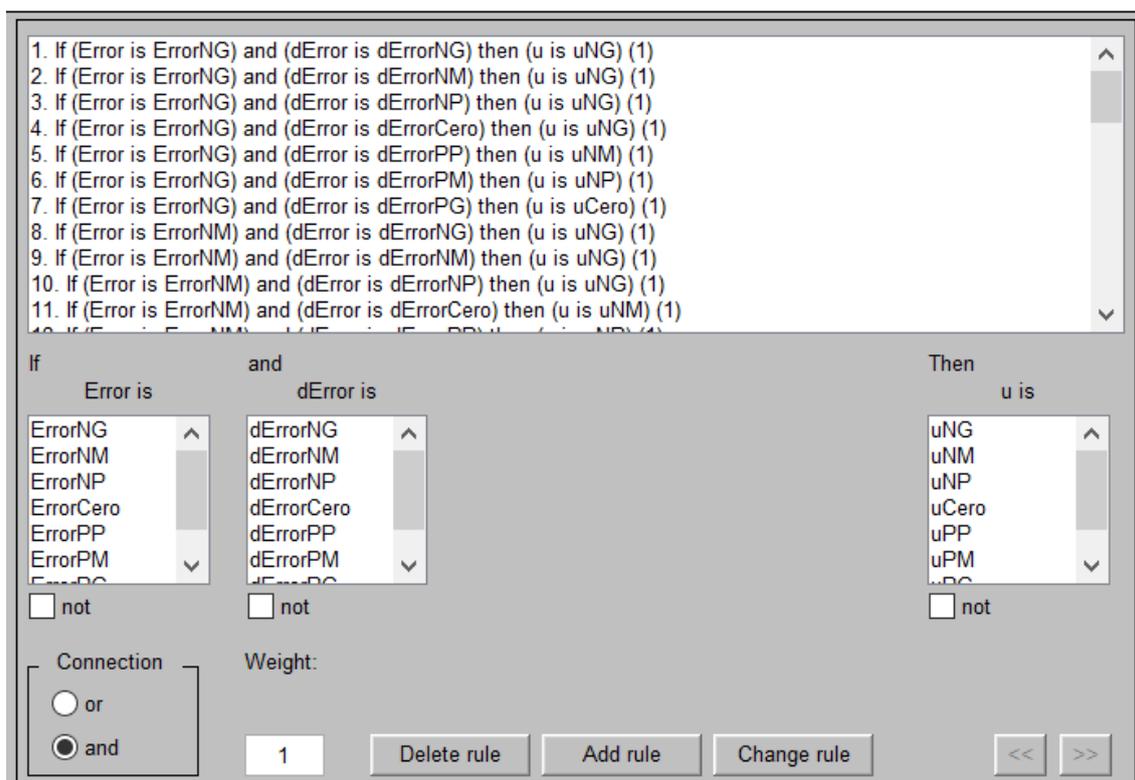


Figura 48. Reglas de inferencia en Matlab.

Como podemos observar en esta ventana aparecen 3 bloques que representan las 3 variables del controlador, en ellos definimos con un "If" y un "And" las variables de entrada al sistema y posteriormente con un "Then" la variable de salida correspondiente en tal caso. Después con el botón de "Add rule" añadiremos la regla que hemos creado, y habrá que repetir el proceso las veces que sean necesarias.

6.2. Comprobación de resultados por simulación

Una vez tenemos el bloque controlador de lógica difusa totalmente definido, podemos comenzar la simulación del sistema, para ello vamos a definir el tiempo de simulación en 30ms y mostraremos la salida del sistema respecto al valor de referencia propuesto.

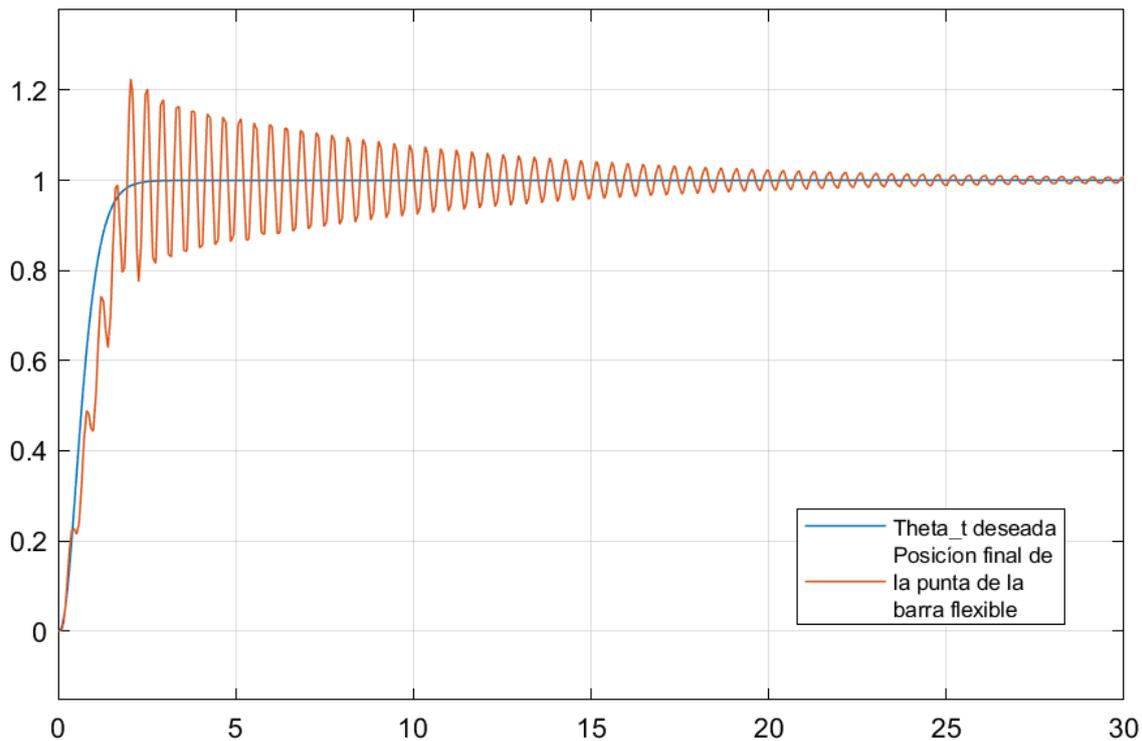


Figura 49. Valor de la salida deseada frente a la salida obtenida.

Como podemos observar la punta a lo largo del recorrido describe numerosas oscilaciones, siguiendo las reglas propuestas en el controlador de lógica difusa, estas oscilaciones eran inexistentes en el controlador GPI.

A continuación aumentaremos el tiempo de simulación y aumentaremos en los instantes finales para ver la posición de la punta.

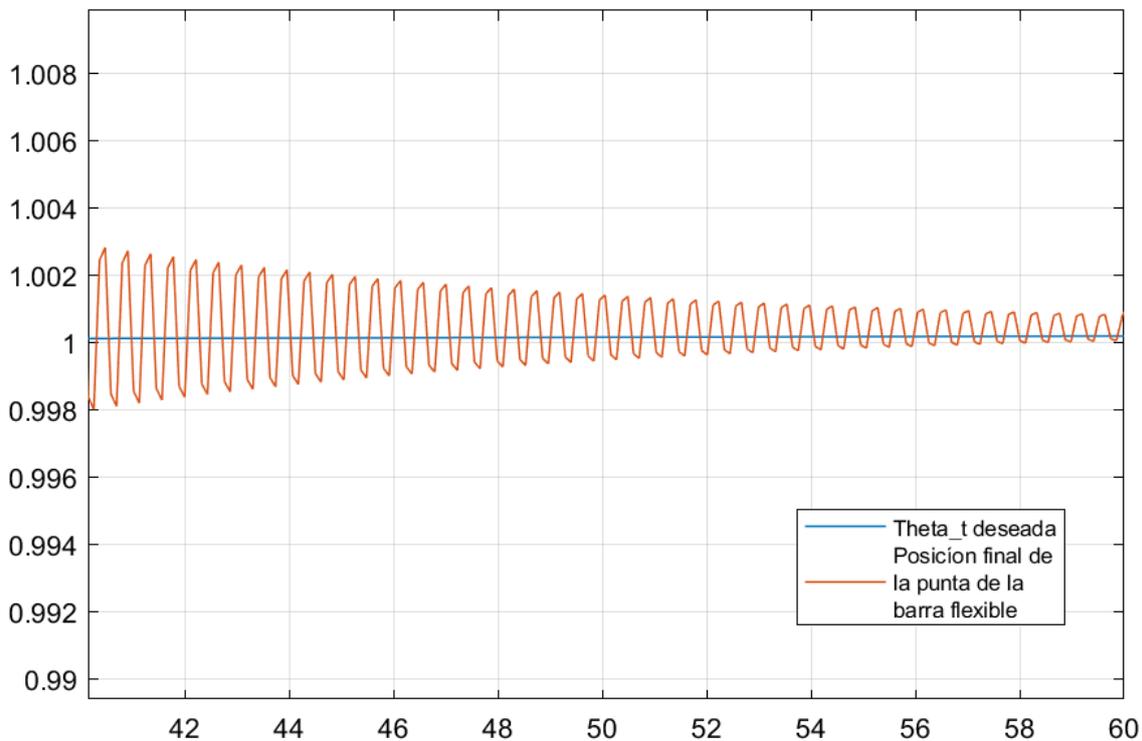


Figura 50. Valor de la salida deseada frente a la salida obtenida de forma aumentada.

Aunque podemos observar que existe alguna diferencia entre la salida deseada y la obtenida, esta cada vez es menor y acaba por no superar los 0.001 radianes de diferencia, teniendo en cuenta las oscilaciones producidas por el controlador de tipo fuzzy, es una salida bastante parecida.

Por último vamos a obtener la gráfica del error para ver el transcurso de dicha variable durante la simulación y comprobar si nuestras aproximaciones previas eran correctas.

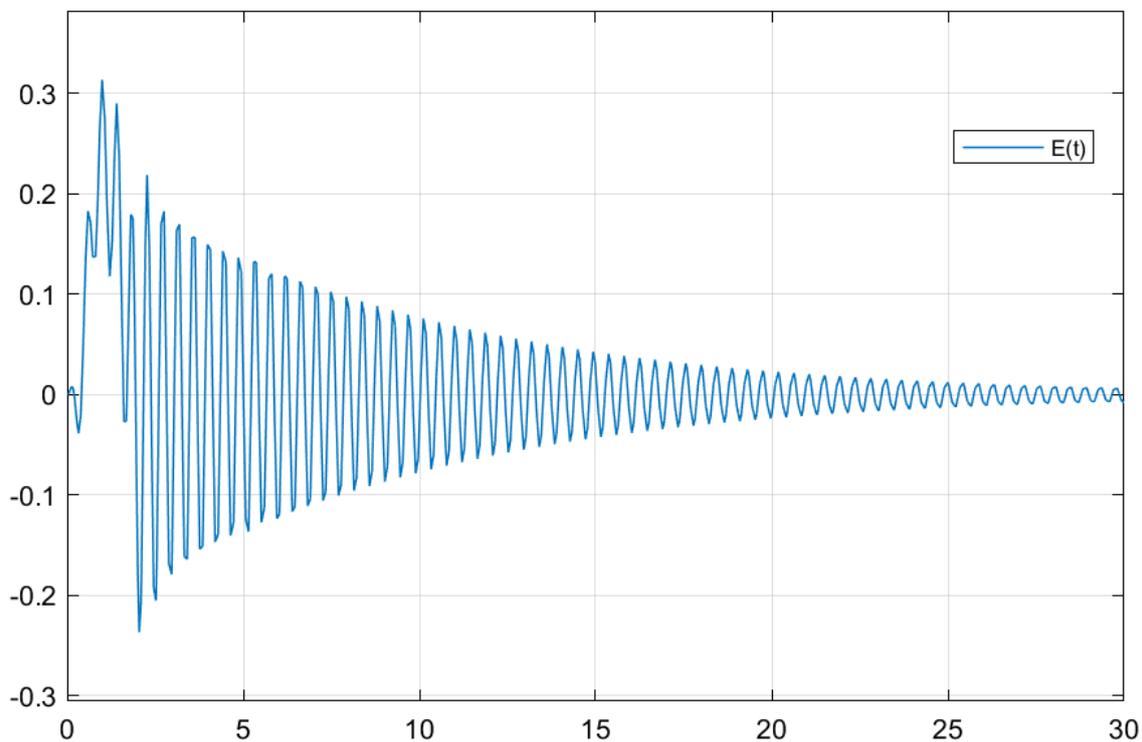


Figura 51. Valor del error a lo largo de la simulación.

Aunque nosotros definimos que el error podría ser de hasta π radianes, al ser la salida deseada 1 radián, podemos comprobar como el error toma como valor máximo aproximadamente 0.3 radianes.

7. CONTROL CON REDES NEURONALES

El concepto de redes neuronales (RNA) fue utilizado por primera vez en 1943 por Warren McCullough y Walter Pitts, y fue Frank Rosenblatt, quien en 1957 diseñó la primera red neuronal entrenable denominada "Perceptron". El perceptron es un algoritmo sencillo de clasificación binaria con una sola capa y una salida de 0 o 1.

En la década de los años noventa del siglo pasado comenzaron a surgir las redes neuronales de más de una capa tratando de resolver problemas más complejos, como la percepción multicapa, tratando de imitar el modo en que el cerebro humano procesa la información, dado que en definitiva, las redes neuronales artificiales (RNA) son la denominación general referida a los sistemas informáticos diseñados para imitar el modo en que el cerebro humano procesa la información (Katz et al., 2017).

Los componentes de una RNA están formados por un conjunto de nodos de computación interconectados, más concretamente se trata de neuronas artificiales, organizadas en capas. De acuerdo con el flujo de las señales, la RNA puede presentar arquitecturas de avance o de retroalimentación, como se muestra en la siguiente figura:

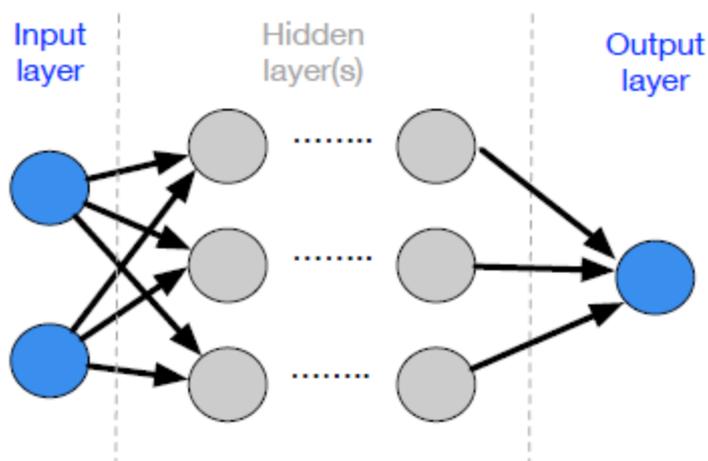


Figura 52. Arquitectura simplificada de una RNA

Fuente: Zhang y Li (2020)

La figura muestra una red neuronal simple, en la que cada neurona artificial tiene entradas ponderadas, una función de activación y una salida. Los pesos de las interconexiones se ajustan en función de las reglas de aprendizaje. Existen tres modelos importantes de reglas de aprendizaje que son, el aprendizaje no supervisado, el aprendizaje supervisado y el aprendizaje por refuerzo (LeChun, Bengio y Hinton, 2015). La elección de las reglas de

aprendizaje corresponde a la tarea de aprendizaje concreta y las funciones de activación comunes son la sigmoidea, la tangente hiperbólica, la función de bases radiales (RBF) y la función de transferencia lineal a trozos, como la unidad lineal rectificadora (ReLU) (Kruse et al., 2013).

Teniendo en cuenta lo expuesto, una RNA se puede definir a través de tres factores, esto es, la estructura de interconexión entre las capas, el tipo de función de activación y el procedimiento de actualización de los pesos (Zhang y Li, 2020).

Por otra parte, el perceptrón multicapa (MLP) se considera como una clase de RNA de avance. Consta de una capa de entrada, una o varias capas ocultas y una capa de salida, en la que cada neurona de una capa está totalmente conectada con cada nodo de la capa siguiente, a través de la técnica de retropropagación (LeChun, Bottou y Bengio, 1998).

En otro orden, la red neuronal convolucional (CNN), se define como un tipo especial de red multicapa, formada por una o más capas convolucionales. Cada capa convolucional incluye varios mapas de características con diferentes vectores de pesos, en la que una implementación secuencial de un mapa de características, escanea la imagen de entrada con una sola unidad y un campo receptivo, y almacena los estados de esa unidad en los lugares correspondientes del mapa de características. Esta operación es el equivalente a una convolución, seguida de un sesgo aditivo y una función de aplastamiento, que le otorga el nombre de convolucional (LeChun, Bottou y Bengio, 1998).

Las CNN son superiores para procesar datos bidimensionales (en concreto, imágenes de cámaras), gracias a las operaciones de convolución. En la actualidad, las CNN se utilizan para desarrollar vehículos completamente autónomos, y representan o son, una parte de una RNA con múltiples capas ocultas entre las capas de entrada y salida (Zhang y Li, 2020).

Las DNN (por ejemplo, una MLP con más de tres capas o una CNN) se diferencian de las NN superficiales (por ejemplo, una MLP de tres capas) en el número de capas, las funciones de activación que pueden emplearse y la disposición de la capa oculta.

En comparación con las NN superficiales, las DNN pueden ser entrenadas con mayor profundidad para encontrar patrones con alto rendimiento incluso para relaciones no lineales complejas. Una NN puede ser entrenada offline u online. Una NN entrenada fuera de línea significa que sólo aprende durante el desarrollo. Después del entrenamiento, los pesos de la NN serán fijos y la NN actuará de forma determinista. Por lo tanto, los métodos de verificación estática podrían ser posibles. En cambio, el entrenamiento en línea permitirá que la NN siga

aprendiendo y evolucionar durante el funcionamiento, lo que requiere métodos de verificación en tiempo de ejecución. En algunas aplicaciones, como el Sistema de Control de Vuelo Inteligente desarrollado por la NASA (Taylor, Darrah y Moats, 2003) se emplean tanto estrategias de entrenamiento offline como online para cumplir con los 150 requisitos del sistema.

Las NN son fundamentalmente diferentes a los programas algorítmicos, una metodología de desarrollo formal para un sistema NN. El proceso de desarrollo de un sistema NN puede incluir seis fases (Rodvold, 1999):

- 1. Formulación de requisitos y objetivos.
- 2. Selección de los conjuntos de datos de entrenamiento y de prueba.
- 3. Selección de la arquitectura de la NN.
- 4. Entrenamiento de la red.
- 5. Prueba de la red
- 6. Aceptación y uso por parte del cliente.

Al igual que (Rodvold, 1999), Falcini et al.(2017) introdujeron un ciclo de vida de desarrollo similar para las DNNs en el software de automoción, (Falcini et al., 2017) propusieron un modelo que integraba el desarrollo de datos integrado con el desarrollo de software estándar para resaltar la importancia de los datos en el desarrollo de las DNN. Falcini et al. (2017) también resumieron que el comportamiento funcional depende tanto de su arquitectura como de su resultado de aprendizaje a través del entrenamiento.

7.1. Diseño y simulación de un controlador de NN

Siguiendo las fases de desarrollo de un sistema NN propuestas por Rodvold (1999), necesitamos un bloque de NN que al igual que en el controlador Fuzzy, reciba un ángulo θ_t^* y nos responda con una entrada de voltaje para el motor $u(t)$.

Para ello necesitamos entrenar dicho bloque, tenemos distintas opciones, tales como usar datos aleatorios, usar los datos obtenidos en el controlador Fuzzy o usar los datos conseguidos con el controlador GPI. Ya que queremos que nuestro controlador sea lo más preciso posible, lo entrenaremos con los datos hallados en la simulación realizada en el capítulo 5 de GPI.

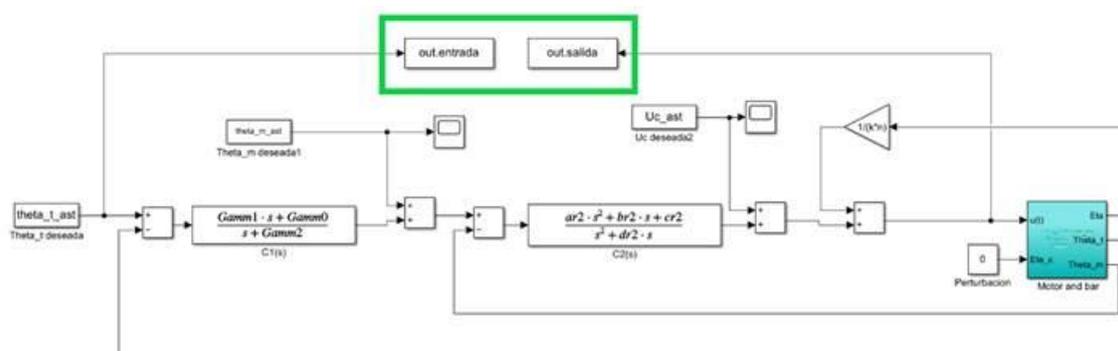


Figura 53. Datos obtenidos del controlador GPI.

Gracias a los bloques señalados, podemos convertir ambas señales en parámetros, para posteriormente mediante Matlab convertirlos en parámetros de una columna independientes del tiempo, de la siguiente manera:

```
Ent=out.entrada.Data;
```

```
Sal=out.salida.Data;
```

Una vez tenemos todos los datos que usaremos para el entrenamiento de la red neuronal abrimos la toolbox de redes neuronales que está disponible en Matlab con el comando "nftool".

Lo primero que debemos realizar en este proceso es importar las variables que vamos a usar, estas deben ser del mismo tamaño.

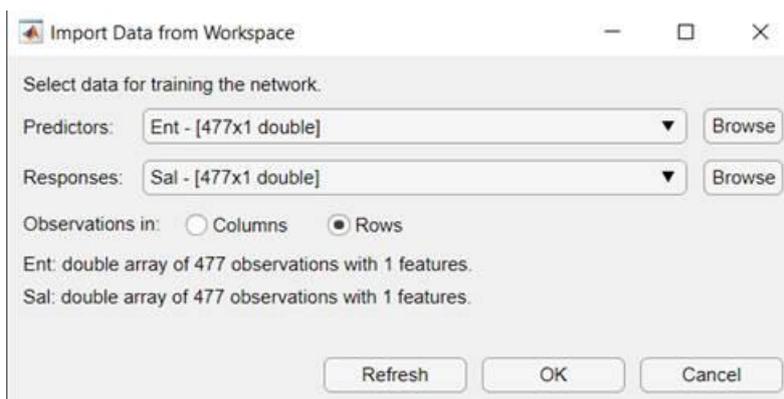


Figura 54. Introducción de las variables en nuestra red neuronal.

Con los variables ya importadas, podemos elegir que porcentaje de estos valores se utilizan para el entrenamiento, que porcentaje para la validación y que porcentaje para el testeo del controlador.

También podemos elegir que tamaño deseamos en la capa intermedia.

En este caso hemos utilizado los siguientes datos y usaremos el algoritmo de Levenberg-Marquardt para entrenar la red.

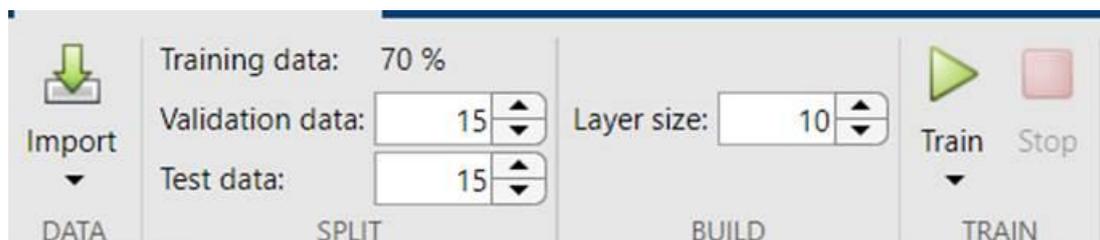


Figura 55. Opciones de entrenamiento para nuestra red neuronal.

Asignamos θ_t^* como entrada al bloque, y la salida será $u(t)$.

Con el sistema completo podemos realizar las simulaciones pertinentes.

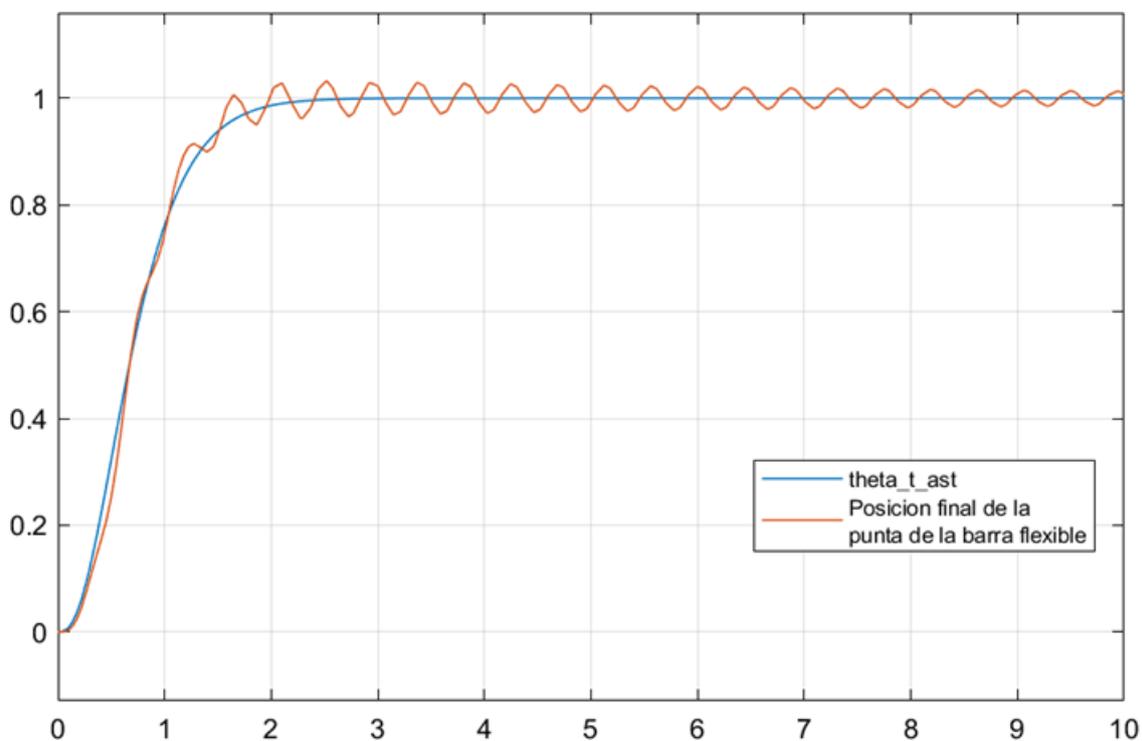


Figura 58. Valor de la salida deseada frente a la salida obtenida.

Podemos observar que al estar la red entrenada con los datos del controlador GPI, que son prácticamente idénticos a los deseados, conseguimos una salida con ciertas oscilaciones, al igual que con el controlador fuzzy, pero con menos error, y más rápida que este.

Cabe mencionar que el entrenamiento de la red neuronal es numérico pero no exacto, por lo que cada vez que volvamos a entrenar nuestra red podemos obtener distintos valores que podrían alterar ligeramente la gráfica final.

7.2. Diseño y simulación de dos controladores NN

Después de comprobar que todos los valores usados en el entrenamiento son válidos para un controlador que proporciona una salida parecida a la deseada, podemos dividir el bloque de redes neuronales en dos distintos que realizarían el mismo trabajo que los controladores C1 y C2 en el control de GPI. Para ello volvemos a nuestro circuito de GPI y conseguimos un tercer valor, intermedio, entre el primer y segundo controlador

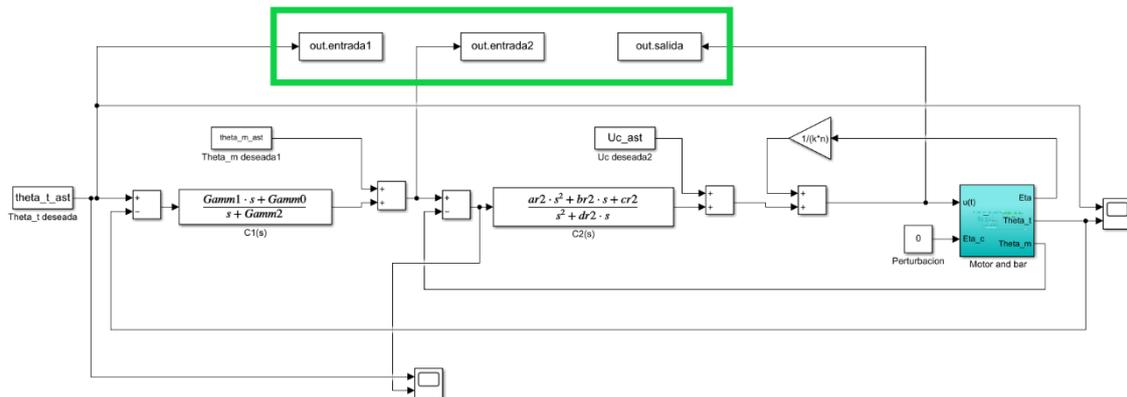


Figura 59. Nuevos datos obtenidos del controlador GPI.

Al igual que antes debemos convertir las las señales obtenidas en parámetros de una sola columna (independientes del tiempo), para ello usamos los siguientes comandos:

```
Ent1=out.entrada1.Data;
```

```
Ent2=out.entrada2.Data;
```

```
Sal=out.salida.Data;
```

A partir de aquí, todo el proceso es idéntico al realizado anteriormente pero ejecutado 2 veces, ya que el primer controlador tendrá como entradas “Ent1” y como salida “Ent2” y el segundo controlador tendrá como entrada “Ent2” y como salida “Sal”.

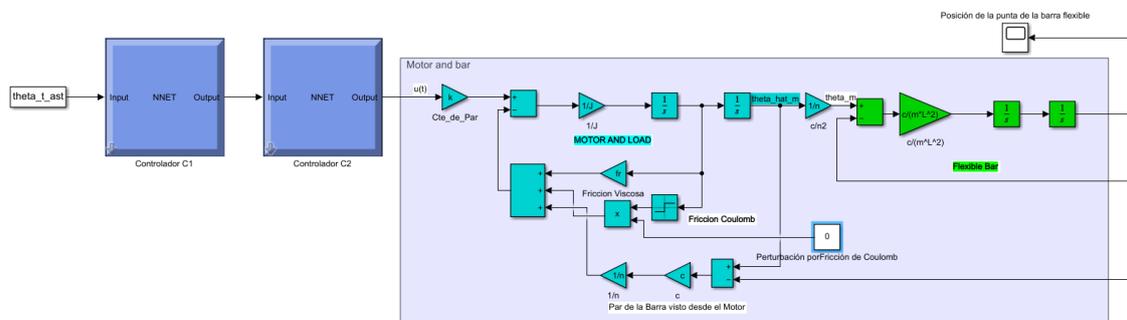


Figura 60. Diagrama de bloques con dos controladores de redes neuronales.

Al igual que en el experimento anterior la entrada será θ_t^* y la salida final será $u(t)$, pero en este caso los controladores estarán unido entre ellos como ocurría en el GPI.

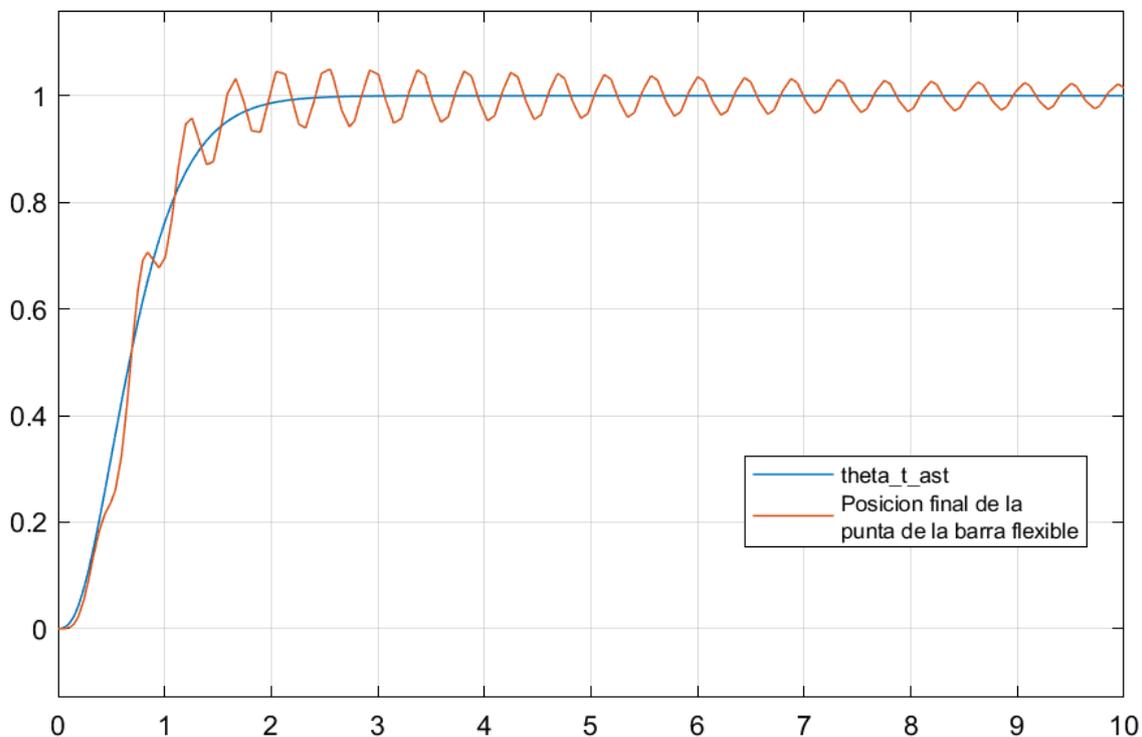


Figura 61. Valor de la salida deseada frente a la salida obtenida.

Como podemos comprobar la oscilación general es mayor, sobre todo al comienzo del movimiento, esto se debe a que como hemos dicho con anterioridad, el entrenamiento de redes neuronales no es exacto, y dos controladores generan mas error que uno solo. También existen ventajas, como que no hay que cambiar el circuito completo si se quiere editar algo de la etapa de control, valdría con editar simplemente el controlador en cuestión.

8. REFERENCIAS BIBLIOGRÁFICAS

- AILON, A. y ORTEGA, R. (1993). An observer-based set-point controller for robot manipulators with flexible joints, *Systems and Control Letters*, 21, 329-335.
- ALINALBU, S.; y GERD, H. (2001). A globally stable state feedback controller for flexible joint robots. *Adv. Robot.*, 15, 799–814. Doi: <https://doi.org/10.1163/156855301317198133>.
- ALVANDAR, S., y NIGAM, M.J. (2008). Adaptative Neuro-Fuzzy Inference based control of six DOF robot manipulator. *Journal of Engineering and Technology Review*, 1, 106-111.
- ASHRAF, A.M. (2020). Data-based PID control of flexible joint robot using adaptive safe experimentation dynamics algorithm. *Bulletin of Electrical Engineering and Informatics*, 10 (1), 79- 85. Doi: 10.11591/eei.v10i1.2472.
- BARBOSA, H., SIMÕES, M.A., SCHNEIDER, A., RAMOS, L.V., y NECES, F. (2017). Quasi-Omnidirectional Fuzzy Control of a Climbing Robot for Inspection Tasks. *J Intell Robot Syst*, 91, 333–347 (2018). <https://doi.org/10.1007/s10846-017-0672-9>
- BARRIENTOS, A. (2007). *Fundamentos de robótica*. Mc Graw Hill.
- BECEDAS, J., PAYO, I., FELIU, V., y SIRA, H. (2008). Generalized Proportional Integral Control for a Robot with Flexible Finger Gripper. Proceedings of the 17th World Congress, The International Federation of Automatic Control, *17th IFAC World Congress (IFAC'08)*. Seoul, Korea, July 6-11, 2008.
- BELÉNDEZ, T., C. NEIPP y BELÉNDEZ, A. (2002). Large and small deflections of a cantilever beam. *European J. Physics*, 23 (3), 371-379.
- BENNET, S. (1996). History of Automatic Control to 1960: An Overview. *IFAC Proceedings Volumes*, 29 (1), 3008-3013. Doi: [https://doi.org/10.1016/S1474-6670\(17\)58135-5](https://doi.org/10.1016/S1474-6670(17)58135-5).
- BENTALBA, S., EI HAJJAJI, A., y RACHID, A. (1997). Fuzzy control of a mobile robot: a new approach. *Proceedings of the 1997 IEEE International Conference on Control Applications*, 69-72.
- BERGHNIS, H., y NIJMEIJER, H. (1993). Global regulation of robots using only position measurements, *Systems and Control Letters* 21, 289-293.

- BOOK, W. J., y MAJETTE, M. (1983). Controller Design for Flexible, Distributed Parameter Mechanical Arms via Combined State Space and Frequency Domain Techniques, *Trans ASME J. Dyn. Syst. Meas. And Control*, 105, 245-254.
- BOOK, W.J., MAIZZA, O., y WHITNEY, D.E. (1976). Feedback Control of Two Beam, Two Joint Systems With Distributed Flexibility, *Trans ASME J.Dyn. Syst. Meas. and Control*, 97 G (4), 424-431. Disponible en <https://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.465.7068&rep=rep1&type=pdf> (ref. de 15 de marzo de 2022).
- BUFFINTON, K.W. (1992). Dynamics of Elastic Manipulators with Prismatic Joints. *J. Dyn. Syst. Meas. Control.*, 114, 41–49. Doi: <https://doi.org/10.1115/1.2896506>.
- CAO, H., ZHOU, J., JIANG, P., KEUNG, K., HON, B., YI, H., y DONG, C. (2020). An integrated processing energy modeling and optimization of automated robotic polishing system. *Robotics and Computer-Integrated Manufacturing*, 65. Doi: <https://doi.org/10.1016/j.rcim.2020.101973>
- CARDON, S. Z., e IBERALL, A. S (1970). Oscillations in biological systems, *Biosystems*, 3, (3), 237-249.
- CASTAÑO, J. (2006). La lógica difusa en ingeniería: Principio, aplicaciones y futuro. *Ciencia y Tecnología*, 24(2): 87-107. 2006 - ISSN: 0378-0524
- CHAPNIK, B.V., HEPPLER, G.R., y APLEVICH, J.D. (1993). Controlling the impact response of a one-link flexible robotic arm. *IEEE Transactions on Robotics and Automation*, 9(3):346–351.
- CHEONG, J., CHUNG, W., y YOUM, Y. (2000). Bandwidth Modulation of Rigid Subsystem for the Class of Flexible Robots, *Proceedings Conference on Robotics & Automation*, San Francisco, 1478-1483.
- CHEONG, J., CHUNG, W.K., y YOUM, Y. (2002). PID Composite Controller and Its Tuning for Flexible Link Robots, *Proceedings of the 2002 IEEE/RSJ, Int. Conference on Intelligent Robots and Systems EPFL*, Lausanne, Switzerland.
- CHIOU, B.C., y SHAHINPOOR, M. (1988). Dynamics stability analysis of a one-link force-controlled flexible manipulator. *Journal of Robotic Systems*, 5(5):443–451.

- CHOI, J.W., y CHO, N.I. (2002). Suppression of narrow-band interference in DS-spread spectrum systems using adaptive IIR notch filter. *Signal Processing*, 82(12), 2003–2013.
- CHU, Z.; y CUI, J. (2015). Experiment on vibration control of a two-link flexible manipulator using an input shaper and adaptive positive position feedback. *Adv. Mech. Eng.*, 7, 1–13. Doi: <https://doi.org/10.1177/1687814015610466>.
- COHEN, D. S., y NEU, J. C. (1979). Interacting oscillatory chemical reactors, *Annals of the New York Academy of Sciences*, 316, (1), 332- 337.
- COLLINS, D. (2021). What is dual-loop control for servo systems and when is it used? Motion Control Tips. Disponible en <https://www.motioncontroltips.com/what-is-dual-loop-control-for-servo-systems-and-when-is-it-used/> (ref. de 15 de marzo de 2022)
- COMITÉ ESPAÑOL de AUTOMÁTICA (CEA, 2015). *El libro blanco de la robótica en España. investigación, tecnologías y formación*. Disponible en https://www.ceautomatica.es/wp-content/uploads/2015/08/LIBRO-BLANCO-DE-LA-ROBOTICA-2_v2.pdf (ref. de 25 de octubre de 2021).
- DAXWANGER, W.A. , y SCHMIDT, G.K. (1995). Skill based visual parking control using neural and fuzzy networks. *1995 IEEE International Conference on Systems, Man and Cybernetics. Intelligent Systems for the 21st Century*, 1659-1664.
- De LUCA, A. (2000). Feedforward/Feedback Laws for the Control of Flexible Robots, *IEEE Int. Conf. on Robotics and Automation*, ,233-240.
- De LUCA, A., y SICILIANO, B. (1991). Closed-Form Dynamic Model of Planar Multilink Lightweight Robots, *IEEE Transactions on Systems, Man, and Cybernetics*, 21, 826-839.
- DJURIC, A. M., y ELMARAGHY, W. H. (2007). Unified Reconfigurable Robots Jacobian, *Proc. of the 2nd Int. Conf. on Changeable, Agile, Reconfigurable and Virtual Production*, 811-823.
- DOU, B.; y YUE, X.K. (2018). Collision dynamics analysis and control with consideration of friction in space capture process. *China Space Sci. Technol.*, 38, 54–62.
- DWIVEDY, S. K., y EBERHARD, P. (2006). Dynamic analysis of flexible manipulators, a literature review, *Mechanism and Machine Theory*, 41, (7),749–777.
-

- ELPROCUS (2022). What is a PID Controller: Working & Its Applications. Disponible en <https://www.elprocus.com/the-working-of-a-pid-controller/> (ref. de 21 de enero de 2022)
- ESCOLANO, F., CAZORLA, M., ALFONSO, M., COLOMINA, O., y LOZANO, M. (2001). *Inteligencia Artificial Modelos, Técnicas y Áreas de Aplicación*. Thomson Editores Spain.
- ESCRIBANO, U. (2019). *Robots de estructura flexible: análisis modelado y diseño de controladores*. [Tesis doctoral]. Universidad de Buenos Aires.
- FALCINI, F., LAMI, G., y COSTANZA, A.M. (2017). Deep learning in automotive software, *IEEE Software* 34, 56–63. Doi: 10.1109/MS.2017.79.
- FELIU, V. (2006). Robots flexibles: hacia una generación de robots con nuevas prestaciones. *Revista Iberoamericana de Automática e Informática Industrial*, 3, (3), 24-41.
- FELIU, V. (2007). Diseño y control de robots flexibles. Workshop Español de Robótica, ROBOT 2007. Disponible en <http://webdiis.unizar.es/~neira/docs/VFeliu-CEDI2007.pdf> (ref. de 15 de marzo de 2022).
- FILIPOVIC, M. (2008). New form of the Euler-Bernoulli rod equation applied to robotic systems. *Theoret. Appl. Mech.*, 35, (4), 381–406. Doi: 10.2298/TAM0804381F.
- FILIPOVIC, M. (2009). Euler- Bernoulli Equation Today. *The 2009 IEEE/RSJ International Conference on Intelligent Robots and Systems*, October 11-15, St. Louis, USA. Disponible en https://www.researchgate.net/publication/47749367_New_form_of_the_Euler-Bernoulli_rod_equation_applied_to_robotic_systems (ref. de 16 de marzo de 2022).
- FILIPOVIC, M., y VUKOBRATOVIC, M. (2005). Modelling of Flexible Robotic Systems, *Computer as a Tool, EUROCON 2005, The International Conference, Belgrade, Serbia and Montenegro*, 2, 1196 -1199.
- FILIPOVIC, M., y VUKOBRATOVIC, M. (2006). Contribution to modelling of elastic robotic systems, Engineering & Automation Problems, *International Journal*, 5, (1), 22-35.
- FILIPOVIC, M., y VUKOBRATOVIC, M. (2008a). Complement of Source Equation of Elastic Line, *Journal of Intelligent & Robotic Systems, International Journal*, 52, (2), 233 – 261.
-

- FILIPOVIC, M., y VUKOBRATOVIC, M. (2008b). Expansion of source equation of elastic line, *Robotica, International Cambridge Journal*, 26, (6), 739-751.
- FLIESS, M. (2000). Sur de nouveaux penser faissons de uers anciens. Iere. *Conference Francophone d'Automatique (CIFA-2000)*. Lille, France.
- FLIESS, M., MARQUEZ, R., DELALEAU, E., SIRA, H. (2002). Correcteurs proportionnels-intégraux généralisés. *ESAIM: Control, Optimisation and Calculus of Variations*, 7, 23-41. Doi: 10.1051/cocv:200202
- FLORES-ABAD, A.; MA, O.; PHAM, K.; ULRICH, S. (2014). A review of space robotics technologies for on-orbit servicing. *Progress in Aerospace Sciences*, 68, 1-26. Doi: <https://doi.org/10.1016/j.paerosci.2014.03.002>.
- FUKUDA, T., KITAMURA, N., y TANIE, K. (1986). Flexible handling by gripper with consideration of characteristics of objects. En *IEEE Proceedings on Robotics and Automation*, 3.
- FUKUDA, T., y ARAKAWA, A. (1987). Modeling and Control Characteristics for a Two-Degrees-of-Freedom Coupling System of Flexible Robotic Arms. *JSME International Journal*, 30. Doi: <https://doi.org/10.1299/jsme1987.30.1458>.
- GHORBEL, F., y SPONG, W.M. (1992). *Adaptive Integral Manifold Control of Flexible Joint Robot Manipulators*, IEEE International Conference on Robotics and Automation, Nice, France.
- GILARDI, G., y SHARF, I. (2002). Literature survey of contact dynamics modelling. *Mechanical and Machine Theory*, 37:1213– 1239.
- GORCE, P., y FONTAINE, J.G. (1996). Design methodology approach for flexible grippers. *Journal of Intelligent and Robotic Systems*, 15(3):307–328.
- HE, Y.; MEI, J.G.; FANG, Z.W. (2017). Study on energy consumption of a single joint servo system for a robot. *Manuf. Technol. Mach. Tools*, 9, 84–90.
- HERNANDEZ V.M. y SIRA, H. (2002b). Generalized Proportional Integral Control of Rigid Robots. *Proc. 41st Conference on Decision and Control*, 2050-2055. Las Vegas, December 2002.

- HERNANDEZ V.M. y SIRA, H. (2002c). Flexible Joint Robot Control without Mechanical Sensors. Proc. *IFAC Latinamerican Conference on Automatic Control*, CLCA 2002.
- HERNANDEZ, V.M. y SIRA, H. (2001). Position Control of an Inertia-Spring DC-Motor System without Mechanical Sensors: Experimental Results. Proc. *40th Conference on Decision and Control*, 1386- 1391.
- HERNÁNDEZ, V.M. y SIRA, H. (2002a). Sliding Mode Generalized PI Tracking Control of a DC Motor-Pendulum System. *Proceedings of the 7th International Workshop on Variable Structure Systems VSS'02*, 265-274.
- HOUTZAGER, I., van WINGERDEN, J. W. y VERHAEGEN, M. (2013). Rejection of periodic wind disturbances on a smart rotor test section using lifted repetitive control. *Control Systems Technology, IEEE Transactions on*, 21(2), 347-359.
- HUANG, X., CUI, X., LIAO, X., y HU, G. (2011). Pattern control in oscillatory systems with invisible controllers, *EPL*, 95, 1-6.
- HUGHES, P. C. (1977). *Dynamic of a flexible manipulator arm for the space shuttle*, AIAA Astrodynamics Conference, Grand Teton National Park, Wyoming, USA.
- IFR (2018). *Robots and the Workplace of the Future*. International Federation of Robotics Frankfurt, Germany March 2018.
- INTERNATIONAL FEDERATION OF ROBOTICS (IFR, 2020). World Robotics Report 2020. Disponible en <https://ifr.org/ifr-press-releases/news/record-2.7-million-robots-work-in-factories-around-the-globe> (ref. de 26 de octubre de 2021).
- INTERNATIONAL FEDERATION OF ROBOTICS (IFR, 2021). World Robotics 2021. Disponible en https://ifr.org/img/worldrobotics/World_Robotics_2021_Sales_Flyer.pdf (ref. de 26 de octubre de 2021).
- JANG, H., KRISHNAN, H., y ANG, M.H. (1997). A simple rest-to-rest control command for a flexible link robot, *IEEE Int. Conf. on Robotics and Automation*, 3312-3317.
- JARAMILLO, V.H. (2015). *Nuevas técnicas para el control de robots flexibles con carga variable*. [Tesis doctoral]. Universidad de Castilla-La Mancha.

- KAOUK JH, HABER GP, AUTORINO R, CROUZET S, OUZZANE A, FLAMAND V, VILLERS A. (2014). A novel robotic system for single-port urologic surgery: first clinical investigation. *Eur Urol.*66(6):1033-1043. Doi: 10.1016/j.eururo.2014.06.039.
- KARIM, N.A., y ARDESTANI, M.A. (2016). Takagi-Sugeno fuzzy formation control of nonholonomic robots. En *2016 4th International Conference on Control, Instrumentation, and Automation (ICCIA)*, 178–183.
- KATZ, G., BARRETT, C., DILL, D. L., JULIAN, K., KOCHENDERFER, M. J. (2017). Reluplex: 1206 An efficient smt solver for verifying deep neural networks, En *International 1207 Conference on Computer Aided Verification, Springer*, 97–117.
- KELLY, R., ORTEGA, R., AILON, A., y LORIA, A. (1994). Global regulation of flexible joint robots usign approximate differentiation. *IEEE Transactions on Automatic Control*, 39, (6), 1222-1224.
- KHADEM, S. E., y PIRMOHAMMADI, A. A. (2003). Analytical Development of Dynamic Equations of Motion for a Three-Dimensional Flexible Link Manipulator with Revolute and Prismatic Joints, *IEEE Transactions on Systems, Man and Cybernetics, part B: Cybernetics*, 33, (2).
- KHAIRUDIN, M.; MOHAMED, Z.; HUSAIN, A.; MAMAT, R. (2014). Dynamic characterisation of a two-link flexible manipulator: Theory and experiments. *Adv. Robot. Res.*, 1, 61–79. Doi: <https://doi.org/10.12989/arr.2014.1.1.061>.
- KHAIRUDIN, M.; MOHAMED, Z.; HUSAIN, A.R.; y AHMAD, M.A. (2010). Dynamic Modelling and Characterisation of a Two-Link Flexible Robot Manipulator. *J. Low Freq. Noise Vib. Act. Control.*, 29, 207–219. Doi: <https://doi.org/10.1260/0263-0923.29.3.207>.
- KIM, J. S., SIUZUKI, K., KONNO, A. (1996). Force Control of Constrained Flexible Manipulators, *International Conference on Robotics and Automation, Minneapolis, Minnesota*, 635-640.
- KIM, M. BERTRAM, M., POLLMANN, M., von QERTZEN, A., MIKHAILOV, A. S., ROTERMUND, H. H., y ERTL, G. (2001). Controlling chemical turbulence by global delayed feedback: pattern formation in catalytic CO oxidation on Pt (110), *Science*,292, (5520), 1357-1360.

- KOH, K.H., SREEKUMAR, M., y PONNAMBALAM, S.(2016). Hybrid electrostatic and elastomer adhesion mechanism for wall climbing robot. *Mechatronics* 35, 122–135
Disponible en <http://www.sciencedirect.com/science/article/pii/S0957415816000210>
- KRISHNAN, H. (1995). *An Approach to Regulation of Contact Force and Position in Flexible-Link Constrained Robots*, IEEE, Vancouver.
- KRUSE, R., BORGELT, C., KLAWONN, F., MOEWES, C., STEINBRECHER, M. , HELD, P. (2013). Multi-layer perceptrons, En *Computational Intelligence*, Springer, 47–81.
- LECUN, Y., BENGIO, Y., y HINTON, G. (2015). Deep learning. *Nature* 521, 436–444.
<https://doi.org/10.1038/nature14539>.
- LECUN, Y., BOTTOU, L., BENGIO, Y., y HAFFNER, P. (1998). Gradient-based learning applied to document recognition, En *Proceedings of the IEEE*, 86, (11), 2278-2324, Doi: 10.1109/5.726791.
- LEE, G., KIM, H., SEO, K., KIM, J., y KIM, H.S. (2015) Multitrack: a multi-linked track robot with suction adhesion for climbing and transition. *Robot. Auton. Syst.* 72, 207–216.
- LEE, K., J.F. WILSON y OH, S.J. (1993). Elastica of cantilever beams with variable cross section. *Int. J. Non-Linear Mechanics*, 28 (5), 579-589.
- LI, R.J.; LIU, H.Z.; y LI, P.F. (2008). Dynamic modeling of planar two-degree-of-freedom five-bar mechanism with joint dissipation considered. *J. Agric. Mach.*, 39, 157–161.
- LIU, F.C.; LIU, C.; FENG, Q.L. (2014). Influence of friction under gravity on motion control of space manipulator. *High Technol. Commun.*, 24, 65–71.
- LIU, N. (2009). Intelligent path following method for nonholonomic robot using fuzzy control. En *Second International Conference on Intelligent Networks and Intelligent Systems, 2009. ICINIS '09*. 282–285.
- LOONEY., C.G. (1988). Fuzzy Petri Nets for Rule Based Decision making. *IEEE Transactions on Systems, Man and Cybernetics - Part A: Systems and Humans*, 18, 178-183.
- LOW, K. H. (1987). A Systematic Formulation of Dynamic Equations for Robot Manipulators with Elastic Links, *J Robotic Systems*, 4, 3, 435-456.

- LOW, K.H. (1989). Solution Schemes for the System Equations of Flexible Robots, *J. Robotics Systems*, 6, 4, 383-405.
- LOW, K.H., VIDYASAGAR, M., y LAGRANGIAN, A. (1988). Formulation of the Dynamic Model for Flexible Manipulator Systems, *ASME J Dynamic Systems, Measurement, and Control*, 110, No. 2, 175-181.
- MAHAMOOD, R.M.; PEDRO, J.J. (2011). Hybrid PD/PID Controller Design for Two-Link Flexible Manipulators. En *Proceedings of the 8th Asian Control Conference*, Kaohsiung, Taiwan, 15–18 May 2011; 1358–1363.
- MARQUEZ, R. (2001). A Propos de Quelques Methodes Classiques de Commande LinCaire. PhD *Thesis*. Universitb Paris-Sud, Centre d'Orsay.
- MARQUEZ, R. DELALEAU, E., y FLIESS, M. (2000). Commande par PID GiniralixC d'un Moteur Electrique sans Capteur Me'canique. *Actes Conference Internationale Francophone d'Automatique (CIFA-2000)*. Lille, Rancia.
- MARQUEZ, R., FLIESS, M., y MOUNIER, H. (2001). A non- conventional robust PI-controller for the Smith predictor. *40th conference on Decision and Control*. Orlando, FL, December.
- MATSUNO, F. WAKASHIRO, K., y IKEDA, M. (1994). Force Control of a Flexible Arm, *International Conference on Robotics and Automation*, 2107-2112.
- MATSUNO, F., y KANZAWA, T. (1996). Robust Control of Coupled Bending and Torsional Vibrations and Contact Force of a Constrained Flexible Arm, *International Conference on Robotics and Automation*, 2444-2449.
- MATSUNO, F.; y YAMAMOTO, K. (2007). Dynamic hybrid force/position control of a two degree-of-freedom flexible manipulator. *J. Robot. Syst.* 2007, 11, 355–366.
- MAZHARI, S., y KUMAR, S. (2008). Heuristic Search Algorithms for Tuning PUMA 560 Fuzzy PID Controller. *International Journal of Electrical and Computer Engineering*, 3.
- MCCULLOCH, W. S. y PITTS, W. (1943). A logical calculus of the ideas immanent in 1200 nervous activity, *The bulletin of mathematical biophysics* 5, 115–133.
- MEIROVITCH, L. (1967). *Analytical Methods in Vibrations*. New York, Macmillan.

- MOALLEM, M., KHORASANI, K., y PATEL, V.R. (1996). Tip Position Tracking of Flexible Multi-Link Manipulators, An Integral Manifold Approach, *International Conference on Robotics and Automation*, 2432-2436.
- MOBERG, S. (2010). *Modeling and Control of Flexible Manipulators*. Linköping studies in science and technology. Dissertations, nº 1349. Linköping University.
- MOHAMED, Z.; KHAIRUDIN, M.; HUSAIN, A.; y SUBUDHI, B. (2014). Linear matrix inequality-based robust proportional derivative control of a two-link flexible manipulator. *J. Vib. Control.*, 22, 1244–1256. Doi: <https://doi.org/10.1177/1077546314536427>.
- MORALES, R., FELIU, V., JARAMILLO, V. (2012). Position control of very lightweight single-link flexible arms with large payload variations by using disturbance observers. *Robotics and Autonomous Systems*, 60, 532-547. Doi: 10.1016/j.robot.2011.11.016
- MORALES, R., SIRA, H., FELIU, V. (2014). Adaptive control based on fast online algebraic identification and GPI control for magnetic levitation systems with time-varying input gain. *International Journal of Control*, 87, 1604-1621. Doi: 10.1080/00207179.2014.880129.
- MORENO, J.; KELLY, R. (2005). Pose Regulation of Robot Manipulators with Dynamic Friction Compensation. En *Proceedings of the 44th IEEE Conference on Decision and Control*, Seville, Spain. 4368–4372. Doi: 10.1109/CDC.2005.1582849.
- NANDONG, J. (2015). Double-loop control structure for oscillatory systems: Improved PID tuning via multi-scale control scheme, 2015 10th Asian Control Conference (ASCC), 1-6, Doi: 10.1109/ASCC.2015.7244476.
- NANDONG, J., SAMYUDIA, Y., y TADE, M.O. (2006). Dynamic simulation and optimization of two-stage extractive alcoholic fermentation process: design impact on controllability, *Chemical Product Process Modeling*, 1, (1), 1-22.
- O'DWYER, A. (2006). *Handbook of PI and PID controller tuning rules*, London: Imperial College Press.
- ORTEGA, R., LORIA, A., KELLY, R., y PRALY, L (1994). On passivity-based output feedback global stabilization of Euler-Lagrange systems. *Proc. 33rd CDC*, pp. 381- 386. Lake Buena Vista, FL.

- OZGOLI, S., y TAGHIRAD, H.D. (2006). A survey on the control of flexible joint robots. *Asian J Control*, 8(4), 1–15.
- PATANE, S., JOYCE, E. R., SNYDER, M.P., y SHESTOPLE, P. (2017). Archinaut: In-Space Manufacturing and Assembly for Next-Generation Space Habitats, AIAA 2017-5227. *AIAA SPACE and Astronautics Forum and Exposition*. September 2017. Doi: <https://doi.org/10.2514/6.2017-5227>.
- PILTAN, F., SULAIMAN, N., GAVAHIAN, A., SOLTANI, S., y ROOSTA, S. (2011). Design mathematical tunable gain PID-like sliding mode fuzzy controller with minimum rule base. *International Journal of Robotics and Automation (IJRA)*, 2.
- PORRAS, A., y LLINAS, R. (2014). Bio-inspired coupled oscillatory phase reset control system applied to movement in an underwater vehicle, *Robotics Autonomous Systems*, 62, 257-266. MATSUNO, F., y KASAI, S. (1998). Modeling and robust force control of constrained one-link flexible arms. *Journal of Robotic Systems*, 15(8), 447–464.
- POTKONJAK, V., y VUKOBRATOVIC, M. (1999). Dynamics in Contact Tasks in Robotics, Part I General Model of Robot Interacting with Dynamic Environment, *Mechanism and Machine Theory*, 33.
- PRADHAN, S.K.; SUBUDHI, B. (2011). NARMAX Modeling of a Two-link Flexible Robot. En *Proceedings of the Annual IEEE India Conference*, Hyderabad, India, 16–18, 1–5.
- RAHIMI, H.; NAZEMIZADEH, M. (2013). Dynamic analysis and intelligent control techniques for flexible manipulators: A review. *Adv. Robot.*, 28, 63–76. Doi: <https://doi.org/10.1080/01691864.2013.839079>.
- RAHIMI, H.N. y NAZEMIZADEH, M. (2014). Dynamic analysis and intelligent control techniques for flexible manipulators: a review. *Advanced Robotics*, 28, (2), 63-76, Doi: <http://dx.doi.org/10.1080/01691864.2013.839079>.
- RAIMONDI, F.M., MELLUSO, M., y CIANCIMINO, L.S. (2005). A New Kinematic and Dynamic Direct Adaptive Fuzzy Control of Constrained Mobile Wheeled Vehicles. *2005 IEEE Conference on Emerging Technologies and Factory Automation*. 2, 181-188.
- RAMALINGAM, S., y MOHIDEEN, R. (2019). Composite materials for advanced flexible link robotic manipulators- An Investigation, *International Journal of Ambient Energy*, DOI: 10.1080/01430750.2019.1613263.

- READMAN M. (1994). *Flexible joint robots*. CRC Press.
- RODVOLD, D. M. (1999). A software development process model for artificial neural networks in critical applications, En *IJCNN'99. International Joint Conference on Neural Networks. Proceedings (Cat. No.99CH36339)*, 5, 3317–3322. Doi: 10.1109/IJCNN.1999.836192.
- ROSENBLATT, F. (1958). The perceptron: a probabilistic model for information storage and organization in the brain., *Psychological review* 65, 386.
- SAWADA, M.; y ITAMIYA, K. (2012). A Position Control of 2 DOF Flexible Link Robot Arms Based on Computed Torque Method. En *Proceedings of the 2012 International Conference on Advanced Mechatronic Systems*, Tokyo, Japan, 547–552.
- SAYAHKARAJY, M., MOHAMED, Z., y MOHD FAUDZI, A. A. (2016). Review of modelling and control of flexible-link manipulators. *Proceedings of the Institution of Mechanical Engineers, Part I: Journal of Systems and Control Engineering*, 230(8), 861–873. Doi: <https://doi.org/10.1177/0959651816642099>
- SCAGLIONI, B.; y FERRETTI, G. (2015). Closed form model of manipulators with highly flexible links. *IFAC-Papers Online*, 48, 653–654. Doi: <https://doi.org/10.1016/j.ifacol.2015.05.013>.
- SCHMIDT, D., y BERNS, K. (2013). Climbing robots for maintenance and inspections of vertical structures—a survey of design aspects and technologies. *Robot. Auton. Syst.* 61(12), 1288–1305.
- SHARKAWY, A.B. (2010). Genetic fuzzy self-tuning PID controllers for antilock braking systems. *Eng Appl Artif Intell*, 23, 1041–1052.
- SHI, L., y TRABIA, M. (2005). Comparison of distributed pd-like and importance based fuzzy logic controllers for a two-link rigid flexible manipulator. *Journal of Vibration and Control*, 11: 723–747.
- SHIMOGA, K.B. (1996). Robot grasp synthesis algorithms: a survey. *International Journal of Robotics Research*, 15, 230–266.
- SICILIANO, B., SCIAVICCO, L., VILLANI, L., y ORIOLO, G. (2009). *Robotics: modelling, planning and control*. Springer.

- SIRA, H. (2003). On the generalized PI sliding mode control of DC-to-DC power converters: a tutorial. *International Journal of Control*, 76, 1018-1033. Doi: 10.1080/0020717031000099047.
- SPECTOR, V.A., y FLESHNER, H. (1990). Modeling and Design Implications of Noncollocated Control in Flexible Systems. *J. Dyn. Sys., Meas., Control.*, 112(2): 186-193. Doi: <https://doi.org/10.1115/1.2896125>
- SPONG, M.W., HUTCHINSON, S., VIDYASAGAR, M. (2006). *Robot modeling and control*. Wiley.
- SPONG, W. M. (1989). On the Force Control Problem for Flexible Joint Manipulators, *IEEE Transactions on Automatic Control*, 34,1.
- SURDILOVIC, D., y VUKOBRATOVIC, M. (1996a). One method for efficient dynamic modeling of flexible manipulators, *Mechanism and Machine Theory*, 31, 3, 297-315.
- SURDILOVIC, D., y VUKOBRATOVIC, M. (1996b). Deflection compensation for large flexible manipulators, *Mechanism and Machine Theory*, 31,3, 317-329.
- TAKEGAKI, M., y S. ARIMOTO (1981). A new feedback method for dynamic control of manipulators, *Journal of Dynamic Systems, Measurements and Control*, 102, 119-125.
- TALOLE, S.E., KOLHE, J.P., y PHADKE, S.B. (2010). Extended-state observer-based control of flexible-joint system with experimental validation. *IEEE Trans Ind Electron*, 57(4), 1411–1419.
- TANAKA, M., CHONAN, S., y JIANG, Z. (1996). Force control of a flexible finger with distributed sensors and piezoelectric actuators. *Journal of Intelligent Material Systems and Structures*, 7(3): 301–306.
- TAYEBI, A., ABDUL, S., ZAREMBA, M. y YE, Y. (2008). Robust iterative learning control design: Application to a robot manipulator. *IEEE/ASME Transactions on Mechatronics*, 13(5), 608-613.
- TAYLOR, B., DARRAH, M., y MOATS, C. (2003). Verification and validation of neural networks: a sampling of research in progress. En RIDDY, K.L., y ANGELINE, P. J. (ed), *Intelligent Computing: Theory and Application*, 5103.

- TIAN, L., WANG, J., y MAO, Z. (2004). Constrained motion control of flexible robot manipulators based on recurrent neural networks. *IEEE Transaction on Systems, Man and Cybernetics.*, 34, 1541–1552.
- TIBADUIZA, D., y DAZA, C. (2008). Control Fuzzy de Velocidad para una Transmisión Hidrostática. *Revista Facultad de Ingeniería Universidad de Antioquia*. No. 44, 137-143.
- TIBADUIZA, D.A., AMAYA, I., RODRÍGUEZ, S., MEJÍA, N., y FLÓREZ, M. (2011). Implementation of a fuzzy control for the direct kinematic control of a robot manipulator. *Ingeniare. Revista chilena de ingeniería*, 19, 3, 312-322
- TOKNI, M.O., y AZAD., A.K.M. (2008). *Flexible Robot Manipulators. Modelling, simulating and control*. IET.
- TOMEI, P. (1991). A Simple PD Controller for Robots with Elastic Joints, *IEEE Transactions on Automatic Control*, 36, (10), 1208-1213.
- ULRICH, S., y SASIADEK, J.Z.(2014). On the simple adaptive control of flexible-joint space manipulators with uncertainties. En: *Book chapter: aerospace robotics II*. Springer; 13–23.
- VALIZADEH, A., y SHARIATEE, M. (2021). PZT Actuators' Effect on Vibration Control of the PRRRP 2-DOF Flexible Parallel Manipulator. *Hindawi, Shock and Vibration*, Doi: <https://doi.org/10.1155/2021/6985661>.
- VANDINI, A.; SALERNO, A.; PAYNE, C.J.; y YANG, G. Vision-based Motion Control of a Flexible Robot for Surgical Applications. En *Proceedings of the 2014 IEEE International Conference on Robotics and Automation*, Hong Kong, China, 6205–6211. Doi: 10.1109/ICRA.2014.6907774
- WALSH, A.; y RICHARD, F.J. (2015). Modeling and control of flexible telescoping manipulators. *IEEE Trans. Robot.*, 31, 936–947. Doi: 10.1109/TRO.2015.2441473.
- WANG, C Y. (1981). Large deflections of an inclined cantilever with an end load. *Int. J. Non-Linear Mechanics*, 16 (2), 155-164.
- WANG, F.; y GAO, Y. (2033). Advanced Studies of Flexible Robotic Manipulators Modeling, Design, Control and Applications. *Series in Intelligent Control and Intelligent Automation*, 4; World Scientific: Tuck Link, Singapore.
-

- WOLF, P. (2021). P, I, D, PI, PD, and PID control. University of Michigan. Disponible en [https://eng.libretexts.org/Bookshelves/Industrial_and_Systems_Engineering/Book%3A_A_Chemical_Process_Dynamics_and_Controls_\(Woolf\)/09%3A_Proportional-Integral-Derivative_\(PID\)_Control/9.02%3A_P%2C_I%2C_D%2C_PI%2C_PD%2C_and_PID_control](https://eng.libretexts.org/Bookshelves/Industrial_and_Systems_Engineering/Book%3A_A_Chemical_Process_Dynamics_and_Controls_(Woolf)/09%3A_Proportional-Integral-Derivative_(PID)_Control/9.02%3A_P%2C_I%2C_D%2C_PI%2C_PD%2C_and_PID_control) (ref. de 15 de marzo de 2022).
- WU, S.C. y TOMIZUKA, M. (2010). An iterative learning control design for self-servowriting in hard disk drives. *Mechatronics*, 20(1), 53-58.
- WU, W.X.; ZHU, S.Q.; y WANG, X.Y. (2013). Based on friction fuzzy modeling and compensating robot low speed control. *J. Electr. Mach. Control.*, 17, 46–53.
- XIN, X. (2011). Swing-up Control for a Two-link Underactuated Robot with a Flexible Elbow Joint: New Results Beyond the Passive Elbow Joint. En *Proceedings of the 50th IEEE Conference on Decision and Control and European Control Conference*, Orlando, FL, USA, 12–15 December 2011; 2481-2486.
- XU, P., CHEUNG, C.F., WANG, C., y ZHAO, C. (2020). Novel hybrid robot and its processes for precision polishing of freeform surfaces, *Precision Engineering*, 64, 53-62. Doi: <https://doi.org/10.1016/j.precisioneng.2020.03.013>.
- YIM, W. (1996). Modified Nonlinear Predictive Control of Elastic Manipulators, *International Conference on Robotics and Automation*, 2097-2102.
- ŽÁKOVÁ, K., y HUBA, M. (1997). Theoretical Analysis of Ziegler-Nichols Conclusions, *IFAC Proceedings Volumes*, 30, (21), 195-200. Doi: [https://doi.org/10.1016/S1474-6670\(17\)41438-8](https://doi.org/10.1016/S1474-6670(17)41438-8).
- ZHANG, B., ZHOU, K., WANG, Y. y WANG, D. (2010). Performance improvement of repetitive controlled pwm inverters: A phase-lead compensation solution. *International Journal of Circuit Theory and Applications*, 38 (5), 453-469.
- ZHANG, C.-Y.; BAI, C.C. (2012). Extremum response surface method of reliability analysis on two-link flexible robot manipulator. *J. Cent. South Univ.*, 19, 101–107. Doi: <https://doi.org/10.1007/s11771-012-0978-5>.

- ZHANG, F., y ZHAOHUI, Y. (2021). The Study of Dynamic Modeling and Multivariable Feedback Control for Flexible Manipulators with Friction Effect and Terminal Load. *Sensors*, 21, 1522. Doi: <https://doi.org/10.3390/s21041522>.
- ZHANG, J., y LI, J. (2020). Testing and verification of neural-network-based safety-critical control software: A systematic literature review, *Information and Software Technology*. Doi: <https://doi.org/10.1016/j.infsof.2020.106296>
- ZHANG, N.; FENG, Y.; YU, X. (2004). Optimization of Terminal Sliding Control for Two-link Flexible Manipulators. En *Proceedings of the 30th Annual Conference of IEEE Industrial Electronics Society*, Busan, Korea, 2–6 November 2004; 1318- 1322.
- ZHENG-JIE, W., YANG-BO, L., JING, L., y WEI, S. (2010). Modeling and Combined Control of Aerial Vehicle Based on Aerodynamic Force and Lateral Thrust. *The 2010 International Conference on Modelling, Identification and Control (ICMIC)*, 106-111.
- ZHU, H., GUAN, Y., WU, W., ZHANG, L., ZHOU, X., ZHANG, H. (2015). Autonomous pose detection and alignment of suction modules of a biped wall-climbing robot. *IEEE/ASME Trans. Mechatron.* 20(2), 653–662.
- ZURITA, E.W., LINARES, J., GUZMÁN, E., SIRA, H. (2011). A comparison between the GPI and PID controllers for the stabilization of a DC-DC “buck” converter. *IEEE Transactions on Industrial Electronics*, 58, 5251-5262. Doi: 10.1109/TIE.2011.2123857.