

UNIVERSIDAD DE INGENIERÍA Y TECNOLOGÍA

CARRERA DE INGENIERÍA MECATRÓNICA



**DISEÑO E IMPLEMENTACIÓN DE UN
CONTROLADOR DIFUSO PARA UN ACTUADOR
BLANDO UTILIZADO EN UN EXOESQUELETO
ROBÓTICO FLEXIBLE PARA LA REHABILITACIÓN
DE EXTREMIDADES INFERIORES EN RECIÉN
NACIDOS CON MIELOMENINGOCELE**

TESIS

Para optar el título profesional de Ingeniero Mecatrónico

AUTOR:

Luis Esteban Centeno Untiveros 

ASESOR:

Emir Vela Saavedra, Ph. D. 

Lima - Perú

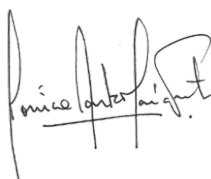
2023

DECLARACIÓN JURADA

Yo, Mónica Cecilia Santa María Fuster identificada con DNI No 18226712 en mi condición de autoridad responsable de validar la autenticidad de los trabajos de investigación y tesis de la UNIVERSIDAD DE INGENIERIA Y TECNOLOGIA, DECLARO BAJO JURAMENTO:

Que la tesis denominada “DISEÑO E IMPLEMENTACIÓN DE UN CONTROLADOR DIFUSO PARA UN ACTUADOR BLANDO UTILIZADO EN UN EXOESQUELETO ROBÓTICO FLEXIBLE PARA LA REHABILITACIÓN DE EXTREMIDADES INFERIORES EN RECIÉN NACIDOS CON MIELOMENINGOCELE” ha sido elaborada por el señor Luis Esteban Centeno Untiveros, con la asesoría de Emir Augusto Vela Saavedra, identificado con DNI 40063139, y que se presenta para obtener el título profesional de Ingeniero mecatrónico, ha sido sometida a los mecanismos de control y sanciones anti plagio previstos en la normativa interna de la universidad, encontrándose un porcentaje de similitud de 0%.

En fe de lo cual firmo la presente.



Dra. Mónica Santa María Fuster
Directora de Investigación

En Barranco, el 12 de abril 2024

TABLA DE CONTENIDO

	Pág.
RESUMEN	12
ABSTRACT	13
INTRODUCCIÓN	14
Presentación del tema de investigación	14
Descripción de la situación problemática	14
Formulación del problema	17
Objetivos de investigación	18
Justificación	19
Alcance y limitaciones / restricciones	20
CAPÍTULO I REVISIÓN CRÍTICA DE LA LITERATURA	22
1.1 Dispositivos de rehabilitación para bebés	22
1.2 Métodos de control utilizados para actuadores flexibles	25
CAPÍTULO II MARCO TEÓRICO	30
2.1 Mielomeningocele	30
2.1.1 Fisioterapia	31
2.2 Músculos Artificiales Blandos o VPAM (Músculos Artificiales Neumáticos Activados por Vacío)	32
2.3 Sistema de control mecatrónico a lazo cerrado	33
2.4 Controladores automáticos	33
2.4.1 Controlador PID	34
2.4.2 Controlador difuso (<i>Fuzzy</i>)	36
CAPÍTULO III MARCO METODOLÓGICO	40

3.1 Actuator flexible utilizado	40
3.2 Modelo de rehabilitación utilizado	41
3.3 Sistema de retroalimentación - actuación	45
3.4 Entorno de implementación del controlador	49
3.5 Requerimientos de control	52
3.6 Diseño del controlador difuso	54
CAPÍTULO IV RESULTADOS	59
4.1 Comportamiento del sistema en lazo abierto	59
4.2 Diseño del controlador difuso	62
4.3 Control en el modelo de pierna simplificado para 6 meses	79
4.4 Prueba de control frente a incertezas antropométricas	82
4.5 Comparación del método de control diseñado frente a otras variantes	87
4.6 Comportamiento del controlador para otras señales de referencia	92
4.7 Prueba de control frente a entradas tipo escalón	96
4.8 Prueba de resistencia frente a disturbios	100
4.9 Efecto de la configuración del regulador electrónico en el control	103
4.10 Efecto de la fabricación de otro actuator flexible	106
CONCLUSIONES	110
REFERENCIAS BIBLIOGRÁFICAS	117
ANEXOS	118
4.11 Código Arduino	118
4.12 Diagrama de bloques Simulink	123
4.13 Función de costo Matlab	123

ÍNDICE DE TABLAS

Tabla 3.1	Equipos utilizados.	49
Tabla 3.2	Métodos de implementación del controlador.	57
Tabla 3.3	Tabla de requerimientos definidos en base a recomendaciones de especialistas del INSN San Borja	58
Tabla 4.1	Valores optimizados de σ_1 y σ_2 para las MF del error.	75
Tabla 4.2	Valores optimizados de σ_1 y σ_2 para las MF de la derivada del error.	75
Tabla 4.3	Valores optimizados de σ_1 y σ_2 para las MF de la derivada de presión.	75
Tabla 4.4	Reglas difusas definidas para el controlador.	78
Tabla 4.5	Resumen de las características del controlador diseñado.	79
Tabla 4.6	Desempeño de cada variación del controlador difuso. (<i>Experimento realizado en el banco de pruebas configurado para 6 meses</i>)	94
Tabla 4.7	Resumen de los resultados para distintas referencias. (<i>Experimento realizado en el banco de pruebas configurado para 0 meses</i>)	96
Tabla 4.8	Características de la respuesta del controlador. (<i>Experimento realizado en el banco de pruebas configurado para 6 meses</i>)	102
Tabla 4.9	Variación de la presión para cada disturbio. (<i>Experimento realizado en el banco de pruebas configurado para 6 meses</i>)	102

ÍNDICE DE FIGURAS

Figura 1.1	Prototipo funcional diseñado por Park et al. [25].	23
Figura 1.2	Vista general del chaleco diseñado por Kokkoni et al. [26].	24
Figura 1.3	Prototipo final diseñado por Kriti et al. [4].	24
Figura 1.4	Aproximación de resorte, amortiguador y elemento contráctil de un PAM [13].	26
Figura 1.5	Montaje experimental utilizado para la implementación del control adaptativo propuesto [13].	26
Figura 1.6	Digrama de bloques del método de control propuesto por Tsung-Chin et al.[10].	27
Figura 1.7	Digrama de bloques del método de control propuesto por TU Diep C. y Kyoung K.[27].	28
Figura 2.1	Formación de mielomeningocele [6].	30
Figura 2.2	Ejercicios de fisioterapia para mielomeningocele [6].	31
Figura 2.3	Diagrama de funcionamiento de un sistema en lazo cerrado.	33
Figura 2.4	Diagrama de bloques de un proceso con controlador PID. Donde: K_p es la ganancia proporcional, K_i la ganancia integral, K_d la ganancia diferencia, u la ley de control, T_s el tiempo de muestro y z la transformada Z.	34
Figura 2.5	Diagrama de bloques de un controlador difuso.	36
Figura 2.6	Fuzzificación de dos entradas numéricas [34]. Donde A_1 y A_2 pertenecen al conjunto universo A , y B_1 y B_2 al conjunto universo B	37
Figura 2.7	Base de reglas de error e y el cambio de error Δe . Siendo las particiones: NG = Negativo Grande, NP = Negativo Pequeño, ZO = cero, PP = Positivo Pequeño, PG = Positivo Grande [34].	38
Figura 2.8	Defuzzificación del conjunto C' inferido utilizando el método de centro de área [34].	39

Figura 2.9	Diagrama de bloques de un proceso con controlador difuso PD.	39
Figura 3.1	Estructura y funcionamiento del actuador flexible [28].	41
Figura 3.2	Resultados de la prueba experimental de fuerza-contracción con el actuador flexible.	42
Figura 3.3	Modelo simplificado de pierna de bebé fabricado. Donde cada parte sería equivalente a: a) peso del pie, b) hueso de la pierna, c) IMU, d) soporte para evitar pandeo, e) femur, f) peso de la pierna, g y h) soportes para el actuador flexible y i) actuador flexible.	43
Figura 3.4	Equivalencia del modelo simplificado de pierna de bebé. Donde: a) representa el muslo del bebé, b) la pierna y c) el pie.	43
Figura 3.5	Maniquí utilizado para las pruebas experimentales de cero meses. Donde cada letra hace referencia a: a) soporte para el IMU, b y c) anclaje para el actuador y d) el actuador flexible utilizado	44
Figura 3.6	Maniquí utilizado para las pruebas experimentales de seis meses.	44
Figura 3.7	Sensor de orientación absoluta [35]	45
Figura 3.8	Locación del IMU en la pierna del bebé. Donde θ es el ángulo de inclinación de la pierna.	46
Figura 3.9	Diagrama de los componentes utilizados.	48
Figura 3.10	Circuito Neumático realizado.	49
Figura 3.11	Diagrama de flujo Arduino Mega.	51
Figura 3.12	Diagrama de bloques referencial utilizado en Simulink.	52
Figura 3.13	Trayectoria trapezoidal propuesta.	53
Figura 3.14	Trayectoria sinusoidal propuesta.	54
Figura 3.15	Estructura del controlador difuso. Donde: θ es el ángulo de inclinación de la pierna, θ_d el ángulo deseado, u la ley de control y e el error.	55
Figura 3.16	Estructura de la metodología planteada.	56

Figura 4.1	Respuesta del sistema en lazo abierto frente a una entrada sinusoidal. (<i>Experimento realizado en el banco de pruebas</i>)	60
Figura 4.2	Comportamiento de la presión de salida del regulador electrónico junto con la presión ideal. (<i>Experimento realizado en el banco de pruebas</i>)	61
Figura 4.3	Respuesta del sistema en lazo abierto frente a incrementos de presión cada 5 segundos. (<i>Experimento realizado en el banco de pruebas</i>) . .	63
Figura 4.4	Respuesta del sistema en lazo abierto frente a incrementos de presión cada 20 segundos. (<i>Experimento realizado en el banco de pruebas</i>) .	63
Figura 4.5	Respuesta del sistema en lazo abierto frente a incrementos de presión cada 30 segundos. (<i>Experimento realizado en el banco de pruebas</i>) .	64
Figura 4.6	Controlador PID siguiendo la trayectoria deseada para la rehabilitación con ganancias $K_p = 0.45$, $K_i = 0.02$ y $K_d = 0.002$. (<i>Experimento realizado en el banco de pruebas</i>)	65
Figura 4.7	Ley de control para el seguimiento de la trayectoria deseada para la rehabilitación con ganancias $K_p = 0.45$, $K_i = 0.02$ y $K_d = 0.002$. (<i>Experimento realizado en el banco de pruebas</i>)	65
Figura 4.8	Error del PID frente a distintas entradas tipo <i>step</i> con ganancias $K_p = 0.45$, $K_i = 0.02$ y $K_d = 0.002$. (<i>Experimento realizado en el banco de pruebas</i>)	66
Figura 4.9	Histograma de valores de error durante las pruebas.	67
Figura 4.10	Histograma de valores de la derivada de error durante las pruebas. .	68
Figura 4.11	Histograma de valores de la derivada de presión durante las pruebas.	68
Figura 4.12	Existencia de redundancia al utilizar 7 conjuntos difusos para la derivada del error.	69
Figura 4.13	Forma de las funciones de membresía obtenidas para el error utilizando <i>fcm</i>	70
Figura 4.14	Forma de las funciones de membresía obtenidas para la derivada del error utilizando <i>fcm</i>	71

Figura 4.15 Forma de las funciones de membresía obtenidas para la derivada de la presión utilizando f_{cm}	71
Figura 4.16 Efecto de los parámetros σ	74
Figura 4.17 Aproximación con los valores de σ estimados por f_{mincon}	74
Figura 4.18 Aproximación de las MF con los parámetros optimizados para el error.	76
Figura 4.19 Aproximación de las MF con los parámetros optimizados para la derivada del error.	76
Figura 4.20 Aproximación de las MF con los parámetros optimizados para la derivada de la presión.	77
Figura 4.21 Superficie difusa generada.	78
Figura 4.22 Respuesta del sistema controlado junto con la referencia utilizada. (<i>Experimento realizado en el banco de pruebas</i>)	81
Figura 4.23 Comportamiento de la ley de control durante el experimento. (<i>Experimento realizado en el banco de pruebas</i>)	82
Figura 4.24 Comportamiento de la presión real en el sistema durante el experimento. (<i>Experimento realizado en el banco de pruebas</i>)	83
Figura 4.25 Comparación entre la respuesta del controlador frente a distintas condiciones antropométricas. (<i>Experimento realizado en el maniquí</i>)	84
Figura 4.26 Comparación de la ley de control para cada variación antropométrica. (<i>Experimento realizado en el maniquí</i>)	85
Figura 4.27 Desempeño del controlador PID frente a variaciones antropomórficas utilizando una misma ganancia. (<i>Experimento realizado en el maniquí</i>)	86
Figura 4.28 Ley de control del controlador PID frente a variaciones antropomórficas, utilizando una misma ganancia. (<i>Experimento realizado en el maniquí</i>)	87
Figura 4.29 MF del error para la variante $Fuzzy_2$	88
Figura 4.30 MF la derivada del error para la variante $Fuzzy_2$	88
Figura 4.31 MF la derivada de presión para la variante $Fuzzy_2$	88

Figura 4.32	<i>MF</i> del error para la variante <i>Fuzzy</i> ₃	89
Figura 4.33	<i>MF</i> la derivada del error para la variante <i>Fuzzy</i> ₃	89
Figura 4.34	<i>MF</i> la derivada de presión para la variante <i>Fuzzy</i> ₃	90
Figura 4.35	<i>MF</i> del error para la variante <i>Fuzzy</i> ₄	90
Figura 4.36	<i>MF</i> la derivada del error para la variante <i>Fuzzy</i> ₄	91
Figura 4.37	<i>MF</i> la derivada de presión para la variante <i>Fuzzy</i> ₄	91
Figura 4.38	Comparación de desempeño de las variaciones del controlador. (<i>Experimento realizado en el banco de pruebas configurado para 6 meses</i>)	93
Figura 4.39	Comparación las leyes de control de las variaciones del controlador. (<i>Experimento realizado en el banco de pruebas configurado para 6 meses</i>)	93
Figura 4.40	Comportamiento del controlador para una referencia sinusoidal con 5 segundos de subida y de retorno. (<i>Experimento realizado en el banco de pruebas configurado para 0 meses</i>)	95
Figura 4.41	Comportamiento del controlador para una referencia trapezoidal con 10 segundos de subida y de retorno. (<i>Experimento realizado en el banco de pruebas configurado para 0 meses</i>)	95
Figura 4.42	Error de control frente a entradas <i>step</i> utilizando el control difuso. (<i>Experimento realizado en el banco de pruebas configurado para 6 meses</i>)	97
Figura 4.43	Error de control frente a entradas <i>step</i> utilizando el control PID. (<i>Experimento realizado en el banco de pruebas configurado para 6 meses</i>)	98
Figura 4.44	Ley de control frente a entradas <i>step</i> utilizando el control difuso. (<i>Experimento realizado en el banco de pruebas configurado para 6 meses</i>)	99
Figura 4.45	Ley de control frente a entradas <i>step</i> utilizando el control PID. (<i>Experimento realizado en el banco de pruebas configurado para 6 meses</i>)	99
Figura 4.46	Respuesta del controlador frente a un disturbio generado. (<i>Experimento realizado en el banco de pruebas configurado para 6 meses</i>)	101

Figura 4.47	Respuesta de la ley de control frente a un disturbio generado. (<i>Experimento realizado en el banco de pruebas configurado para 6 meses</i>) . . .	103
Figura 4.48	Efecto de los parámetros del regulador electrónico en el control. (<i>Experimento realizado en el banco de pruebas configurado para 0 meses</i>)	104
Figura 4.49	Seguimiento del regulador electrónico (línea roja) a la ley de control (línea negra) para una ganancia y sensibilidad iguales a 0. (<i>Experimento realizado en el banco de pruebas configurado para 0 meses</i>)	105
Figura 4.50	Seguimiento del regulador electrónico (línea roja) a la ley de control (línea negra) para una sensibilidad igual a F y una ganancia igual a 0. (<i>Experimento realizado en el banco de pruebas configurado para 0 meses</i>)	105
Figura 4.51	Seguimiento del regulador electrónico (línea roja) a la ley de control (línea negra) para una ganancia y sensibilidad iguales a F. (<i>Experimento realizado en el banco de pruebas configurado para 0 meses</i>)	106
Figura 4.52	Seguimiento del regulador electrónico (línea roja) a la ley de control (línea negra) para una sensibilidad igual a F y una ganancia igual a F. (<i>Experimento realizado en el banco de pruebas configurado para 0 meses</i>)	107
Figura 4.53	Efecto de los parámetros del regulador en el seguimiento de la ley de control. (<i>Experimento realizado en el banco de pruebas configurado para 0 meses</i>)	107
Figura 4.54	Diferencia de desempeño del controlador difuso utilizado dos actuadores distintos. (<i>Experimento realizado en el banco de pruebas configurado para 3 meses</i>)	109
Figura 4.55	Diferencia de la ley de control del controlador difuso utilizado dos actuadores distintos. (<i>Experimento realizado en el banco de pruebas configurado para 3 meses</i>)	109
Figura 4.56	Diagrama de bloques Simulink.	124

Dedicatoria:

A mi amada familia, cuyo apoyo incondicional y guía han iluminado cada paso de este arduo camino. Agradezco profundamente el sacrificio y la dedicación que han demostrado para allanar el camino hacia la realización de mis metas.

A quienes han estado a mi lado, testigos leales de mi viaje, a quienes agradezco por compartir risas, alegrías y experiencias únicas. Aprecio la fortaleza de nuestros lazos y la amistad que ha sido un faro en los momentos oscuros.

Agradecimientos:

Este trabajo fue apoyado por PROCENCIA Perú bajo el contrato N° 105-2021-FONDECYT Proyectos de Investigación Aplicada y Desarrollo Tecnológico.

Agradecemos al Dr. Segundo Cruz del Instituto Nacional de Salud del Niño por la discusión sobre las necesidades clínicas.

RESUMEN

El presente trabajo formó parte del proyecto financiado por PROCENCIA, cuyo objetivo principal fue automatizar la rehabilitación de las piernas de bebés de hasta seis meses de edad con mielomeningocele. La rehabilitación de los bebés afectados consiste en realizar movimientos de flexión y extensión en la pierna por un especialista en terapia física. Además, en colaboración con un especialista médico, se establecieron las siguientes especificaciones a cumplir: tres segundos de flexión, cinco de extensión y cinco de reposo entre cada movimiento de la rodilla, asegurando que el movimiento sea suave en todo momento. Para ello, se utilizó un actuador flexible accionado por vacío como parte del sistema de rehabilitación.

El controlador se enfrentó a dos desafíos principales: las variaciones antropométricas y la falta de un modelo matemático. El primero se refiere a la necesidad de que el controlador pueda funcionar dentro de un rango de pesos y longitudes que se asemejen a los de un recién nacido de hasta seis meses de edad. Por otro lado, el segundo desafío radica en la ausencia de un modelo matemático, lo que limita considerablemente las opciones de controladores disponibles para su implementación. Teniendo en cuenta estos desafíos, se propuso el uso de un controlador difuso.

Los resultados de la implementación demostraron que el controlador difuso es capaz de cumplir con los requisitos de movimiento dentro del rango de cero a seis meses sin necesidad de ajustar sus parámetros internos para cada mes. También se implementó un controlador PID para realizar una comparación, sin embargo, este mostró ser muy sensible a las variaciones antropométricas entre los distintos meses, lo que dificultó (en algunos casos) obtener las ganancias adecuadas para cumplir con los requisitos propuestos.

PALABRAS CLAVES:

Mielomeningocele; robótica blanda; control difuso; rehabilitación; actuador flexible

ABSTRACT

DESIGN AND IMPLEMENTATION OF A FUZZY CONTROLLER FOR FLEXIBLE ROBOTIC EXOSKELETON USED IN REHABILITATION OF LOWER LIMBS IN NEWBORNS WITH MYELOMENINGOCELE

The present work was part of a founded project by PROCENCIA, whose main objective was to automate the rehabilitation of the legs of infants up to six months of age with myelomeningocele. The rehabilitation of the affected babies consists of performing smooth flexion/extension movements on the leg by a physical therapy specialist. In addition, with a medical specialist, the following specifications were established: three seconds of flexion, five of extension and five of rest between each knee movement. For this purpose, a flexible vacuum actuator was used as part of a rehabilitation system.

The controller faced two main challenges: anthropometric variations and the lack of a mathematical model. The first relates to the need for the controller to be able to operate within a range of weights and lengths that resemble those of a newborn up to six months of age. On the other hand, the second challenge lies in the absence of a mathematical model, which considerably limits the controller options available for implementation. Taking these challenges into account, the use of a fuzzy controller was proposed.

The results of the implementation showed that the fuzzy controller is able to meet the motion requirements within the range of zero to six months without the need to adjust its internal parameters for each month. A PID controller was also implemented for comparison, however, it was shown to be very sensitive to anthropometric variations between months, making it difficult (in some cases) to obtain adequate gains to meet the proposed requirements.

KEYWORDS:

Myelomeningocele; soft robotics; fuzzy control; rehabilitation; flexible actuator

INTRODUCCIÓN

Presentación del tema de investigación

La robótica blanda es un campo emergente que está ganando popularidad debido a sus numerosas ventajas en aplicaciones que requieren una manipulación delicada de objetos. La capacidad de utilizar actuadores ligeros, flexibles, económicos, seguros y fáciles de fabricar ha captado la atención de la comunidad científica [1]. Estos dispositivos se utilizan principalmente en sistemas de rehabilitación [2] [3]. Sin embargo, el estudio de estos dispositivos se ve limitado debido a la falta de un sistema de sensado y control incorporado [4] [5], lo cual dificulta la realización de pruebas en condiciones cercanas a la realidad. Trabajar en lazo abierto, por ejemplo, dificulta la generación de trayectorias definidas por el usuario. Además, cualquier variación en la planta o entorno puede llevar a un funcionamiento no deseado debido a la falta de retroalimentación. Además, la ausencia de un modelo matemático para los actuadores blandos es un problema común debido a la compleja relación entre todas las variables físicas que interactúan durante su funcionamiento, lo que a su vez limita el diseño de un controlador [5]. Por lo tanto, se requiere la búsqueda de métodos que permitan trabajar con estas incertidumbres o que prescindan de un modelo matemático para su diseño.

Descripción de la situación problemática

El mielomeningocele es la forma más grave de espina bífida, una enfermedad congénita que afecta a aproximadamente 350 bebés cada año en Perú. Esta condición se caracteriza por un cierre defectuoso en la columna vertebral durante la cuarta semana de gestación, lo que resulta en la formación de un pequeño saco que contiene líquido cefalorraquídeo, nervios y parte de la columna vertebral misma [6]. Como consecuencia de esta enfermedad, se produce una disminución o incluso una abolición de la función

motora en las extremidades inferiores, lo que se traduce en una incapacidad parcial o total para el movimiento de los pies, las piernas, los muslos e incluso las caderas [7].

Como parte del tratamiento para el mielomeningocele, se lleva a cabo una intervención quirúrgica seguida de un programa de rehabilitación física después de 30 días. Este enfoque tiene como objetivo prevenir los efectos perjudiciales de la parálisis, que puede causar retracciones y deterioro de las condiciones musculares, y promover la presencia de músculos sanos. Hasta los seis meses de edad, la terapia consiste en cuidados posturales y movimientos pasivos de las extremidades inferiores [8].

Esta terapia requiere la intervención manual de especialistas capacitados, lo que genera una demanda de recursos humanos especializados capaces de cubrir las necesidades existentes. Además, se ha comprobado que una respuesta temprana brinda mejores resultados en estos casos. Lamentablemente, tanto en Lima como en otras partes de Perú, existe una escasez de profesionales capacitados para llevar a cabo esta terapia y comenzarla lo antes posible. Esta falta de recursos especializados es una de las razones por las cuales los niños con esta enfermedad no reciben la terapia adecuada y, como resultado, pueden presentar problemas de movilidad a lo largo de sus vidas [6].

Ante esta falta de recursos, se han propuesto alternativas para llevar a cabo la terapia. En la tesis titulada "Diseño y Fabricación de un exoesqueleto robótico flexible para la rehabilitación de extremidades inferiores en recién nacidos con espina bífida" por M. Mendoza [6], se planteó la posibilidad de automatizar el proceso de rehabilitación, específicamente la flexión y extensión de la rodilla, utilizando un sistema robótico. Como resultado, se diseñó e implementó un exoesqueleto con un actuador flexible de bajo perfil accionado por vacío (LP-VPAM) como actuador, controlado en lazo abierto. Sin embargo, la tesis señala la necesidad de mejoras para garantizar la efectividad y seguridad del exoesqueleto para el bebé, lo cual se lograría a través de un control adecuado.

Los actuadores de este tipo tienen la capacidad de generar fuerzas significativas, las cuales dependen tanto de la presión de succión utilizada como del material con el que

están fabricados [9]. Aunque el principio de funcionamiento parece ser simple, donde la fuerza del actuador se genera mediante la presión negativa, la relación entre estas dos variables no es trivial debido a la influencia de otras propiedades físicas internas. Esto hace que el sistema sea altamente no lineal. Además, factores como la histéresis y la variabilidad en el tiempo del sistema [10] dificultan la obtención de un modelo matemático preciso.

En la literatura, se destacan dos métodos de modelado de actuadores flexibles: el modelo teórico y el modelo fenomenológico. El primero se basa en la obtención de un modelo matemático utilizando propiedades físicas de los materiales utilizados y un análisis geométrico del actuador en sí [11]. En cambio, el segundo método se basa en una aproximación del sistema a un modelo de masa, resorte y amortiguador, relacionando las constantes de cada componente en función de la presión de entrada, lo que simplifica considerablemente el análisis [12]. Sin embargo, las ecuaciones resultantes son muy complejas de manipular debido a las no linealidades presentes, lo cual dificulta el diseño de un controlador basado en un modelo matemático. Además, el modelado requiere el uso de sensores precisos y costosos, así como configuraciones experimentales complicadas de fabricar [13].

Por otro lado, existen métodos computacionales ampliamente utilizados en la literatura para obtener este modelo matemático. Entre los más destacados se encuentran las redes neuronales [14], el modelado difuso [15], los algoritmos genéticos [16] y las combinaciones de estos, como el modelo difuso inverso NARX [17]. Si bien estos métodos proporcionan mejores aproximaciones al modelo del actuador, suelen ser complicados de implementar en una aplicación real y requieren una gran capacidad computacional. Además, se necesita una gran cantidad de datos para lograr una aproximación precisa. Otra opción popular en la investigación es la realización de simulaciones utilizando, por ejemplo, el método de elementos finitos [18] [19]. La gran ventaja de las simulaciones es que tienen en cuenta las propiedades físicas de los materiales, lo que produce resultados

más cercanos a la realidad. Sin embargo, muchas de estas pruebas se realizan de manera quasi-estática [20], lo que implica considerar movimientos lentos o cambios de condiciones lentos. Si bien esto permite una mejor relación entre las variables analizadas, no se tienen en cuenta los efectos dinámicos que pueden surgir al trabajar a velocidades más altas.

Algunos autores emplean métodos de control basados en modelos analíticos o experimentales, pero debido a las incertidumbres, simplificaciones y no linealidades del actuador, el rendimiento del controlador se ve afectado [12]. Otros controlan el actuador en función de simulaciones, pero esto requiere una alta capacidad computacional. Otro enfoque es el control por modo deslizante [21], que requiere al menos un modelo matemático base y una selección cuidadosa de parámetros. Sin embargo, el sistema de sensores utilizado en estos controles tiende a ser voluminoso, lo que podría resultar incómodo en una aplicación práctica [9].

Como se mencionó anteriormente, la obtención de un modelo matemático preciso del actuador resulta complicada debido a las simplificaciones y consideraciones necesarias [12]. Esto dificulta el diseño de un controlador, ya que muchos de los métodos existentes requieren un modelo matemático para un rendimiento óptimo. En contraste, el control difuso ofrece una alternativa más sencilla de implementar, ya que se basa en el criterio experto del usuario y no depende directamente de los datos disponibles. Además, la literatura ha demostrado la robustez y efectividad de este tipo de controladores para trabajar con actuadores similares a los utilizados en esta investigación [22].

Formulación del problema

En el marco de este trabajo de investigación, se toma como punto de partida el trabajo previo realizado por Mijail Mendoza [6], con el objetivo de diseñar e implementar un

controlador difuso que permita la automatización del proceso de movimientos terapéuticos de la rodilla utilizando el exoesqueleto desarrollado. Este enfoque se elige debido a las ventajas del control difuso, como su capacidad para manejar incertidumbres y no depender directamente de un modelo matemático preciso. Siguiendo este enfoque, se espera lograr una terapia más efectiva y accesible para los recién nacidos con espina bífida, para demostrar esto se hace uso de maniquís que simulan las condiciones reales de un bebé de 3 y 6 meses.

Objetivos de investigación

Objetivo General:

- Diseñar e implementar un controlador difuso para el movimiento de un actuador flexible que forma parte de un prototipo funcional de exoesqueleto destinado a la rehabilitación de extremidades inferiores en recién nacidos menores a seis meses de edad con mielomeningocele, y validar su desempeño al realizar la flexión y extensión de la rodilla de maniquís de bebés con dimensiones correspondientes a cero y seis meses de edad.

Objetivos Específicos:

- Implementar un sistema electrónico que permita obtener retroalimentación de las variables físicas de interés del sistema robótico, incluyendo posición y velocidad angular, presión interna del actuador flexible y presión del regulador.
- Diseñar un controlador difuso que cumpla con los requisitos específicos del ejercicio de rehabilitación para infantes de cero a seis meses de edad.
- Diseñar un controlador PID y comparar su desempeño con el controlador difuso.

- Validar el controlador difuso en laboratorio con un actuador flexible accionado con vacío para mover una pierna de un maniquí de infante simulando un escenario real.

Justificación

Aunque el número de recién nacidos afectados por mielomeningocele es relativamente bajo en comparación con otros países [6], es crucial contar con un sistema especializado de rehabilitación que satisfaga la demanda de esta enfermedad. La terapia de rehabilitación es esencial para evitar que los niños dependan de cuidados a lo largo de su vida, ya que la falta de recuperación de la fuerza en las extremidades inferiores después de la cirugía puede ocasionar dificultades en su movilidad [23]. Además, muchos de los dispositivos de rehabilitación existentes están diseñados para un público adulto, ya que requieren la capacidad del usuario de ser consciente, seguir instrucciones y evitar lesiones por parte del dispositivo.

Existen diversas razones por las cuales los recién nacidos operados por esta enfermedad a menudo no reciben la rehabilitación adecuada. En primer lugar, el costo de este tratamiento suele ser elevado. Aunque no se disponen de cifras exactas para el Perú, en países como Chile, el costo promedio anual de la rehabilitación se estima en alrededor de mil dólares [24]. En segundo lugar, la efectividad de la rehabilitación depende en gran medida de la experiencia del fisioterapeuta a cargo. Desafortunadamente, en Perú hay una escasez de especialistas en esta enfermedad, lo que dificulta aún más el acceso a la rehabilitación adecuada, especialmente para aquellos recién nacidos que viven fuera de Lima.

Por otro lado, se han desarrollado dispositivos específicos para la rehabilitación de diversas extremidades en recién nacidos utilizando actuadores flexibles. Sin embargo, la mayoría de estos dispositivos operan en un sistema de lazo abierto, lo que significa que

carecen de un sistema de retroalimentación que permita obtener información en tiempo real sobre el estado del sistema, así como de una interfaz para el usuario.

Motivado por estas limitaciones, el presente trabajo de investigación se centra en el diseño e implementación de un controlador difuso para el sistema de accionamiento de un exoesqueleto robótico de rehabilitación destinado a infantes con mielomeningocele. La elección de un controlador difuso se debe a las características del sistema a controlar, que incluye ruidos e incertezas paramétricas, como el peso y la longitud de la pierna del bebé, así como perturbaciones generadas por el propio bebé y la naturaleza no lineal del actuador flexible utilizado. Además, se realizaron mejoras en comparación con el sistema robótico presentado en [6], que incluirán un sistema de sensado en tiempo real, una interfaz de visualización más intuitiva y una mayor compatibilidad. El impacto de este trabajo radica en el potencial de mejorar la calidad de vida de los bebés afectados por mielomeningocele, al permitir que cualquier fisioterapeuta pueda supervisar y guiar los ejercicios en todo momento, reduciendo así la necesidad de contar con especialistas en la enfermedad de manera presencial. Además, en el futuro, esta tecnología podría extenderse para ser utilizada en otras enfermedades que requieran ejercicios de rehabilitación similares, como la parálisis cerebral o lesiones en la médula espinal.

Alcance y limitaciones / restricciones

Este trabajo de investigación se enfoca en el diseño e implementación de un controlador difuso para controlar el ángulo de flexión y extensión de la rodilla durante el proceso de rehabilitación. El objetivo principal es proporcionar una herramienta de rehabilitación específicamente diseñada para bebés con mielomeningocele en la región sacrolumbar, con una edad máxima de hasta seis meses. Se utilizó un actuador flexible preexistente para el sistema de actuación, por lo que el enfoque principal se centró en otros aspectos del diseño. El prototipo se centra en satisfacer las necesidades y características

únicas de este grupo de pacientes, brindando un enfoque especializado y adaptado a su condición médica.

Asimismo, se puso especial atención en el diseño del sistema de sensado, visualización y la compatibilidad del prototipo de sistema robótico de rehabilitación. El sistema de sensado se limitó a la medición de variables como la presión de salida de la válvula, la presión de entrada del actuador y el ángulo de inclinación de la rodilla. Estas variables fueron consideradas suficientes para describir de manera precisa el comportamiento del sistema durante la rehabilitación.

El diseño del controlador se enfocó en garantizar un rendimiento óptimo en un rango de flexión que va desde 0° hasta 64° , el cual es el rango común flexión sin generar ningún daño brindado por el Dr. Segundo Cruz considerando las características comunes en bebés de hasta seis meses de edad. Para lograr este objetivo, se realizaron pruebas y se diseñó un controlador específico utilizando un modelo impreso en 3D físico simplificado de una pierna de bebé. Este modelo simplificado fue creado mediante impresión 3D y permite al usuario ajustar su longitud y peso. Gracias a ello, se logró simular diferentes condiciones en las que el controlador pudo operar. La utilización de este modelo simplificado nos permitió realizar experimentos sin las limitaciones inherentes a las pruebas con bebés reales, ofreciéndonos un entorno controlado y libre de riesgos.

Finalmente, tanto el sistema robótico junto con su controlador fueron validados de manera funcional con dos maniquís, uno equivalente a una edad de cero y otro a la de seis meses provisto por el equipo dirigido por el docente Cesar Delgado de la Universidad Nacional de Trujillo (UNT). Este modelo físico logró caracterizar de manera fiel la dinámica real de la pierna y nos permitió observar el desempeño del controlador bajo condiciones más realistas.

CAPÍTULO I

REVISIÓN CRÍTICA DE LA LITERATURA

Se realiza una revisión crítica de la literatura de dispositivos de rehabilitación diseñados exclusivamente para bebés y accionados por un actuador blando, con el objetivo de extraer tanto las ventajas como las carencias que tienen estos dispositivos en el momento de su implementación. Asimismo, se investiga sobre los distintos métodos de control utilizados para este tipo de actuadores flexibles, ya que, debido a su no linealidad, no todos los métodos de control muestran un desempeño adecuado en la implementación.

1.1 Dispositivos de rehabilitación para bebés

Park et al.[25] diseñaron un sistema robótico *wearable*, dispositivos electrónicos que se usan sobre el cuerpo, con actuadores flexibles para asistir a la flexión y extensión de la rodilla. Este dispositivo se diseñó para infantes con hasta 1 año de edad, además como se puede observar en la Fig. 1.1 el sistema es compacto y ligero, lo cual es una característica importante a tener en cuenta; ya que esto simplificará significativamente el proceso de manufactura y su compactibilidad para ser un *wearable*.

No obstante, aunque el sistema logró llevar a cabo el movimiento de flexión y extensión de la rodilla, carece de un controlador interno y un sistema de sensores propio del dispositivo portátil. Por esta razón, los autores mencionan la posibilidad de que alguno de los actuadores genere un aumento indeseado de fuerza, lo que podría resultar en daños potenciales para el usuario final.

En otro estudio realizado por Kokkoni et al. [26], se desarrolló un chaleco diseñado para asistir el movimiento de las extremidades superiores en infantes de doce meses con

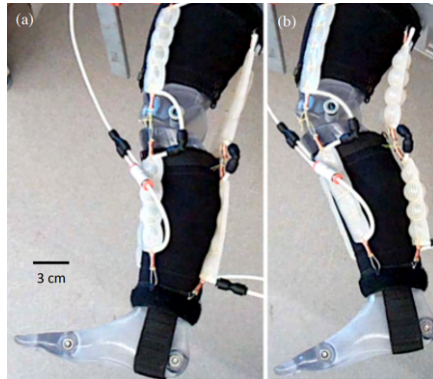


FIGURA 1.1: Prototipo funcional diseñado por Park et al. [25].

limitaciones de movilidad. Este chaleco empleaba dos actuadores flexibles ubicados en la parte superior del antebrazo para proporcionar apoyo en la flexión completa del brazo (Fig. 1.2). Los autores resaltan que eligieron utilizar actuadores flexibles debido a su ligereza, tamaño compacto y facilidad para su colocación. Los resultados obtenidos mostraron que esta configuración con actuadores flexibles puede generar trayectorias complejas y asistir al infante en la realización de movimientos de flexión sin problemas. Sin embargo, al igual que el dispositivo mencionado anteriormente, este chaleco carece de un sistema de sensores o una interfaz con el usuario. El control de cada uno de los actuadores se realiza manualmente utilizando un circuito electroneumático y pulsadores. Además, los autores señalan que el rendimiento del dispositivo es óptimo para ese caso específico, lo que implica que cualquier cambio en el peso, tamaño o incluso la geometría del infante requeriría modificaciones en el sistema, ya que este carece de capacidad de adaptación.

Finalmente, Kriti et al. [4] planteó el diseño de un dispositivo wearable que permitiría asistir en los movimientos de abducción y aducción de la cadera y de flexión del tobillo y la rodilla en niños de entre tres y nueve meses de edad con parálisis cerebral, utilizando actuadores blandos (Fig. 1.3). Sus resultados demostraron que es posible utilizar este tipo de configuraciones para realizar esta clase de ejercicios de rehabilitación, enfatizando en que la principal ventaja es el uso de actuadores flexibles que permiten reducir

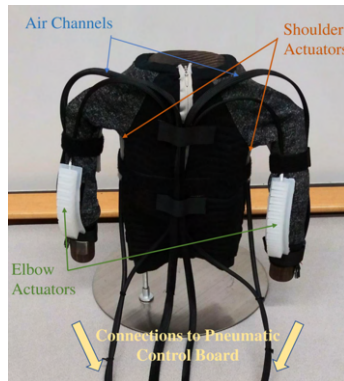


FIGURA 1.2: Vista general del chaleco diseñado por Kokkoni et al. [26].



FIGURA 1.3: Prototipo final diseñado por Kriti et al. [4].

considerablemente el peso general y que estos son desarrollados para que tengan cuidado con lo que se desee interactuar. Si bien los resultados demuestran que sí se puede realizar el ejercicio de rehabilitación, un aspecto faltante es la adición de un sistema de control, sensado y una interfaz con el usuario.

Los proyectos previamente descritos demuestran que es posible realizar de manera correcta la rehabilitación de las extremidades de un infante utilizando un sistema accionado por actuadores flexibles. Esto debido no solo a su bajo costo y poca dificultad de fabricación, sino también a su bajo peso y su particularidad de brindar 'delicadeza' a sus

acciones, perfecto para utilizarlo en personas y más aún, en bebés. Otro aspecto que mencionan los autores es que los movimientos que se realizan deben ser lentos y que siempre se debe tener un sistema de seguridad que evite que se generen movimientos bruscos o indeseados. Asimismo, recalcan en la necesidad de implementar un sistema de control para todo el sistema, ya que los experimentos fueron realizados ajustando la presión de entrada manualmente y observando el comportamiento de la extremidad a asistir.

1.2 Métodos de control utilizados para actuadores flexibles

Ning S. et al. [13] propuso junto con su equipo utilizar un controlador adaptativo para controlar un musculo neumático de Festo, más específicamente el DMSP-20-60N-RM-CM-DN. Ellos mencionan que el control de un actuador flexible no es un problema trivial debido a que estos presentan un comportamiento no lineal, histéresis compleja y variación de su geometría en función de la presión de entrada lo que complica significativamente el modelado matemático de la planta. Es por esto que utilizan el método fenomenológico, para realizar la aproximación matemática del actuador flexible como una combinación de un resorte, amortiguador y un elemento contráctil (Fig. 1.4). Esta aproximación se hizo con el fin de poder demostrar matemáticamente la estabilidad y robustez del controlador frente a disturbios e incertezas paramétricas. Los resultados obtenidos demuestran que el controlador propuesto funciona de manera correcta frente a disturbios generados durante el proceso, ya que este trata de compensar algún tipo de disturbio externo sin tener que afectar significativamente a la ley de control. Asimismo, se realizaron experimentos variando la carga que el actuador debía levantar y sin modificar algún parámetro del controlador diseñado, el resultado fue que el controlador funcionó correctamente para este cambio, demostrando de igual forma que tiene un correcto desempeño para incertezas paramétricas. Sin embargo, todos estos experimentos fueron realizados dentro de un montaje muy voluminoso, costoso y difícil de replicar (Fig. 1.5) debido a que se utilizan

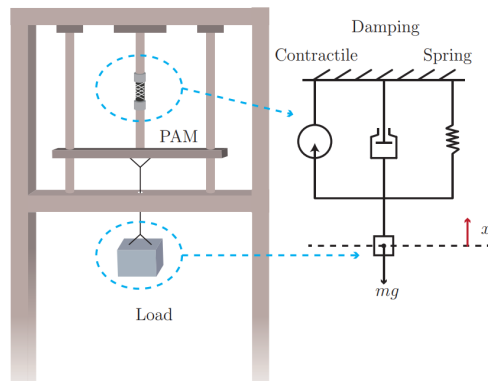


FIGURA 1.4: Aproximación de resorte, amortiguador y elemento contráctil de un PAM [13].

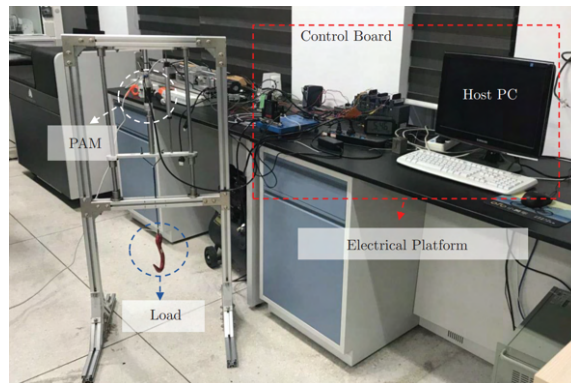


FIGURA 1.5: Montaje experimental utilizado para la implementación del control adaptativo propuesto [13].

máquinas y sensores de gran precisión. De igual manera, el controlador solamente se utiliza para controlar la posición de un bloque de masa suspendido y no es probado dentro de un sistema robótico donde se utilice el actuador flexible generador de movilidad.

Por otro lado, Tsung-Chin et al. [10] propuso utilizar un *Fuzzy Sliding Mode Controller* (FSMC) (Fig. 1.6) debido a que, al igual que la investigación anterior, mencionan que el principal problema al trabajar con controladores para este tipo de actuadores son las no linealidades; ya que los controladores lineales suelen tener un bajo desempeño para

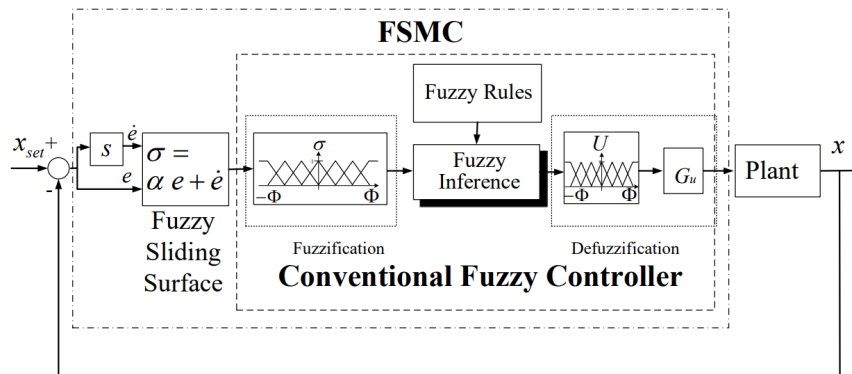


FIGURA 1.6: Digrama de bloques del método de control propuesto por Tsung-Chin et al.[10].

distintas condiciones. Asimismo, el modelamiento matemático no siempre es sencillo para esta clase de actuadores, por lo que un FSMC facilitaría el trabajo porque este método no necesita tener el modelo matemático para poder funcionar correctamente pues trabaja con un conjunto de reglas difusas en base a la superficie de deslizamiento. El desempeño del controlador se comparó a la de un PID clásico, mostrando una reducción significativa en el porcentaje de Overshoot (de hasta 4 veces menos), pero su principal diferencia fue al momento de probar señales de referencia con una frecuencia alta; ya que, el FSMC presentaba un error de 2.5° aproximadamente mientras que el PID clásico tenía un error de 5° aproximadamente. Además, se realizaron pruebas cambiando el peso que el actuador debía levantar y se demostró que el FSMC no se ve afectado para estos cambios, mientras que el PID si presenta un mayor porcentaje de error. Sin embargo, este método de control no fue probado frente a disturbios externos lo cual es un aspecto importante teniendo en cuenta que el infante podría generar un movimiento no deseado en el momento de la rehabilitación.

Finalmente, se analizará el trabajo realizado por TU Diep C. y Kyoung K. [27] quienes proponen utilizar un PID no lineal junto con redes neuronales (Fig. 1.7). Debido a las incertezas de los actuadores flexibles ellos proponen utilizar redes neuronales para estimar el modelo de la planta y así evitar realizar un modelamiento matemático de esta.

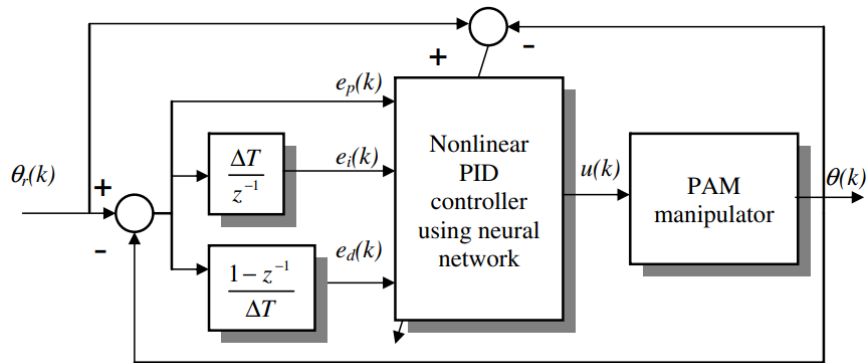


FIGURA 1.7: Digrama de bloques del método de control propuesto por TU Diep C. y Kyoung K.[27].

Además, el PID utilizado será auto ajustable a medida que pasa el tiempo, es decir, que dependiendo del valor de referencia y del error actual los valores de K_p , K_i y K_d variarán hasta llegar a minimizar el error lo máximo posible. El desempeño del controlador se comparó con un PID clásico, demostrando que el PID no lineal tiene un mejor seguimiento a la trayectoria de referencia desde un inicio mientras que el PID clásico presenta un gran error desde el inicio del seguimiento de la trayectoria. Además, los valores de K_p , K_i y K_d terminan convergiendo luego de 5s aproximadamente, demostrando que se encontraron unos valores óptimos para dicha aplicación. Sin embargo, el desempeño del controlador no es puesto a prueba frente a incertezas paramétricas o disturbios externos.

Como se pudo observar, se lograron implementar distintos métodos de control para controlar un actuador flexible; sin embargo, todos estos actuadores eran de presión positiva y además no eran blandos en su totalidad. Pero todos llegan a la misma conclusión: la no linealidad y las incertezas de modelamiento son un problema común para el controlador pero se puede realizar una aproximación matemática como se propone en [13] o utilizar un controlador que no tome en cuenta el modelo matemático de la planta y solamente trabaje con los valores de entrada y salida [10] [27]. Con esta información podemos decir que un controlador PID no es suficiente y siempre es necesario utilizar un control

no lineal para incrementar la precisión del controlador, más teniendo en cuenta que el objetivo es utilizarlo en rehabilitación de infantes debemos tratar de maximizar la seguridad y evitar algún tipo de movimiento brusco indeseado, junto con las oscilaciones.

CAPÍTULO II

MARCO TEÓRICO

2.1 Mielomeningocele

El mielomeningocele es el tipo más grave de espina bífida, enfermedad congénita que surge en la cuarta semana de gestación en donde se encuentra un cierre defectuoso de la columna vertebral [6]. En esta, se forma un saco de líquido cefalorraquídeo con tejido nervioso y partes de la columna vertebral. Una vez nacidos, los bebés enfermos perderán progresivamente sus habilidades motrices en las extremidades inferiores, lo cual podría llegar a una parálisis parcial o total. El tratamiento de esta enfermedad consta de una intervención quirúrgica postparto y una posterior fisioterapia [8]. En la Fig. 2.1 se puede observar la formación de mielomeningocele en el recién nacido.

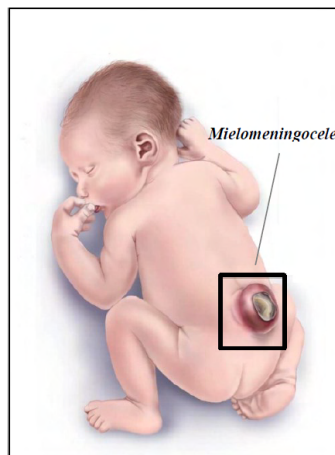


FIGURA 2.1: Formación de mielomeningocele [6].

2.1.1 Fisioterapia

La terapia física dada a un recién nacido con mielomeningocele es dada por un especialista de forma manual 30 días después del procedimiento quirúrgico [8], teniendo como objetivo la preservación del tejido neural viable, la reconstrucción de la anatomía en lo posible y la disminución de la probabilidad de infección [7].

Esta terapia consiste en la realización de distintos movimientos de las articulaciones afectadas como cadera, rodilla, tobillo y la de mayor relevancia para el desarrollo: la rodilla [6]. A continuación, se presenta una lista de los ejercicios realizados en la terapia. Los mismos ejercicios se pueden observar en la Fig. 2.2.

- Extensión de la cadera con flexión de la rodilla en posición prona.
- Flexión de cadera con flexión de rodilla en posición de costado.
- Plantarflexión de tobillo en posición supina.



FIGURA 2.2: Ejercicios de fisioterapia para mielomeningocele [6].

2.2 Músculos Artificiales Blandos o VPAM (Músculos Artificiales Neumáticos Activados por Vacío)

Los Músculos Artificiales Blandos, también conocidos como VPAM (Músculos Artificiales Neumáticos Activados por Vacío), representan una categoría revolucionaria de dispositivos diseñados para replicar la funcionalidad de los músculos biológicos. A diferencia de sus homólogos rígidos, estos materiales ofrecen una flexibilidad y adaptabilidad excepcionales, convirtiéndolos en componentes esenciales para una variedad de aplicaciones en campos como la robótica, la biomecánica y la ingeniería biomédica [16].

Los VPAM se caracterizan por su capacidad para cambiar de forma y experimentar deformaciones cuando se aplica vacío en su interior. Este fenómeno se basa en la respuesta de materiales elastoméricos a la variación de presión interna. Al aplicar o retirar vacío, los VPAM pueden contraerse o expandirse de manera controlada, replicando así el comportamiento muscular [28]. Esta propiedad neumática proporciona a los VPAM una versatilidad única, permitiendo la creación de dispositivos que simulan el movimiento suave y continuo de los músculos biológicos.

Uno de los aspectos más destacados de los VPAM es su capacidad para reproducir tanto la contracción como la relajación de los músculos naturales. Esto permite un control más preciso y una mayor similitud con el comportamiento biomecánico humano [5]. Además, la suavidad y ligereza inherentes a estos materiales los hacen ideales para aplicaciones en las que la interacción segura con entornos y seres humanos es esencial.

Desde una perspectiva de diseño, los VPAM ofrecen la posibilidad de crear estructuras más ergonómicas y adaptables en comparación con otros actuadores. Esto se traduce en la capacidad de desarrollar robots y dispositivos biomiméticos más amigables y seguros para la interacción con el entorno y con seres humanos [29].

2.3 Sistema de control mecatrónico a lazo cerrado

Un sistema en lazo cerrado es una estructura que consiste en utilizar la señal de salida de nuestra planta como una señal de retroalimentación (Fig. 2.3). Esta señal de retroalimentación ingresa a nuestro controlador el cual nos permitirá modificar la señal de entrada de nuestra planta de tal manera que minimicemos el error y podamos realizar un correcto seguimiento a nuestra señal de entrada o referencia. El uso de la retroalimentación hace que la respuesta del sistema sea relativamente insensible a perturbaciones externas y a variaciones internas de los parámetros del sistema [30].

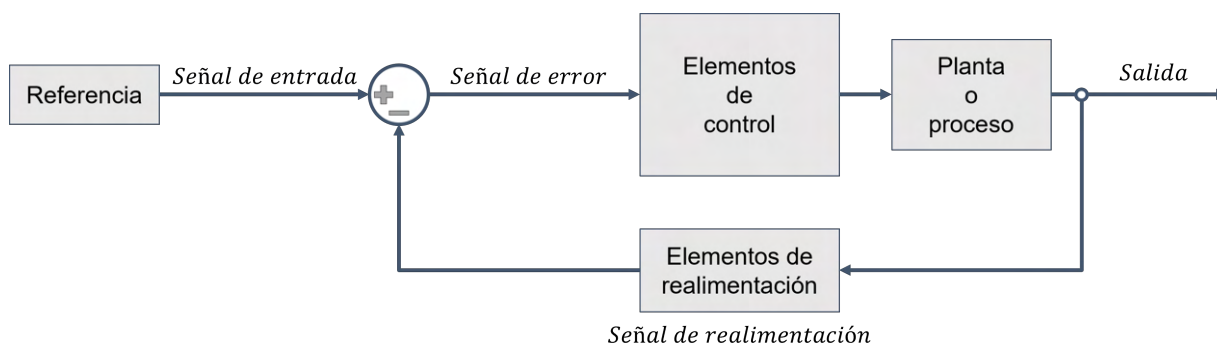


FIGURA 2.3: Diagrama de funcionamiento de un sistema en lazo cerrado.

2.4 Controladores automáticos

Como se mencionó en la subsección anterior, los sistemas en lazo cerrado utilizan el valor real de la salida de la planta como entrada hacia el controlador, esto con el objetivo de minimizar el error con respecto a la señal de referencia o deseada. Esto se logra gracias a que el controlador modifica la fuerza de control de la planta para que logre cumplir ciertas especificaciones de diseño, tales como: tiempo de estabilización, error en estado estacionario, porcentaje de sobrepaso, entre otros. Sin embargo, existen distintos tipos de controladores utilizados actualmente tanto en la industria como en la investigación, el uso

de esos depende directamente de la complejidad de la planta y de las especificaciones a cumplir. Para esta investigación se trabajarán principalmente con controladores especializados en sistemas no lineales con incertezas [30].

2.4.1 Controlador PID

El controlador PID (Proporcional-Integral-Derivativo) debido a su facilidad de implementación y buen desempeño para ciertas condiciones de trabajo es el más utilizado en muchos sistemas en lazo cerrado [31]. Este algoritmo de control hace uso de la idea más simple de retroalimentación, ya que los tres elementos de los que está compuesto dependen solamente del error. La formulación final de la fuerza de control utilizando un controlador PID se puede apreciar en la Fig. 2.4, donde: e el error de control, K_p la ganancia proporcional, K_i la ganancia integral y K_d la ganancia derivativa.

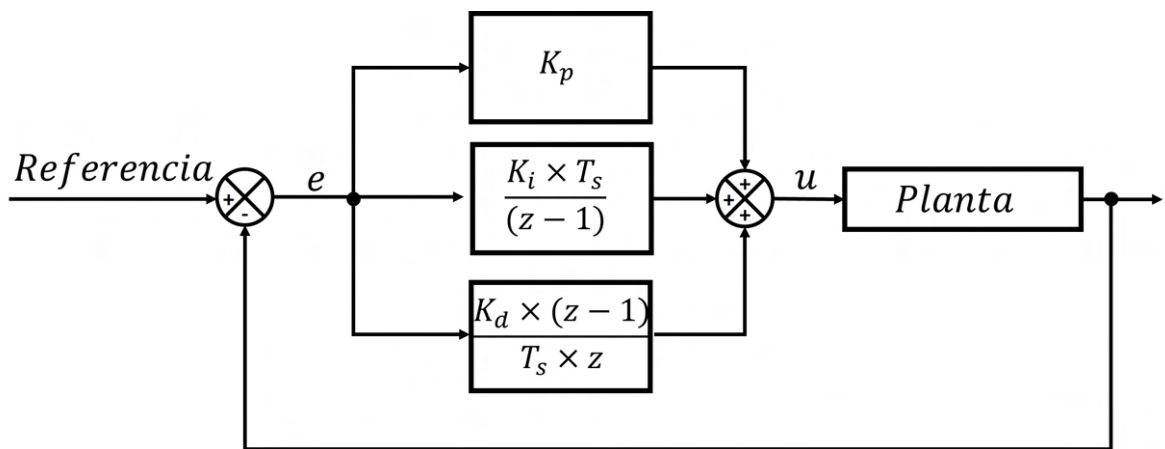


FIGURA 2.4: Diagrama de bloques de un proceso con controlador PID. Donde: K_p es la ganancia proporcional, K_i la ganancia integral, K_d la ganancia diferencial, u la ley de control, T_s el tiempo de muestreo y z la transformada Z.

Sin embargo, la complejidad del controlador PID se ve al momento de sintonizar: seleccionar los valores adecuados para cada uno de estos componentes (proporcional,

integral y derivativo). Es por esto, por lo que es importante conocer la función que tiene cada uno de estos parámetros en la señal controlada final, las cuales son descritas a continuación:

- **Ganancia proporcional K_p :**

La parte proporcional es equivalente al producto entre la ganancia proporcional y la señal de control. Acelera la respuesta del proceso y reduce el error estacionario [32]. Con un valor muy alto, el tiempo de respuesta es muy corto; sin embargo, aumenta el sobrepaso y las oscilaciones, inclusive pudiendo provocar inestabilidad a la planta.

- **Ganancia integral K_i :**

La parte integral de la ley de control elimina el error en estado estacionario [32]. Su ventaja es que este dependerá de los valores pasados del error por lo que al multiplicarlo por la constante integral este continuará aumentando su valor a menos que el error sea cero. Por un lado, con valores muy pequeños de K_i se tendrá una respuesta muy lenta. Por otro lado, con valores muy altos, se tendrá una respuesta más rápida, pero muy oscilatoria, por lo que debe tener cuidado con llegar a la inestabilidad.

- **Ganancia derivativa K_d :**

La parte derivativa del algoritmo se podría interpretar como el análisis del error futuro. Esta componente depende de la tasa de cambio del error, el cual nos permitirá mantener el error al mínimo, corrigiéndolo proporcionalmente con la misma velocidad que se produce, evitando así, que el error se incremente. Esta componente es la que nos permitirá tener cierto nivel de robustez al brindar a la planta la capacidad de corregir o volver a la señal de referencia si es que se modifica por algún factor externo. Añadir esta ganancia reduce tanto el grado de oscilación como el tiempo de respuesta [32]. Sin embargo, esta ganancia es muy sensible al ruido, por lo que este afecta mucho la respuesta en valores muy grandes, pudiendo volverla oscilatoria.

Para alcanzar una sintonización óptima de los valores existen muchos métodos o reglas para hacerlo, ya sea con un enfoque analítico o experimental y tanto el uso de métodos automáticos en línea como métodos para una sintonización delicada y fina [33].

2.4.2 Controlador difuso (*Fuzzy*)

Las técnicas de control convencionales generalmente requieren modelos matemáticos de sistemas para diseñar un controlador. Por otro lado, la mayoría de los modelos matemáticos de los sistemas de la vida real no son muy fáciles de obtener. Incluso, si se puede desarrollar un modelo relativamente preciso de un sistema dinámico, generalmente es demasiado complejo de utilizar en el desarrollo del controlador, especialmente para muchos procedimientos de diseño de control convencionales que requieren algunas suposiciones. En tales casos, el control difuso proporciona una estructura eficiente para incluir información lingüística de expertos humanos en información numérica. Esto no es posible en las técnicas de control convencionales. En este tipo de casos, se pueden preferir controladores difusos.

La estructura del controlador difuso se puede observar en la Fig. 2.5, donde se observa los cinco componentes o módulos funcionales del control difuso [34]:

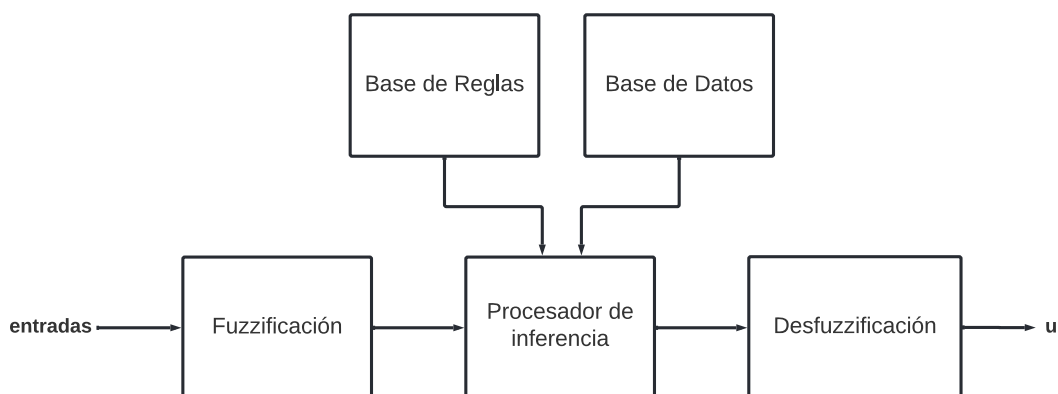


FIGURA 2.5: Diagrama de bloques de un controlador difuso.

- **Fuzzificador:**

Es la interfaz de entrada del controlador. Por este componente ingresan los parámetros de entrada (normalmente numéricos) que serán procesados por el controlador. Cada una de estas entradas está asociada a conjuntos difusos A y B que forman particiones de sus conjuntos universos. En la Fig. 2.6 se puede observar un ejemplo de fuzzificación de dos entradas numéricas, donde: μ representa el valor de pertenencia de cada conjunto, x la señal de entrada y y la señal de salida.

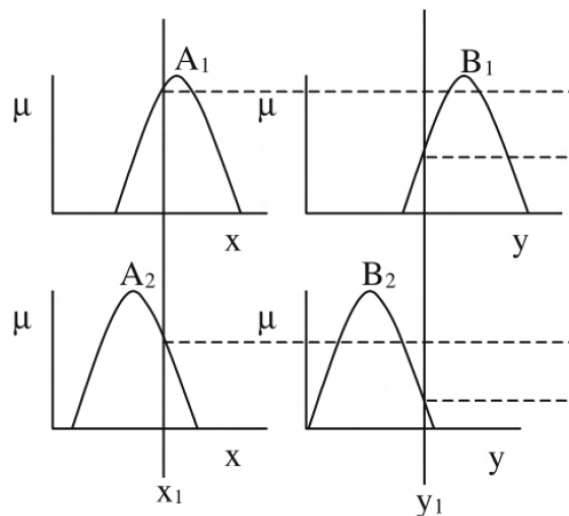


FIGURA 2.6: Fuzzificación de dos entradas numéricas [34]. Donde A_1 y A_2 pertenecen al conjunto universo A , y B_1 y B_2 al conjunto universo B

- **Base de datos:**

Este módulo establece los parámetros del sistema y las particiones de dominio.

- **Base de reglas:**

En la base de reglas se establece las relaciones difusas entre entradas y salidas. Estas relaciones tienen una estructura condicional como la siguiente: “Si x es A_k , y es B_k entonces z es C_k ”; donde x y y son entradas numéricas; z es una salida numérica y A_k , B_k y C_k son conjuntos difusos. Esta estructura condicional aplica para todo

tipo de combinaciones y va aumentando de acuerdo con el número de entradas y conjuntos existentes.

■ **Procesador de inferencia difusa:**

Este componente calcula un conjunto difuso a partir de una o más entradas difusas, basándose en las reglas difusas. Un ejemplo se puede observar en la Fig. 2.7.

	NG	NP	Ze	PP	PG
NG	NG	NG	NG	NP	Ze
NP	NG	NP	NP	Ze	PP
Ze	NG	NP	Ze	PP	PG
PP	NP	Ze	PP	PP	PG
PG	Ze	PP	PG	PG	PG

FIGURA 2.7: Base de reglas de error e y el cambio de error Δe . Siendo las particiones: NG = Negativo Grande, NP = Negativo Pequeño, ZO = cero, PP = Positivo Pequeño, PG = Positivo Grande [34].

■ **Desfuzzificador:**

Al igual que con las entradas del fuzzificador, las salidas de estos módulos son normalmente numéricas. En este, se calcula un elemento representativo z del conjunto difuso C' inferido por el módulo del procesamiento. Esto se puede ver gráficamente en la Fig. 2.8.

Al ser el control difuso tan flexible en cuanto al número de entradas, reglas, conjuntos difusos, etc. permite que haya variedad de tipos de este. Uno de estos es controlador difuso de tipo PD (Proporcional-Derivativo), el cual está basado en el controlador PD clásico, una variante del controlador PID que no cuenta con la parte integral. La familia de reglas difusas de este controlador es la siguiente [34]: “Si e es A_i y Δe es B_j entonces

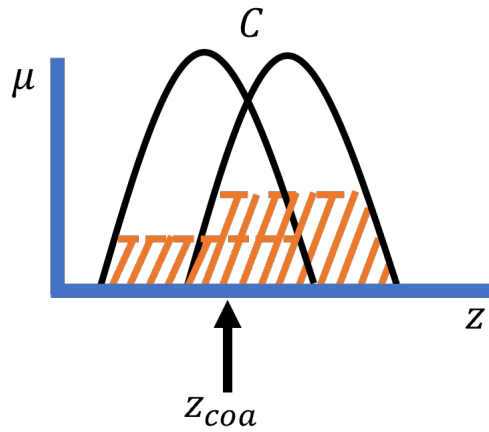


FIGURA 2.8: Defuzzificación del conjunto C' inferido utilizando el método de centro de área [34].

u es C_k ". En la Fig. 2.9 se puede observar el diagrama de bloques de un proceso con controlador difuso PD.

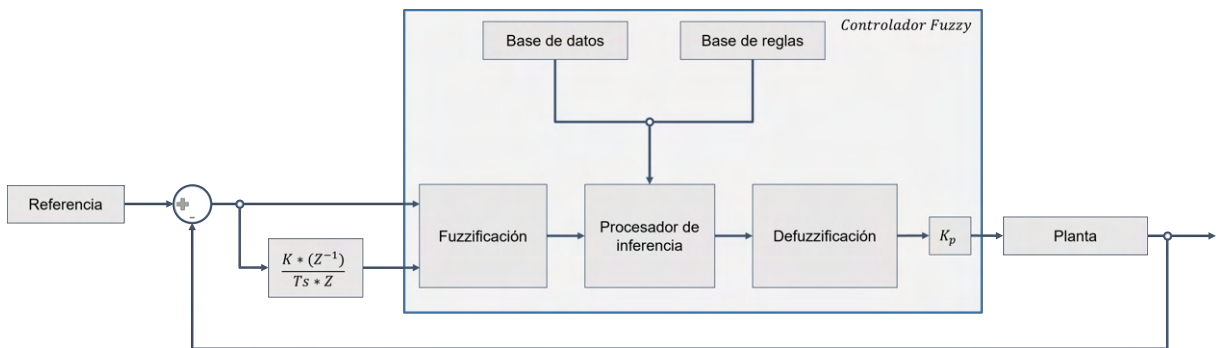


FIGURA 2.9: Diagrama de bloques de un proceso con controlador difuso PD.

CAPÍTULO III

MARCO METODOLÓGICO

En este capítulo, se describirá la metodología utilizada en el trabajo para alcanzar los objetivos propuestos y obtener los resultados correspondientes. En la primera sección, se presentarán los entornos donde se llevarán a cabo las distintas pruebas del controlador. En la segunda sección, se seleccionarán los sensores que se implementarán en el sistema para obtener retroalimentación sobre las variables físicas del sistema robótico en los ejercicios de flexión y extensión de la rodilla. La tercera sección abordará el entorno en el que se implementará el controlador, junto con las configuraciones que se tendrán en cuenta. Por último, las dos últimas secciones explicarán los requisitos del controlador y la metodología utilizada para su diseño e implementación

3.1 Actuador flexible utilizado

El actuador flexible accionado por vacío utilizado (VPAM) para las pruebas experimentales era ligero y tenía una sección transversal rectangular (30×15 mm). Se fabricó teniendo en cuenta las especificaciones clínicas para la rehabilitación. Cada uno de los anillos (piezas azules) era ligeramente mayor que la sección transversal ($34,2 \times 21,2$ mm), con una anchura de 6 mm. Además, se imprimieron en 3D con una Prusa MK3s+ y se fabricaron con ácido poli láctico (PLA). Los anillos se unieron con un filme de polietileno de 0,1 mm de grosor, lo que permitió al VPAM contraerse y expandirse. El VPAM está formado por 8 celdas, cada celda está formada por dos anillos separados por 15 mm por el filme (ver Fig. 3.1). Obteniendo una longitud total del actuador de 174 mm.

El actuador no fue caracterizado de manera matemática pero una investigación previa realizada por el grupo MabLab de UTEC [28] demuestra que el actuador puede generar 25 N de fuerza trabajando a -20 kPa pero su relación no es tan directa (ver Fig. 3.2). Sin embargo, para caracterizar la dinámica del actuador se tendría que realizar experimentos para cada valor de presión que se encuentre dentro del rango de trabajo, aunque incluso con esta información no tendría mucho impacto porque no se posee un sensor de fuerza en el exoesqueleto.

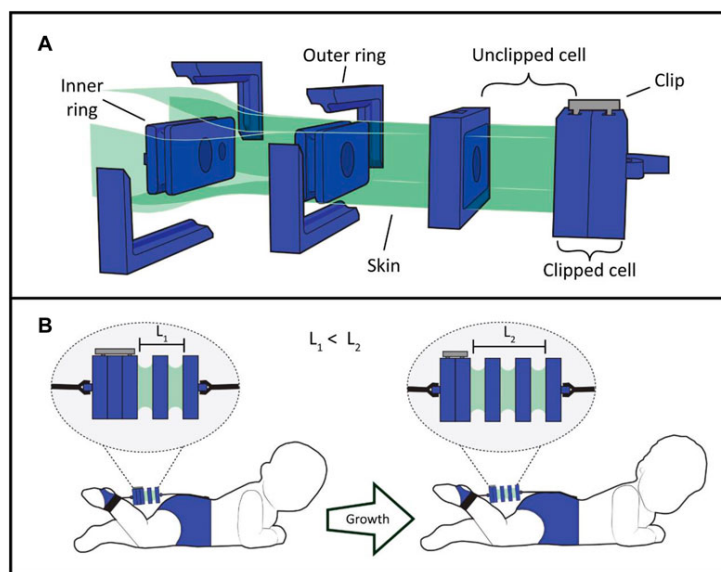


FIGURA 3.1: Estructura y funcionamiento del actuador flexible [28].

3.2 Modelo de rehabilitación utilizado

Debido a las limitaciones mencionadas, no se llevó a cabo pruebas con bebés reales. En su lugar, todas las pruebas se realizaron utilizando un modelo de pierna simplificado impreso en 3D (ver Fig. 3.3) y dos maniqués fabricados de silicona que tienen un esqueleto impreso con PLA que simulan las extremidades inferiores de un bebé de cero meses (ver Fig. 3.5) y de un bebé de seis meses (ver Fig. 3.6). El modelo de pierna simplificado (ver Fig. 3.3) es un banco de pruebas modular impreso en 3D que permitió

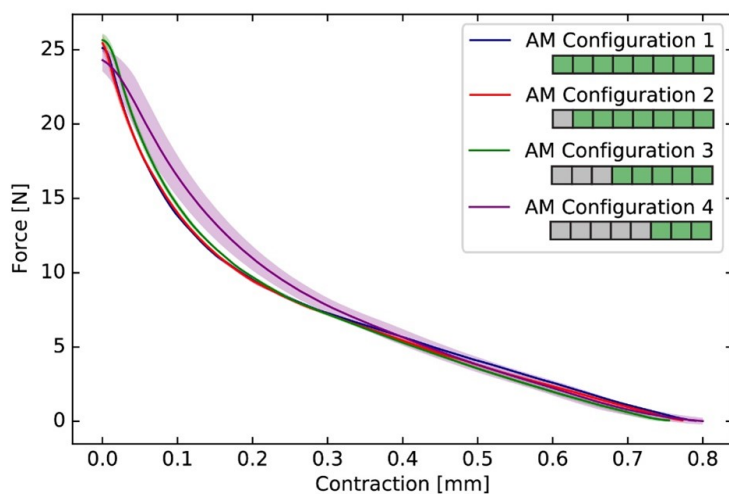


FIGURA 3.2: Resultados de la prueba experimental de fuerza-contracción con el actuador flexible.

variar las longitudes y pesos según sea necesario, dentro del rango correspondiente a la edad de cero a seis meses. Este sistema tuvo un rango de movimiento de 0° a 90° . Para simular el peso de la pierna y el pie, se utilizaron discos metálicos colocados en el extremo y en el medio de la barra, respectivamente. La Fig. 3.4 ilustra la equivalencia entre el modelo de pierna simplificado y las extremidades inferiores reales, así como la ubicación del actuador en el modelo.

Por otro lado, los maniqués utilizados en el estudio fueron desarrollados por el departamento de mecánica de la Universidad Nacional de Trujillo como parte del proyecto *ExoFlex*, liderado por el profesor Cesar Pinedo. El objetivo principal de este proyecto fue replicar con la mayor precisión posible las condiciones antropométricas de un bebé cero y seis meses. Sin embargo, una limitación de los maniqués es que no se pueden modificar para simular diferentes condiciones, ya que su peso y longitud están fijos debido a su construcción. En contraste, el modelo de pierna simplificado proporcionó la capacidad de realizar pruebas que simulaban diversas combinaciones de longitud y peso. Esto permitió evaluar el rendimiento del controlador en diferentes escenarios y realizar un análisis

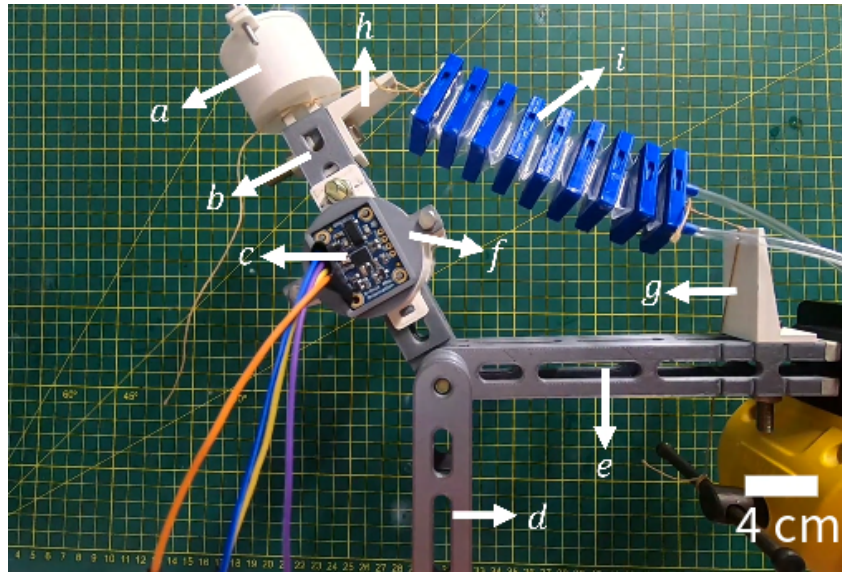


FIGURA 3.3: Modelo simplificado de pierna de bebé fabricado. Donde cada parte sería equivalente a: a) peso del pie, b) hueso de la pierna, c) IMU, d) soporte para evitar pandeo, e) femur, f) peso de la pierna, g y h) soportes para el actuador flexible e i) actuador flexible.

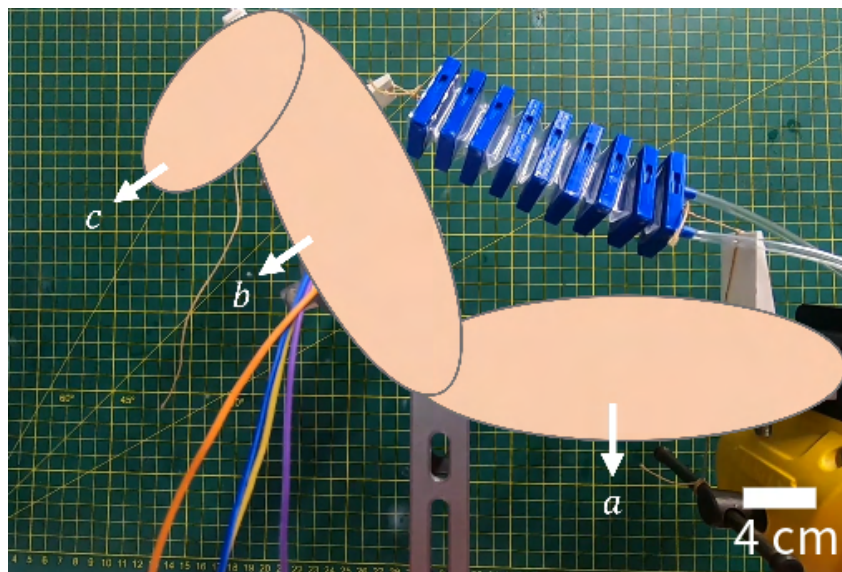


FIGURA 3.4: Equivalencia del modelo simplificado de pierna de bebé. Donde: a) representa el muslo del bebé, b) la pierna y c) el pie.

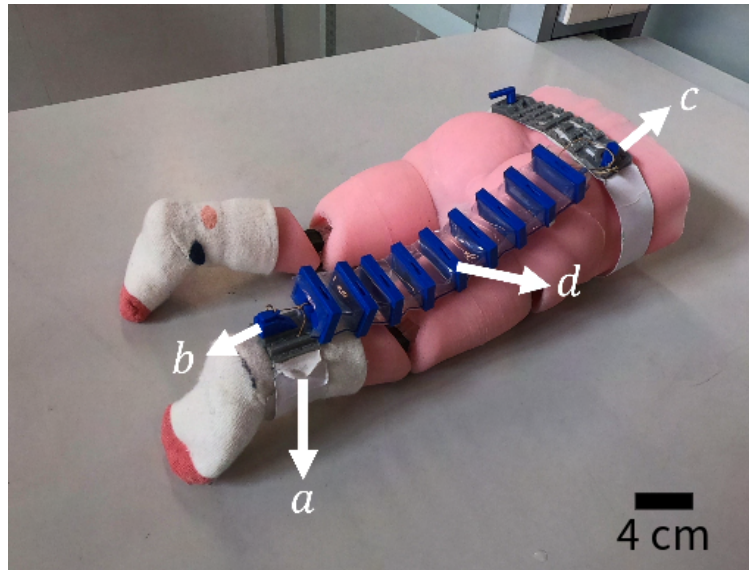


FIGURA 3.5: Maniquí utilizado para las pruebas experimentales de cero meses. Donde cada letra hace referencia a: a) soporte para el IMU, b y c) anclaje para el actuador y d) el actuador flexible utilizado



FIGURA 3.6: Maniquí utilizado para las pruebas experimentales de seis meses.

más completo, como observar su comportamiento a medida que se incrementa el peso y determinar sus límites de funcionamiento.

El modelo simplificado fue utilizado para realizar pruebas del controlador en distintas combinaciones de longitud y pesos debido a su modularidad. Mientras que los maniquís fueron utilizados para la validación final del controlador siendo las pruebas con las condiciones más reales a las que el controlador se estaría desempeñando. Además mencionando que los maniquís llegaron en la etapa final del proyecto y no fue posible realizar un análisis profundo en estos.

3.3 Sistema de retroalimentación - actuación

Para el ejercicio de rehabilitación es importante saber el ángulo en el que se encuentra la pierna del bebé en todo instante porque este no debe superar los 64° y es el valor que buscaremos controlar para que siga un patrón definido por el usuario, el cual permitirá realizar la rehabilitación de forma automática. Para obtener información de este ángulo se utilizó el sensor de orientación absoluta Adafruit BNO055 (Fig. 3.7).

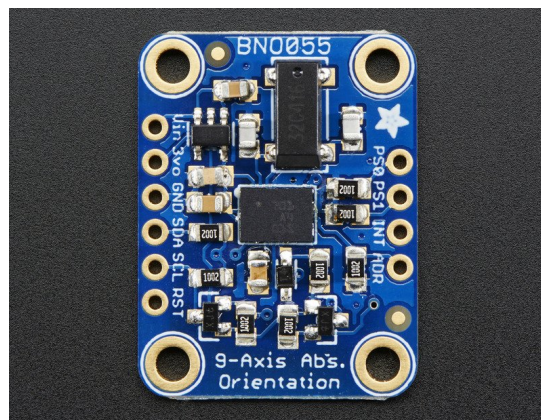


FIGURA 3.7: Sensor de orientación absoluta [35]

Este sensor estuvo compuesto por un IMU (unidad de medición inercial) de 9 grados de libertad, la cual contó internamente con acelerómetros, magnetómetros y giroscopios en los 3 ejes del espacio [36]; y un procesador ARM Cortex-M0 que permitió, mediante un algoritmo integrado, filtrar (Kalman) y calibrar la data generada por los sensores de la IMU, brindando al usuario las medidas en el espacio de rotación angular/orientación 3D, velocidad y aceleración lineal y angular directamente [35] sin necesidad de implementar un algoritmo propio para obtenerlos. Además, contó también con un sensor de temperatura interno, el cual no se utilizó para la investigación. Este sensor de orientación absoluta fue colocado en el tobillo del bebé, o lo que es equivalente, en el extremo de la pierna fabricada (Fig. 3.8).

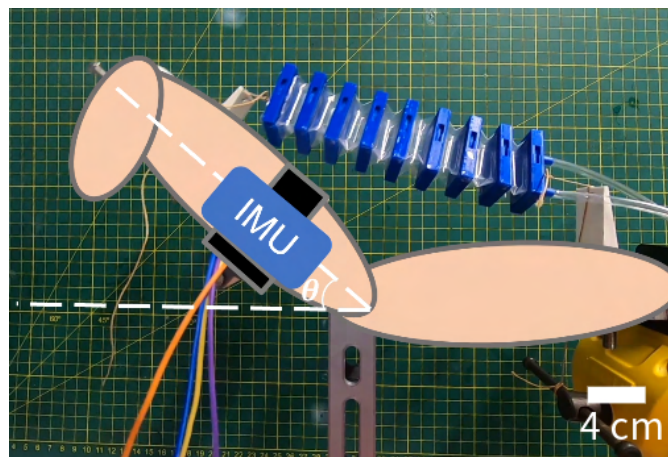


FIGURA 3.8: Locación del IMU en la pierna del bebé. Donde θ es el ángulo de inclinación de la pierna.

El actuador para la aplicación fue el regulador de presión electrónico ITV2091-2, el cual fue el encargado de suministrarle una presión deseada definida por el usuario. Es importante configurar el regulador ajustando su ganancia y su sensibilidad, esto para que pueda tener un tiempo de reacción rápida y pueda llegar a la presión deseada lo más rápido posible, ya que el controlador necesita que el regulador llegue a la presión deseada en un pequeño instante de tiempo. Para esta aplicación se configuró el regulador para que

trabaje a una ganancia de F (lo cual significa la respuesta más rápida) y una sensibilidad de 5 (lo que significa que trabajará a la mayor precisión posible).

El regulador de presión electrónico ITV2091-2 necesitó una señal de voltaje de entrada para generar un valor de presión de salida. La relación existente entre la presión de salida y el voltaje de entrada se muestra en (3.1).

$$Presión_{salida} = \frac{Voltaje_{entrada} \cdot Presión_{máxima}}{5V} \quad (3.1)$$

Dentro del *datasheet* del regulador [37] se recomendó trabajar con una resolución de 12 bits por lo que se utilizó un módulo DAC (Adafruit MCP4725) que nos permitió llegar a dicha resolución y tener un buen desempeño. Adicionalmente, se utilizó un regulador manual (IRV10A-C06LZN) conectado a la entrada del regulador electrónico, esto para que el regulador electrónico tuviera una presión de entrada constante y así pudiera realizar el control interno de manera más eficiente, tal y como se recomendó en el *datasheet* de este. El regulador manual se fijó para un valor de presión constante de -50 kPa mientras que el regulador electrónico se trabajó en un rango de 0 kPa hasta -30 kPa. Se eligió el límite de -30 kPa debido a que este fue el máximo valor de presión necesario para obtener un ángulo de flexión de 60° luego de realizar distintas pruebas en lazo abierto con pesos y longitudes.

De igual forma, fue importante tener en cuenta la presión que llega al actuador flexible en cada instante de tiempo. Conocer cuál fue la presión interna del actuador flexible nos permitió saber si es que existe una fuga en este o si hay pérdida de presión en las tuberías neumáticas. Además, idealmente lo sensado dentro del actuador debería ser igual o muy similar a la ley de control que se requiera, lo que nos indicó si es que el control de presión interno que tiene el regulador electrónico es capaz de generar la curva de presión necesaria en cada instante de tiempo.

Finalmente, los sensores de presión, el IMU y el *DAC* fueron conectados a un Arduino Mega 2056 que fue el encargado de enviar los datos a una computadora mediante comunicación serial. El Arduino fue utilizado como un dispositivo de adquisición de datos (*DAQ*), ya que este recibirá los datos de los sensores, los enviará a la computadora, este los procesará y generará un valor de presión u que el regulador electrónico debe generar en ese instante, el Arduino recibirá dicho valor u mediante comunicación serial y lo enviará al regulador electrónico mediante la *DAC*. El Arduino no realizó ningún tipo de procesamiento de las señales recibidas, más que la obtención de los valores finales usando las librerías de cada sensor junto a una transformación de los valores recibidos a bits que envió mediante comunicación serial a la computadora.

La Tabla 3.1 muestra un resumen de todos los dispositivos utilizados junto con una breve descripción de estos. Por otro lado, la Fig. 3.9 muestra, gráficamente, la conexión entre todos los dispositivos físicos junto con el tipo de señal de entrada y salida de los componentes. Mientras que la Fig. 3.10 muestra una representación de la parte neumática del sistema.

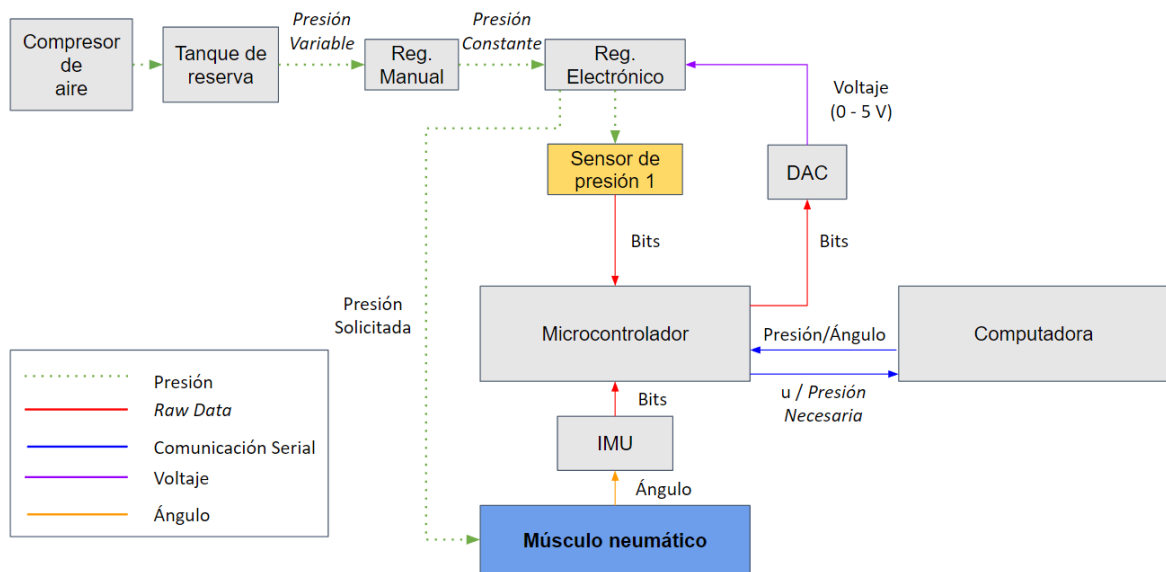


FIGURA 3.9: Diagrama de los componentes utilizados.

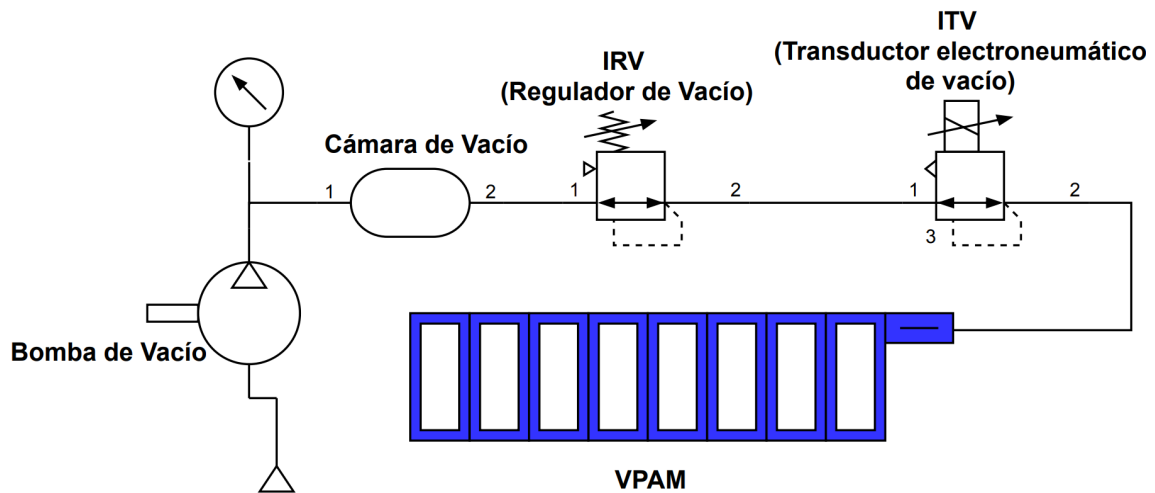


FIGURA 3.10: Circuito Neumático realizado.

Equipo	Utilidad	Tipo
SMC ITV2091-21N2BS5	Suministrar una presión deseada por el usuario al actuador. Obtener información de la presión que el regulador está suministrando a través de las tuberías.	Actuador / Sensor
Mini sensor de presión negativa	Obtener información de la presión que el actuador está recibiendo.	Sensor
SMC IRV10A-C06LZN	Suministrar una presión constante al regulador.	Actuador
Adafruit BNO055	Sensar el ángulo de flexión y extensión de la pierna.	Sensor
Arduino Mega 2056	Encargado de la recepción y envío de datos que el sistema genere.	Controlador
Módulo DAC MCP4725 de 12 bits	Encargado de generar una señal analógica a partir de una señal digital.	DAC

TABLA 3.1: Equipos utilizados.

3.4 Entorno de implementación del controlador

Como se mencionó, el Arduino funcionó como una *DAQ*, es decir, solamente recibió y envió datos a la computadora y no realizó ningún tipo de procesamiento adicional. Su lógica de funcionamiento se comparte en el diagrama de flujo de la Fig. 3.11, el cual

se resalta que este solamente recibe y envía los datos, siendo su único proceso importante la lectura por los pines analógicos y transformación de los valores a bits, los cuales se enviaron por serial a la computadora. Para la implementación del controlador se utilizó Simulink debido a que una computadora tenía un mayor poder de procesamiento que un microcontrolador y nos dio más facilidades al momento de implementar el controlador difuso, ya que se tuvo un *toolbox* llamado *Fuzzy Logic Designer* (FLD) que simplificó el proceso de implementación de las reglas, funciones de membresía y los métodos de fuzzificación y defuzzificación. Adicionalmente Simulink permitió realizar operaciones como derivar, integrar, visualizar y almacenar variables en tiempo real, lo cual facilitó el análisis del desempeño del controlador lo cual sería complicado de realizar solamente con un Arduino.

Se evaluaron otros métodos para la implementación del controlador, una lista de los métodos junto a sus ventajas y desventajas se muestran en la Tabla 3.2. Al lograr independizar el microcontrolador con la computadora podemos aprovechar el poder de procesamiento de una computadora y ejecutar gran cantidad de algoritmos complejos (computacionalmente) y enviarle al microcontrolador solamente el resultado o el valor que el necesita para seguir su flujo de trabajo. La estructura del programa realizado en Simulink se muestra en la Fig. 3.12 y se puede dividir en cuatro partes: Generador y almacenamiento de variables (color amarillo), algoritmo de control (color verde), visualización de las variables (color morado) ,y envío y recepción datos por comunicación serial. Una gran ventaja de trabajar con Simulink es que, por ejemplo, si se desea implementar algún otro algoritmo de control solamente se modificaría el bloque verde de la Fig. 3.12.

Finalmente, otro aspecto importante es la frecuencia a la que se ejecutó todo el lazo cerrado. El *datasheet* del sensor BNO055 indica que este trabaja a una frecuencia máxima de 100 Hz [36] y que es recomendable que el Arduino esté configurado a 115200 bps; sin embargo, se realizaron pruebas enviando una señal de voltaje al regulador electrónico a una frecuencia de 100Hz y este mostró fallos como fuertes vibraciones y ruidos. Se

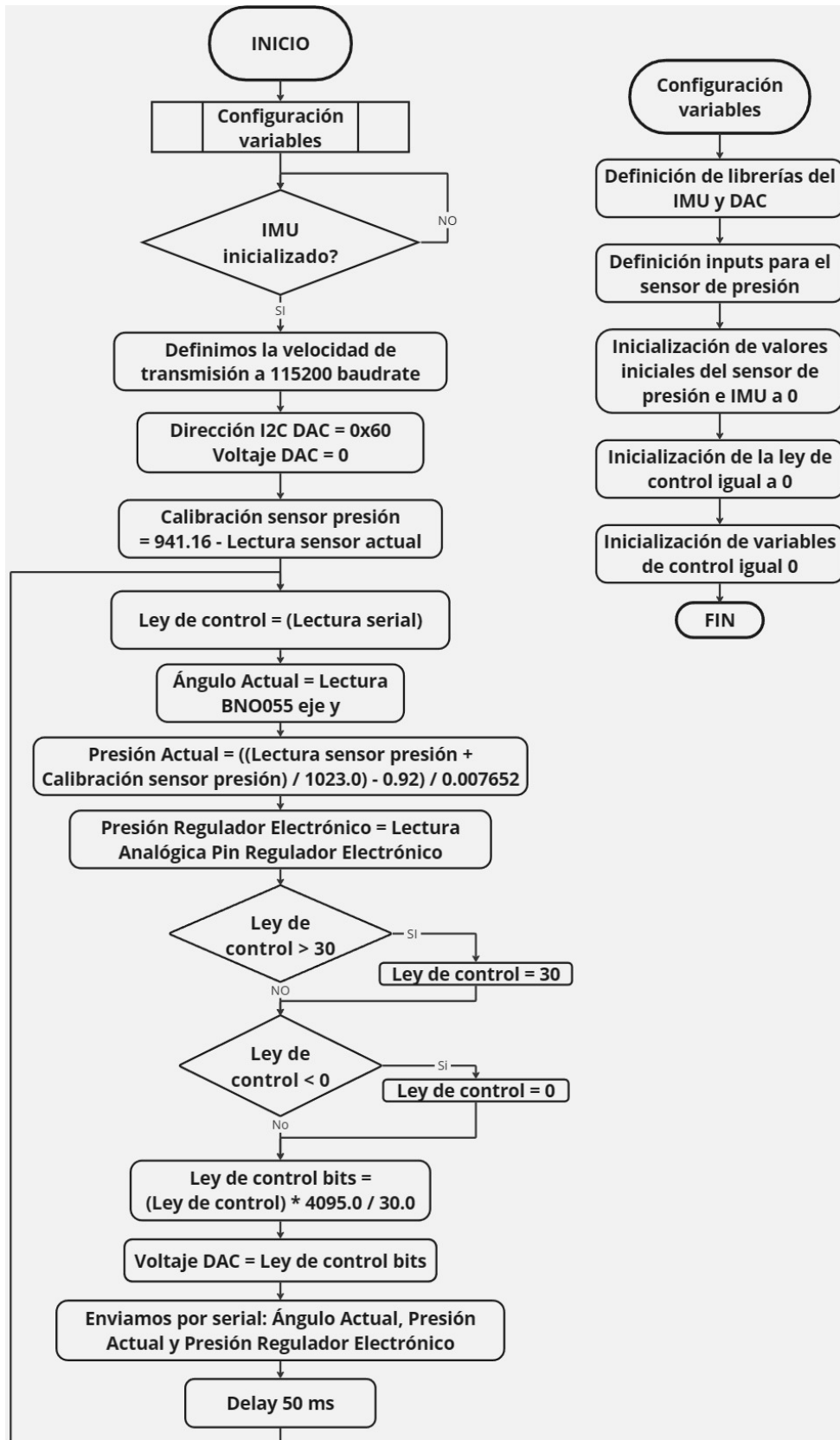


FIGURA 3.11: Diagrama de flujo Arduino Mega.

realizaron pruebas variando la frecuencia de trabajo y se observó que trabajar a una frecuencia de 20Hz todos los componentes presentan un correcto funcionamiento. Además, para esta aplicación no es necesario trabajar a grandes frecuencias, ya que el movimiento deseado es lento (tres segundos) y el sensor a esta frecuencia es capaz de captar de buena manera la dinámica del sistema sin mucha pérdida.

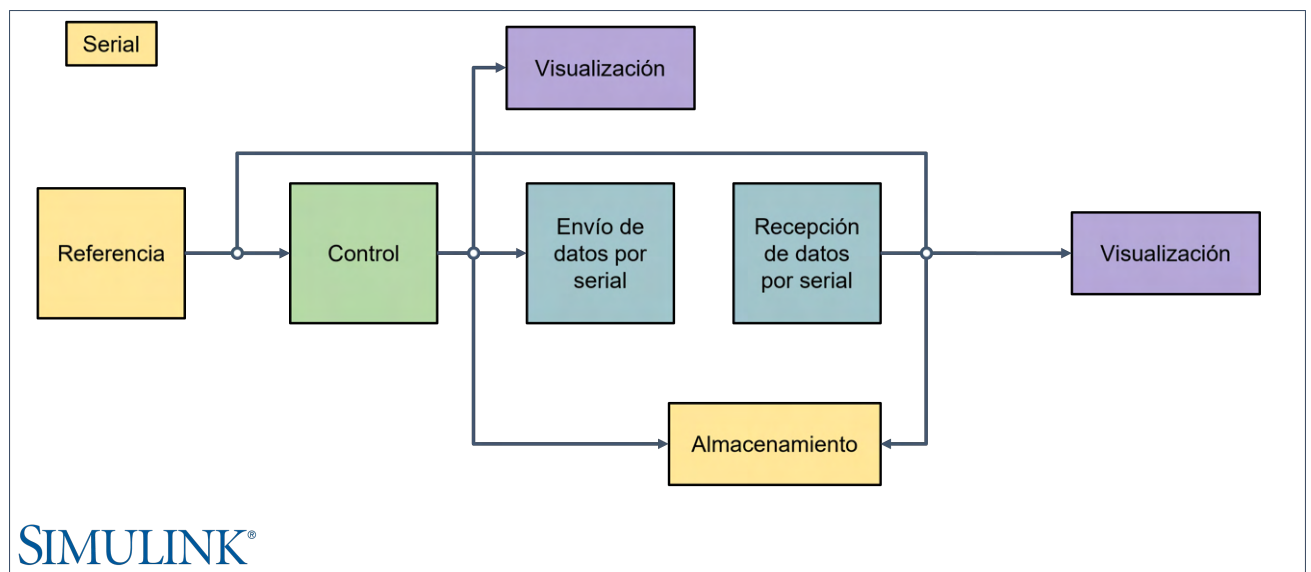


FIGURA 3.12: Diagrama de bloques referencial utilizado en Simulink.

3.5 Requerimientos de control

Los requerimientos del controlador se obtuvieron en base a la opinión de un doctor experto que es parte del proyecto *ExoFlex*, llamado Segundo Cruz quien es neonatólogo del Instituto Nacional de Salud del Niño - San Borja. Se le consultó al doctor sobre las especificaciones que debería tener la trayectoria deseada para que cumpla con su función de estimular las extremidades inferiores del bebé. Asimismo, con su apoyo se obtuvo el rango aproximado de variación de pesos y longitudes de los bebés entre cero y seis meses. La Tabla 3.3 muestra un resumen de los requerimientos obtenidos con apoyo del doctor Segundo Cruz y será una manera adicional de evaluar el desempeño del controlador.

Si bien no existe una trayectoria ideal para realizar la rehabilitación lo importante es plantear una que cumpla con los tiempos mostrados en la Tabla 3.3. La Fig. 3.13 presenta la trayectoria elegida la cual tiene una forma trapezoidal con: tres segundos de flexión, cinco segundos de extensión y cinco segundos de reposo entre flexión y extensión junto con una amplitud de 50 desde 10 hasta 60 siendo el valor máximo valor de flexión tolerable 64 según el doctor. El movimiento deseado es suave en todo instante con un comportamiento lineal en flexión y extensión lo cual facilitará al controlador. Sin embargo, esta no es la única trayectoria posible, es posible utilizar otras que cumplan principalmente el rango de tiempos de extensión y flexión junto a la amplitud del movimiento, la Fig. 3.14 muestra otra posible trayectoria haciendo uso de una función sinusoidal. Por simplicidad se optará por utilizar la trayectoria trapezoidal y en futuros trabajos se evaluará la trayectoria óptima para realizar el ejercicio.

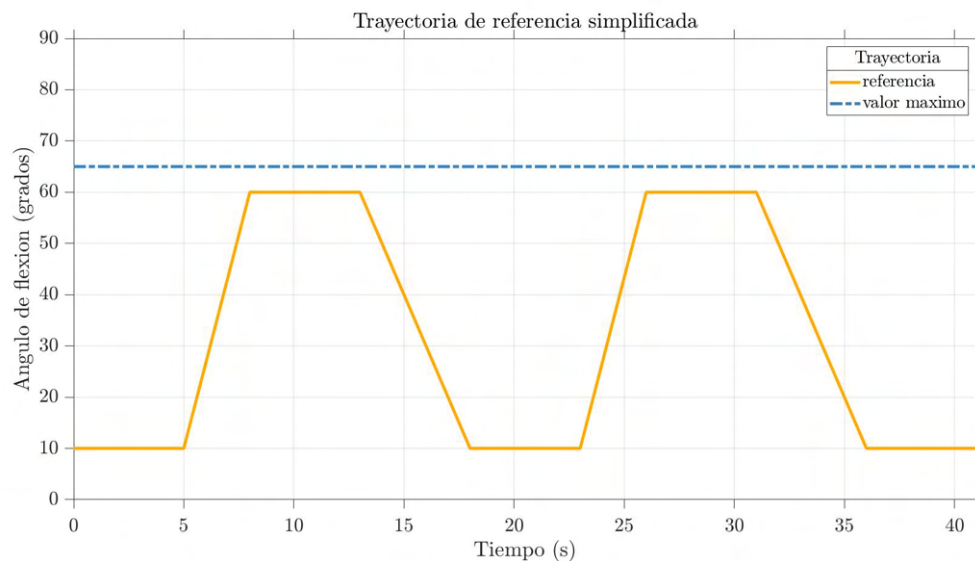


FIGURA 3.13: Trayectoria trapezoidal propuesta.

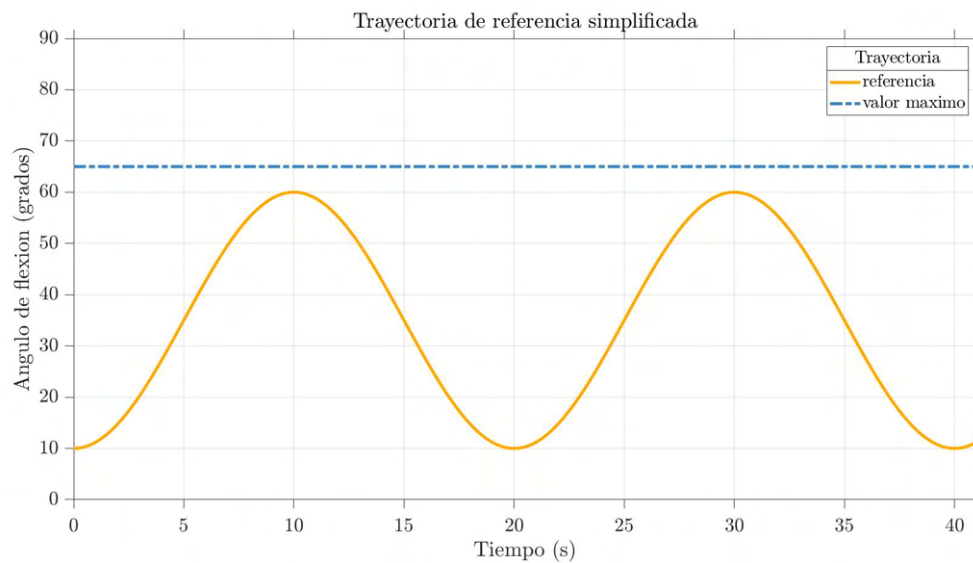


FIGURA 3.14: Trayectoria sinusoidal propuesta.

3.6 Diseño del controlador difuso

El controlador difuso a utilizado fue diseñado en Simulink aprovechando de que se logró realizar una conexión entre el Arduino con Matlab. Se utilizó el *Fuzzy Logic Toolbox* para facilitar el diseño del controlador aprovechando la conexión que se realizó con Matlab. Las entradas del controlador se basaron en el error y la derivada del error utilizando un integrador en la salida del controlador, una estructura propuesta en [22] y que se puede observar en la Fig. 3.15. Esta estructura nos permitió trabajar de manera similar a la de un controlador $PD - Fuzzy$ con la principal diferencia de las reglas que se utilizaron internamente y al integrador que se tuvo a la salida. [22] menciona que utilizar un controlador PI implementado de esta manera tiene un desempeño similar a la de trabajar con la integral del error y que no es tan costoso computacionalmente debido a que los valores de la derivada del error no toman valores muy elevados. Además, el integrador discreto permite tener una ley de control suave pese a que du tenga cambios muy bruscos.

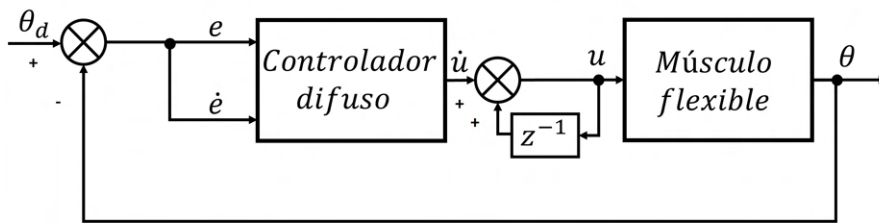


FIGURA 3.15: Estructura del controlador difuso. Donde: θ es el ángulo de inclinación de la pierna, θ_d el ángulo deseado, u la ley de control y e el error.

Se utilizó el controlador difuso del tipo Mandami por su facilidad de implementación y que es el más recomendado para este tipo de aplicaciones [22]. Sin embargo, la elección de la forma de las funciones de membresía (MF) y de las reglas no se realizaron completamente de forma empírica o basadas en prueba y error, ya que se utilizó el algoritmo *fuzzy - c - means* para realizar un mejor análisis de datos y tomarlo como punto de partida. Lo que se hizo primero será implementar un controlador PID realizando distintas pruebas para distintas condiciones de trabajo y trayectorias. Luego se utilizó el algoritmo *fuzzy - c - means* para poder observar como serían las formas de las funciones de membresía y si es necesario utilizar una gaussiana, triangular, trapezoidal o sigmoideal y de igual forma se observó la cantidad de funciones de membresía necesarias gráficamente. Una vez que se tenga el número de funciones de membresía y el tipo que serían cada una de estas lo que se hizo será obtener los mejores parámetros que permitan aproximar dichas funciones a su representación en una ecuación matemática, es decir, por ejemplo, obtener los mejores parámetros de la norma y varianza para representar cada una de las funciones de membresía de manera independiente. Esto se hace porque el *FLD* necesita los parámetros de la ecuación matemática de cada función de membresía numéricamente para poder funcionar de manera óptima. Paso seguido se diseñaron las reglas en base a la experiencia que se obtuvo luego de realizar cada uno de los experimentos y luego se implementó en el *FLD*.

La selección de los límites para la derivada del error y el error se podrían realizar de

manera empírica, sin embargo para mayor precisión se optó por realizar un análisis de la distribución de los datos con histogramas. Mediante inspección visual se pudo determinar cuál es el rango de los valores con más incidencia durante los experimentos. Un esquema que indica cada uno de los pasos realizados para el diseño del controlador difuso se puede apreciar en la Fig. 3.16.

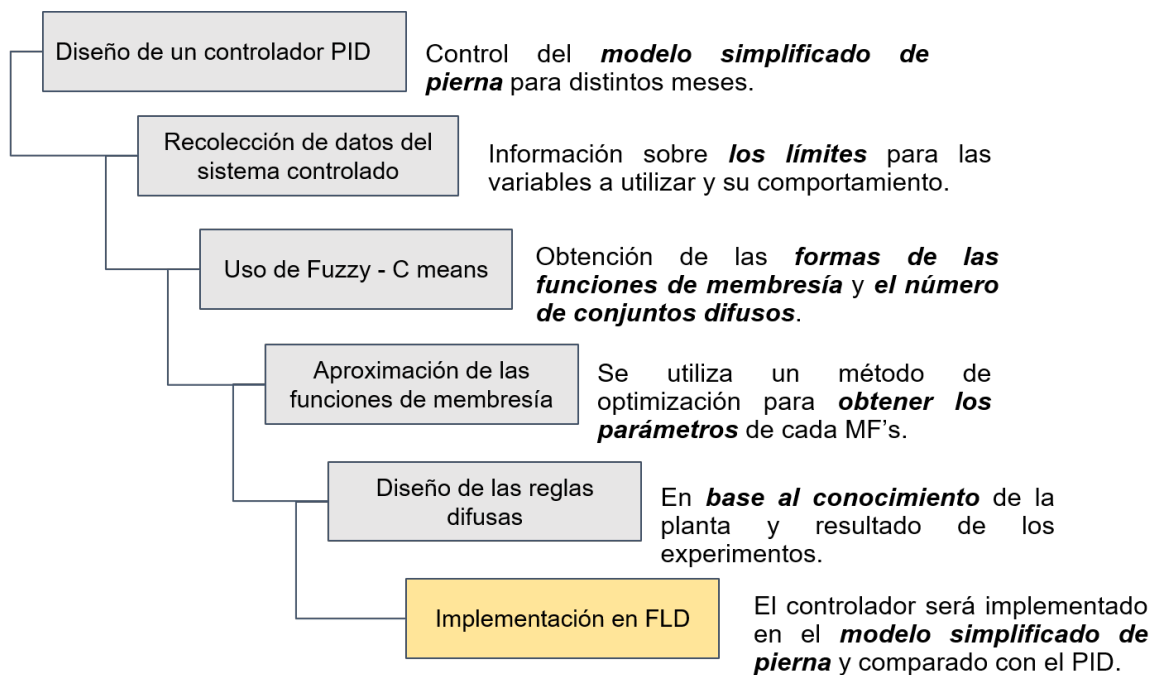


FIGURA 3.16: Estructura de la metodología planteada.

Método	Ventaja	Desventaja
Arduino - Simulink alto nivel [38]	<ul style="list-style-type: none"> ■ Toda la programación del Arduino se realiza mediante los bloques de Simulink y se tiene bloques exclusivos para Arduino. ■ Se pueden hacer modificaciones <i>on – line</i>. 	<ul style="list-style-type: none"> ■ Todo el programa funciona utilizando el microcontrolador, por lo que hacer uso de varios bloques como <i>scopes</i> o incluso el <i>FLD</i> se mostró un delay de hasta 5 segundos. ■ Simulink no posee un bloque para el uso de la <i>DAC</i> y se limita a usar los bloques exclusivos que posee.
Arduino - MATLAB [38]	Es posible utilizar las librerías de Arduino y muestra una programación muy similar al IDE de Arduino.	<ul style="list-style-type: none"> ■ Es necesario diseñar el control difuso manualmente desde cero. ■ Visualizar los datos deseados en tiempo real es complicado. ■ No se pueden hacer modificaciones <i>on – line</i>.
Solamente Arduino	<ul style="list-style-type: none"> ■ Acceso directo a todas las librerías de los sensores. ■ Mayor libertad para configurar o realizar operaciones. ■ No es necesario de una computadora para hacer uso del sistema. 	<ul style="list-style-type: none"> ■ Poco poder de procesamiento. ■ No se pueden hacer modificaciones <i>on – line</i>. ■ Es necesario diseñar el control difuso manualmente desde cero.
Arduino - Simulink Serial	<ul style="list-style-type: none"> ■ El Arduino y el Simulink tienen programas independientes. ■ Todos los algoritmos <i>pesados</i> se ejecutan en la computadora y no en el Arduino. ■ Uso de los bloques de Simulink. ■ Es posible hacer modificaciones <i>on – line</i>. 	<ul style="list-style-type: none"> ■ Se debe sincronizar la frecuencia de trabajo del Simulink con el Arduino. ■ No es posible hacer uso del sistema sin una computadora. ■ Existe la posibilidad de algún tipo de pérdida de datos.

TABLA 3.2: Métodos de implementación del controlador.

Parámetro	Valor
Posición del recién nacido	Prono (boca abajo)
Rango de edad	Desde cero a seis meses
Rango de trabajo (flexión y extensión)	desde 10° hasta 60°.
Valor máximo de flexión	64°.
Características del movimiento	<ul style="list-style-type: none"> ■ Tres segundos flexión. ■ Cinco segundos de extensión. ■ Entre tres a cinco segundos de pausa. ■ Movimiento suave sin cambios bruscos principalmente en la extensión. ■ Movimiento periódico.
Rango aproximado de pesos	<ul style="list-style-type: none"> ■ Parte inferior de la pierna: [160 – 260] gramos. ■ Pierna: [52 – 103] gramos.

TABLA 3.3: Tabla de requerimientos definidos en base a recomendaciones de especialistas del INSN San Borja

CAPÍTULO IV

RESULTADOS

4.1 Comportamiento del sistema en lazo abierto

Es crucial realizar una verificación inicial del correcto funcionamiento de la comunicación entre todos los dispositivos que se utilizaron en los experimentos posteriores. Para ello, se ha establecido un esquema detallado de conexión entre los equipos, que se muestra en la Figura 3.9. En esta fase inicial, se llevó a cabo una prueba a lazo abierto del sistema. El objetivo de esta prueba era monitorear simultáneamente todas las señales deseadas mientras se realizaban los experimentos. Al finalizar esta etapa, todos los datos recopilados se almacenarán para su posterior análisis. Cabe señalar que para esta investigación se utilizó una convención que consta de mostrar de manera positiva la presión de vacío que genera la bomba de vacío en todas las gráficas para una mejor interpretación.

Se envió una referencia sinusoidal con una amplitud de 13.5 kPa y una frecuencia de 1.0472 Hz para observar la respuesta del sistema y verificar el funcionamiento de cada componente. El resultado de aplicar esta referencia de presión se muestra en la Figura 4.1, donde la línea azul representa las lecturas del sensor de orientación absoluta (BNO055) y la línea naranja representa el valor de presión ingresado. Cabe destacar que, debido a la naturaleza del lazo abierto, la señal de entrada se limitó a un máximo de 20 kPa, ya que a esta presión el actuador alcanza su contracción máxima.

La respuesta obtenida revela la necesidad de implementar un sistema de control, ya que la relación entre la presión de entrada del sistema y el ángulo de salida no es directa. Observamos que el ángulo de salida sigue un patrón repetitivo, pero presenta un desfase respecto al valor deseado. Es importante destacar que los ángulos óptimos para nuestro

sistema deberían estar en el rango de 0° y 70° , y la presión suministrada no debe superar los 20 kPa.

Además, se observó un desfase de aproximadamente 0.8 segundos entre el valor máximo de presión enviado y el valor máximo del ángulo generado. Esto indicó que no siempre alcanzamos el ángulo máximo cuando la presión está en su punto más alto y que, incluso cuando la presión comienza a disminuir, el ángulo puede seguir aumentando. Estos puntos mostraron la complejidad del sistema y enfatizaron la importancia de implementar estrategias de control para lograr un funcionamiento preciso y consistente.

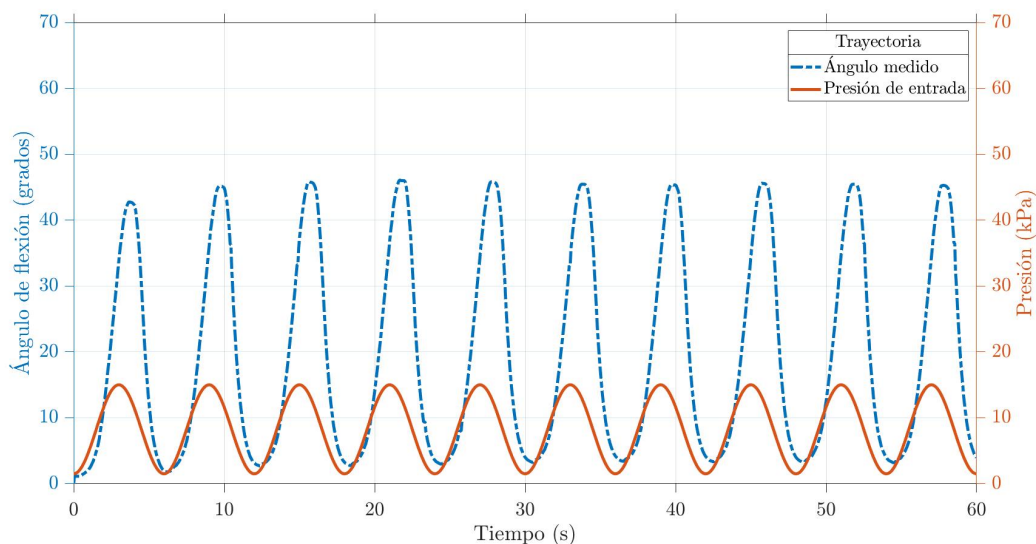


FIGURA 4.1: Respuesta del sistema en lazo abierto frente a una entrada sinusoidal. *(Experimento realizado en el banco de pruebas)*

Por otro lado, hemos podido monitorear con precisión la presión de salida del regulador, la cual debería ser similar al valor de presión enviado por la computadora, como se muestra en la Figura 4.2. Es crucial notar que las lecturas del regulador electrónico exhiben cierto nivel de ruido y oscilaciones. Estos fenómenos pueden ser atribuidos al control interno del regulador y a las posibles interferencias en la transmisión de datos a través de los cables conectados al Arduino.

Es relevante destacar que la similitud entre las señales medida y deseada en esta gráfica es esencial. Al implementar el controlador difuso, asumimos que el regulador de presión opera correctamente. Si el regulador no puede seguir la referencia de presión enviada, esto puede afectar el rendimiento de nuestro controlador difuso u otros controladores utilizados. Además, consideramos la posibilidad de que el sistema experimentó pérdida de presión en las tuberías, porosidad del material PLA o incluso internamente en los reguladores. Estas imperfecciones no pueden corregirse directamente. Por lo tanto, se volvió necesario emplear un modelo que tome en cuenta estos detalles.

Sin embargo, la complejidad de la dinámica del sistema dificultó la obtención de una ecuación que relacione la presión con el ángulo de salida de manera directa. Por lo tanto, se requirió un enfoque que integre estas complejidades para lograr un control efectivo y preciso (error < 4 %) del sistema.

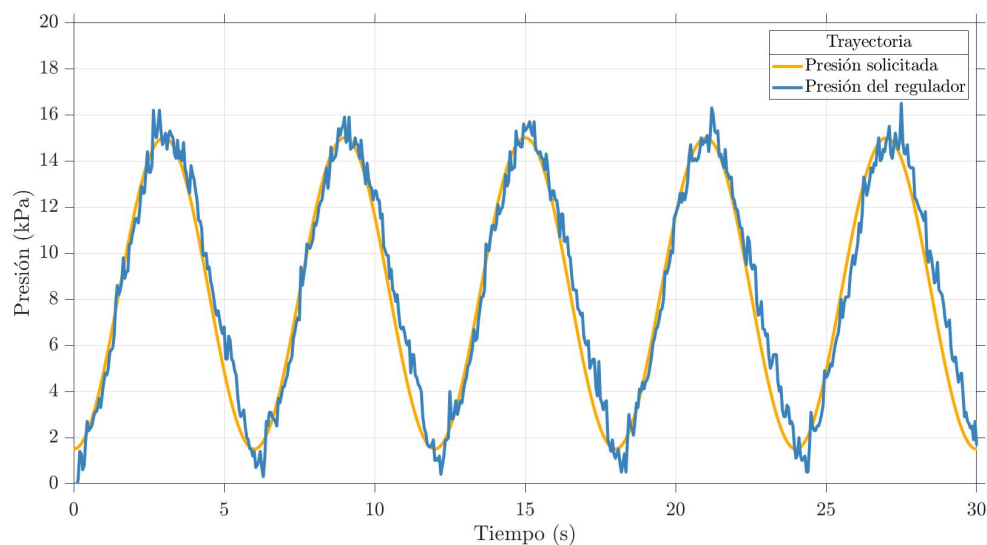


FIGURA 4.2: Comportamiento de la presión de salida del regulador electrónico junto con la presión ideal. (*Experimento realizado en el banco de pruebas*)

Además, pudimos observar gráficamente la histéresis presente en el actuador. Para este propósito, se realizó un experimento en el que se aplicó un escalón de presión que se incrementaba después de intervalos uniformes de tiempo. Idealmente, el comportamiento

de compresión debería ser muy similar al de extensión. El experimento se llevó a cabo utilizando las características de un recién nacido de cero meses en un entorno de lazo abierto. La presión inicial fue de 0 kPa, y se incrementó gradualmente hasta alcanzar un máximo de 12 kPa. Esta prueba se repitió tres veces, variando la duración del incremento en 5, 20 y 30 segundos, respectivamente

Los resultados del experimento se presentan en las Figuras 4.3, 4.4 y 4.5. Es evidente que el comportamiento de extensión difiera del de compresión, ya que para una misma presión, el ángulo de establecimiento varía. Además, se observó que el tiempo no parece ser un factor determinante, ya que todas las gráficas mostraron un comportamiento muy similar. La única diferencia notable es que la curva correspondiente a un incremento de 30 segundos se mostró más suave, indicando que se le dio al sistema suficiente tiempo para estabilizarse en su valor final.

Un hallazgo adicional es que la relación entre la presión y el ángulo no fue lineal. A partir de aproximadamente 8 kPa, se observó un aumento significativo en el ángulo, tanto en flexión como en extensión. Este fenómeno resaltó la complejidad de la respuesta del sistema y subrayó la necesidad de un control preciso para lograr un movimiento estable y controlado en el exoesqueleto.

4.2 Diseño del controlador difuso

Para implementar el controlador difuso, se utilizó el tipo Mandami, como se ilustra en la Figura 3.16. Para obtener las formas aproximadas de las funciones de membresía necesarias para el controlador, se empleó el algoritmo de *fuzzy - C - means*. La determinación del número de conjuntos difusos se realizó de manera empírica, respaldada por el método de *fuzzy - C - means*. Elegir un número excesivamente alto de conjuntos difusos podría redundar y complicar la formulación de las reglas debido a todas las posibles combinaciones.

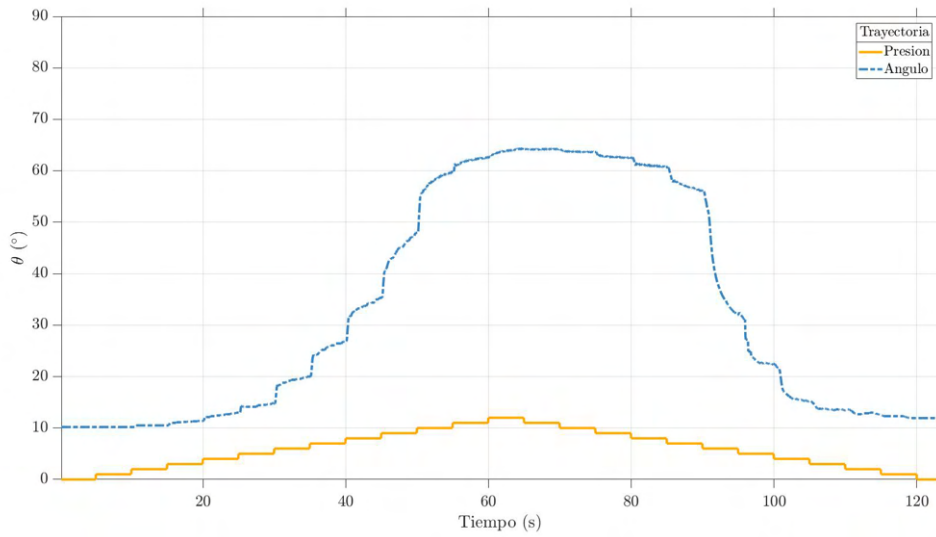


FIGURA 4.3: Respuesta del sistema en lazo abierto frente a incrementos de presión cada 5 segundos. (*Experimento realizado en el banco de pruebas*)

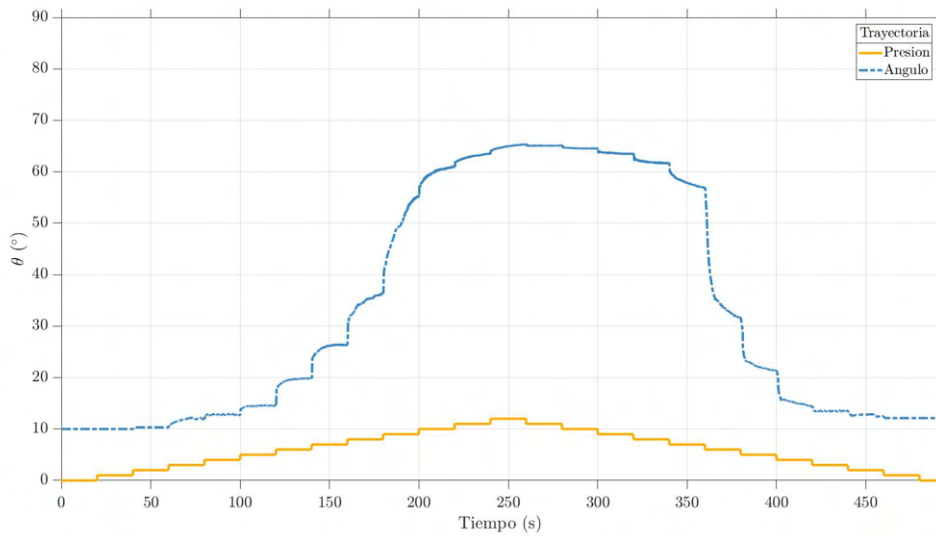


FIGURA 4.4: Respuesta del sistema en lazo abierto frente a incrementos de presión cada 20 segundos. (*Experimento realizado en el banco de pruebas*)

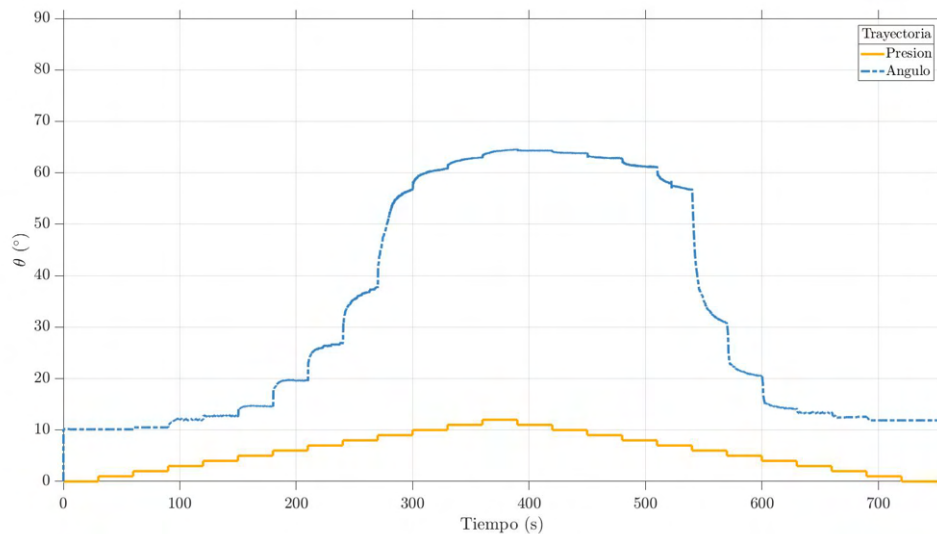


FIGURA 4.5: Respuesta del sistema en lazo abierto frente a incrementos de presión cada 30 segundos. (*Experimento realizado en el banco de pruebas*)

Se llevaron a cabo varias pruebas con diferentes configuraciones, utilizando una de las combinaciones óptimas de ganancias para un controlador PID. Estas pruebas se realizaron con entradas tipo escalón y con la trayectoria deseada para un bebé de seis meses. La elección de esta configuración se basa en la premisa de que, si el controlador puede manejar el caso más extremo (máximas longitudes y pesos), podrá controlar correctamente los casos de meses inferiores. Las ganancias para este controlador son: $K_p = 0.45$, $K_i = 0.02$ y $K_d = 0.002$. La Fig. 4.6 muestra su rendimiento al seguir la trayectoria deseada. Se observó un seguimiento generalmente bueno, con un ángulo máximo de 60.75° . El controlador comenzó a seguir la trayectoria en aproximadamente 5 segundos y su comportamiento fue suave en todo momento, con ligeras oscilaciones en el retorno.

Es relevante notar la ley de control, mostrada en la Fig. 4.7, ya que una ley de control que presente cambios bruscos podría generar problemas en los equipos utilizados. Además, se buscó que la ley de control fuera suave para permitir un seguimiento adecuado por parte del controlador interno del regulador electrónico.

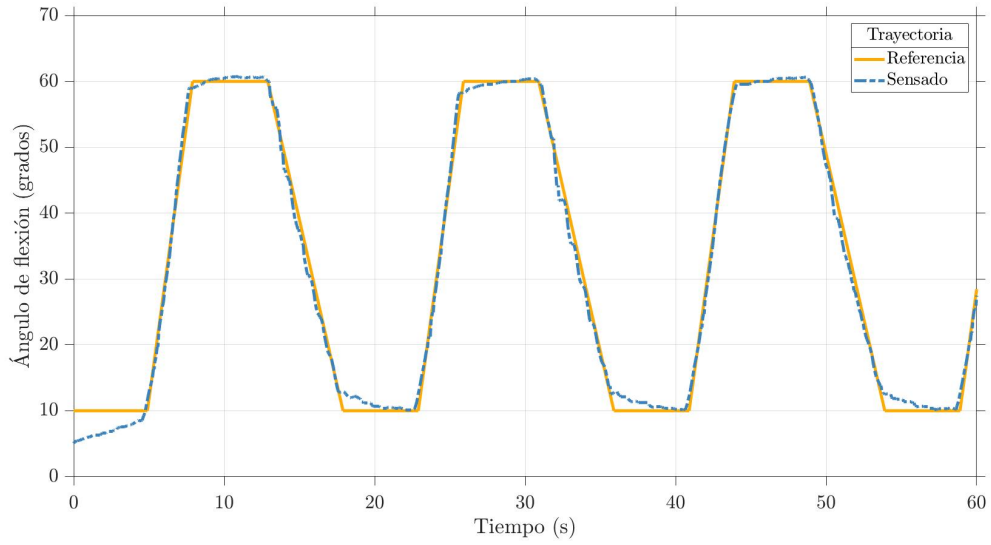


FIGURA 4.6: Controlador PID siguiendo la trayectoria deseada para la rehabilitación con ganancias $K_p = 0.45$, $K_i = 0.02$ y $K_d = 0.002$. (Experimento realizado en el banco de pruebas)

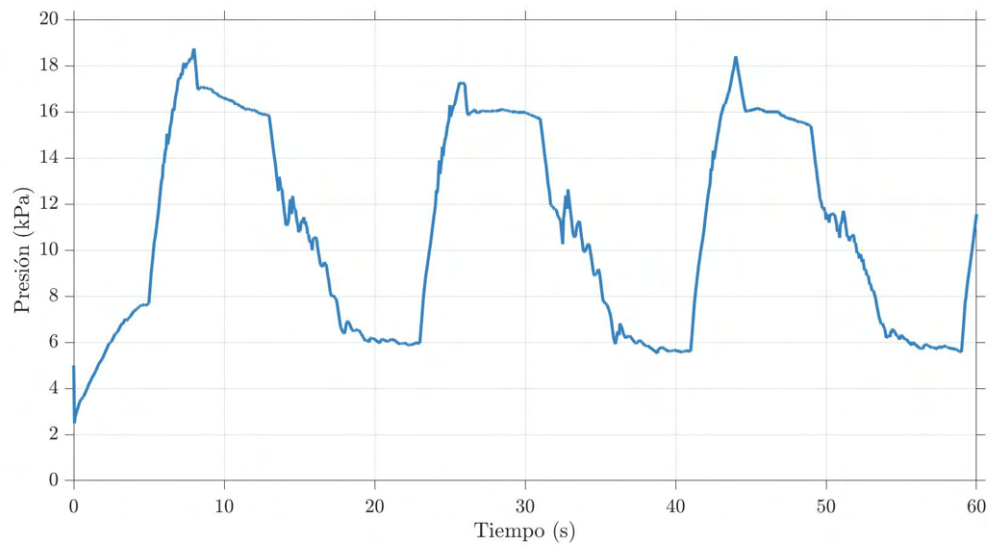


FIGURA 4.7: Ley de control para el seguimiento de la trayectoria deseada para la rehabilitación con ganancias $K_p = 0.45$, $K_i = 0.02$ y $K_d = 0.002$. (Experimento realizado en el banco de pruebas)

Además, se evaluó el controlador para entradas del tipo escalón en un rango desde 10° hasta 60° , con incrementos de 2° para cada prueba individual. Los resultados de estas pruebas se presentan en la Fig. 4.8. Observamos que el sistema tarda aproximadamente 8 segundos en alcanzar y mantener un error muy cercano a cero grados. El mayor *overshoot* registrado es de aproximadamente 6.03% , ocurriendo cuando la referencia es de 60° . Es importante destacar que la presencia de *overshoot* se observó para valores de referencia superiores a 50° .

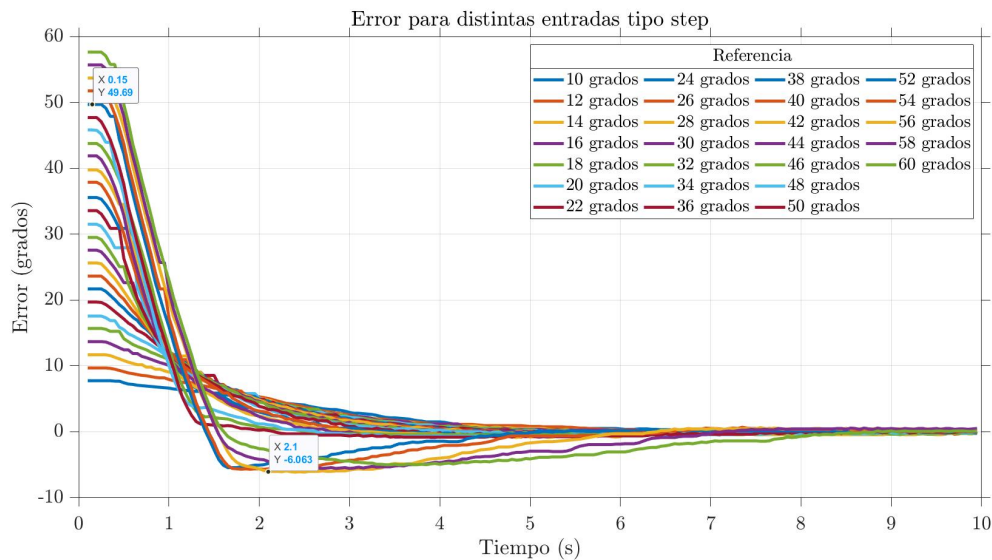


FIGURA 4.8: Error del PID frente a distintas entradas tipo *step* con ganancias $K_p = 0.45$, $K_i = 0.02$ y $K_d = 0.002$. (Experimento realizado en el banco de pruebas)

Todos los datos recopilados se utilizaron para realizar un análisis detenido. Primero, se determinó el universo de discurso para cada variable lingüística. Para este propósito, se utilizaron histogramas para identificar los valores atípicos y definir el rango que contiene los valores más significativos para cada variable. Los histogramas correspondientes al error, la derivada del error y la derivada de presión se presentan en las Fig. 4.9, 4.10 y 4.11, respectivamente. Es importante mencionar que la salida de la estructura de control que se utilizará es la derivada de presión, no la presión directamente.

Al analizar el histograma del error, se observa que la mayoría de los valores se encuentran en el rango de $[-1, 0]^\circ$. Aunque no hay una tendencia clara en la distribución de los datos, se encuentran dentro del rango de $[-8, 10]^\circ$. Por otro lado, la derivada del error mostró un comportamiento disperso, con una concentración de valores en el rango de $[0, 2] \frac{^\circ}{seg}$. Sin embargo, para el diseño, se consideró el rango de $[-30, 30] \frac{^\circ}{seg}$, ya que, aunque hay pocos valores más allá de los $\pm 20 \frac{^\circ}{seg}$, estos podrían ser útiles en situaciones donde la referencia es alta.

Al analizar el histograma de la derivada de presión, se observó un comportamiento decreciente con una mayor concentración en el rango de $[-0.05, 0] \frac{kPa}{seg}$. Sin embargo, todo el rango presenta valores significativos, por lo que se tomó el rango de $[-0.5, 0.7] \frac{kPa}{seg}$ para el diseño del controlador.

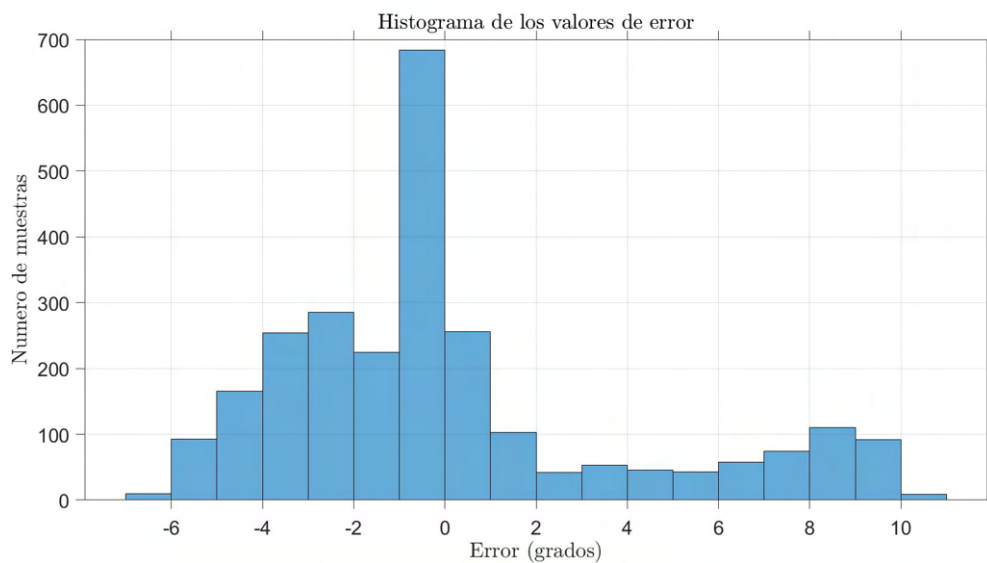


FIGURA 4.9: Histograma de valores de error durante las pruebas.

Para implementar el algoritmo de *fuzzy - C - means*, se utilizó la función *fcm* de Matlab, que realiza todos los cálculos teniendo en cuenta el número de conjuntos difusos definidos por el usuario y el grado de pertenencia entre conjuntos. En este caso, se estableció un valor de pertenencia entre conjuntos igual a 2, excepto para la derivada del



FIGURA 4.10: Histograma de valores de la derivada de error durante las pruebas.

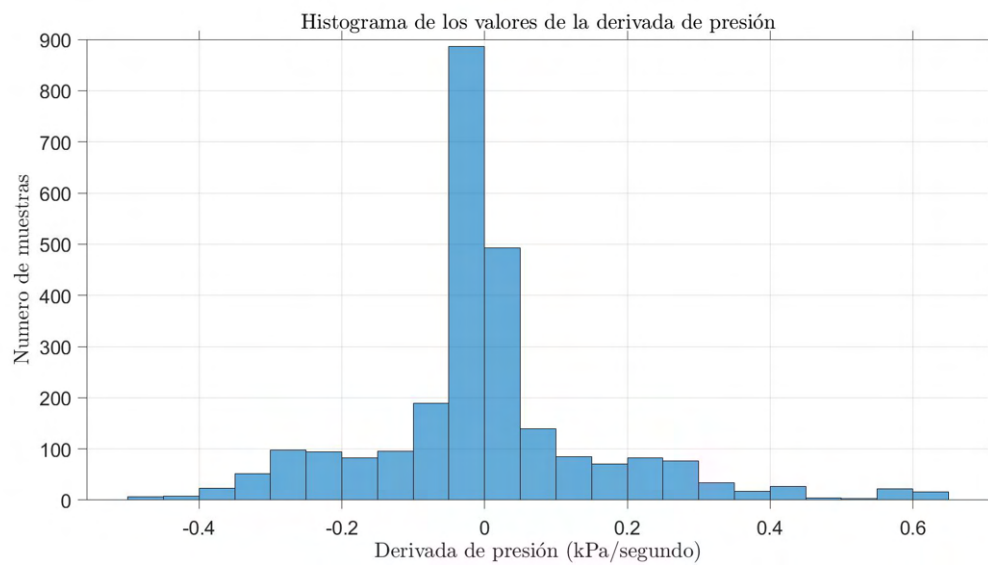


FIGURA 4.11: Histograma de valores de la derivada de presión durante las pruebas.

error, donde se utilizó un valor de 1.8 debido a que la forma de la función de membresía no se aproximó a alguna conocida. Se limitó el número de iteraciones a un máximo de 100 y se utilizó 5 conjuntos difusos para todos los casos.

Es relevante mencionar que considerar 7 conjuntos difusos podría llevar a la existencia de redundancias entre ellos. Un ejemplo de esto se puede observar en la Fig. 4.12, donde los conjuntos $C6$, $C7$ y $C4$ abarcan un rango de valores muy similar, y sus centros están muy cercanos entre sí. En este escenario, sería preferible combinar estos conjuntos para cubrir un rango mayor y evitar redundancias innecesarias en la representación del sistema.

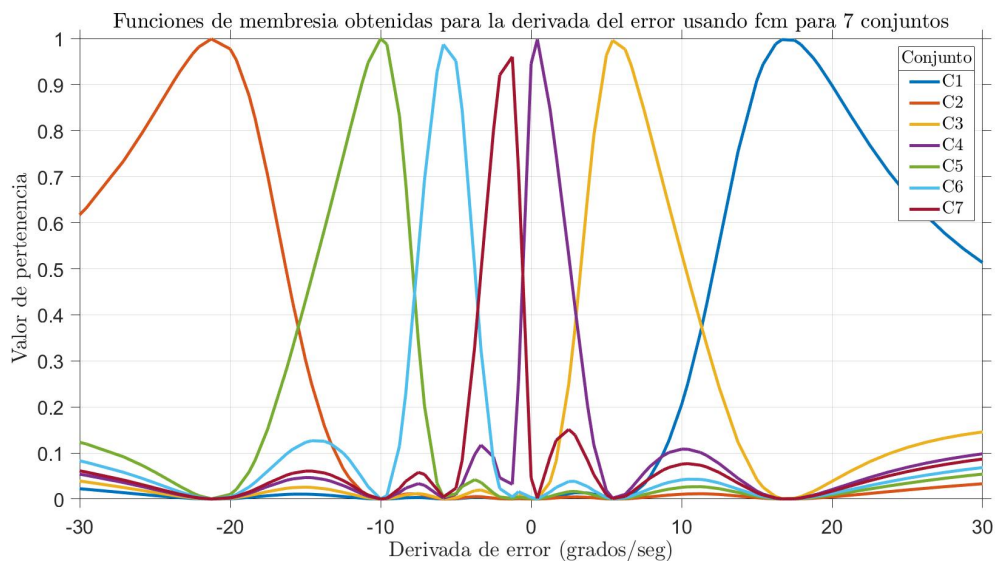


FIGURA 4.12: Existencia de redundancia al utilizar 7 conjuntos difusos para la derivada del error.

Las funciones de membresía obtenidas utilizando el algoritmo de *fuzzy - C - means* para el error, la derivada del error y la derivada de presión se presentan en las Fig. 4.13, 4.14 y 4.15, respectivamente. Se observó que todas estas funciones tuvieron un comportamiento similar al de las gaussianas de dos lados, ya que no fueron simétricas con respecto a su centro. Además, los extremos de las funciones de membresía en todos los

casos tuvieron un lado que se extiende hacia el infinito, a diferencia de los otros conjuntos difusos.

Debido a que las funciones de membresía tuvieron un comportamiento casi simétrico con respecto a cero en cada variable lingüística, se optó por utilizar una convención uniforme para nombrar los conjuntos difusos. Estos se denominaron: Positivo Grande (*PG*), Positivo Pequeño (*PP*), Cero (*Ze*), Negativo Pequeño (*NP*) y Negativo Grande (*NG*). Esta elección se hizo por simplicidad y para facilitar el diseño de las reglas del sistema de control difuso.

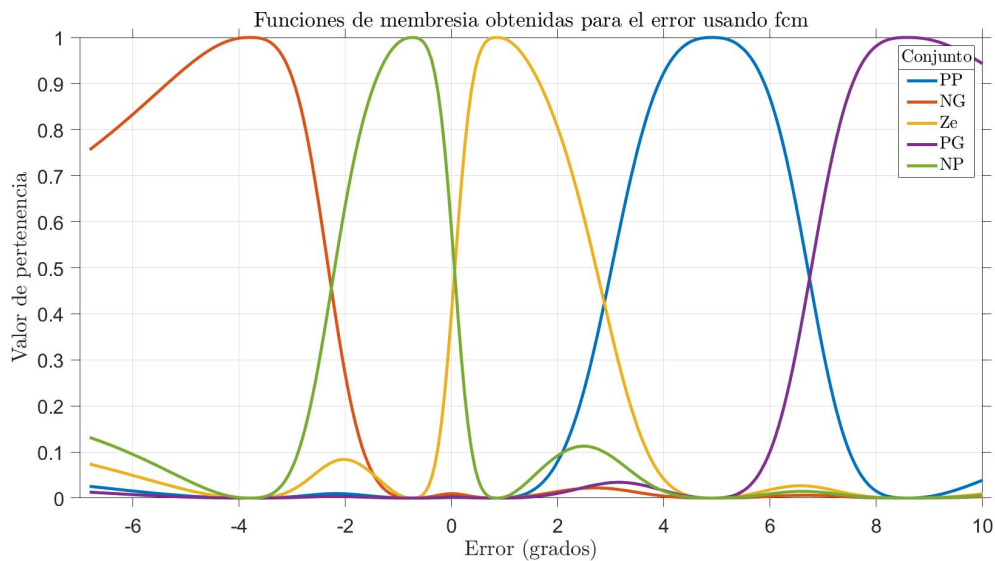


FIGURA 4.13: Forma de las funciones de membresía obtenidas para el error utilizando *fcm*.

Como se mencionó anteriormente, para la implementación del controlador difuso en Simulink se utilizó el toolbox de Matlab llamado *Fuzzy – Logic – Designer*. Sin embargo, dentro de esta herramienta, es necesario ingresar los parámetros que describen la función de membresía de cada conjunto. En el caso de utilizar una gaussiana de dos lados, estos parámetros incluyen el centro de la función, σ_1 y σ_2 . El objetivo fue encontrar los mejores valores de σ que aproximen cada función de membresía (*MF*), dado que la

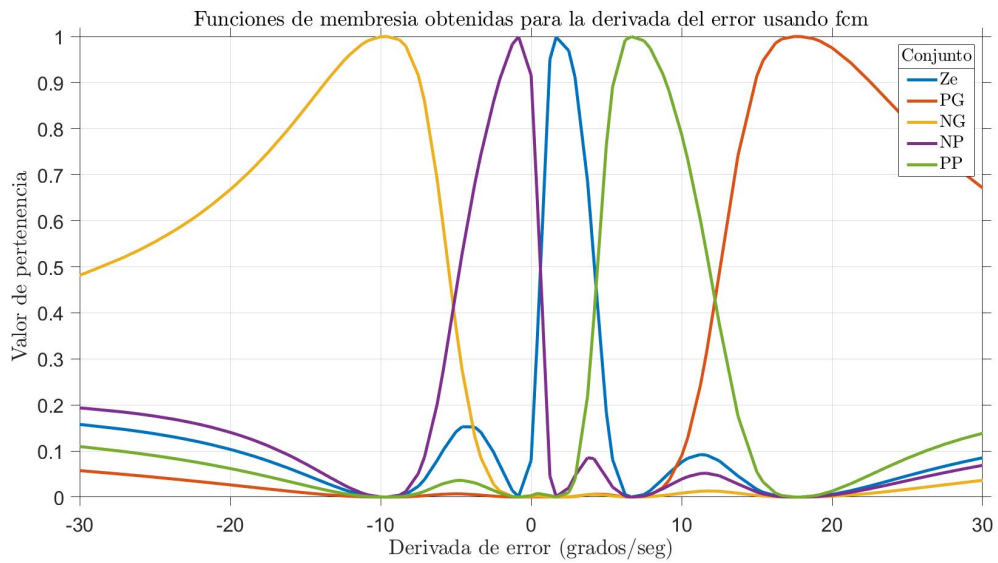


FIGURA 4.14: Forma de las funciones de membresía obtenidas para la derivada del error utilizando *fcm*.

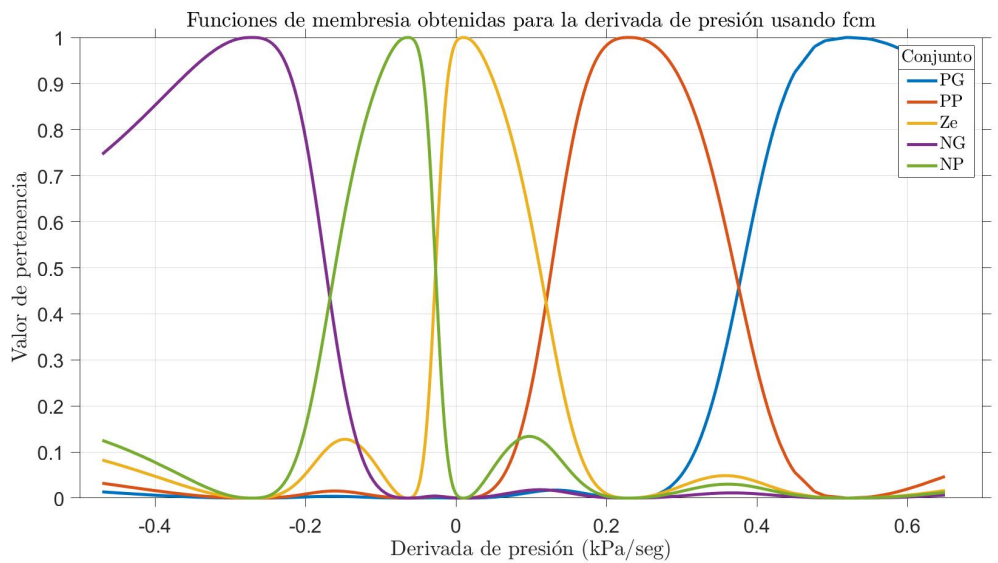


FIGURA 4.15: Forma de las funciones de membresía obtenidas para la derivada de la presión utilizando *fcm*.

función f_{cm} proporcionó el centro de cada MF como dato conocido. Por lo tanto, la única variable desconocida fue σ .

Además de los centros de cada MF , la función f_{cm} también nos proporcionó el valor de pertenencia de cada punto con respecto a cada MF . Conociendo la fórmula matemática que describía una gaussiana de dos lados (como se mostró en la ecuación 4.3), se realizó una aproximación utilizando estos datos proporcionados.

$$f_1(x) = \begin{cases} \text{si } x \leq m_1 & exp[-\frac{1}{2}(\frac{x-m_1}{\sigma_1})^2] \\ \text{Cualquier otro caso} & 1 \end{cases} \quad (4.1)$$

$$f_2(x) = \begin{cases} \text{si } x \geq m_1 & exp[-\frac{1}{2}(\frac{x-m_2}{\sigma_2})^2] \\ \text{Cualquier otro caso} & 1 \end{cases} \quad (4.2)$$

$$\mu(x) = f_1(x) * f_2(x) \quad (4.3)$$

Donde: x es la variable de entrada, m_1 el centro de la gaussiana izquierda, m_2 el centro de la gaussiana derecha, σ_1 la varianza de la gaussiana izquierda, σ_2 la varianza de la gaussiana derecha y μ la función de membresía final.

Para ilustrar el proceso de aproximación, la Fig. 4.16 muestra un ejemplo de variación de los parámetros σ , buscando la mejor combinación para que se asemejen lo máximo posible a la función original (línea azul). Aunque este método puede aplicarse manualmente para cada conjunto difuso de cada variable lingüística, es tedioso y no garantiza precisión completa.

Para automatizar y optimizar este proceso, se propuso utilizar un método de optimización que reduzca la función de costo definida por el error cuadrático medio entre la función original y la aproximada (ecuación 4.4). Los únicos parámetros variables en este

proceso fueron σ_1 y σ_2 . Se utilizó la función de Matlab llamada *fmincon* para minimizar esta función de costo sin restricciones. Los valores iniciales para σ_1 y σ_2 fueron ambos 0.5, y se emplearon las opciones predeterminadas de *fmincon*. Estas incluyeron: el algoritmo *interior – point*, una tolerancia de $1e - 6$ y un número máximo de iteraciones de 1000.

La Fig. 4.17 presenta un ejemplo de aproximación utilizando este método, donde la línea de color rojo representa la función aproximada que busca parecerse a la línea azul, que representa la salida original o deseada. Es importante destacar que el error de aproximación nunca será 0, ya que en los datos originales se observan pequeñas oscilaciones fuera del rango de la campana, las cuales son imposibles de moderar únicamente con una función de gaussiana.

$$J(x) = \frac{(\mu(x) - Fcm(x))^2}{n} \quad (4.4)$$

Donde: $J(x)$ es el valor de la función de costo, x es la variable de entrada, μ es la salida de la gaussiana de dos lados, Fcm la salida de la función de Matlab y n el número de datos totales utilizados.

En las Tablas 4.1, 4.2 y 4.3, se presentan los valores optimizados de σ , junto con el costo inicial y final, que permitieron estimar adecuadamente las funciones de membresía (*MF*) de nuestras variables lingüísticas. Los resultados de las estimaciones para el error, la derivada del error y la derivada de la presión se visualizan en las Fig. 4.18, 4.19 y 4.20, respectivamente.

Es notable que las *MF* de la derivada del error son las que mejor se lograron predecir, evidenciado por la proximidad entre las líneas azul (original) y roja (aproximada) en cada caso. Un indicador numérico de la calidad de la optimización es observar el costo inicial y final obtenido, que se presentan en cada tabla respectiva. Se observó que el costo

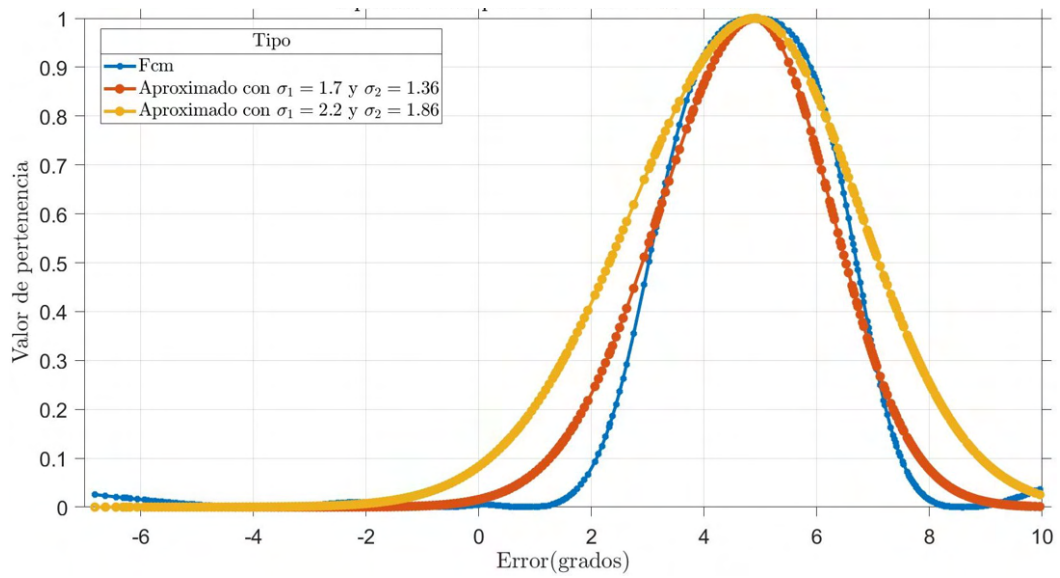


FIGURA 4.16: Efecto de los parámetros σ .

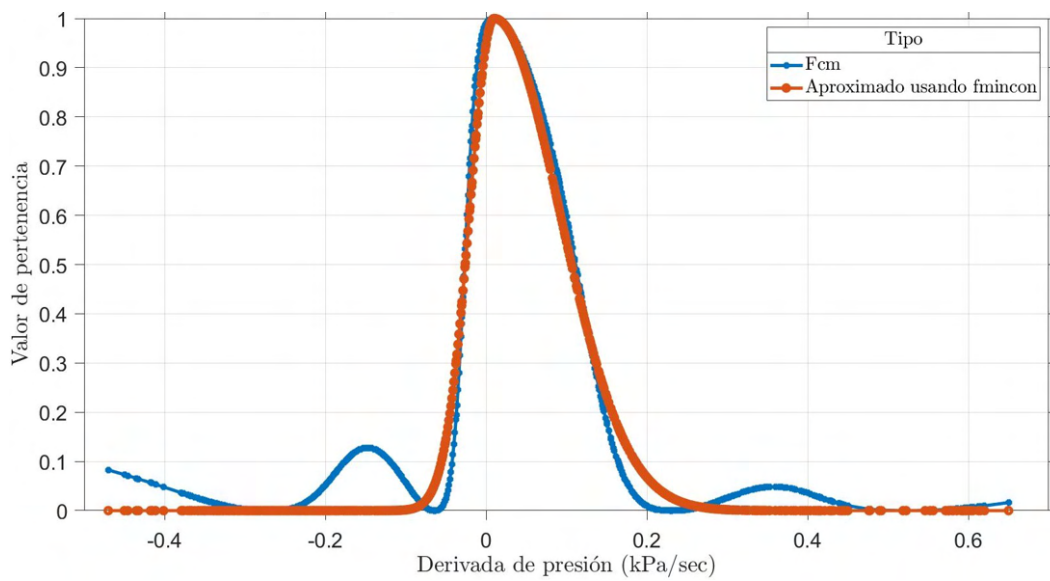


FIGURA 4.17: Aproximación con los valores de σ estimados por *fmincon*.

MF (error)	Centro	σ_1	σ_2	Costo Inicial	Costo Final
PP	4.906	1.4884	1.3999	0.1674	0.0315
NG	-3.7903	3.5509	1.1857	0.3141	0.0439
Ze	0.8513	0.5811	1.4893	0.1377	0.0534
PG	8.5791	1.4521	4.1977	0.1934	0.0247
NP	-0.732	1.1527	0.6765	0.2155	0.0636

TABLA 4.1: Valores optimizados de σ_1 y σ_2 para las *MF* del error.

MF (derivada del error)	Centro	σ_1	σ_2	Costo Inicial	Costo Final
Ze	1.8229	0.8691	1.9145	0.2448	0.0698
PG	17.6582	3.9004	13.1242	0.1615	0.0148
NG	-9.6437	12.8328	3.1952	0.2492	0.0329
NP	-0.6862	3.2345	1.1013	0.4199	0.0796
PP	6.7938	1.8619	4.0346	0.2163	0.0349

TABLA 4.2: Valores optimizados de σ_1 y σ_2 para las *MF* de la derivada del error.

MF (derivada de la presión)	Centro	σ_1	σ_2	Costo Inicial	Costo Final
PG	0.5222	0.1074	100.3321	0.5675	0.0210
PP	0.2298	0.0783	0.1153	0.7845	0.0332
Ze	0.01	0.0305	0.0819	0.6093	0.0690
NG	-0.271	99.8976	0.0774	0.7633	0.0349
NP	-0.0638	0.0766	0.028	0.7754	0.0714

TABLA 4.3: Valores optimizados de σ_1 y σ_2 para las *MF* de la derivada de presión.

final es significativamente menor que el costo inicial en todos los casos. Numéricamente, el error es la variable que pudo aproximarse de mejor manera, dado su costo inicial considerablemente mayor en comparación con las otras variables.

Es importante señalar que, aunque el error mostró una mayor diferencia entre la estimación y la realidad en las visualizaciones gráficas, esta discrepancia puede mitigarse explorando otros tipos de *MF*, como una sigmoideal de dos lados. Sin embargo, el aspecto más crucial fue capturar la mejor forma posible variando los parámetros σ_1 y σ_2 .

El último paso en el proceso implicaron definir las reglas difusas que determinaron

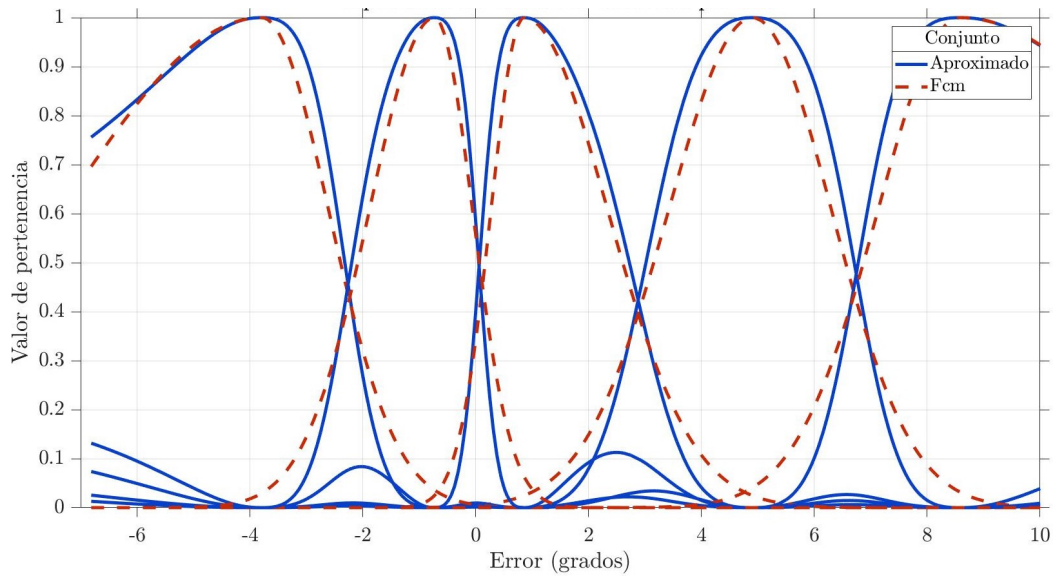


FIGURA 4.18: Aproximación de las MF con los parámetros optimizados para el error.

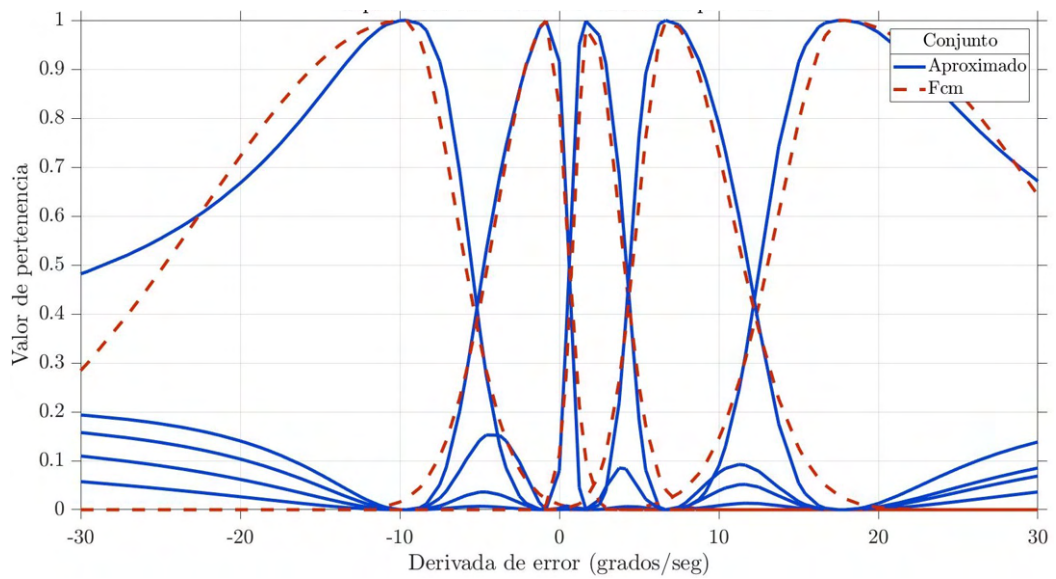


FIGURA 4.19: Aproximación de las MF con los parámetros optimizados para la derivada del error.

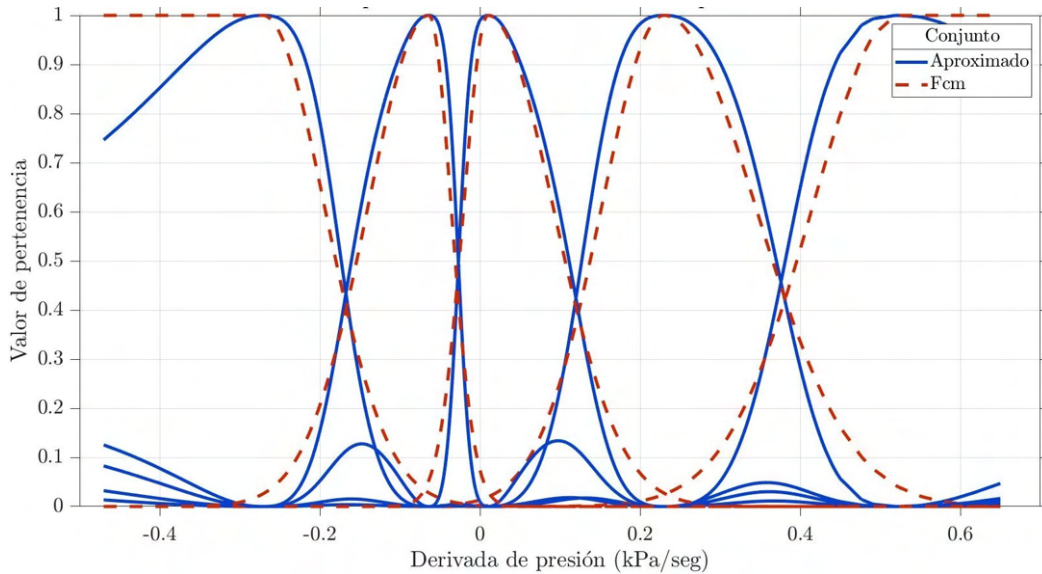


FIGURA 4.20: Aproximación de las MF con los parámetros optimizados para la derivada de la presión.

el resultado del controlador en función de las condiciones de entrada y salida durante la fusión y defusión. Estas reglas fueron establecidas empíricamente, basándose en la experiencia adquirida durante diversas pruebas realizadas en el actuador. Cada caso fue minuciosamente analizado para determinar la mejor salida o identificar la opción más común observada en las pruebas. Este análisis detenido llevó a la creación de la tabla presentada en la Tabla 4.4.

Cabe destacar que estas reglas siguen la convención previamente definida: Positivo Grande (PG), Positivo Pequeño (PP), Cero (Ze), Negativo Pequeño (NP) y Negativo Grande (NG). La aplicación de estas reglas difusas se convirtió en el último paso crucial para la implementación del controlador, ya que permitieron traducir las condiciones del sistema en acciones precisas y efectivas.

Finalmente, la Tabla 4.5 presenta un resumen completo de todas las características importantes de nuestro controlador diseñado. Estas características fueron esenciales y se utilizaron como parámetros en el toolbox de Matlab para implementar el controlador en

e \ de	NG	NP	Ze	PP	PG
NG	NG	NG	NG	NP	Ze
NP	NG	NP	NP	Ze	PP
Ze	NG	NP	Ze	PP	PG
PP	NP	Ze	PP	PP	PG
PG	Ze	PP	PG	PG	PG

TABLA 4.4: Reglas difusas definidas para el controlador.

el sistema. La tabla proporcionó una visión clara y detallada de los elementos clave que componen nuestro controlador difuso.

Adicionalmente, la Fig. 4.21 muestra la superficie difusa equivalente generada por el controlador. Como se pudo observar, esta superficie exhibe un comportamiento similar al de una sigmoïdal en tres dimensiones. Se mantiene constante en los extremos y presenta un aumento exponencial pero suave en todo momento. La forma específica de esta superficie está directamente influenciada por las reglas que se han utilizado en su diseño, representando visualmente el comportamiento del controlador en diferentes situaciones.

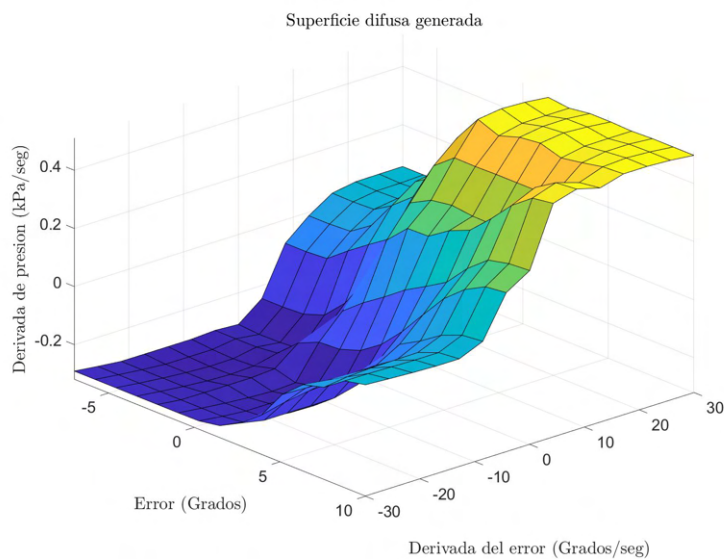


FIGURA 4.21: Superficie difusa generada.

Característica	Valor
Tipo de control difuso	Mandami
Método de defusificación	Centroide
Método <i>and</i>	min
Método <i>or</i>	max
Implicación	min
Agregación	max
Universo de discurso del error	$[-7, 10]^\circ$
Universo de discurso de la derivada del error	$[-30, 30]^\circ/\text{seg}$
Universo de discurso de la derivada de presión	$[-0.47, 0.65]\text{kPa}/\text{seg}$
# de conjuntos difusos para el error	5
# de conjuntos difusos para la derivada del error	5
# de conjuntos difusos para la derivada de presión	5
Forma de las funciones de membresía	Gaussiana de dos lados
Ganancia K_p	1

TABLA 4.5: Resumen de las características del controlador diseñado.

4.3 Control en el modelo de pierna simplificado para 6 meses

Cada una de las pruebas del controlador se llevó a cabo utilizando el modelo de pierna simplificado. A diferencia del maniquí, este modelo puede modificarse para simular las condiciones de un bebé de cero, tres o seis meses, según sea necesario para las pruebas específicas. Para evaluar el desempeño del controlador, se configuró inicialmente el modelo de pierna para simular las condiciones de un bebé de seis meses.

En estas pruebas, la referencia utilizada fue la función trapezoidal descrita en la sección anterior. Esta función cumplió con los requisitos establecidos, con una amplitud de 50° y una periodicidad que implica 3 segundos de tiempo de subida y 5 segundos de tiempo de retorno, con breves pausas entre cada movimiento.

Los resultados obtenidos del controlador difuso se presentan en la Fig. 4.22. En esta figura, la línea amarilla representa la referencia o el comportamiento deseado, mientras que la línea azul muestra la salida real del sistema. Observamos que la señal controlada logra emular la forma deseada con un ligero desfase. Durante la subida, se observa un

comportamiento ligeramente exponencial en lugar de seguir exactamente la referencia lineal. Sin embargo, este cambio es suave y durante el retorno, se notan pequeñas caídas de ángulo en ciertos puntos de la trayectoria, pero el sistema logra recuperarse y seguir la referencia de manera adecuada.

Es interesante destacar que en el primer ciclo de movimiento, se observó el mayor pico de *overshoot*, pero después de este pico, el ángulo máximo alcanzado se aproxima más al valor deseado, con un error de aproximadamente 1.2° . A pesar de estas pequeñas variaciones, el controlador demostró un rendimiento adecuado. Cumplió con el requisito de llegar al valor máximo en 3 segundos, mantenerse durante 5 segundos en los extremos y volver a la posición inicial en otros 5 segundos. Además, el movimiento fue suave en todo momento.

El valor máximo de flexión alcanzado fue de 62.4375° , que estuvo dentro del rango permitido por el médico (máximo de 64°). En general, a pesar de las ligeras discrepancias, el controlador logró mantener el movimiento en línea con las especificaciones requeridas y demostró una respuesta satisfactoria en términos de suavidad y precisión.

Además, es crucial tener en cuenta los primeros segundos después de que el control entra en funcionamiento, ya que se espera que esta transición sea suave y controlada, sin movimientos bruscos al inicio. En el caso del control difuso implementado, esta transición tomó aproximadamente 5 segundos para completarse y alcanzar su estado final, el cual se mantuvo constante a lo largo de todo el experimento.

Por otro lado, es fundamental analizar cuidadosamente nuestra ley de control, ya que una ley inadecuada podría ocasionar un desgaste acelerado en los equipos. La Fig. 4.23 muestra nuestra ley de control diseñada para lograr el comportamiento que se muestra en la Fig. 4.22. Esta ley de control se caracterizó por su suavidad y continuidad, evitando cambios abruptos en cualquier momento. Además, se mantuvo dentro del rango de trabajo establecido (0 - 30 kPa), eliminando la necesidad de usar un saturador de seguridad.

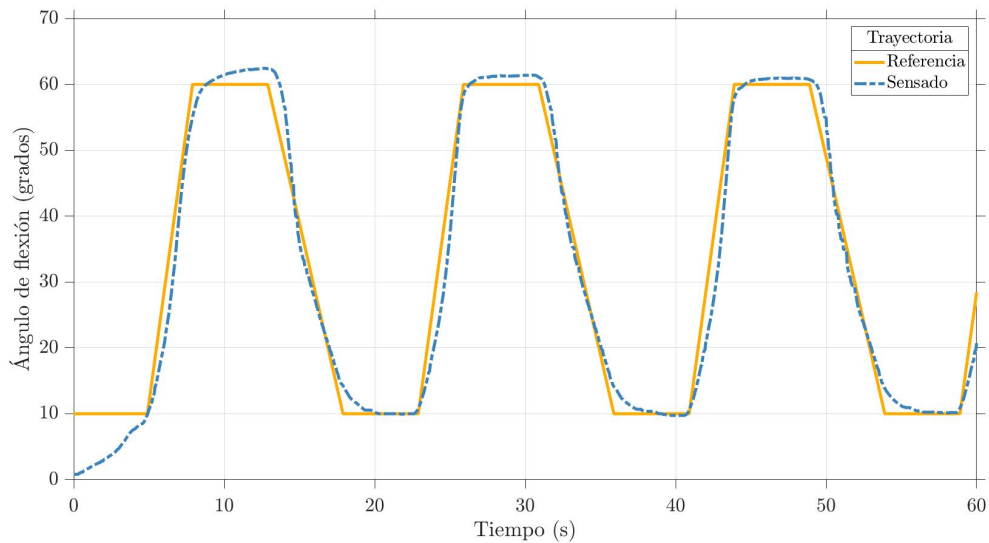


FIGURA 4.22: Respuesta del sistema controlado junto con la referencia utilizada. (*Experimento realizado en el banco de pruebas*)

La forma de la ley de control se asemejó a una trapezoidal. Durante la subida, presentó un comportamiento lineal, mientras que en el retorno, adquirió una dinámica diferente, mostrando una oscilación cercana al punto medio de la trayectoria. Este fenómeno pudo atribuirse a la histéresis del actuador y al hecho de que el actuador solo puede funcionar a presión negativa. Por lo tanto, regresar a su estado inicial se vuelve complicado, ya que está directamente influenciado por el peso de la pierna.

Es interesante observar que el primer ciclo de movimiento fue diferente de los subsiguientes. La principal disparidad radicó en que este ciclo requiere un máximo de 23 kPa, mientras que los ciclos posteriores necesitaron un máximo de 20.7 kPa.

No obstante, lo que se muestra en la Fig. 4.23 representa la presión ideal necesaria en cada momento según el controlador. Fue fundamental contrastar esta presión ideal con la presión real del sistema en cada instante de tiempo. Esta comparación se ilustra en la Fig. 4.24, donde se presentan simultáneamente la presión ideal generada por el controlador (línea negra), la presión del regulador (línea naranja) y la presión que entra al actuador

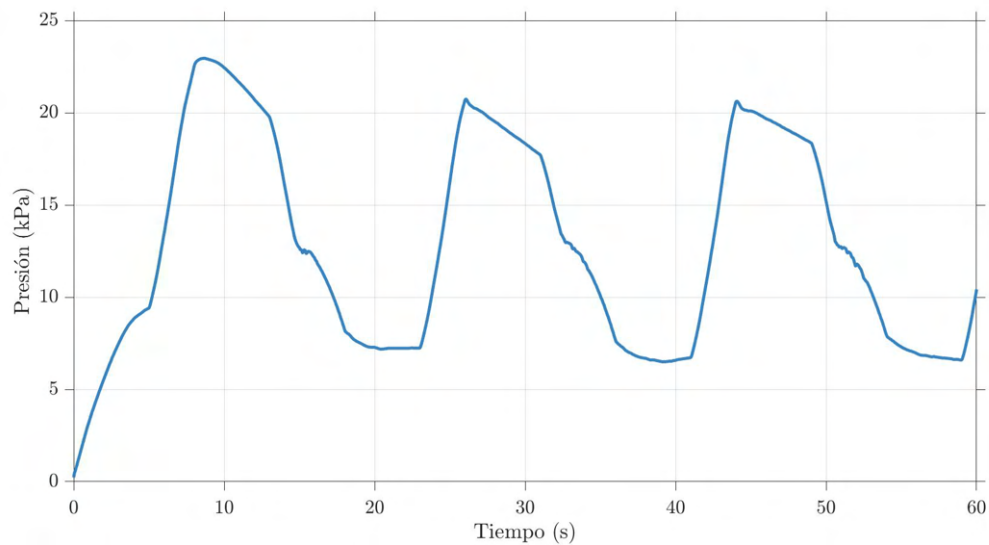


FIGURA 4.23: Comportamiento de la ley de control durante el experimento. (*Experimento realizado en el banco de pruebas*)

(línea azul). Al visualizar estas tres señales simultáneamente, pudimos verificar que el regulador está generando la presión requerida por el controlador. Además, se observó la pérdida de presión en la entrada al actuador; sin embargo, el controlador compensó esta pérdida aumentando la presión necesaria de manera adecuada.

4.4 Prueba de control frente a incertezas antropométricas

Otro requisito fundamental del controlador fue su capacidad para operar en bebés dentro del rango de edad de cero a seis meses, lo que implicó enfrentar un ambiente altamente incierto. Dada la variabilidad en el peso de las piernas de los bebés, que osciló entre aproximadamente 212 g y 363 g, y las diferencias en las longitudes que eran desde aproximadamente 93 mm hasta 117 mm, el controlador debía ser capaz de mantener su funcionamiento óptimo a pesar de estas fluctuaciones.

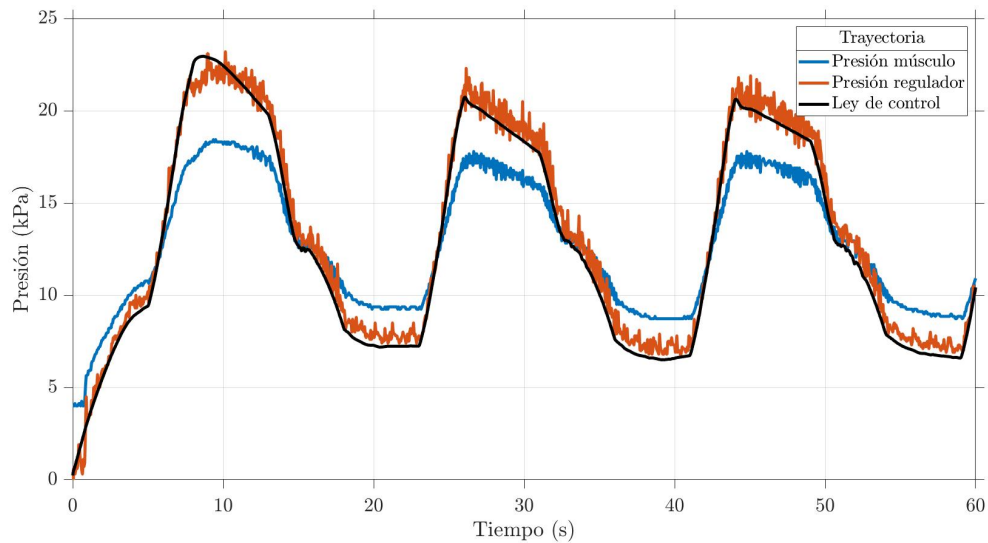


FIGURA 4.24: Comportamiento de la presión real en el sistema durante el experimento.
(Experimento realizado en el banco de pruebas)

Para esta prueba se utilizó los maniquís de cero y seis meses para simular las piernas de un bebé de entre cero y seis meses, respectivamente. Los resultados de estos experimentos se presentan en la Fig. 4.25. Es crucial señalar que todas las pruebas se realizaron utilizando la misma referencia trapezoidal, sin realizar ninguna modificación en los parámetros del controlador diseñado. La notable similitud en los resultados de ambos experimentos demostró la capacidad del controlador para funcionar correctamente ante variaciones en los pesos y longitudes de las piernas simuladas. Tres características principales se destacaron en estos experimentos:

- En el caso de seis meses, se necesitaron aproximadamente 2.9 segundos para la flexión y 5.4 segundos para la extensión. Mientras tanto, en el caso de cero meses, se requirieron 3.2 y 6 segundos para llevar a cabo la flexión y extensión, respectivamente, manteniendo un movimiento fluido en todo momento.

- Respecto al tiempo de respuesta, el caso de seis meses tomó 5 segundos para comenzar a seguir la referencia, mientras que el caso de cero meses solo necesitó 2.3 segundos.
- El caso de cero meses mostró un error estacionario más pequeño en la parte superior de la referencia (aproximadamente 1.63° en comparación con 0.6° en el caso de seis meses). Sin embargo, el caso de cero meses experimentó un overshoot del 10 % en la etapa estacionaria inferior de la referencia, a diferencia del caso de seis meses, que logró evitar el overshoot.
- Ambos casos se estabilizaron después de 0.3 segundos, alcanzando un error muy cercano a cero (0.1°) en ese momento.

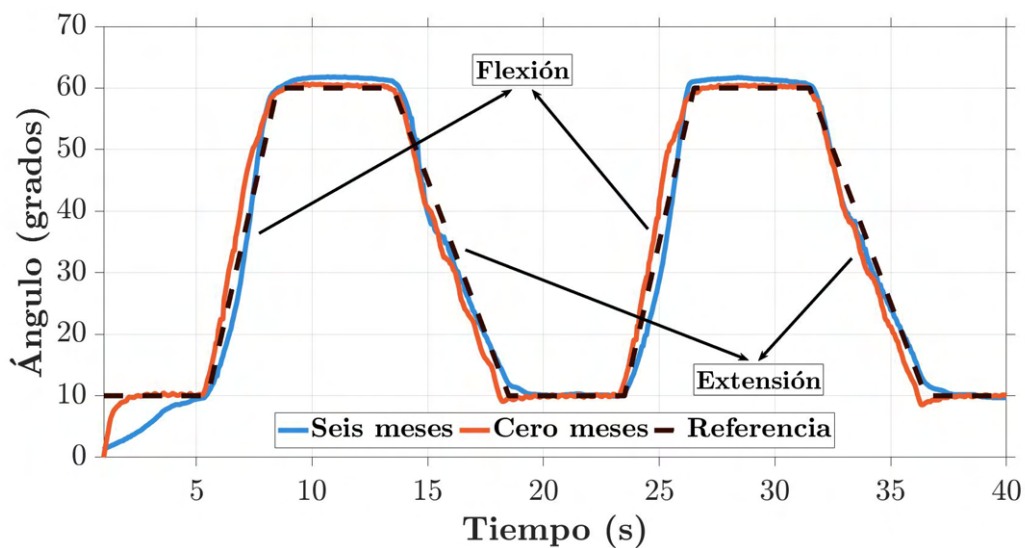


FIGURA 4.25: Comparación entre la respuesta del controlador frente a distintas condiciones antropométricas. (*Experimento realizado en el maniquí*)

De manera similar, la ley de control generada para cada caso se presenta en la Fig. 4.26. Es crucial destacar que esta ley de control siempre exhibió un comportamiento suave y constante, adaptado al rango de trabajo del regulador electrónico y evitando la saturación. En los casos de cero y seis meses, se utilizaron valores de presión máxima

de -16.99 y -22 kPa respectivamente. Además, ambas señales de control mostraron un comportamiento sorprendentemente similar y coherente, como era de esperar, dado que operaban bajo la misma señal de referencia, mostrando la capacidad del controlador para trabajar en dos configuraciones distintas de la planta.

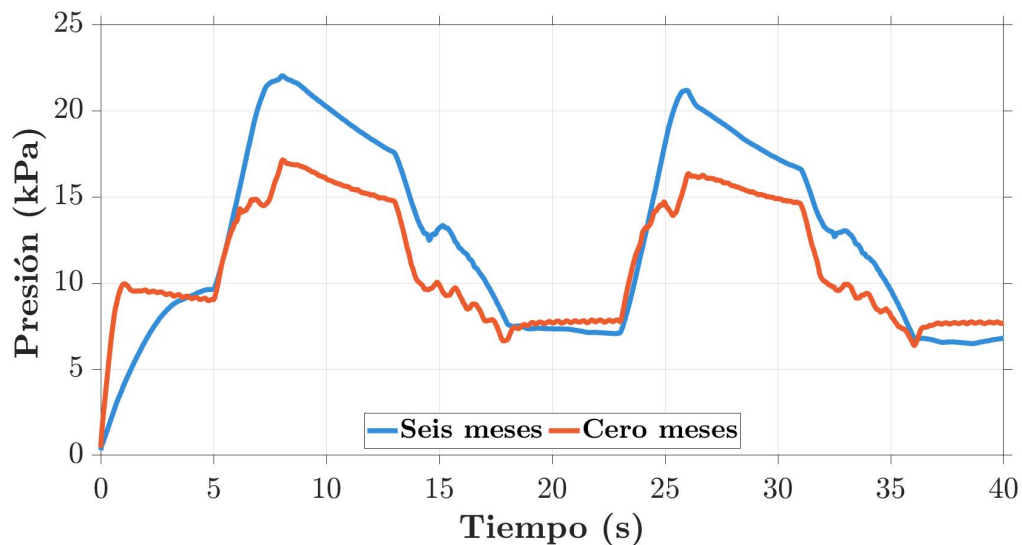


FIGURA 4.26: Comparación de la ley de control para cada variación antropométrica. *(Experimento realizado en el maniquí)*

Se realizó un análisis exhaustivo del desempeño del controlador difuso frente a variaciones antropométricas, y el mismo análisis se llevó a cabo para evaluar el controlador PID y determinar su capacidad para manejar el sistema en las mismas condiciones. Los resultados se presentan en la Fig. 4.27. En el caso del controlador PID, se utilizó la misma ganancia para cada caso ($K_p = 0.4$, $K_i = 0.02$ y $K_d = 0.002$). Los resultados indicaron que el PID no pudo controlar efectivamente los casos en diferentes meses usando las mismas ganancias, mostrando oscilaciones significativas a lo largo de la trayectoria y señalando un riesgo potencial para el bebé.

La principal disparidad se observó en el movimiento de extensión y durante los

primeros segundos de la flexión-extensión de la rodilla. Se notó que la extensión era particularmente propensa a oscilaciones ante pequeñas variaciones en el sistema, ya que diferencias de tan solo 151 g y 24 mm (diferencia entre pesos y longitudes entre las piernas de los maniquís) fueron suficientes para generar oscilaciones. Este comportamiento se atribuye a la naturaleza no lineal del actuador flexible utilizado, donde las ganancias fijas del PID tuvieron un impacto significativo en su dinámica y comportamiento.

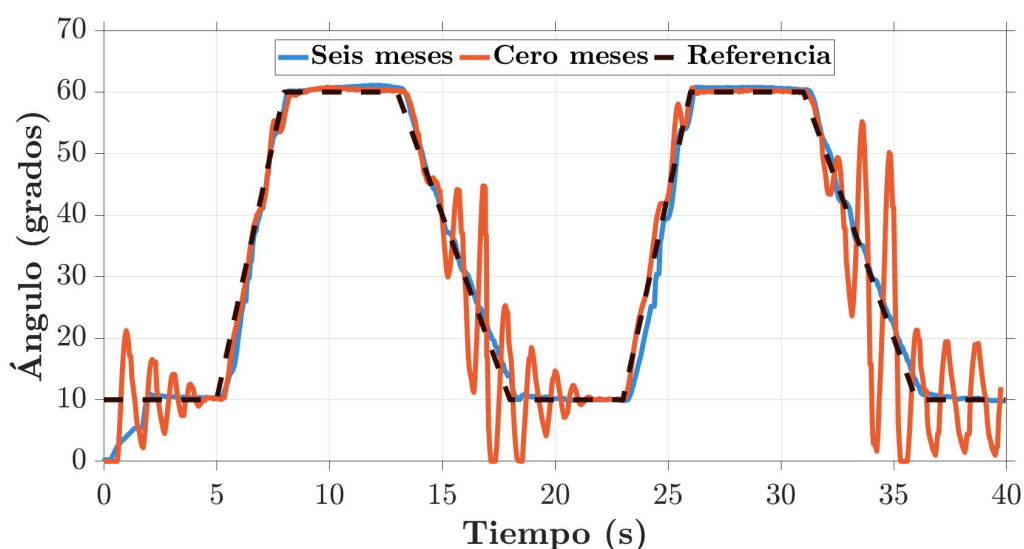


FIGURA 4.27: Desempeño del controlador PID frente a variaciones antropomórficas utilizando una misma ganancia. (*Experimento realizado en el maniquí*)

La ley de control generada por el controlador PID parecía ser sorprendentemente similar en ambos casos, como se puede observar en la Fig. 4.28. No obstante, se notó una diferencia significativa en los resultados obtenidos. En el caso de cero meses, se observó un comportamiento oscilatorio con una presión máxima de -20 kPa. Por otro lado, el caso de seis meses mostró una presión muy diferente en comparación con el controlador difuso (ver Fig. 4.26). Mientras el controlador difuso requirió -22 kPa para alcanzar el ángulo máximo, el PID necesitó -17.57 kPa para lograr el mismo resultado

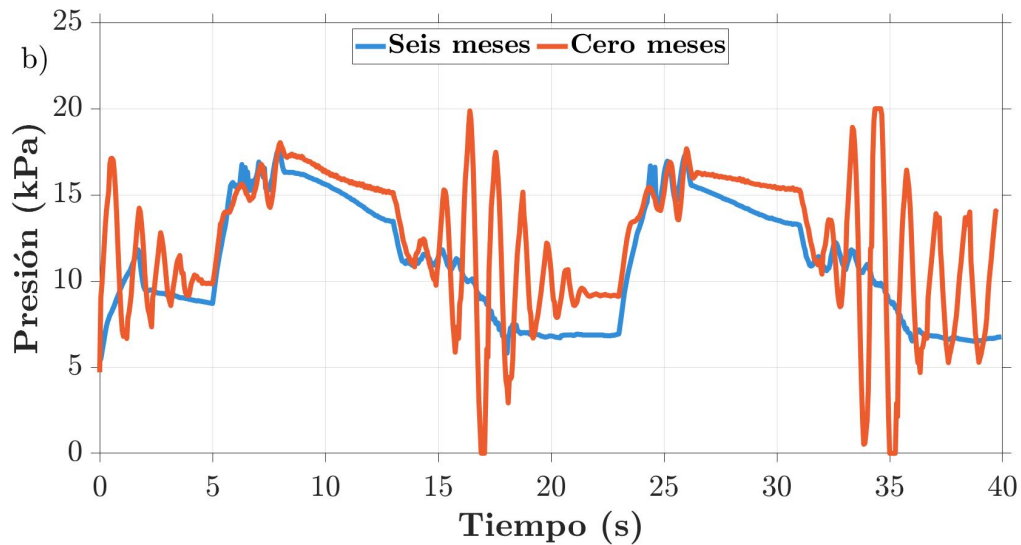


FIGURA 4.28: Ley de control del controlador PID frente a variaciones antropomórficas, utilizando una misma ganancia. (*Experimento realizado en el maniquí*)

4.5 Comparación del método de control diseñado frente a otras variantes

En esta sección, se evaluaron tres variantes adicionales del controlador diseñado, así como el efecto de la ganancia K_p (ver Fig. 2.9) en cada una de estas variantes. La primera variante, denominada $Fuzzy_2$, se diferenció de la versión original en sus funciones de membresía (MF). Estas funciones fueron modificadas a discreción, ajustando los centros y las varianzas siguiendo un criterio empírico para definir los rangos que debería contener cada conjunto difuso. Las funciones de membresía para el error, la derivada del error y la derivada de presión se pueden observar en las Fig. 4.29, 4.30 y 4.31, respectivamente. Se mantuvieron los rangos de los universos de discurso y las formas generales de las funciones de membresía, siendo la única diferencia las funciones de membresía de los extremos. Originalmente, estas funciones mostraron un comportamiento *abierto* para valores fuera del rango, pero en este caso, las funciones de membresía *cierran* justo en los extremos.

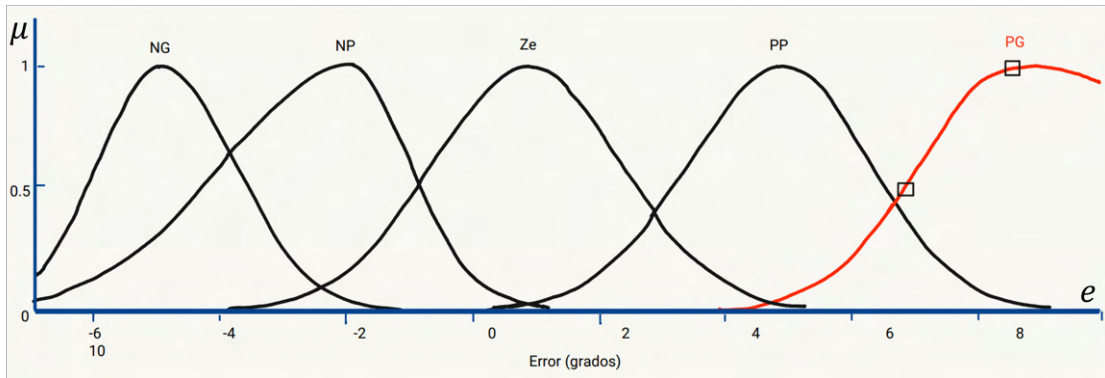


FIGURA 4.29: MF del error para la variante $Fuzzy_2$.

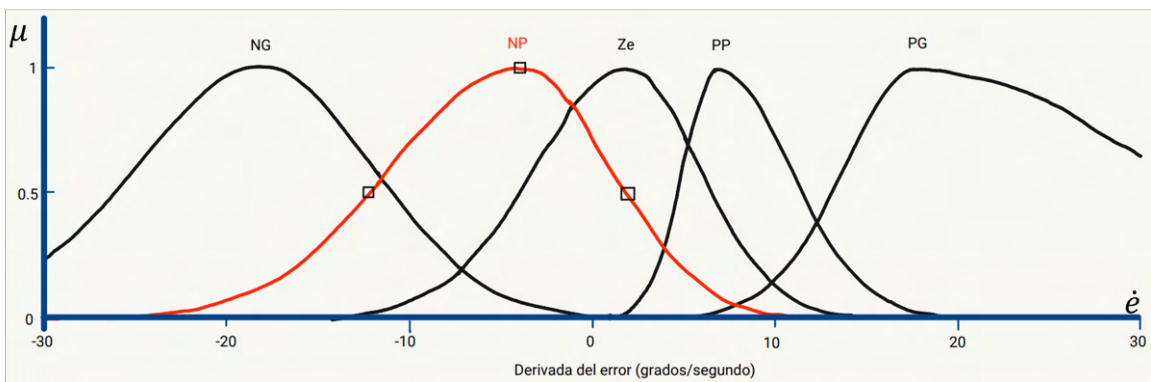


FIGURA 4.30: MF la derivada del error para la variante $Fuzzy_2$.

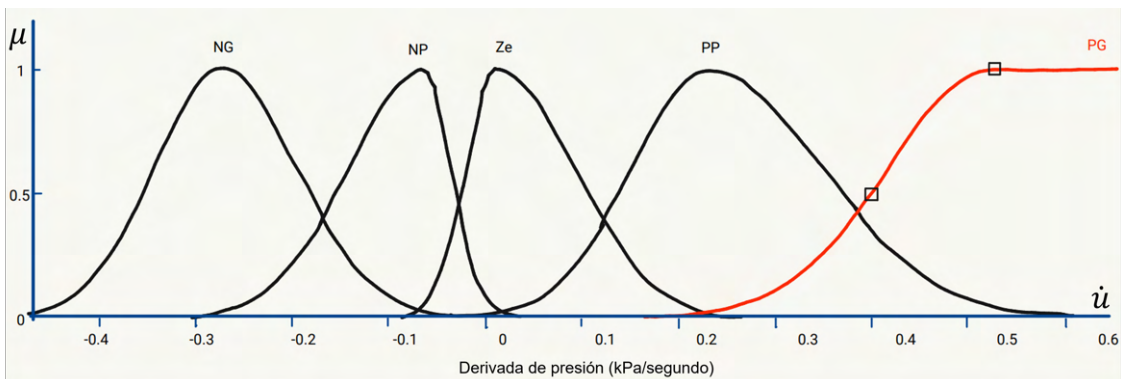


FIGURA 4.31: MF la derivada de presión para la variante $Fuzzy_2$.

La segunda variante, denominada $Fuzzy_3$, fue esencialmente $Fuzzy_2$ pero utilizando solo funciones de membresía (MF) triangulares. Esto se hizo para evaluar el efecto de utilizar funciones de membresía lineales en un sistema no lineal. Las funciones de membresía para el error, la derivada del error y la derivada de presión se pueden observar en las Fig. 4.32, 4.33 y 4.34, respectivamente.

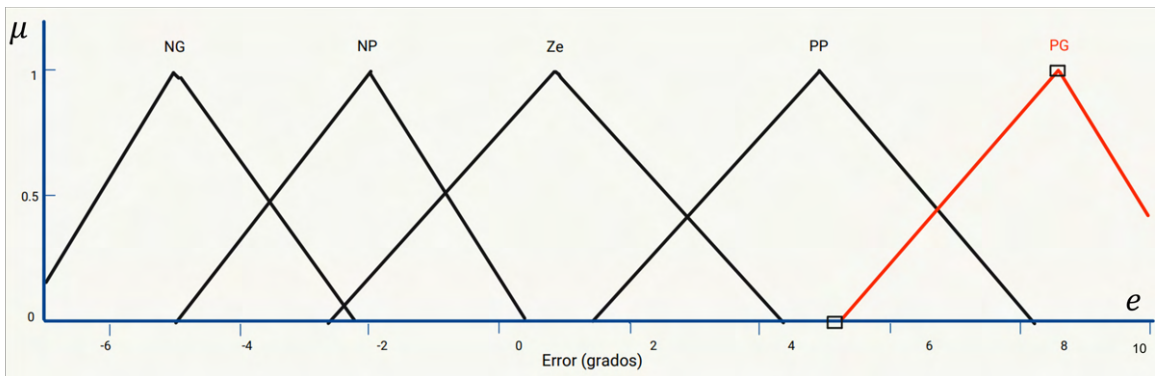


FIGURA 4.32: MF del error para la variante $Fuzzy_3$.

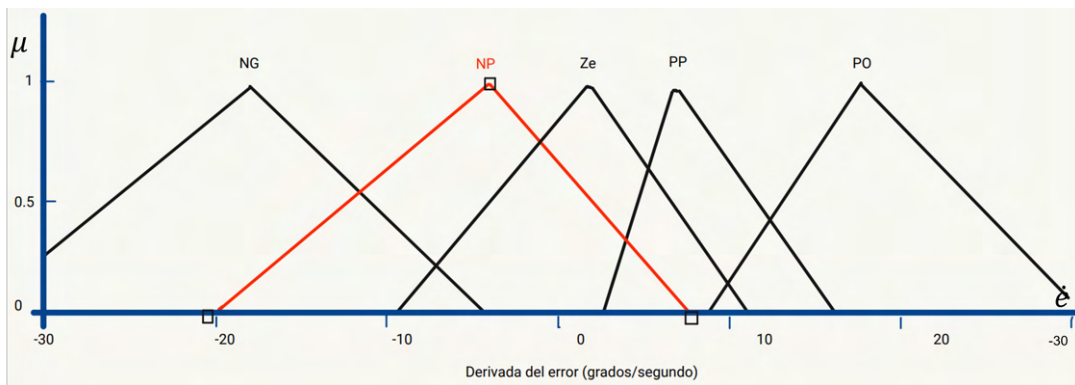


FIGURA 4.33: MF la derivada del error para la variante $Fuzzy_3$.

Finalmente, la última variante ($Fuzzy_4$) representó un caso que incluye 7 conjuntos difusos con funciones de membresía triangulares y separados de forma equidistante para un rango de valores diferente. El nuevo rango de trabajo para este caso es: $[-30, 40]^\circ$ para el error (ver Fig. 4.35), $[-80, 80] \frac{^\circ}{seg}$ para la derivada del error (ver Fig. 4.36) y $[-10, 10] \frac{kPa}{seg}$ para la derivada de presión (ver Fig. 4.37). Los 7 conjuntos difusos

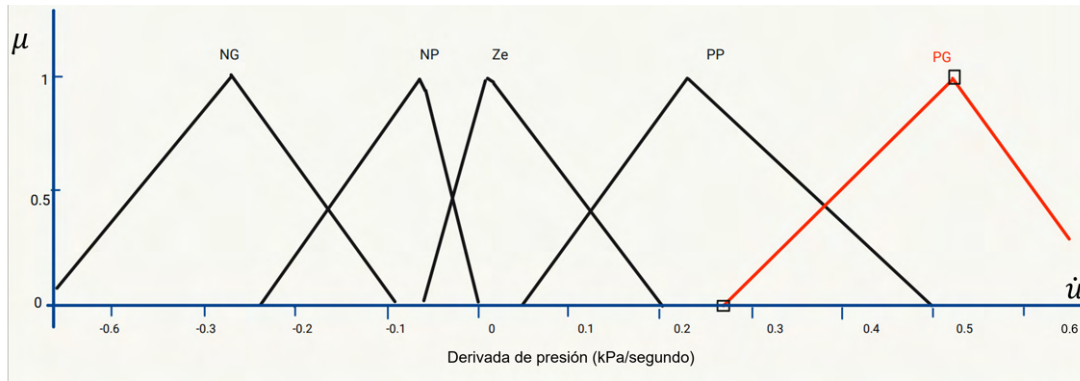


FIGURA 4.34: MF la derivada de presión para la variante $Fuzzy_3$.

con los que se trabajará ahora son: Negativo Grande (NG), Negativo Mediado (NM), Negativo Pequeño (NP), Cero (Z), Positivo Pequeño (PP), Positivo Mediado (PM) y Positivo Grande (PG).

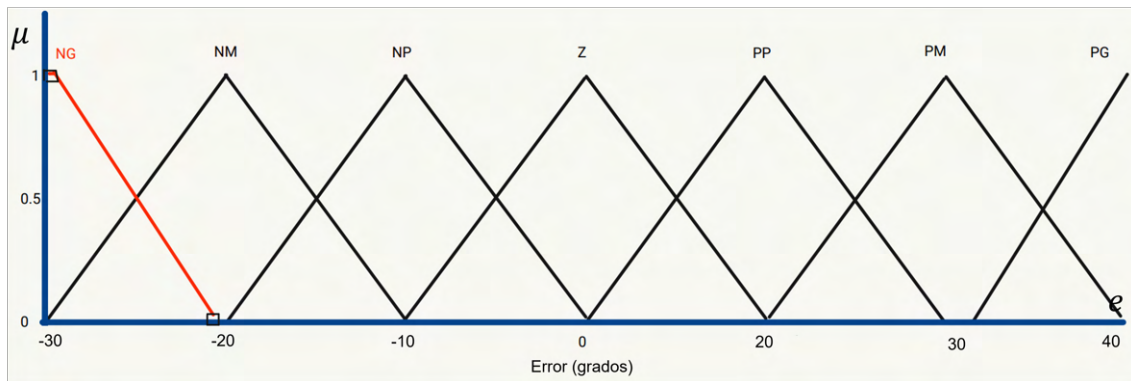


FIGURA 4.35: MF del error para la variante $Fuzzy_4$.

Cada una de estas variaciones se evaluó utilizando la misma referencia trapezoidal que se utilizó en experimentos anteriores. Además, se varió ligeramente la ganancia del controlador para observar su impacto en el rendimiento del sistema. La evaluación de su desempeño se realizó mediante la medición del Error Cuadrático Medio (RMS) y el valor máximo del ángulo. Estos experimentos se llevaron a cabo utilizando un modelo simplificado de bebé con una configuración de seis meses y manteniendo un peso constante en todo momento.

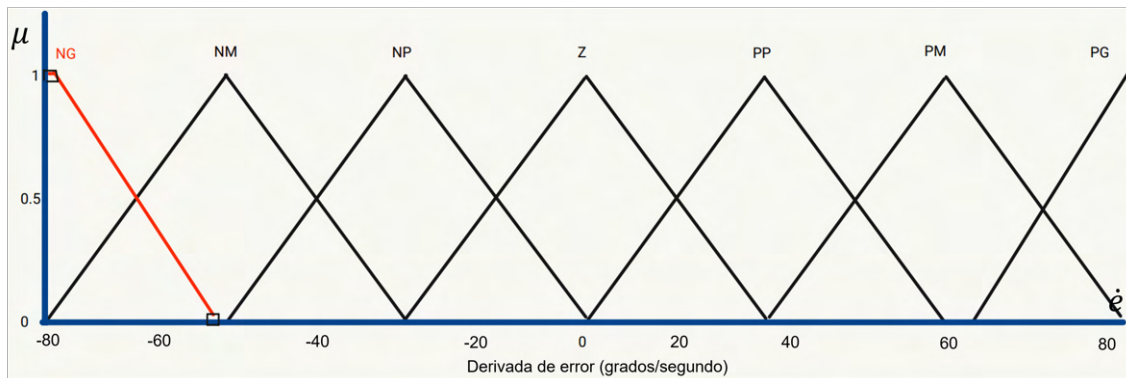


FIGURA 4.36: MF la derivada del error para la variante $Fuzzy_4$.

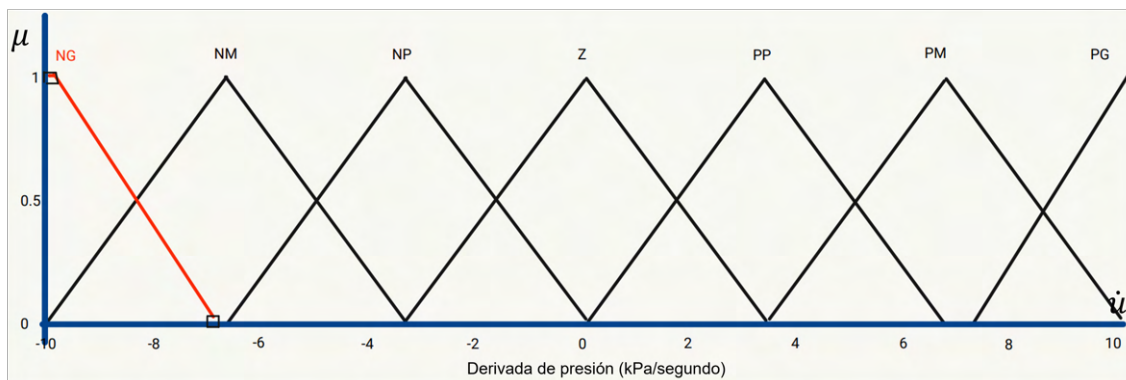


FIGURA 4.37: MF la derivada de presión para la variante $Fuzzy_4$.

La Tabla 4.6 presenta un resumen de los resultados obtenidos para cada combinación. Es evidente que a medida que la ganancia aumentó, el valor del RMS disminuyó. Sin embargo, el valor del ángulo máximo aumentó al mismo tiempo que el RMS disminuyó. La combinación 4 ($Fuzzy_4$) es un ejemplo de esto, con una ganancia de 15, logró un ángulo máximo de 65.87° . Aunque el valor numérico del RMS ofreció una idea del rendimiento, es crucial observar el comportamiento de cada caso para descartar posibles errores de funcionamiento.

La Fig. 4.38 ilustra gráficamente el comportamiento temporal de las variaciones

del control con sus mejores valores de ganancia. A grandes rasgos, todas muestran similitudes, pero se distinguen por ciertos detalles:

- $Fuzzy_4$ tarda aproximadamente 4 segundos en alcanzar el valor de referencia, mientras que los demás lo logran después de aproximadamente 5.5 segundos.
- $Fuzzy_3$ no muestra un buen comportamiento durante el retorno; a pesar de su movimiento suave, no logra seguir completamente la referencia y, al acercarse al ángulo mínimo, sufre una caída abrupta.
- Todas las combinaciones, excepto $Fuzzy_2$, coinciden en el valor máximo del ángulo que pueden alcanzar.
- En el momento de la subida, todas las variaciones muestran un comportamiento muy similar.

Es interesante destacar que el uso de funciones de membresía lineales proporcionó un mejor rendimiento en este caso particular. Esto se evidenció en la diferencia significativa observada en $Fuzzy_2$ en comparación con las demás variaciones.

4.6 Comportamiento del controlador para otras señales de referencia

En esta sección, se realizó un análisis exhaustivo del rendimiento del controlador difuso para entradas sinusoidales y trapezoidales con la misma amplitud (variando desde 10° hasta 60°) pero con diferentes tiempos de subida y retorno: 3, 5, 8 y 10 segundos. Los casos evaluados fueron simétricos, es decir, los tiempos de subida y retorno son iguales. Además, se agregó una masa de 50 gramos en el extremo de la pierna para simular una condición de un bebé más pesado, y se trabajó con una configuración de cero meses en el modelo de pierna simplificado.

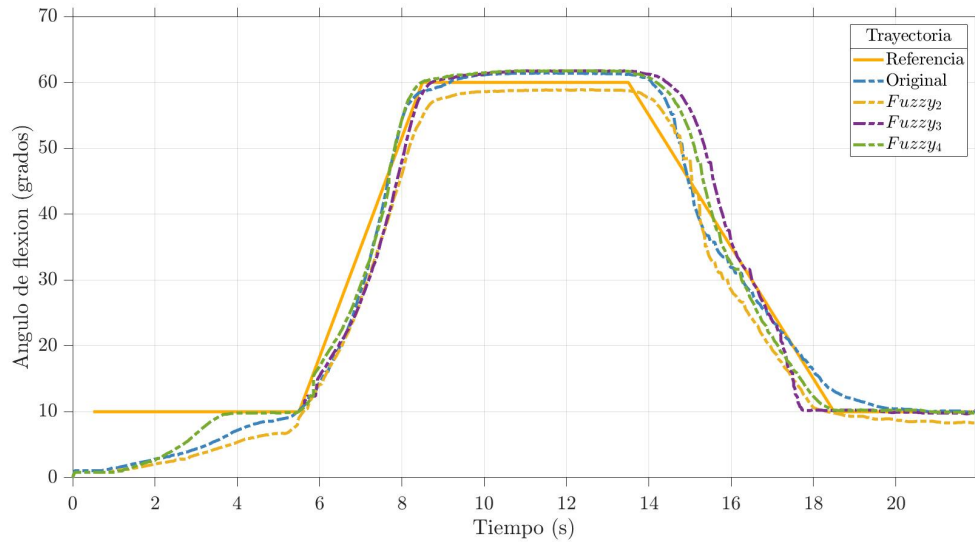


FIGURA 4.38: Comparación de desempeño de las variaciones del controlador. (*Experimento realizado en el banco de pruebas configurado para 6 meses*)

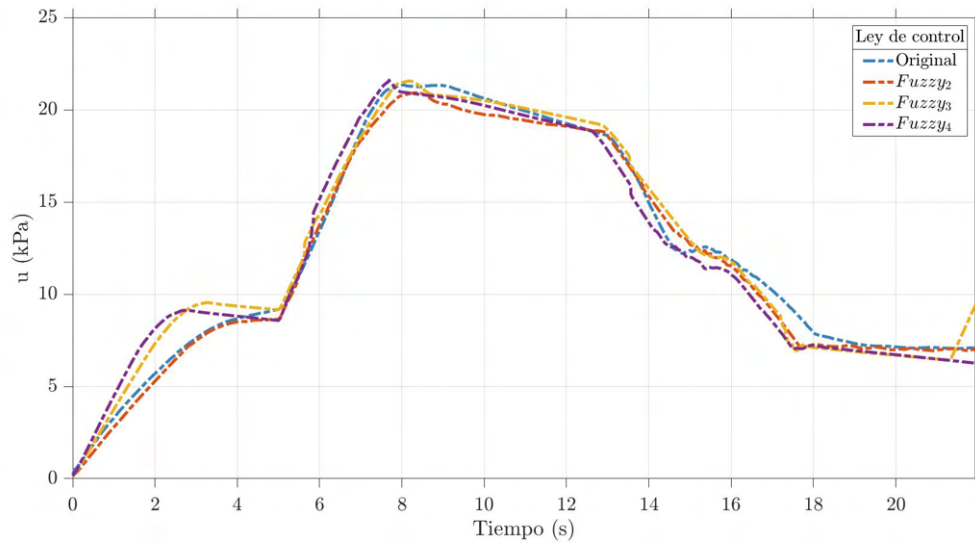


FIGURA 4.39: Comparación las leyes de control de las variaciones del controlador. (*Experimento realizado en el banco de pruebas configurado para 6 meses*)

Referencia	Ganancia	RMS	Ángulo máximo	¿Suave subida?	¿Suave retorno?
Original	0.8	12.5366	64.5625	Sí	Sí
	1	7.3035	62.2500	Sí	Sí
	1.1	6.3233	64.8750	Sí	Sí
<i>Fuzzy₂</i>	10	8.1057	59.8750	Sí	Sí
	12	6.1677	60.6875	Sí	Sí
	15	4.7158	60.7500	Sí	Sí
<i>Fuzzy₃</i>	10	9.6803	61.8125	Sí	Sí
	12	6.6602	61.6875	Sí	Sí
	15	5.0605	61.8125	Sí	Sí
<i>Fuzzy₄</i>	10	10.6264	62.5000	Sí	Sí
	12	7.2529	61.6875	Sí	Sí
	15	7.1568	65.8750	Sí	Sí

TABLA 4.6: Desempeño de cada variación del controlador difuso. (*Experimento realizado en el banco de pruebas configurado para 6 meses*)

La Tabla 4.7 presenta un resumen de los resultados obtenidos, incluyendo el Error Cuadrático Medio (*RMS*), que proporciona una medida del rendimiento del controlador a lo largo del tiempo, el ángulo máximo alcanzado y la suavidad del movimiento durante la subida y el retorno. Se observó que en todos los casos, el *RMS* en la prueba sin carga es menor que en la prueba con carga adicional. Este fenómeno se explica por la presencia de un mayor *overshoot* durante el retorno en los casos con carga adicional, lo que aumenta el error.

También se observó que el *RMS* tiende a disminuir a medida que se aumenta el tiempo de subida y retorno. Esta disminución se debe a que los cambios de presión se realizan de manera gradual en situaciones más lentas, y los efectos dinámicos tienen menos impacto en el movimiento. Por último, se notó que tanto la subida como el retorno son suaves en la mayoría de los casos, como se ilustra en la Fig. 4.41. Sin embargo, en los casos de 3 y 5 segundos con carga adicional, se observó un ligero *overshoot* (con un error máximo de 4.88°) durante el retorno, como se muestra en la Fig. 4.40.

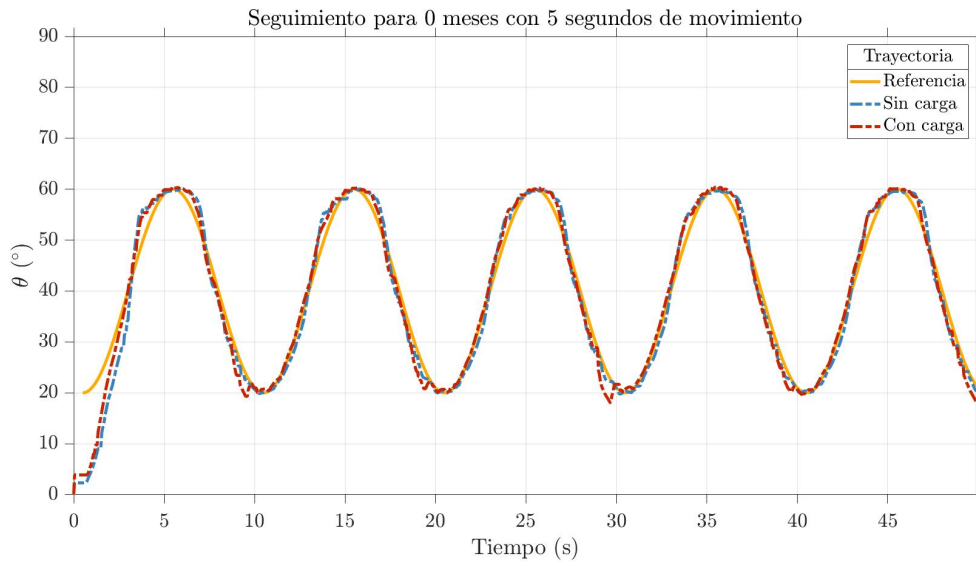


FIGURA 4.40: Comportamiento del controlador para una referencia sinusoidal con 5 segundos de subida y de retorno. (*Experimento realizado en el banco de pruebas configurado para 0 meses*)

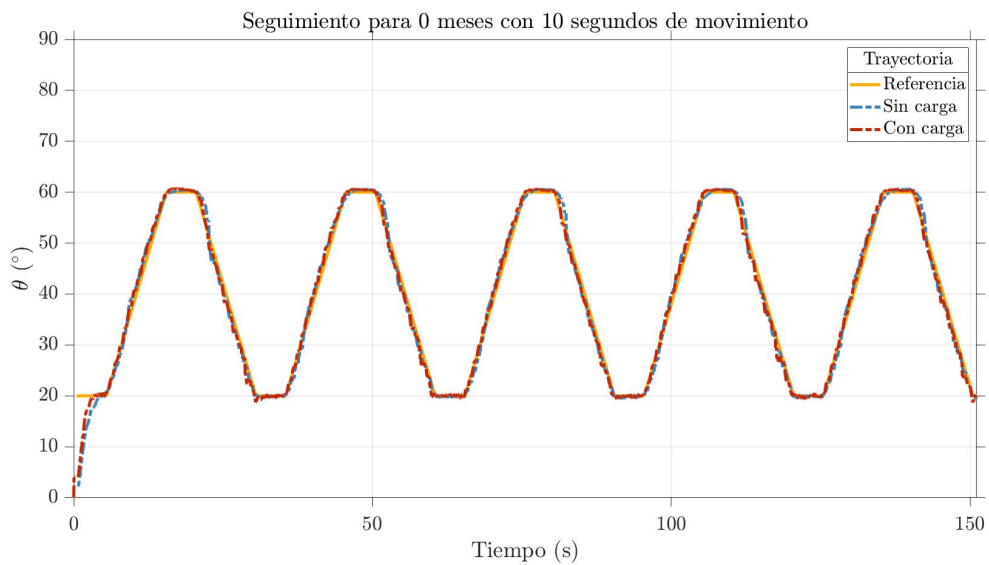


FIGURA 4.41: Comportamiento del controlador para una referencia trapezoidal con 10 segundos de subida y de retorno. (*Experimento realizado en el banco de pruebas configurado para 0 meses*)

Referencia	Tiempo de subida y retorno	RMS	Ángulo máximo	¿Suave subida?	¿Suave retorno?	Carga adicional	
Sinusoidal	3 segundos	7.3873	60.9375	Sí	Sí	-	
		8.3418	62.3125	Sí	Sí, con overshoot	50 gramos	
	5 segundos	4.8565	60.4375	Sí	Sí	-	
		5.9050	60.0625	Sí	Sí, con un pequeño overshoot	50 gramos	
	8 segundos	3.1061	60.4375	Sí	Sí	-	
		3.8424	60.4375	Sí	Sí	50 gramos	
	10 segundos	2.5564	60.5625	Sí	Sí	-	
		3.2093	60.5000	Sí	Sí	50 gramos	
	Trapezoidal	3 segundos	5.1736	60.8750	Sí	Sí	-
			6.0276	60.9375	Sí	Sí, con overshoot	50 gramos
5 segundos		3.1668	60.9375	Sí	Sí	-	
		4.0619	60.8750	Sí	Sí, con un pequeño overshoot	50 gramos	
8 segundos		2.1868	60.9375	Sí	Sí	-	
		2.8914	60.8750	Sí	Sí	50 gramos	
10 segundos		1.8624	60.8750	Sí	Sí	-	
		2.2435	60.8750	Sí	Sí	50 gramos	

TABLA 4.7: Resumen de los resultados para distintas referencias. (*Experimento realizado en el banco de pruebas configurado para 0 meses*)

4.7 Prueba de control frente a entradas tipo escalón

Es crucial evaluar el rendimiento de ambos controladores (difuso y PID) para el caso de entradas tipo escalón (*step*), ya que estas pueden ser necesarias para otro tipo de ejercicios de rehabilitación. Esta prueba se llevó a cabo utilizando la configuración de seis meses, con los valores de PID establecidos en $K_p = 0.5$, $K_i = 0.009$ y $K_d = 0.0005$, y el controlador difuso con funciones de membresía triangulares distribuidas de manera equidistante.

En las pruebas, las referencias enviadas al controlador variaron desde 10° hasta 70° , y cada experimento se realizó de forma independiente, manteniendo constantes las condiciones sin variar las ganancias o las masas utilizadas. Los resultados se presentan en las Fig. 4.42 y 4.43 para los controladores difuso y PID respectivamente.

En el caso del control PID, se observó que este logra aproximarse al valor de la referencia en aproximadamente 3.5 segundos, reduciendo gradualmente este error hasta alcanzar un valor muy cercano a 0° en alrededor de 10 segundos. Sin embargo, es importante notar que para valores de referencia superiores a 40° , comenzó a aparecer un fenómeno de *overshoot*, alcanzando un máximo de alrededor del 10.6167 % para una referencia de 60° .

Por otro lado, en la Fig. 4.42 se muestra el error generado por el controlador difuso, que presentó un comportamiento diferente al PID. Se observó que todos los casos, excepto para la referencia de 70° , mostraron un patrón de comportamiento muy similar, aunque con diferentes amplitudes. En todos los casos, el error permaneció casi constante durante el primer segundo y, curiosamente, todos los casos mostraron intersecciones alrededor de los 2.5 segundos, excepto para la referencia de 70° . No obstante, después de aproximadamente 6 segundos, el error se redujo a casi 0° , y el mayor valor de *overshoot* es de alrededor del 5.85 %.

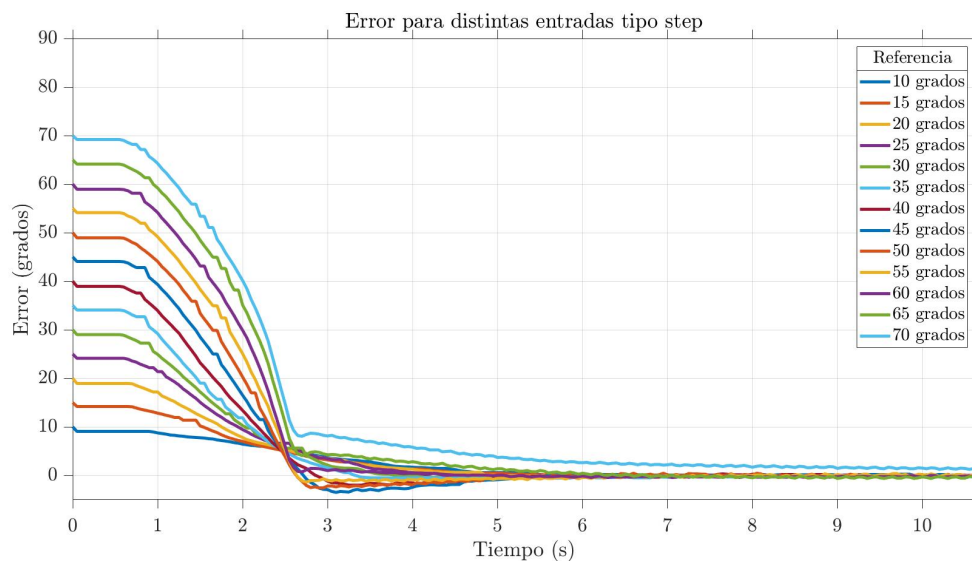


FIGURA 4.42: Error de control frente a entradas *step* utilizando el control difuso. (Experiencia realizada en el banco de pruebas configurado para 6 meses)

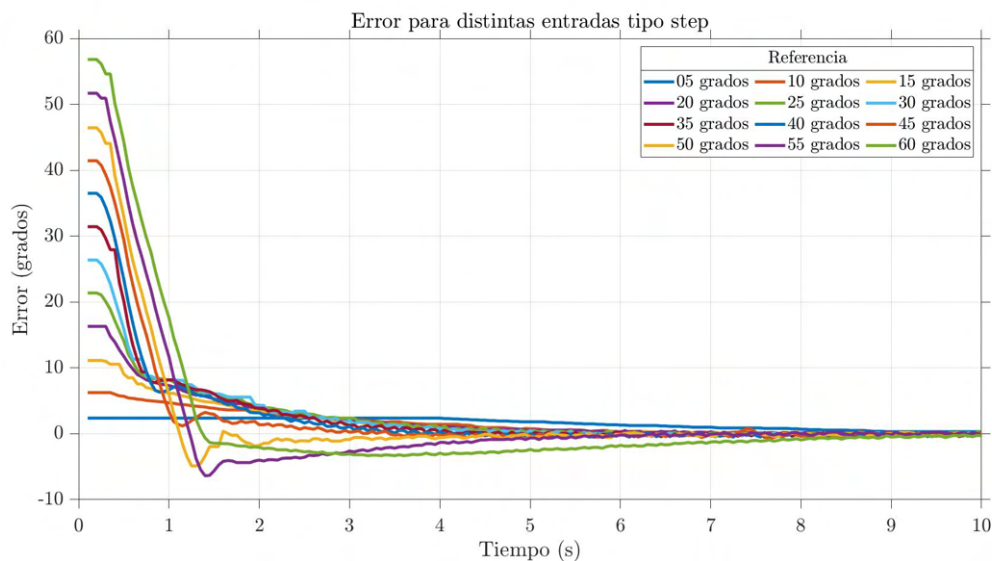


FIGURA 4.43: Error de control frente a entradas *step* utilizando el control PID. (*Experimento realizado en el banco de pruebas configurado para 6 meses*)

A pesar de que ambos controladores lograron alcanzar el valor de referencia deseado, es esencial evaluar la ley de control implementada. Estas leyes se ilustran en las Fig. 4.44 y 4.45 para el controlador difuso y el PID respectivamente. Aquí se destacan notables diferencias entre ambos enfoques.

En el caso del PID, se observó que utiliza un valor de presión alto desde el principio, lo que implica una respuesta inicial agresiva. Esta estrategia se traduce en múltiples saturaciones de presión a lo largo del tiempo. Por otro lado, el controlador difuso aumentó suavemente el valor de presión a medida que pasa el tiempo, evitando saturaciones significativas. Además, es interesante notar que la presión se saturaba solo en el caso extremo de 70° en el controlador difuso.

Una diferencia clave radicó en la presencia de oscilaciones en la ley de control del PID, que no se observaron en el enfoque difuso. Esto sugiere que el controlador difuso ofreció una respuesta más estable y suave, lo que pudo ser crucial en aplicaciones de rehabilitación donde la precisión y la suavidad del movimiento fueron fundamentales

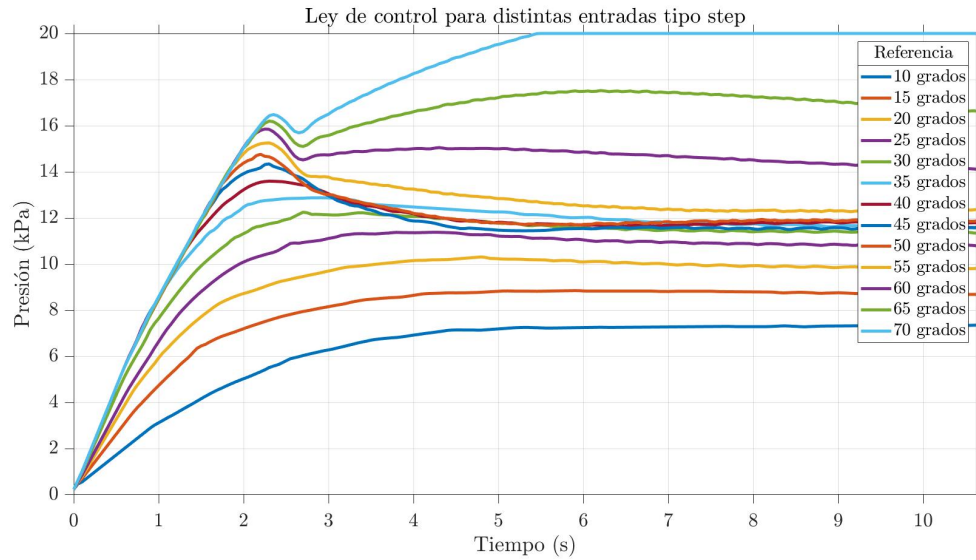


FIGURA 4.44: Ley de control frente a entradas *step* utilizando el control difuso. (Experimento realizado en el banco de pruebas configurado para 6 meses)

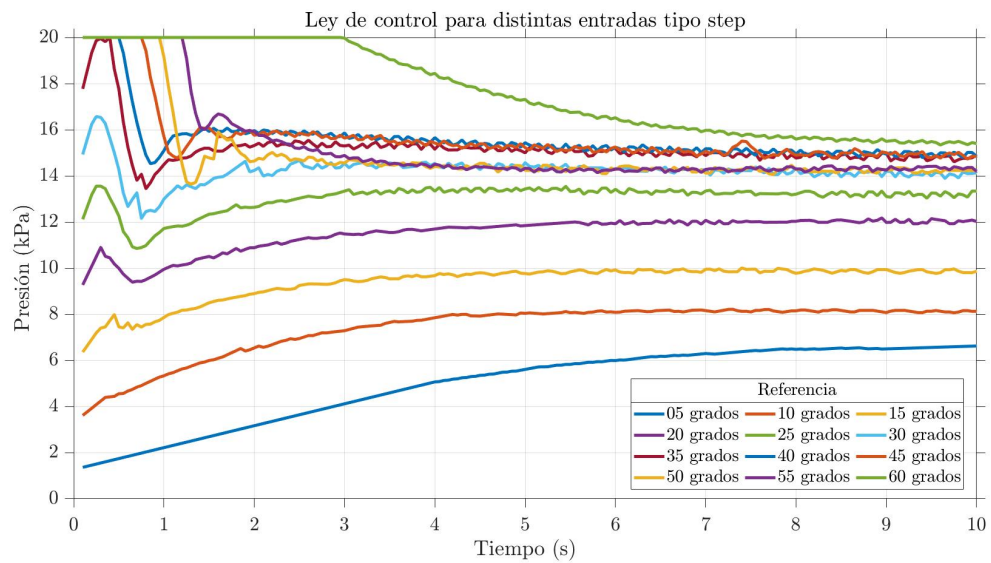


FIGURA 4.45: Ley de control frente a entradas *step* utilizando el control PID. (Experimento realizado en el banco de pruebas configurado para 6 meses)

4.8 Prueba de resistencia frente a disturbios

Una prueba adicional esencial que se llevó a cabo para evaluar la capacidad del controlador para corregir disturbios consistió en simular pequeños golpes realizados en la pierna durante la ejecución del experimento. La Fig. 4.46 ilustra cómo, en un instante específico, se aplicó un golpe manualmente a la pierna (señalado por el rectángulo negro), lo que provocó un aumento drástico del error en ese período de tiempo. Se realizaron un total de cinco de estos *empujones*, simulando así pequeñas patadas de un bebé, en diferentes momentos del movimiento del sistema:

- Tres de estos golpes, identificados como A , B y C en la gráfica, se aplicaron cuando la pierna estaba en su posición más alta. Estos golpes variaron en magnitud, desde un impacto leve hasta uno un poco más fuerte, con el objetivo de simular distintas intensidades de disturbios.
- Otro golpe, marcado como D en la gráfica, se aplicó durante la fase de subida del movimiento.
- El último golpe, señalado como E en la gráfica, se generó durante el período de reposo cuando la pierna estaba en su ángulo mínimo.

El propósito de estas pruebas fue observar cómo el sistema respondió a disturbios en diferentes etapas del movimiento y evaluar la capacidad del controlador para corregir el error inducido por estos disturbios.

El controlador difuso demostró una respuesta rápida frente a los disturbios, logrando retomar la trayectoria deseada en un corto periodo de tiempo. Los resultados se resumieron en la Tabla 4.8, que muestra el tiempo que tomó al controlador volver a la referencia, junto con los valores iniciales y finales del ángulo. Se observó que para los casos en los que la pierna se encontró en el ángulo máximo (A , B y C) y se aplicó un

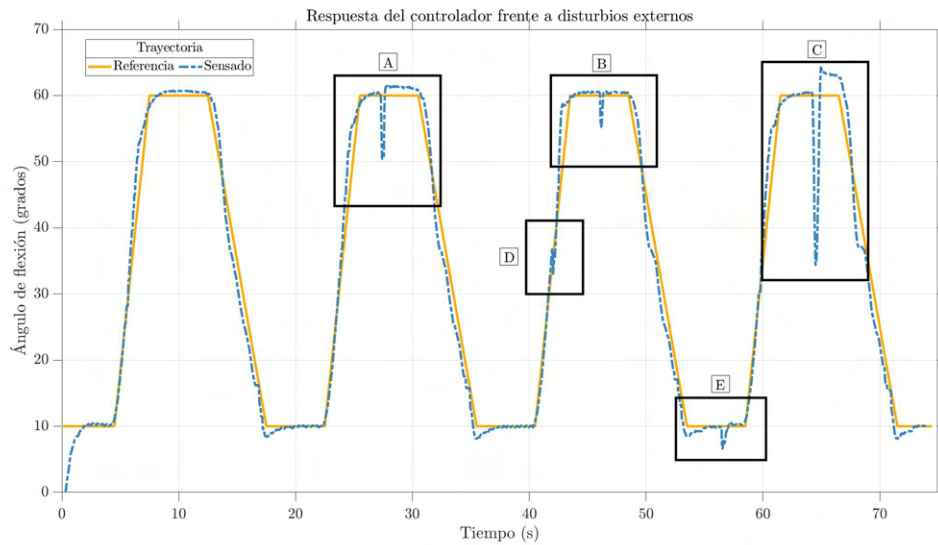


FIGURA 4.46: Respuesta del controlador frente a un disturbio generado. (*Experimento realizado en el banco de pruebas configurado para 6 meses*)

golpe de intensidad leve, el controlador recuperó la trayectoria en aproximadamente 0.55 segundos.

Importante destacar que en estos casos no hubo una gran diferencia entre el ángulo inicial y final. Esto es positivo, ya que implicó un retorno suave y la ausencia de un *overshoot* significativo. Sin embargo, las diferencias aparecieron en los valores finales del ángulo entre *A* y *C*. Mientras *A* mantuvo una diferencia menor a 1° entre el ángulo inicial y final, *C* superó nuestro límite (64°) con un ángulo final de 64.25° . El *overshoot* en *C* se atribuyó a la velocidad del cambio brusco en la presión y la velocidad resultante, lo que complicó detener el sistema en la referencia.

Por otro lado, el caso *E* mostró un comportamiento similar al de *B*, con un rápido retorno (0.622 segundos) y una diferencia no drástica entre los ángulos inicial y final, considerando que la referencia es de 10° en ese momento. En contraste, el caso *D* presentó una respuesta mucho más rápida que los demás (0.3 segundos para estabilizarse) y una diferencia mínima de aproximadamente 0.3° entre los valores iniciales y finales. Esto se pudo deber a que, como la referencia estuvo en constante crecimiento en ese período

Disturbio	Tiempo de respuesta (segundos)	Valor inicial (grados)	Valor final (grados)
A	0.88	60.5	61.38
B	0.55	60.56	60.69
C	0.8	60.38	64.25
D	0.3	41.95	42.25
E	0.622	9.938	10.56

TABLA 4.8: Características de la respuesta del controlador. (*Experimento realizado en el banco de pruebas configurado para 6 meses*)

Disturbio	Presión inicial (kPa)	Pico de presión (kPa)
A	13.67	15.36
B	13.1	13.89
C	13.45	16.83
D	6.422	6.969
E	10.02	-

TABLA 4.9: Variación de la presión para cada disturbio. (*Experimento realizado en el banco de pruebas configurado para 6 meses*)

de tiempo, la presión requerida para compensar el movimiento externo generado por el disturbio fue suficiente, aprovechando el movimiento ya existente para compensar el disturbio de manera eficaz.

Al analizar el comportamiento de la presión necesaria, o ley de control, como se muestra en la Fig. 4.47, se observó un aumento en la presión en los mismos periodos en los que se generaron los disturbios. Este fenómeno fue lógico, ya que un incremento en el error requiere una mayor presión por parte del controlador para corregir dicho error. A pesar de los disturbios, el controlador mantiene un comportamiento suave en todo momento, evitando alcanzar los límites de saturación. Un resumen de estos valores se presenta en la Tabla 4.9. Se destaca que la mayor variación en la presión fue de aproximadamente 3.4 kPa, observada en el caso *C*, que experimentó el disturbio más significativo. En los demás casos, las variaciones fueron leves, siendo *D* el de menor variación (aproximadamente 0.5 kPa), y *E* que no generó ninguna variación en la trayectoria de presión.

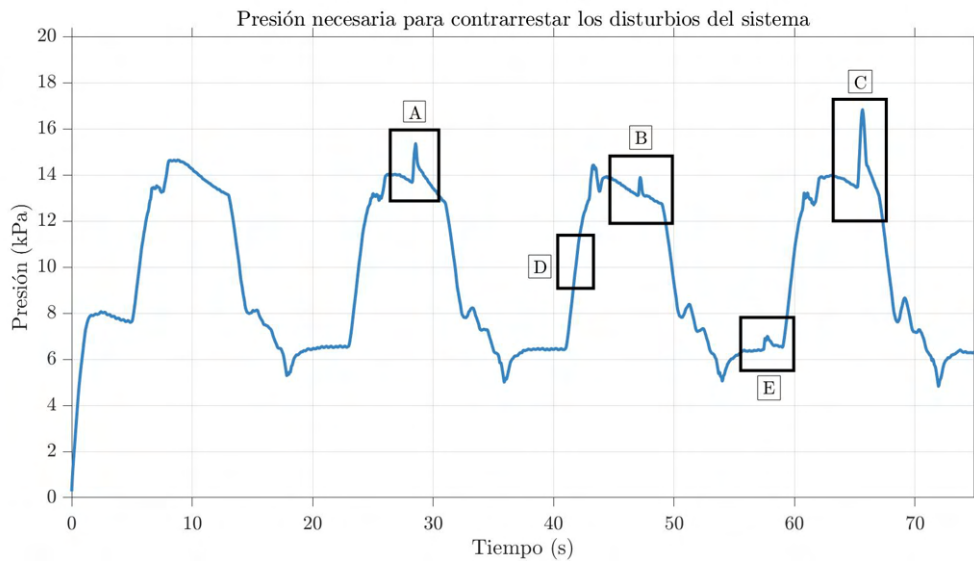


FIGURA 4.47: Respuesta de la ley de control frente a un disturbio generado. (*Experimento realizado en el banco de pruebas configurado para 6 meses*)

4.9 Efecto de la configuración del regulador electrónico en el control

Es esencial destacar la importancia de la configuración del regulador electrónico utilizado, ya que esta configuración pudo mejorar o deteriorar el desempeño del controlador, incluso con la posibilidad de llevar al sistema a la inestabilidad. En esta sección, se presenta un análisis general del efecto que tienen cuatro configuraciones diferentes del regulador en el controlador. Las dos variables que se modificaron fueron la ganancia y la sensibilidad, y para las pruebas se empleó el controlador difuso con una configuración de 0 meses para el modelo de pierna simplificado.

La Fig. 4.48 muestra el comportamiento temporal del controlador para una referencia de 40° (representada por la línea amarilla) con las cuatro posibles combinaciones, utilizando los valores máximos y mínimos para la ganancia y sensibilidad. Todas las pruebas se realizaron con el mismo controlador, sin modificar ningún parámetro ni variar la planta. Se observó que existen dos combinaciones donde el control se volvió completamente oscilatorio. Este fenómeno ocurrió cuando la ganancia tiene un valor igual a 0. Esta

situación tuvo sentido, ya que la ganancia indicó qué tan rápido el regulador debe alcanzar el valor deseado. Al estar configurada con un valor de 0, el regulador demoró en alcanzar el *set-point*, lo que resultó en una incapacidad para seguir correctamente nuestra ley de control.

Este comportamiento se ilustra de manera más detallada en las Fig. 4.49 y 4.50, donde la línea negra representa nuestra ley de control, mientras que la línea roja muestra la presión generada por el regulador electrónico. Idealmente, estas dos líneas deberían ser iguales, pero se puede observar cómo la presión generada por el regulador tendía a aumentar y necesita más tiempo para alcanzar el valor deseado, lo que demostró la importancia crítica de una configuración adecuada del regulador electrónico para el correcto funcionamiento del sistema.

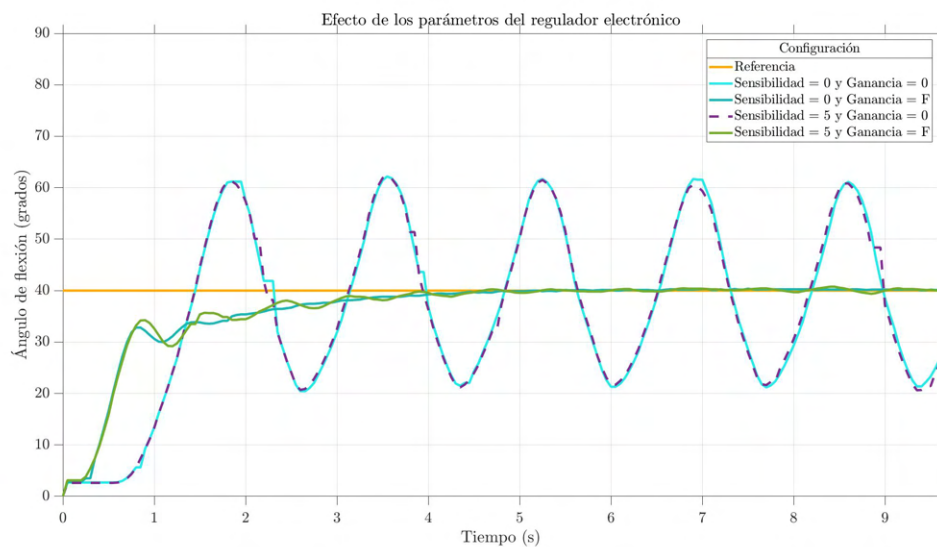


FIGURA 4.48: Efecto de los parámetros del regulador electrónico en el control. (*Experimento realizado en el banco de pruebas configurado para 0 meses*)

Por otro lado, la Fig. 4.48 ilustra que cuando la ganancia se ajustó a un valor específico F (indicado por las líneas de color verde y azul oscuro), el controlador logró alcanzar el valor de referencia sin problemas. Aunque visualmente se observó que el valor de la sensibilidad tuvo un impacto relativamente menor en el rendimiento, se nota que

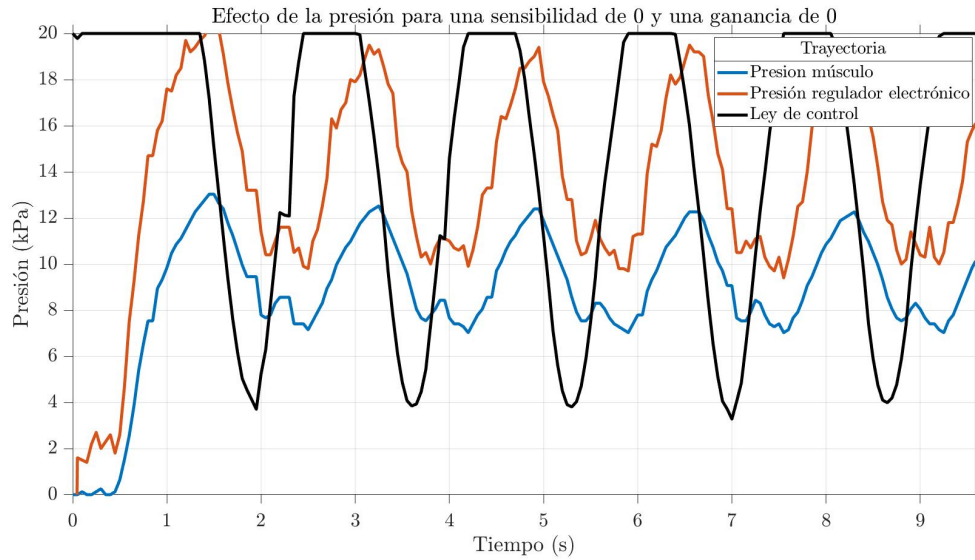


FIGURA 4.49: Seguimiento del regulador electrónico (línea roja) a la ley de control (línea negra) para una ganancia y sensibilidad iguales a 0. (Experimento realizado en el banco de pruebas configurado para 0 meses)

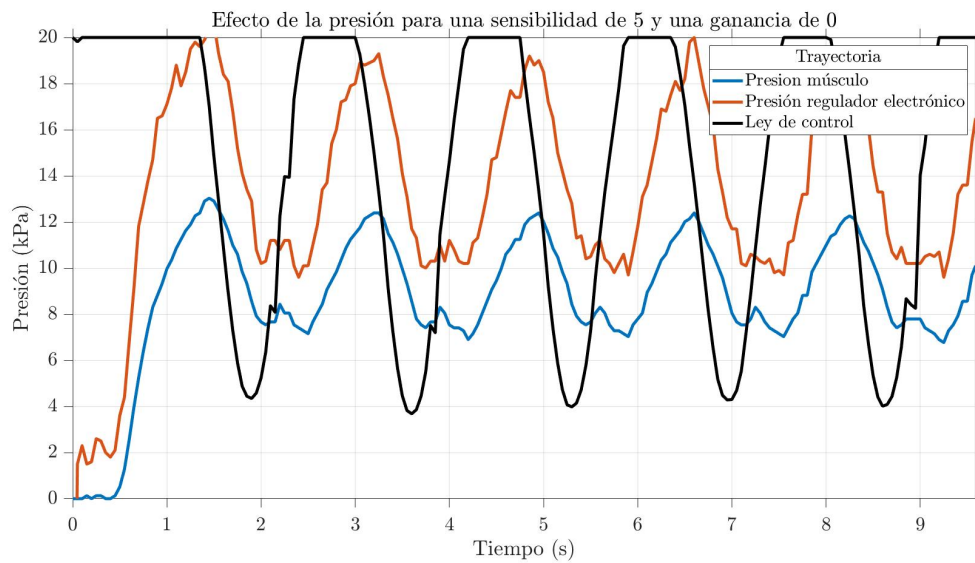


FIGURA 4.50: Seguimiento del regulador electrónico (línea roja) a la ley de control (línea negra) para una sensibilidad igual a F y una ganancia igual a 0. (Experimento realizado en el banco de pruebas configurado para 0 meses)

cuando la sensibilidad se establece en cinco, aparecen ligeras oscilaciones. Este efecto del valor de la sensibilidad se puede apreciar de manera gráfica en las leyes de control representadas en las Fig. 4.51 y 4.52. En el caso de una sensibilidad igual a cero, se observó una mayor presencia de ruido o cambios rápidos, mientras que para un valor de cinco, el controlador siguió de manera más precisa la ley de control requerida. Esto se debió a que la sensibilidad es el parámetro encargado de regular el rango de error permitido, siendo cero equivalente a *no tan preciso* y cinco siendo más preciso. Sin embargo, es crucial mencionar que cuando se utilizó una ganancia igual a cero, el ruido generado por el regulador se volvió significativamente más pronunciado debido a los cambios bruscos que se generan.

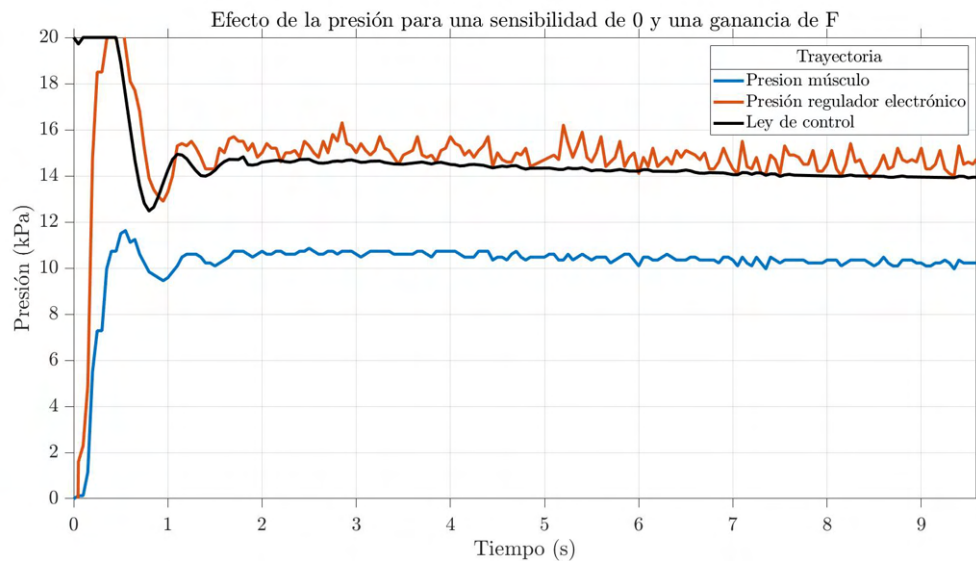


FIGURA 4.51: Seguimiento del regulador electrónico (línea roja) a la ley de control (línea negra) para una ganancia y sensibilidad iguales a F. (*Experimento realizado en el banco de pruebas configurado para 0 meses*)

4.10 Efecto de la fabricación de otro actuador flexible

En la actualidad uno de los mayores desafíos es el hecho que los actuadores utilizados son fabricados manualmente, lo que implica que cada actuador podría presentar

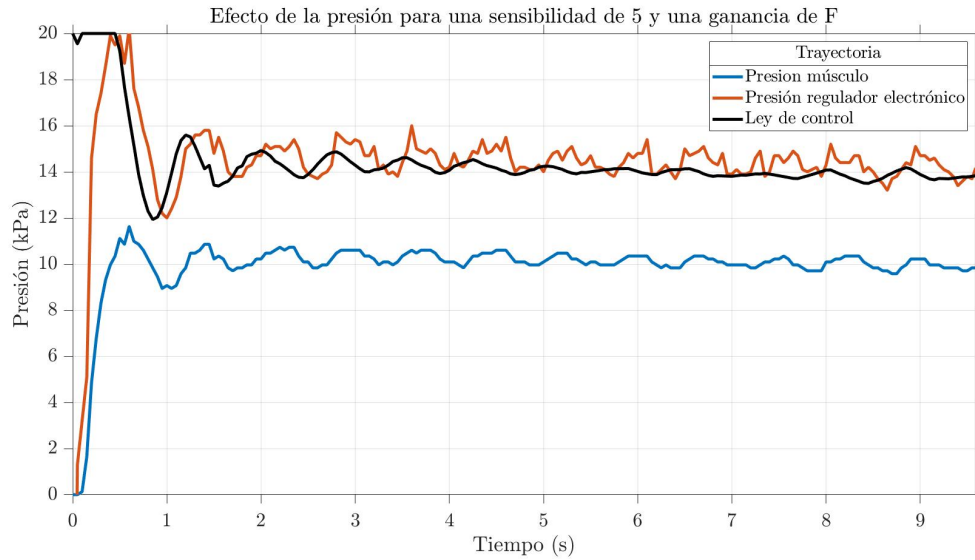


FIGURA 4.52: Seguimiento del regulador electrónico (línea roja) a la ley de control (línea negra) para una sensibilidad igual a F y una ganancia igual a F. (*Experimento realizado en el banco de pruebas configurado para 0 meses*)

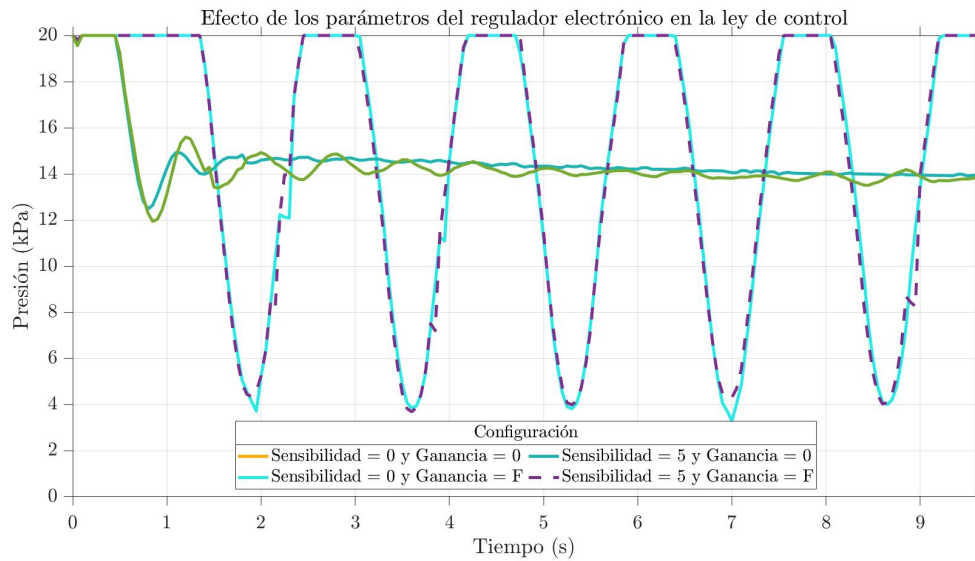


FIGURA 4.53: Efecto de los parámetros del regulador en el seguimiento de la ley de control. (*Experimento realizado en el banco de pruebas configurado para 0 meses*)

diferencias con respecto a los demás. Algunas de las principales irregularidades al fabricar los actuadores incluyen inclinaciones en los anillos, falta de precisión en el sellado del plástico e incluso variaciones en la calidad de impresión de las piezas. Además, al emplear bolsas de polietileno, estas pueden deformarse con el tiempo, estirándose o desarrollando pequeños cortes. Teniendo en cuenta estas limitaciones, esta sección se enfoca en analizar el impacto del uso del controlador difuso en dos actuadores diferentes a los utilizados en experimentos anteriores.

La Fig. 4.54 presentó el rendimiento de dos actuadores distintos utilizando el mismo controlador difuso. Se observó que ambos actuadores mostraron un comportamiento inicial muy similar al comienzo del experimento. Sin embargo, a medida que transcurrió el tiempo, el ángulo máximo del segundo actuador aumentó hasta alcanzar 64.88° , mientras que el primer actuador se mantuvo cerca de los 61° .

Otra similitud se encontró en la fase de subida, donde ambos casos mostraron un comportamiento exponencial similar. No obstante, la fase de retorno presentó ligeras diferencias, ya que el segundo actuador no siguió un movimiento continuo durante esa etapa. En su lugar, experimentó un periodo de aproximadamente 0.5 segundos en el que disminuye su velocidad y parece estar estático, principalmente cerca del centro de la trayectoria de retorno.

En cuanto a la ley de control, se observó una marcada diferencia en ambos casos, lo que subraya el impacto que tiene la variabilidad en la fabricación de los actuadores en el rendimiento del controlador (ver Fig. 4.55). Es esencial destacar que este experimento se realizó bajo las mismas condiciones, siendo la única diferencia los actuadores, que fueron diseñados sin la intención de inducir fallas en el sistema. En teoría, ambas leyes de control deberían ser idénticas o muy similares dado que se están aplicando sobre la misma planta. A pesar de estas diferencias, el controlador mostró un rendimiento aceptable (error $< 5\%$) en ambos casos, aunque se podría mejorar considerando la variación de la ganancia K_p en nuestro controlador.

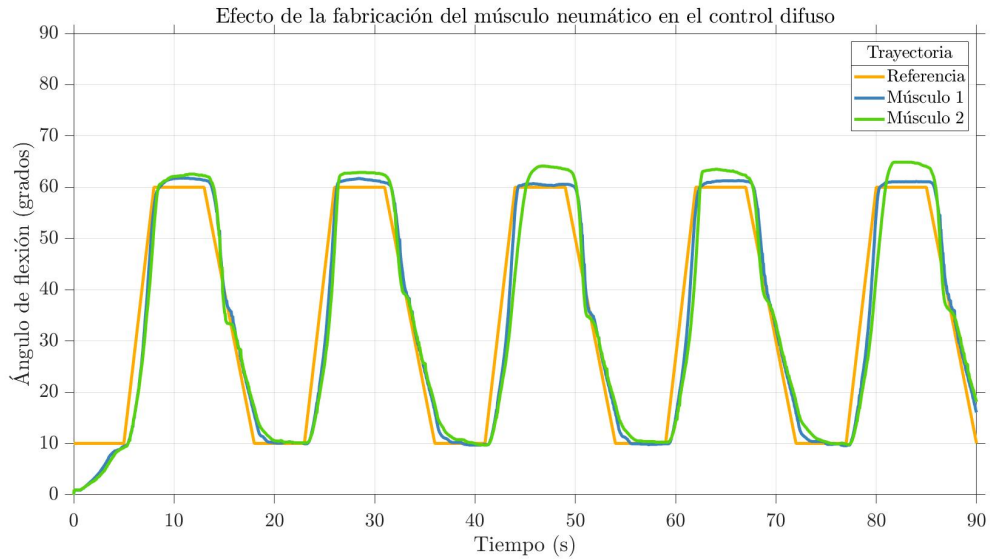


FIGURA 4.54: Diferencia de desempeño del controlador difuso utilizado dos actuadores distintos. (Experimento realizado en el banco de pruebas configurado para 3 meses)

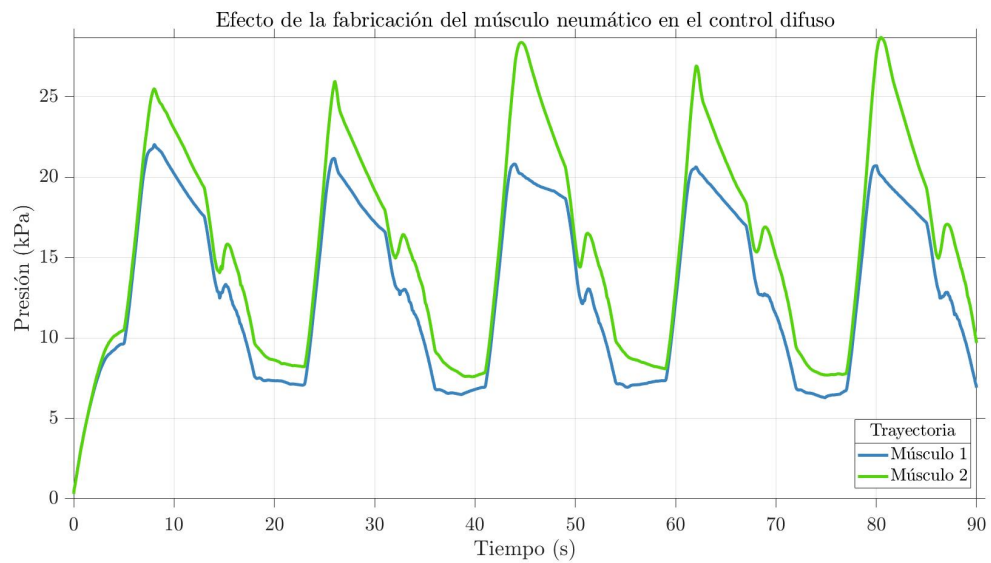


FIGURA 4.55: Diferencia de la ley de control del controlador difuso utilizado dos actuadores distintos. (Experimento realizado en el banco de pruebas configurado para 3 meses)

CONCLUSIONES

1. La implementación del sistema de retroalimentación arroja resultados satisfactorios al lograr la comunicación efectiva entre todos los dispositivos para la transmisión y medición precisa de las variables físicas. Esta conexión adecuada permite el correcto accionamiento de los instrumentos y la obtención de los valores requeridos.
2. Se desarrolla con éxito el control difuso para el sistema robótico utilizando Simulink. Se cumple todos los requisitos del sistema, incluyendo el tiempo de subida, tiempo de retorno, *overshoot* y la resistencia a los disturbios. Además, cumple con las condiciones de seguridad del bebé durante movimientos delicados.
3. La obtención de los valores óptimos para el controlador PID resulta desafiante, especialmente por la sensibilidad del sistema, las variaciones en las ganancias del controlador y a la diferencia de fabricación. Los movimientos de extensión de la pierna de maniquí se ven particularmente afectados, mostrando oscilaciones y cambios abruptos. En contraste, el controlador difuso demuestra mayor flexibilidad, ya que no requiere ajustes significativos en las ganancias para realizar las pruebas del sistema.
4. Se observa que las variaciones en las formas y conjuntos difusos no impactan significativamente en el rendimiento del controlador, generando leyes de control muy similares en cada caso.
5. Se logra cumplir los requisitos de tres segundos de tiempo de flexión, cinco segundos de reposo y cinco segundos de extensión para una pierna en todos los casos, tanto utilizando maniquís como modelos de pierna simplificados. Las leyes de control generadas son suaves, sin cambios bruscos en el movimiento.
6. El controlador difuso demuestra su capacidad para seguir diferentes trayectorias y tiempos de subida y bajada. Se realizan pruebas con referencias sinusoidales y trapezoidales, incluso al añadir una pequeña masa de 50 gramos en los extremos. Los

resultados indican que el controlador puede seguir las referencias de manera suave, destacando un mejor desempeño con tiempos de 10 segundos de subida y bajada, ya que el movimiento lento permite variaciones de presión igualmente suaves.

7. Se subraya la relevancia de las configuraciones de los equipos utilizados, ya que estas configuraciones pueden impactar positiva o negativamente en el rendimiento del controlador. Este factor se evidencia especialmente en el caso del regulador electrónico utilizado en esta investigación.

Sin embargo, debido al alcance del presente trabajo de investigación, se propone:

- Controlar el sistema sin un modelo matemático, obtener dicho modelo podría mejorar significativamente el rendimiento. La obtención del modelo podría basarse en el uso de redes neuronales utilizando los datos generados por los experimentos o aproximándolo a un sistema masa-resorte-amortiguador.
- Se podría mejorar el desempeño del controlador utilizando un microcontrolador o sensor más potente que permita una transmisión más rápida de datos a una frecuencia más alta, como unos Xsens para la toma el sensando.
- Actualmente, el sistema depende de una computadora para llevar a cabo el proceso de control. Sería beneficioso explorar la posibilidad de embeber todo el sistema, brindándole así mayor autonomía al sistema sin depender de una computadora externa.
- Se debería investigar y mejorar el diseño del actuador utilizado para evitar posibles fugas que puedan ocurrir después de múltiples pruebas, prolongando así su vida útil.
- Se podría explorar métodos de optimización avanzados como algoritmos genéticos para obtener funciones de membresía optimizadas y validar su desempeño en el maniquí.

- Evaluar la posibilidad de diseñar una arquitectura que permita la variación automática de la ganancia K_p en función de las condiciones antropomórficas del bebé, lo que podría mejorar la adaptabilidad del controlador a diferentes escenarios.
- Implementación del controlador para poder mover ambas piernas de manera simultánea o alternada, permitiendo al exoesqueleto ser utilizado para distintas terapias.

REFERENCIAS BIBLIOGRÁFICAS

- [1] H. Lipson, “Challenges and opportunities for design, simulation, and fabrication of soft robots,” pp. 21–27, 3 2014.
- [2] P. H. Nguyen and W. Zhang, “Design and computational modeling of fabric soft pneumatic actuators for wearable assistive devices,” *Scientific Reports*, vol. 10, 12 2020.
- [3] Q. Liu, J. Zuo, C. Zhu, and S. Q. Xie, “Design and control of soft rehabilitation robots actuated by pneumatic muscles: State of the art,” *Future Generation Computer Systems*, vol. 113, pp. 620–634, 12 2020.
- [4] K. Subramanyam, E. Rogers, M. Kulesza, D. Holland, J. Gafford, E. Goldfield, and C. Walsh, “Soft wearable orthotic device for assisting kicking motion in developmentally delayed infants,” *Journal of Medical Devices, Transactions of the ASME*, vol. 9, 2015.
- [5] D. Rus and M. T. Tolley, “Design, fabrication and control of soft robots,” pp. 467–475, 5 2015.
- [6] M. Mendoza Flores, “Diseño y fabricación de un exoesqueleto robótico flexible para rehabilitación de extremidades inferiores en recién nacidos con espina bífida,” B.S. Thesis, UTEC, 2021.
- [7] Instituto Nacional de Salud del Niño San Borja, “Guía de práctica clínica para el diagnóstico y tratamiento de mielomeningocele,” *Lima*, p. 1–19, 2018.
- [8] Instituto Mexicano del Seguro Social, “Guía de práctica clínica. prevención, diagnóstico y tratamiento de la espina bífida en niño,” *IMSS*, p. 1–30, 2013.

- [9] C. Lee, M. Kim, Y. J. Kim, N. Hong, S. Ryu, H. J. Kim, and S. Kim, “Soft robot review,” pp. 3–15, 2 2017.
- [10] T.-C. Tsai and M.-H. Chiang, “Design and control of a 1-dof robotic lower-limb system driven by novel single pneumatic artificial muscle,” *Applied Sciences (Switzerland)*, vol. 10, p. 43, 2020.
- [11] B. Tondu, “Modelling of the mckibben artificial muscle: A review,” *Journal of Intelligent Material Systems and Structures*, vol. 23, pp. 225–253, 2 2012.
- [12] D. B. Reynolds, D. W. Repperger, C. A. Phillips, and G. Bandry, “Modeling the dynamic characteristics of pneumatic muscle,” *Annals of Biomedical Engineering*, vol. 31, pp. 310–317, 2003.
- [13] N. Sun, D. Liang, Y. Wu, Y. Chen, Y. Qin, and Y. Fang, “Adaptive control for pneumatic artificial muscle systems with parametric uncertainties and unidirectional input constraints,” *IEEE Transactions on Industrial Informatics*, vol. 16, 2020.
- [14] P. Hyatt, D. Wingate, and M. D. Killpack, “Model-based control of soft actuators using learned non-linear discrete-time models,” *Frontiers Robotics AI*, vol. 6, 2019.
- [15] K. Balasubramanian and K. S. Rattan, “Fuzzy logic control of a pneumatic muscle system using a linearizing control scheme,” *22nd International Conference of the North American Fuzzy Information Processing Society, NAFIPS 2003*, pp. 432–436, 2003.
- [16] G. Runge, J. Peters, and A. Raatz, “Design optimization of soft pneumatic actuators using genetic algorithms,” in *2017 IEEE International Conference on Robotics and Biomimetics (ROBIO)*, 2017, pp. 393–400.
- [17] H. P. H. Anh and K. K. Ahn, “Hybrid control of a pneumatic artificial muscle (pam) robot arm using an inverse narx fuzzy model,” *Eng. Appl.*

- Artif. Intell.*, vol. 24, no. 4, p. 697–716, jun 2011. [Online]. Available: <https://doi.org/10.1016/j.engappai.2010.11.007>
- [18] F. Largilliere, V. Verona, E. Coevoet, M. S. Lopez, J. Dequidt, C. Duriez, and M. Sanz-Lopez, “Real-time control of soft-robots using asynchronous finite element modeling,” 2015. [Online]. Available: <https://hal.inria.fr/hal-01163760>
- [19] F. Faure, C. Duriez, H. Delingette, J. Allard, B. Gilles, S. Marchesseau, H. Talbot, H. Courtecuisse, G. Bousquet, I. Peterlik, and S. Cotin, “Sofa: A multi-model framework for interactive physical simulation sofa: A multi-model framework for interactive sofa, a multi-model framework for interactive physical simulation,” *Physical Simulation. Yohan Payan. Soft Tissue Biomechanical Modeling for Computer Assisted Surgery*, vol. 11, p. 978, 2012. [Online]. Available: <https://hal.inria.fr/hal-00681539>
- [20] P. Polygerinos, S. Lyne, Z. Wang, L. F. Nicolini, B. Mosadegh, G. M. Whitesides, and C. J. Walsh, “Towards a soft pneumatic glove for hand rehabilitation,” 2013, pp. 1512–1517.
- [21] M. Luo, E. H. Skorina, W. Tao, F. Chen, S. Ozel, Y. Sun, and C. D. Onal, “Toward modular soft robotics: Proprioceptive curvature sensing and sliding-mode control of soft bidirectional bending modules,” *Soft Robotics*, vol. 4, pp. 117–125, 6 2017.
- [22] N. Siddique, “Intelligent control nazmul siddique a hybrid approach based on fuzzy logic, neural networks and genetic algorithms foreword by,” pp. 118–119, 2013. [Online]. Available: <http://www.springer.com/series/7092>
- [23] S. D. Apkon, R. Grady, S. Hart, A. Lee, T. McNalley, L. Niswander, J. Petersen, S. Remley, D. Rotenstein, H. Shurtleff, M. Warner, and W. O. Walker, “Advances in the care of children with spina bifida,” p. 33–74, 2014.

- [24] K. Rotter, F. Solís, and M. González, “Costos de atención en pacientes con mielomeningocele en los institutos de rehabilitación infantil teletón,” *Revista Chilena de Pediatría*, vol. 78, p. 35–42, 2007.
- [25] Y. L. Park, J. Santos, K. G. Galloway, E. C. Goldfield, and R. J. Wood, “A soft wearable robotic device for active knee motions using flat pneumatic artificial muscles,” 2014, pp. 4805–4810.
- [26] E. Kokkoni, Z. Liu, and K. Karydis, “Development of a soft robotic wearable device to assist infant reaching,” *Journal of Engineering and Science in Medical Diagnostics and Therapy*, vol. 3, 2020.
- [27] K. K. Ahn and T. U. D. C. Thanh, “Nonlinear pid control to improve the control performance of the pneumatic artificial muscle manipulator using neural network,” *Journal of Mechanical Science and Technology*, vol. 19, pp. 106–115, 2005.
- [28] S. D. Gollob, M. J. Mendoza, B. H. B. Koo, E. Centeno, E. A. Vela, and E. T. Roche, “A length-adjustable vacuum-powered artificial muscle for wearable physiotherapy assistance in infants,” *Frontiers in Robotics and AI*, vol. 10, 2023. [Online]. Available: <https://www.frontiersin.org/articles/10.3389/frobt.2023.1190387>
- [29] S. Sridar, P. H. Nguyen, M. Zhu, Q. P. Lam, and P. Polygerinos, “Development of a soft-inflatable exosuit for knee rehabilitation,” in *2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2017, pp. 3722–3727.
- [30] K. Ogata, *Modern Control Engineering*, 4th ed. USA: Prentice Hall, 2001.
- [31] H. Sánchez, R. Vilanova, and O. Arrieta, “Implementación de controladores pid: Equivalencia y optimalidad,” *XXXV Jornadas de Automática, España*, 2014.
- [32] D. Seborg, T. Edgar, D. Mellicamp, and F. D. III, *Process Dynamics and Control (3rd Edition)*, 2011.

- [33] K. Ogata, *Ingeniería de Control Moderna*, 3rd ed. Prentice-Hall Hispanoamericana, S.A., 1998.
- [34] D. Inga and E. Jara, “Control inteligente - MT0014 sistemas difusos: Control fuzzy,” Material del curso Control Inteligente - MT0014, UTEC, Mayo 2022.
- [35] K. Townsend, “Adafruit bno055 absolute orientation sensor,” *Adafruit learning system*, 2019.
- [36] N. Ahmad, R. A. R. Ghazilla, N. M. Khairi, and V. Kasi, “Reviews on various inertial measurement unit (imu) sensor applications,” *International Journal of Signal Processing Systems*, 2013.
- [37] SMC, “Electro-pneumatic regulator/electronic vacuum regulator series itv,” pp. 805–805, 2015. [Online]. Available: <https://www.smc pneumatics.com/pdfs/ITV.pdf>
- [38] MATLAB, “Programación en arduino con matlab y simulink,” 2020. [Online]. Available: <https://la.mathworks.com/discovery/arduino-programming-matlab-simulink.html>

ANEXOS

4.11 Código Arduino

El código de Arduino utilizado se comparte en IV.1 junto con su respectivo diagrama de flujo mostrado en 3.11. El Arduino utilizado fue un Arduino Mega 2560, las librerías principales son las que el mismo Adafruit diseñó y están compartidas en su repositorio de Github.

```
// Librerías utilizadas:

// Librerías del IMU:
#include <Wire.h>
#include <Adafruit_Sensor.h>
#include <Adafruit_BNO055.h>

// Librerías de la DAC:
#include <Wire.h>
#include <Adafruit_MCP4725.h>

// Variables del sensor de presión:
#define CAP_VCC A0 // Alimentación auxiliar.
#define CAP_GND A1 // Ground auxiliar.

// Variable de la DAC:
Adafruit_MCP4725 dac;

// Variables del sensor de presión:
int analogPin = A2; // Lectura de los datos.
float calibration; // Valor inicial.
int p = 0; // Presión inicial.
```

```

// Variables de IMU:
Adafruit_BNO055 bno = Adafruit_BNO055(55, 0x28);

// Regulador de presi n PIN lectura.
int RegulatorPin = A7;

// Variables controlador:
float setpoint = 35.0;
float currentAngle;
float output;

unsigned long currentTime = 0; //Tiempo actual.

float u = 0; //Ley de control.
int bitsU = 0; // Ley de control en bits.
float bitsU_float = 0.0; // Ley de control en bits en decimal.

//-----
// Definimos la variable de decimal a bits.
typedef union{
    float number;
    uint8_t bytes[4];
} FLOATUNION_t;
//-----

// Definimos variables en bits.
FLOATUNION_t sendAngle;
FLOATUNION_t sendPre_Mus;
FLOATUNION_t sendPre_Reg;
FLOATUNION_t serialU;

// Loop de configuraci n.
void setup(void)

```

```

{
  // Verificamos si es que el IMU funciona:
  Serial.begin(115200);
  if (!bno.begin())
  {
    Serial.print("No BNO055 detected");
    while (1);
  }

  Wire.begin();

  // Configuraci n DAC
  dac.begin(0x60);
  dac.setVoltage(0, false);

  // Configuraci n sensor de presi n .
  pinMode(CAP_VCC, OUTPUT);
  pinMode(CAP_GND, OUTPUT);
  digitalWrite(CAP_VCC, HIGH);
  digitalWrite(CAP_GND, LOW);
  calibration = 941.16 - analogRead(analogPin);

  delay(1000);
}

// Loop principal.
void loop(void)
{
  // Data serial.
  serialU.number = getFloat();
  u = serialU.number;

  //IMU lectura:
  sensors_event_t orientationData;

```

```

bno.getEvent(&orientationData , Adafruit_BNO055 ::VECTOR_EULER);
currentAngle = orientationData.orientation.y;

// Sensor de presi n lectura.
p = analogRead(analogPin);
float Pre_Muscle = raw2kPa(p, calibration);

// Regulador de presi n lectura.
int Pre_Regulator = analogRead(RegulatorPin);

// Saturaci n por seguridad en 30 kPa
if (u > 30.0){
    u = 30.0;
}
else if(u < 0.0){
    u = 0;
}

// Transformamos la ley de control en bits.
bitsU_float = u*4095.0/30.0;

// Cambiamos de tipo de dato.
bitsU = bitsU_float;

// Enviamos los bits a la DAC.
dac.setVoltage(bitsU , false);
//}

// Ahora hacemos la transmision de los datos hacia el otro lado:
// Variable de salida (sensor), angulo:
sendAngle.number = currentAngle;
sendPre_Mus.number = Pre_Muscle;
sendPre_Reg.number = Pre_Regulator;

// Indicador de inicio.

```

```

Serial.write('A');

for (int i=0; i<4; i++){
    Serial.write(sendAngle.bytes[i]);
}

for (int i=0; i<4; i++){
    Serial.write(sendPre_Mus.bytes[i]);
}

for (int i=0; i<4; i++){
    Serial.write(sendPre_Reg.bytes[i]);
}

Serial.print('\n');

delay(50);
}

// Funci n de presi n para transformar los valores en bruto a kPa:
float raw2kPa(int raw, float calibration) {
    return (((raw + calibration) / 1023.0) - 0.92) / 0.007652;
}

// Funci n para transformar a decimales:
float getFloat(){
    int cont = 0;
    FLOATUNION_t f;
    while (cont < 4 ){
        f.bytes[cont] = Serial.read() ;
        cont = cont +1;
    }
    return f.number;
}
}

```

LISTING IV.1: Código de Arduino

Es importante mencionar que se debe definir previamente la forma en la que se realizará la comunicación entre la computadora y el Arduino, para esta investigación se utilizó una velocidad de 115200 baudrates, la letra 'A' y el salto de línea como indicadores de inicio y fin de la comunicación, respectivamente.

4.12 Diagrama de bloques Simulink

En la Fig. 4.56 se comparte el diagrama de bloques utilizado para la implementación del sistema. Para este se utilizó la librería Fuzzy Logic Toolbox para el bloque del controlador difuso y el bloque de comunicación Serial por defecto de Simulink, el cual fue configurado con las características del Arduino.

4.13 Función de costo Matlab

Se comparte los códigos utilizados para la implementación de las funciones de costo, las cuales tienen como parámetros de entrada las desviaciones estandar de cada una de las funciones Gaussianas. Siendo la Gaussiana de dos colas IV.2 y la Gaussiana de una cola IV.3 las mismas que mostraron en el capítulo de metodología. La salida de estas funciones es el costo actual para dicha configuración.

```
%% Funcion de costo para las curvas:  
% input -> S = [S1, S2];  
% output -> CF (valor de costo);  
function CF = costGaussDosColas(S)  
  
% Seleccionamos el conjunto difuso a aproximar: (CAMBIAR)  
muestra = 5;
```

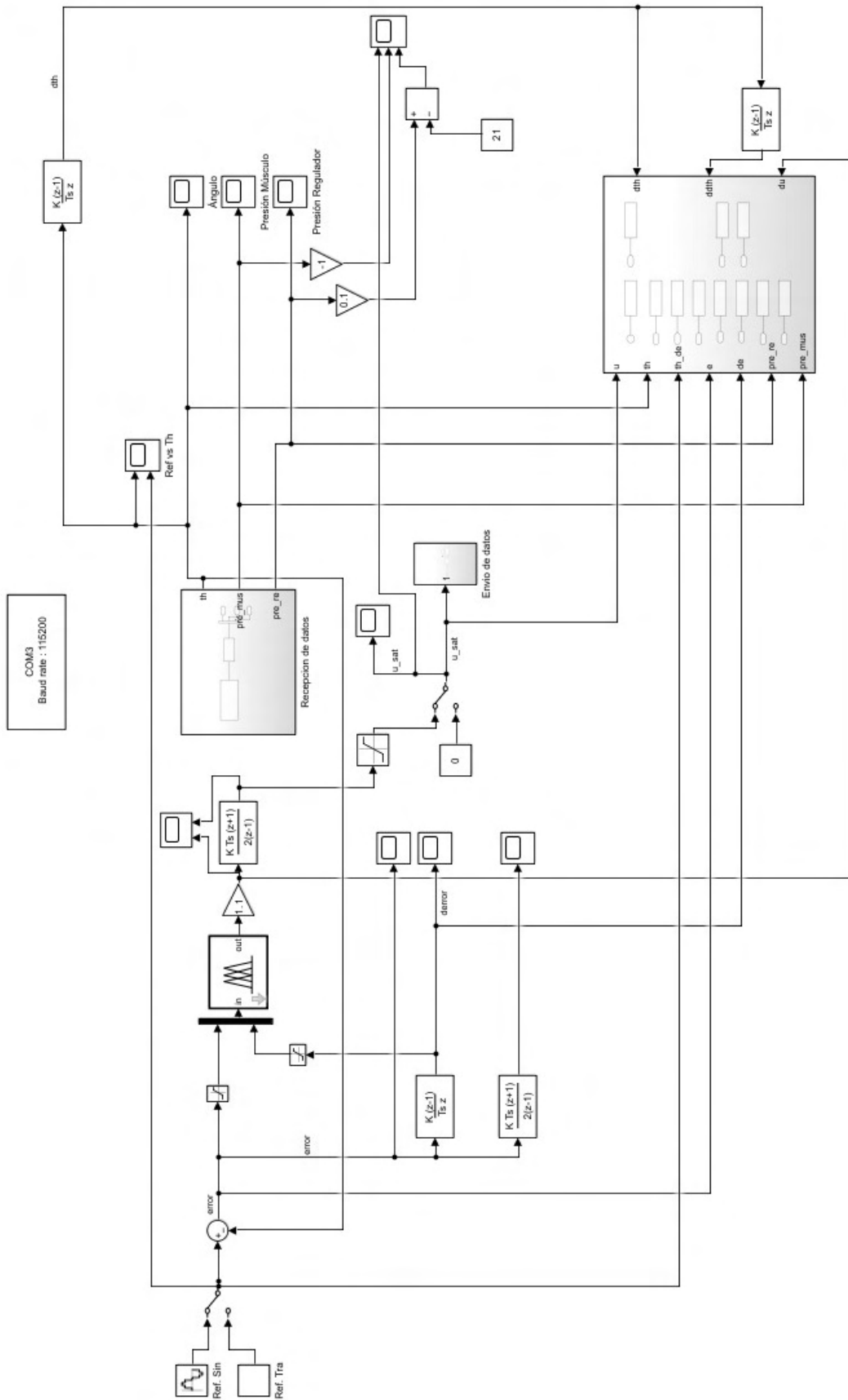


FIGURA 4.56: Diagrama de bloques Simulink.

```

% Cargamos la variable que tiene almacenado todos los valores: (CAMBIAR
)
load 'PID_Cortado_Step.mat' T_F

% Ordenamos en funci n de la variable deseada: (CAMBIAR)
data = sort(T_F(:,4));

% Cargamos los datos de los centros y del fcm. (CAMBIAR)
load 'FCM_u.mat' C_u U_u

% Hacemos un cambio de variable: (CAMBIAR)
C = C_u;
U = U_u;

% Seleccionamos el centro:
Centro = C(muestra);

% Utilizaremos el mismo centro en ambos casos:
m1 = Centro;
m2 = Centro;

% Las desviaciones a utilizar:
s1 = S(1);
s2 = S(2);

% Creamos un vector de ceros para guardar la aproximacion;
G = zeros(length(data),1);

% Nos basamos en la ecuaci n para generar los datos:
for i = 1:length(data)

    if data(i) <= m1
        G(i) = exp(-0.5*((data(i) - Centro)./s1).^2);
    end
end

```



```

end

if data(i) > m2
    G(i) = exp(-0.5*((data(i) - Centro) ./ s2).^2);
end

end

% Utilizamos el error cuadrático:
CF = rms(G-U(muestra, :)' );

end

```

LISTING IV.2: Función de costo de gaussiana de dos colas

```

%% Funcion de costo para las curvas:
% input -> S = [SI];
% output -> CF (valor de costo);
function CF = costGaussSimple(S)

% Seleccionamos el conjunto difuso a aproximar: (CAMBIAR)
muestra = 5;

% Cargamos la variable que tiene almacenado todos los valores: (CAMBIAR)
)
load 'PID_Cortado_Step.mat' T_F

% Ordenamos en función de la variable deseada: (CAMBIAR)
data = sort(T_F(:,4));

% Cargamos los datos de los centros y del fcm. (CAMBIAR)
load 'FCM_u.mat' C_u U_u

% Hacemos un cambio de variable: (CAMBIAR)
C = C_u;

```

```

U = U_u;

% Seleccionamos el centro:
Centro = C(muestra);

% Utilizaremos el mismo centro en ambos casos:
m1 = Centro;

% Las desviaciones a utilizar:
s1 = S(1);

% Creamos un vector de ceros para guardar la aproximacion;
G = exp(-0.5*((data - m1)./s1).^2);

% Utilizamos el error cuadrático:
CF = rms(G-U(muestra, :)' );

end

```

LISTING IV.3: Función de costo de gaussiana de una cola