

UNIVERSIDAD TECNOLÓGICA DE LA MIXTECA  
DIVISIÓN DE ESTUDIOS DE POSGRADO  
MAESTRÍA EN ROBÓTICA



---

“INSTRUMENTACIÓN Y CONTROL DE UN PROTOTIPO DE  
EXOESQUELETO PARA MANO EN REHABILITACIÓN  
ASISTIDA”

---

**TESIS**

PARA OBTENER EL TÍTULO DE MAESTRO EN ROBÓTICA

PRESENTA:

**ING. BERITH ATEMOZTLI DE LA CRUZ SÁNCHEZ**

DIRECTOR DE TESIS:

**DR. MANUEL ARIAS MONTIEL**

CO-DIRECTORA DE TESIS:

**DRA. ESTHER LUGO GONZÁLEZ**

HUAJUAPAN DE LEÓN, OAXACA, MÉXICO, FEBRERO DEL 2020



Tesis presentada en Febrero del 2020 ante los sinodales:

- Dr. Andrés Blanco Ortega  
Centro Nacional de Investigación y Desarrollo Tecnológico (CENIDET)
- Dr. Ricardo Tapia Herrera  
Cátedras CONACYT-UTM
- Dr. Carlos García Rodríguez  
Profesor-Investigador UTM
- Dr. Fermín Hugo Ramírez Leyva  
Profesor-Investigador UTM



# Dedicatoria

*A dios...*  
*A mis padres...*  
*A mi hermana...*  
*A los que han creído en mí.*



# Agradecimientos

A dios, por darme una familia maravillosa, cruzar en mi camino a personas increíbles y por permitirme lograr una meta profesional más.

A mis padres por apoyarme y guiarme con sus consejos a lo largo de mi vida, por enseñarme que todo es posible y luchar por lo que quiero. Por confiar en mi y motivarme cuando no veo salida o cuando nadie más ha creído en mí. Por estar siempre conmigo.

A mi hermana Hameli por tu afecto y amor que me has dado, siempre a mi lado y siendo mi motor para nunca rendirme.

A mis tías, por las atenciones, cariño y amor que siempre me han brindado, dándome fuerzas para seguir adelante.

Al Dr. Manuel Arias Montiel y la Dra. Esther Lugo González, por permitirme trabajar a su lado y guiarme en todo el camino de la maestría, por todas las revisiones y correcciones en mis escritos. Por enseñarme a creer en mí y siempre motivarme a llegar mas alto. Gracias por su apoyo.

A Yubezni Iriani, por tu amistad y ayuda incondicional. Porque a pesar de mis cambios de humor y errores que he cometido siempre me apoyas y estás para mí. Gracias por ser mi amiga y acompañarme en todas las locuras.

A Esther Guadalupe, por dedicarme tu tiempo y talento de modelaje para la recolección de mis fotos, por ser una gran amiga, siempre escuchar y dar ánimos cuando todo parece perdido.

A Heidi, por regañarme y regresarme al camino cuando me vuelvo una oveja descarriada. Porque a pesar de la distancia y los años siempre puedo contar tu amistad incondicional.

I would like to say thanks to my friends Colin and Ali, for your friendship, for helping me to enjoy my stay in Idaho and for your patience dealing with our communication problems. Thank you so much for being there and teaching me that even in the most hostile place I will know incredible people like you.

A la Dra. Alba Pérez y al Dr. P Schoen por darme la oportunidad de trabajar en sus laboratorios, por el apoyo al trabajo desarrollado durante la estancia y por todas las atenciones.

A todos mis amigos y amigas que he conocido durante el trayecto de la maestría, y que al igual que yo, el sentimiento de vacío y desesperación por lo que hacemos muchas veces nos invade, pero es minimizado por nuestra amistad y por aquellos momentos que pasamos juntos.

A la Universidad Tecnológica de la Mixteca por la formación académica recibida.

Al Consejo Nacional de Ciencia y Tecnología ( CONCACYT) por el apoyo económico brindado durante mis estudios de maestría, así mismo, por el apoyo recibido para la realización de la estancia en la Universidad Estatal de Idaho.

A la Universidad Estatal de Idaho por abrirme las puertas durante la realización de mi estancia.

A todas las personas voluntarias que compartieron un poco de su tiempo conmigo para la realización de pruebas.

# Índice

<b>Índice de figuras</b>	<b>XII</b>
<b>Índice de tablas</b>	<b>XV</b>
<b>1. Introducción</b>	<b>1</b>
1.1. Antecedentes . . . . .	2
1.2. Planteamiento del problema . . . . .	9
1.3. Justificación . . . . .	10
1.4. Hipótesis . . . . .	10
1.5. Objetivos . . . . .	10
1.5.1. Objetivo general . . . . .	10
1.5.2. Objetivos específicos . . . . .	10
1.6. Delimitaciones . . . . .	11
1.7. Metodología de desarrollo . . . . .	11
1.8. Estructura de la tesis . . . . .	12
<b>2. Marco teórico</b>	<b>13</b>
2.1. Anatomía de la mano . . . . .	13
2.1.1. Huesos y articulaciones de la mano . . . . .	13
2.1.2. Tipos de agarre de la mano. . . . .	14
2.2. Rehabilitación . . . . .	15
2.2.1. Clasificación de los robots de rehabilitación. . . . .	17
2.2.2. Paradigma de rehabilitación . . . . .	17
2.2.3. Rehabilitación bilateral . . . . .	18
2.3. Instrumentación de dispositivos . . . . .	20
2.3.1. Adquisición de datos . . . . .	20
2.3.2. Procesamiento digital de señales . . . . .	21
2.3.3. Filtrado de señales . . . . .	22
2.4. Electromiografía . . . . .	23
2.5. Aprendizaje automático . . . . .	23
2.5.1. Algoritmos de clasificación . . . . .	24
2.5.2. Lógica difusa . . . . .	26

2.6.	Sistemas de control . . . . .	26
2.6.1.	Control basado en lógica difusa . . . . .	27
<b>3.</b>	<b>Rediseño de prototipo y validación cinemática</b>	<b>29</b>
3.1.	Obtención de requerimientos para el nuevo diseño . . . . .	29
3.2.	Especificaciones de diseño del pulgar . . . . .	30
3.2.1.	Obtención de trayectoria del pulgar con captura de movimiento . . . . .	31
3.2.2.	Análisis de datos y obtención de requerimientos de diseño . . . . .	33
3.3.	Rediseño de componentes . . . . .	35
3.4.	Validación cinemática . . . . .	39
3.5.	Construcción de prototipo . . . . .	41
<b>4.</b>	<b>Diseño e implementación del sistema de instrumentación</b>	<b>45</b>
4.1.	Instrumentación de actuadores . . . . .	46
4.2.	Obtención de señales EMG . . . . .	47
4.3.	Creación de banco de datos . . . . .	50
4.4.	Preprocesamiento de señales EMG . . . . .	51
4.4.1.	Detección de activación muscular . . . . .	51
4.4.2.	Filtrado y normalización de señales . . . . .	53
4.4.3.	Extracción de características . . . . .	54
4.5.	Selección de características . . . . .	57
4.5.1.	Dependencia estadística . . . . .	57
4.5.2.	Información mutua. . . . .	57
4.5.3.	Selección secuencial hacia adelante . . . . .	58
4.5.4.	Selección secuencial flotante hacia adelante . . . . .	58
4.5.5.	Selección de características de subconjunto aleatorio . . . . .	58
4.6.	Clasificación de señales EMG . . . . .	59
<b>5.</b>	<b>Diseño del sistema de control</b>	<b>63</b>
5.1.	Descripción general del sistema . . . . .	63
5.2.	Obtención de requerimientos . . . . .	64
5.2.1.	Adquisición de datos provenientes del encoder . . . . .	66
5.2.2.	Diseño de la tarjeta de circuito impreso . . . . .	67
5.2.3.	Caracterización de motores . . . . .	68
5.3.	Selección de controlador . . . . .	70
5.3.1.	Modelo dinámico y linealización . . . . .	70
5.3.2.	Diseño de un controlador PID . . . . .	72
5.4.	Diseño del controlador difuso . . . . .	73
5.4.1.	Etapa de fuzificación. . . . .	74
5.4.2.	Reglas difusas. . . . .	76
5.4.3.	Máquina de inferencia. . . . .	77
5.5.	Implementación del controlador difuso en Arduino . . . . .	78

---

5.5.1. Biblioteca para el controlador TB6612FNG . . . . .	79
5.5.2. Biblioteca de lógica difusa . . . . .	80
5.6. Diseño de la interfaz gráfica de usuario . . . . .	82
5.6.1. Realización de una nueva prueba . . . . .	84
5.6.2. Agregar prueba a la base de datos y recalibración . . . . .	85
5.7. Resultados del controlador . . . . .	86
<b>6. Conclusiones y trabajos futuros</b>	<b>91</b>
6.1. Conclusiones . . . . .	91
6.2. Trabajos a futuro . . . . .	93
<b>A. Planos constructivos.</b>	<b>95</b>
<b>B. Características más relevantes del banco de datos.</b>	<b>107</b>
<b>C. Rangos de movimiento por falange para diferentes movimientos de la mano.</b>	<b>111</b>
<b>D. Modelo dinámico</b>	<b>115</b>
<b>E. Código Arduino</b>	<b>119</b>
<b>F. Publicaciones derivadas.</b>	<b>135</b>
F.1. 4th IFToMM Symposium on Mechanism Design for Robotics (MEDER 2018) . . . . .	135
F.2. Congreso Nacional de Ingeniería Biomédica (SOMIB 2018) . . . . .	144
F.3. Congreso Mexicano de Robótica 2019 (XXI COMROB) . . . . .	149
<b>Referencias</b>	<b>156</b>



# Índice de figuras

1.1. Aspecto del exoesqueleto de la mano para la rehabilitación. . . . .	3
1.2. Dispositivo HANDEOX . . . . .	3
1.3. Exoesqueleto Maestro. . . . .	4
1.4. Exoesqueleto con mecanismo basado en centros remotos. . . . .	5
1.5. Sistema del exoesqueleto de mano con 5 actuadores lineales. . . . .	5
1.6. Prototipo de exoesqueleto diseñado con un sistema de captura de movimiento. . . . .	6
1.7. Dispositivo ExoKab. . . . .	6
1.8. Prototipo de exoesqueleto. . . . .	7
1.9. Prótesis desarrollada por la Universidad Johns Hopkins . . . . .	8
1.10. Problemas con la colocación de brazaletes Myo. . . . .	9
1.11. Metodología a desarrollar durante el proyecto de tesis. . . . .	12
2.1. Flexión de las tres articulaciones de los dedos. . . . .	14
2.2. Dibujo esquemático del esqueleto de la mano. . . . .	15
2.3. Seis tipos básicos de agarre propuestos por Schlesinger . . . . .	15
2.4. Ejercicios comunes en una terapia de rehabilitación. . . . .	16
2.5. Clasificación de robots rehabilitadores de mano. . . . .	17
2.6. Rehabilitación de espejo de tipo bilateral. . . . .	19
2.7. Partes de un sistema DAQ. . . . .	20
2.8. Diagrama de bloques de un sistema de control. . . . .	26
2.9. Diagrama a bloques de un controlador difuso. . . . .	27
3.1. Prototipo de exoesqueleto a rediseñar. . . . .	30
3.2. Sistema Vicon para captura de movimiento. . . . .	31
3.3. Colocación de las marcas para el sistema de captura de movimiento. . . . .	32
3.4. Vista del movimiento de las marcas con en el software Nexus. . . . .	32
3.5. Posicionamiento de las marcas en el espacio XYZ para el agarre de punta. . . . .	33
3.6. Rango de movimiento de cada falange del dedo índice. . . . .	34
3.7. Movimiento del dedo pulgar en el plano XY . . . . .	34
3.8. Vista lateral de un dedo de exoesqueleto. . . . .	35
3.9. Vista explosionada del bloque encargado del movimiento de la falange proximal. . . . .	38
3.10. Vista lateral de un dedo del exoesqueleto para dos distintos movimientos. . . . .	38

---

3.11. Vista lateral de un dedo del exoesqueleto con la medición de la punta del dedo para calcular el espacio de trabajo. . . . .	39
3.12. Espacio de trabajo obtenido de la simulación de SolidWorks . . . . .	40
3.13. Vista lateral del mecanismo interno del bloque encargado del movimiento de la falange proximal. . . . .	41
3.14. Ensamble final del prototipo. . . . .	42
3.15. Vista general del sistema completo. . . . .	43
4.1. Diagrama esquemático de las conexiones realizadas entre la tarjeta Arduino, controlador y encoder del motor. . . . .	45
4.2. Diagrama esquemático de las conexiones realizadas entre la tarjeta Arduino, controlador y encoder del motor. . . . .	47
4.3. Brazaletes MYO. . . . .	48
4.4. Prueba de funcionamiento y conexión con la interfaz de Matlab®. . . . .	48
4.5. Movimientos de la mano a desarrollar por el usuario. . . . .	49
4.6. Señales obtenidas en Matlab® durante la apertura y el cierre de la mano utilizando el brazaletes MYO. . . . .	49
4.7. Señal EMG obtenida por el Myo Armband al momento de realizar el cierre del puño. . . . .	50
4.8. Señal EMG antes y después del recorte de activación muscular. . . . .	52
4.9. Señal EMG después del procesamiento para la acción de cierre de puño. . . . .	53
4.10. Matriz de características. . . . .	56
4.11. Matriz de confusión resultante. . . . .	60
4.12. Curva ROC para el agarre de punta. . . . .	61
4.13. Curva ROC para la pinza fina. . . . .	61
4.14. Curva ROC para el cierre de puño. . . . .	62
5.1. Diagrama abloques del sistema. . . . .	64
5.2. Vista principal del software Kinovea para la obtención de rangos de movimiento en cada uno de las falanges. . . . .	64
5.3. Trayectoria de posición y velocidad de la falange proximal para el agarre de punta generada con el polinomio de Bézier. . . . .	66
5.4. PCB diseñada en Eagle. . . . .	67
5.5. Interpolación del comportamiento del motor de CD sin carga. . . . .	69
5.6. Interpolación del comportamiento del motor de CD con carga. . . . .	69
5.7. Modelo de un brazo de 2 grados de libertad. . . . .	70
5.8. Diagrama a bloques del controlador PID en Simulink. . . . .	72
5.9. Respuesta en simulación del controlador PID para un agarre de punta. . . . .	73
5.10. Función de membresía de la variable lingüística de entrada Error y Cambio. . . . .	75
5.11. Función de membresía de la salida de un controlador difuso PD. . . . .	76
5.12. Pantalla principal de la herramienta de Matlab® para la creación del sistema difuso. . . . .	77

---

5.13. Relación de variables de entrada y salida del sistema difuso. . . . .	77
5.14. Diagrama del controlador implementado en Arduino . . . . .	78
5.15. Formas de declarar una función difusa en Arduino . . . . .	80
5.16. Pantalla inicial de la Interfaz Grafica de Usuario . . . . .	83
5.17. Notificaciones posibles durante la realización de una nueva prueba. . . . .	84
5.18. Vista de la pantalla principal del GUI al detectar movimiento de agarre de punta. 85	
5.19. Notificaciones posibles durante la recalibración del sistema. . . . .	86
5.20. Pruebas al prototipo con el sistema de captura de movimiento. . . . .	86
5.21. Resultados obtenidos con el sistema de captura de movimiento para el dedo índice en el agarre de punta, para el exoesqueleto y una mano sana. . . . .	87
5.22. Resultados obtenidos con el sistema de captura de movimiento para el dedo índice en el cierre de puño y pinza fina, para el exoesqueleto y una mano sana. 88	
5.23. Vista general de una prueba realizada . . . . .	89
5.24. Posiciones del prototipo de exoesqueleto durante el desarrollo de una prueba de agarre de punta. . . . .	89



# Índice de Tablas

2.1. Diferentes demandas en la robótica de rehabilitación de mano con diferentes modalidades de entrenamiento. . . . .	18
3.1. Valores de los parámetros optimizados para el diseño del exoesqueleto. . . . .	35
3.2. Principales características comparativas de las dos versiones del exoesqueleto de mano. . . . .	36
3.3. Parámetro dimensionales de engranes. . . . .	37
3.4. Elementos de fijación para el prototipo. . . . .	41
4.1. Especificaciones generales del micromotorreductor. . . . .	46
4.2. Especificaciones generales del controlador Toshiba TB6612FNG. . . . .	46
4.3. Cantidad de muestras recolectadas por género y movimiento de la mano. . . . .	51
4.4. Parámetros del filtrado de la señal EMG. . . . .	53
4.5. Características más relevantes obtenidas por diferentes algoritmos. . . . .	59
4.6. Parámetros del clasificador KNN . . . . .	60
5.1. Rangos de movimientos para diferentes movimientos básicos de la mano. . . . .	65
5.2. Tabla comparativa de algunas tarjetas en el mercado. . . . .	66
5.3. Datos obtenidos para la caracterización de los motores de CD sin carga. . . . .	68
5.4. Términos lingüísticos implementados en el controlador difuso. . . . .	74
5.5. Reglas del controlador difuso. . . . .	76
5.6. Términos lingüísticos implementados en el controlador difuso. . . . .	81
5.7. Error obtenido por falange para cada uno de los movimientos con el uso del exoesqueleto. . . . .	88
B.1. Cantidad de muestras recolectadas por género y movimiento de la mano. . . . .	108
B.2. Cantidad de muestras recolectadas por género y movimiento de la mano. . . . .	109
B.3. Cantidad de muestras recolectadas por género y movimiento de la mano. . . . .	109
C.1. Rangos de movimiento de las falanges para cierre de puño. . . . .	112
C.2. Rangos de movimiento de las falanges para agarre de punta. . . . .	112
C.3. Rangos de movimiento de las falanges para pinza fina. . . . .	113

# Capítulo 1

## Introducción

El área de la robótica actualmente tiene una amplia gama de aplicaciones, no es nuevo que se busque el uso de máquinas para realizar actividades que auxilien al ser humano, dentro de ellas se encuentra la robótica de servicio. De acuerdo con la Federación Internacional de Robótica (IFR, por sus siglas en inglés) [1], se tiene la siguiente definición para un robot de éste tipo: *Robot que opera de forma parcial o totalmente autónoma al servicio del bienestar de los seres humanos y de equipamientos, excluyendo operaciones manufactureras*. El área médica actualmente representa uno de los focos de atención para distintas universidades, institutos y empresas privadas, ya que el desarrollo de máquinas de rehabilitación trae consigo grandes ventajas, como ayudar al terapeuta con la carga de trabajo, apoyar en movimiento repetitivos, aumentar el número de terapias y acelerar el proceso de recuperación de los pacientes, de manera segura y confiable.

Algunos de los problemas de salud que se han estudiado por las altas cifras que representan tanto a nivel global como nacional, son los Accidentes Cerebrovasculares (ACV), que son un grupo de desórdenes del corazón y de los vasos sanguíneos, [2]. En México los ACV son la sexta causa de muerte y son generadores en gran medida de discapacidad, la cual depende de: la gravedad del accidente, la región del cerebro que ha sido afectada, la extensión del daño en las células cerebrales y la rapidez con la que zonas del cerebro intactas logran suplir las tareas que antes eran realizadas por la zona lesionada. Un 50 % de las personas que sufren un ACV necesitan rehabilitación para recuperar la capacidad de los miembros afectados; una de las formas de aumentar la movilidad [3], la audición, la visión y las capacidades de comunicación en las personas con discapacidad es a través del uso de dispositivos y el desarrollo de nuevas tecnologías con las que puedan mejorar sus habilidades y, por tanto, estar más capacitados para vivir de forma autónoma y tener una mejor calidad de vida [4]. En México este proceso de rehabilitación se lleva a cabo en las instituciones de salud de cada comunidad y, por lo general, éstas se ven rebasadas en su capacidad ofreciendo a los pacientes sesiones distantes en tiempo una de las otras.

El desarrollo de dispositivos en el área de robótica de rehabilitación se ha centrado en los sobrevivientes de ACV, ya que estos pueden tener graves daños en la movilidad de sus extremidades, año con año el número de casos aumenta rebasando las capacidades de clínicas y centros de salud.

A pesar de que México es uno de los líderes en la manufactura y ensamble de aparatos

médicos a nivel mundial [5], [6], el desarrollo tecnológico en este tipo de dispositivos a nivel nacional es incipiente. Por lo tanto, el desarrollo de dispositivos robóticos para rehabilitación médica es un área de oportunidad para universidades, centros de investigación y empresas mexicanas. En los últimos años, el desarrollo de sistemas robóticos para el apoyo a la rehabilitación ha tenido un auge e incremento considerable, en el mercado se pueden encontrar opciones de este tipo de dispositivos para distintas partes del cuerpo, pero cabe mencionar que esta tecnología en México apenas se está desarrollando, siendo necesaria la importación de países como EUA, Alemania, Japón entre otros.

## 1.1. Antecedentes

La importancia de mejorar la función de la mano después de sufrir alguna lesión ha estimulado el aumento en el desarrollo de dispositivos exoesqueléticos con diferentes enfoques al problema para este miembro del cuerpo, diseñados con diferentes tipos de accionamiento, ya sea neumático [7], [8], [9], [10], hidráulico [11] o por cables [12].

Ejemplos de rehabilitadores comerciales son los desarrollados por la empresa VITIA<sup>®</sup> que cuenta con rehabilitadores para mano, brazo y pierna. Esta empresa desarrolló un guante portátil para terapia física de rehabilitación activa con un software especializado para ayudar en la terapia, permite el tratamiento de la muñeca y de los dedos de forma aislada o en conjunto [13]. La empresa JACE<sup>™</sup>Systems desarrolló el Jace H440 [14], con una interfaz amigable con el usuario que considera aspectos de velocidad y fuerza. Festo desarrolló ExoHand [15], que es un exoesqueleto de accionamiento neumático aplicado para aumentar la fuerza y resistencia del usuario. La fuerza, el ángulo y la dilatación se registran empleando sensores. El control con el uso del entorno de desarrollo CoDeSys adquiere y procesa los valores de posición y fuerza permitiendo la orientación precisa de la segunda mano.

El control de estos dispositivos exoesqueléticos, para su posterior análisis de movimientos y fuerza, se ha realizado con el uso de sensores especializados como el descrito por Loureiro y Harwin [16] donde a través de una arquitectura de control distribuido, el fisioterapeuta puede establecer fácilmente una terapia, en la cual parámetros como inicio de movimiento, fin de movimiento, tiempo para completar, modo de terapia, nivel de asistencia de robot y tipo de agarre se pueden definir y almacenar en una base de datos. Con el avance tecnológico, el área de procesamiento digital de señales biomédicas como las señales electromiográficas superficiales (EMGS) ha incrementado su aplicación en sistemas de control. Ejemplo de esto es el trabajo presentado por Irastorza et al. en [17], donde realizaron un sistema de clasificación para predecir la intención del movimiento basados en la actividad EMG de cinco músculos del brazo superior en personas sanas. La predicción continua de la intención de movimiento en tiempo real durante los movimientos funcionales, es esencial para proporcionar a los usuarios una reatrealimentación continua, donde se busca la obtención de parámetros para controlar en tiempo real dispositivos robóticos de rehabilitación.

El dispositivo desarrollado en [18],[19] es un exoesqueleto de mano para pacientes que sufrieron un accidente cerebrovascular (ver Figura 1.1). Éste resuelve problemas de movimiento y adaptabilidad a la mano del usuario con la propuesta de un nuevo mecanismo de articulación denominado eslabón indirecto, que adopta el mecanismo de piñón y cremallera simétrico

(SPRM, por sus siglas en inglés). Se instalaron 4 sensores flexiforce y se realizó el acondicionamiento de señales con una tarjeta de adquisición de datos MPS-010501 y posteriormente, a la señal obtenida se le hace un ajuste por mínimos cuadrados para su linealización.

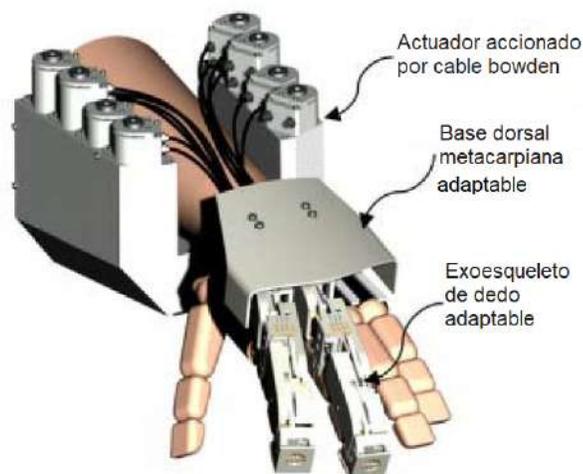


Figura 1.1: Aspecto del exoesqueleto de la mano para la rehabilitación [19].

En [20] se muestra el exoesqueleto para mano llamado HANDEOX (ver Figura 1.2), consta de cinco dedos independientes con una estructura de concha, en cada una de las falanges se integra un sensor de efecto Hall y es controlado por medio de un sistema jerárquico de movimiento independiente, implementa un regulador de lazo cerrado PID clásico con un esquema de anti windup. El dispositivo presentó problemas para la predicción de la posición de los dedos, debido a las pérdidas de fricción que se presentan por el cable Bowden con el cual se realiza el movimiento del mecanismo.

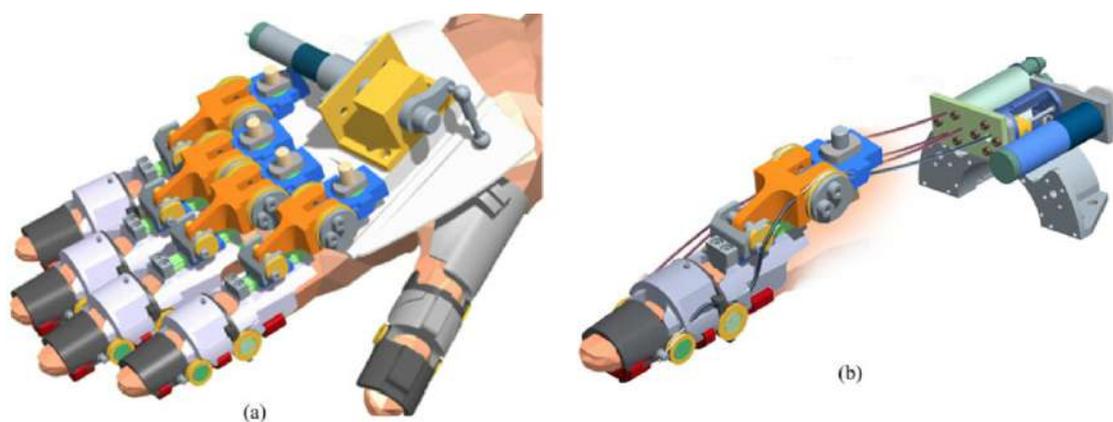


Figura 1.2: (a) Dibujo CAD del sistema HANDEOX con cinco módulos independientes para cada dedo. (b) Detalle del módulo correspondiente al dedo índice [20].

En el dispositivo llamado Maestro, exoesqueleto desarrollado en [21], se presentan dos controladores de asistencia, uno de campo de fuerza aprendida, que es una novedosa técnica

de control mediante redes neurales basada en el aprendizaje fuera de línea para una persona en específico y posteriormente se usa para generar un campo de fuerza que proporciona la asistencia necesaria para lograr el movimiento coordinado en las articulaciones de los dedos. Esta técnica aprende las necesidades de un tema específico y proporciona asistencia apropiada al sujeto, útil para una terapia donde la coordinación entre las articulaciones es importante en lugar de la precisión del movimiento. El segundo controlador desarrollado es adaptable y se ajusta a las necesidades cambiantes del sistema del exoesqueleto de los dedos, realizando la tarea con una velocidad constante, sin embargo, aplica pares incrementados en caso de movimiento restringido, que como consecuencia, es potencialmente menos seguro. El dispositivo físico se muestra en la Figura 1.3.

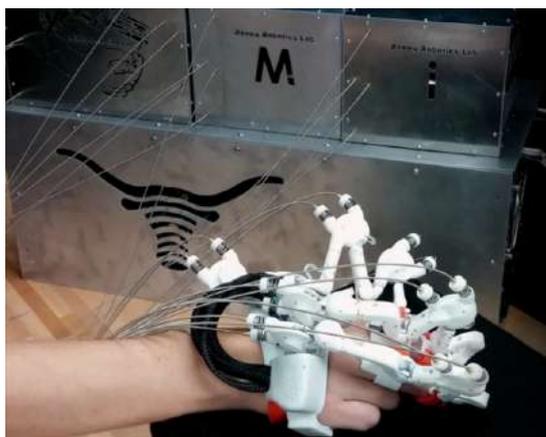


Figura 1.3: Exoesqueleto Maestro, [21].

El diseño presentado en [22] ha sido orientado a optimizar el aprovechamiento de la fuerza y minimizar el peso global del dispositivo. Capaz de ejercer una fuerza neta continua de 5N, tiene sensores en la punta de cada uno de los dedos, que se basan en el principio de detección de un resorte en forma de cruz, el algoritmo empleado es un controlador simple de fuerza de bucle cerrado con lazo de compensación, en donde la señal de realimentación del sensor se limpia con un filtro pasa bajas, posteriormente el error de fuerza impulsa una compensación a través de un circuito de control de velocidad, en la Figura 1.4 se presenta el exoesqueleto.

El dispositivo desarrollado por Ho et al. [23], es un exoesqueleto de mano para rehabilitación de personas que sufrieron un ACV, el cual detecta la intención del paciente al momento de realizar los movimientos utilizando señales electromiográficas (EMG, por sus siglas en inglés) desde el lado hemipléjico y ayuda en la apertura o cierre de la mano. El sistema consiste en un módulo de mano robótica, un controlador embebido para el exoesqueleto y monitorear las señales EMG de los principales grupos musculares. La estrategia de control utilizada en el entrenamiento fue el modo activado por EMG, utilizando un umbral del 20 % de la señal de contracción voluntaria máxima (MVC) para activar los movimientos de apertura de la mano y de cierre de la mano. Las señales EMG se miden al comienzo de cada sesión de entrenamiento, cuando el sistema se encuentra en ejecución, espera a que la señal del músculo abductor corto del pulgar, encargado de la flexión metacarpofalángica, exceda el 20 % de su valor MVC antes de iniciar la acción de cierre manual, así como el 20 % de su valor de MVC proveniente del músculo extensor para la apertura de la mano. El controlador contiene un módulo inalámbrico



Figura 1.4: Exoesqueleto con mecanismo basado en centros remotos [22].

para la comunicación con el exoesqueleto, permitiendo al terapeuta la configuración del modo de entrenamiento utilizando el sistema remoto inalámbrico, como se muestra en la Figura 1.5.

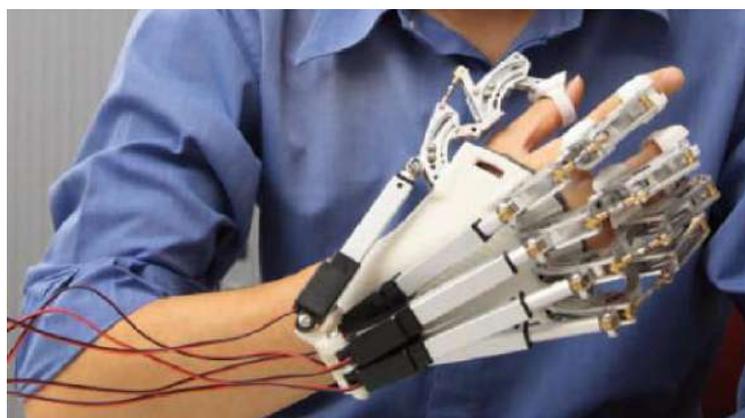


Figura 1.5: Sistema del exoesqueleto de mano con 5 actuadores lineales [23].

En [24] se propone un mecanismo para rehabilitación de mano de 1 GDL por dedo, el mecanismo es construido mediante impresión 3D utilizando acrílico (ABS), se utilizó un sistema de captura de movimiento (MoCap, por sus siglas en inglés) para el diseño de las cadenas cinemáticas del exoesqueleto, utilizando una cámara Full-HD con velocidad de movimiento lenta y un software de adquisición en 2D. El diseño mecánico del exoesqueleto permite la rotación y traslación del efector final ubicado en la falange distal, generando el movimiento del dedo, con una cinemática directa simple que sólo depende del ángulo entre la línea horizontal y la falange distal. La arquitectura de control se basa en la regulación de velocidad del motor con un PID, accionado directamente por el paciente que utiliza botones para la apertura y las fases de cierre, la cual, con un valor medio de la velocidad de apertura para cada dedo, se logra que sea simultánea la apertura en todos los dedos con un tiempo predefinido. El prototipo desarrollado se muestra en la Figura 1.6.

El Instituto Tecnológico de Orizaba en conjunto con la Universidad Nacional Autónoma de México desarrollaron el robot ExoKab [25], el cual es un dispositivo de asistencia para la rehabilitación de pacientes con discapacidades neuromusculares, este dispositivo está



Figura 1.6: Prototipo de exoesqueleto diseñado con un sistema de captura de movimiento [24].

completamente equipado con sensores de fuerza FlexiForce<sup>®</sup> colocados en la falange proximal y medial del dedo pulgar e índice del paciente, que miden cada movimiento de rotación y traslación para controlar el dispositivo durante la rehabilitación pasiva. El sistema electrónico está formado por 4 diferentes etapas, un circuito de control, de sensor de fuerza, de potencia y de comunicación inalámbrica. Los sensores se utilizan en forma resistiva para eliminar el efecto drift y no perder sensibilidad en el sensor al medir fuerza, el procesamiento de la señal analógica se realiza en una tarjeta Arduino 2560. En la Figura 1.7 se muestra el robot ExoKab.

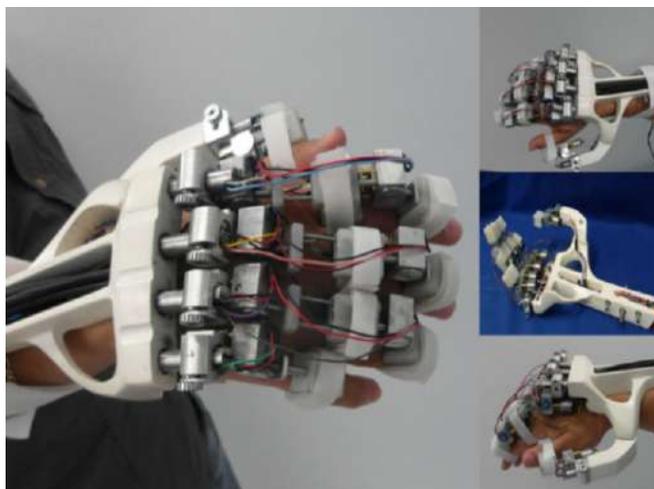


Figura 1.7: Dispositivo ExoKab [25].

En la Universidad Tecnológica de la Mixteca se cuenta con un primer prototipo de exoesqueleto para rehabilitación de mano, desarrollado por [26], el exoesqueleto tiene 2 GDL activos y 2 GDL pasivos en los dedos índice, medio y anular, 1 GDL activo y 1 GDL pasivo para el dedo pulgar y meñique, con movimiento independiente de cada uno de estos. Actualmente se tiene el prototipo mecánico del exoesqueleto con un peso de 703 gr, el cual se utilizará como base de esta tesis para realizar el proceso de control e instrumentación. El prototipo físico está hecho a base de impresión 3D, permitiendo el rediseño de algunas piezas y su fácil adaptación con las piezas existentes. El mecanismo de transmisión de potencia está basado en la utilización de juegos de engranes-cremalleras y piñón-corona. Se utilizaron motores de corriente directa (CD) como actuadores del sistema. En la Figura 1.8, se muestra el montaje real del prototipo de exoesqueleto en la mano de una persona sana [27].

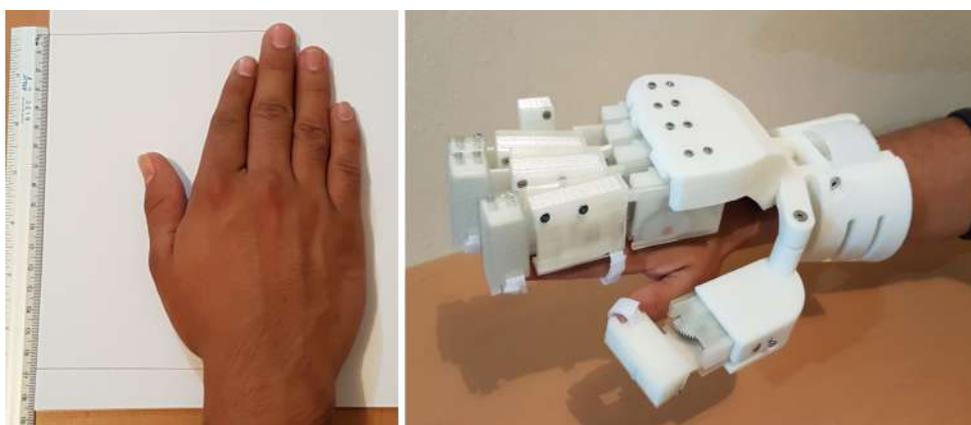


Figura 1.8: Prototipo de exoesqueleto [26].

La adquisición de señales provenientes del usuario mejora la interacción humano-robot, pueden ser a través de comando de voz, análisis y procesamiento de imágenes o por medio de bioseñales como las electromiográficas o encefalografías. Un ejemplo de la aplicación de bioseñales es dado por [28], usándolas para el despegue y el aterrizaje de un cuadricóptero. La aplicación de bioseñales en el área media ha tenido un crecimiento importante en los últimos años. En [29] se describe el proceso para la obtención y tratamiento de señales EMG con el objetivo de controlar una prótesis de tipo comercial. En el año 2015 el Laboratorio de Física Aplicada (APL, por sus siglas en inglés) de la Universidad Johns Hopkins, desarrollo un miembro protésico modular con 22 grados de libertad, la prótesis fue diseñada para un usuario con una amputación arriba del codo, para detección de la intención del paciente se utilizaron dos brazaletes de la marca Myo para la adquisición de señales electromiográficas [30]. En la Figura 1.9 se muestra la prótesis desarrollada durante la realización de pruebas de agarre.



Figura 1.9: Prótesis desarrollada por la Universidad Johns Hopkins [30].

Para la adquisición de señales electromiográficas existen varias opciones en el mercado como el dispositivo iMedilogger [31], el cual está equipado con cuatro sensores electromiográficos que están conectados a un sistema de adquisición capaz de transmitir los datos de forma inalámbrica a la computadora, haciendo del sistema no portable y con el requerimiento de un espacio de trabajo asignado. Otros dispositivos como el DataLITE [32] y el Trigno<sup>TM</sup> Quattro Sensor [33] realiza la transferencia de datos inalámbricamente en tiempo real al ordenador, con precisiones similares a la de un sistema tradicional, pero compacto en pequeñas dimensiones con la posibilidad de ser portable.

Uno de los problemas relacionados con la detección de la intención del usuario es la creación de banco de datos pequeños o enfocados a una sola persona, ocasionando que los resultados obtenidos de los algoritmos de entrenamiento y clasificación sean entrenados cada vez que sean usados en un usuario nuevo. En el trabajo desarrollado en [34] se observa las complicaciones de no generar una base de datos adecuada, obteniendo como resultados que el usuario tenga que colocar el dispositivo de adquisición de señales en la posición exacta. En la Figura 1.10 se muestra que para trabajar con los mismos algoritmos sin reentrenar en una nueva prueba, el autor tiene que dibujar la colocación exacta del brazalete en el brazo del usuario.



Figura 1.10: Problemas con la colocación de brazalete Myo.

## 1.2. Planteamiento del problema

Con base en los datos obtenidos del INEGI [35], se puede observar que hay un área muy amplia para la aplicación de dispositivos de rehabilitación para la mano. La rehabilitación por deterioro de la mano es uno de los problemas más comunes, se tienen distintos avances sobre la aplicación de sistemas inteligentes para la rehabilitación de mano, la mayoría de ellos de origen extranjero y de alto costo, que sólo se encuentran disponibles en laboratorios y no comercialmente. Países como Suiza, Holanda, Alemania y otros de primer mundo, son pioneros en la robótica de rehabilitación. Sin embargo, en México se encuentra en sus inicios y la mayoría de los desarrollos no trascienden la fase de diseño. La cantidad de dispositivos que son llevados a una etapa de instrumentación, control y pruebas tienen índices menores en contraste con la información reportada proveniente de otros países.

La instrumentación y control de los dispositivos desarrollados a nivel investigación, muestran una tendencia enfocada en la posición del mecanismo sin tomar en cuenta el tipo de terapia a realizar, implementando sistemas de control básicos que no consideran la seguridad, comodidad del paciente y la correcta realización de las tareas de rehabilitación, sin tener reportes en la literatura de estos datos. Los aportes por parte de la comunidad científica se enfocan en resultados obtenidos al tratar de cubrir el rango de movimiento de una mano sana, abarcando únicamente terapias en modo pasivo o activo haciendo que los dispositivos en cierto momento de la terapia no sean útiles, ya que sus controles desarrollados resultan inservibles para otras tareas, tal como es el caso de la rehabilitación asistida donde es necesaria la obtención de señales que permitan detectar la intención del usuario para asistir en los movimientos. La selección y aplicación de una técnica de control debe de estar en función del enfoque y tipo de rutina de rehabilitación para el cual se empleará el dispositivo, esto debe realizarse en conjunto con la implementación de un sistema de instrumentación para adquirir las variables necesarias que permitan desarrollar controles precisos y funcionales en el dispositivo.

Con esta tesis se plantea la instrumentación del prototipo físico para rehabilitación asistida de mano con pacientes que sufrieron daño motriz en la mano, disponible en la Universidad Tecnológica de la Mixteca, para la obtención de datos y parámetros que permitan valorar el desempeño del sistema y la implementación de algoritmos de control para el seguimiento de

trayectorias de movimientos básicos necesarios para realizar terapias de rehabilitación asistida utilizando sensores de electromiografía .

### **1.3. Justificación**

El desarrollo de la robótica de rehabilitación en México se encuentra en sus inicios, de los dispositivos reportados en la literatura sólo existe un dispositivo para rehabilitación de mano que ha sido llevado más allá de la fase de diseño. De manera puntual, en Oaxaca no se han desarrollado exoesqueletos enfocados a la rehabilitación y por los altos costos que estos dispositivos tienen, su adquisición por parte de clínicas y centros de salud resulta difícil. Con el desarrollo del sistema de control e instrumentación, se espera obtener datos cuantitativos de fuerza, velocidad y aceleración de la aplicación del mecanismo al paciente que permitan evaluar el progreso de la rehabilitación, teniendo como eje central al usuario para el desarrollo de un control y una interfaz usuario- robot. La selección e implementación del control estará en función de obtener la mejor respuesta del dispositivo y evitar lesionar al usuario, esto se logrará tomando en cuenta el enfoque de rehabilitación asistida, permitiendo obtener requerimientos para la selección de un control que se adapte a las necesidades del dispositivo y a sus propósitos finales, utilizando sensores de fuerza y electromiografía para obtener un control de los actuadores y la detección de los movimientos del usuario.

Con el desarrollo de este proyecto se obtendrá el conocimiento que sirva de base para la reproducción del prototipo y de sus subsistemas, mejoramiento y desarrollo en diversas aplicaciones. Al contar con un prototipo funcional en la universidad, se pueden expandir las aplicaciones y el desarrollo de proyectos que promuevan la generación de conocimiento en el área de robótica de rehabilitación.

### **1.4. Hipótesis**

Un sistema de control implementado en el exoesqueleto de rehabilitación de mano con que cuenta la UTM, permitirá realizar movimientos básicos de rehabilitación asistida así como obtener datos cuantitativos en dichos movimientos para evaluar su desempeño.

### **1.5. Objetivos**

#### **1.5.1. Objetivo general**

Instrumentar el prototipo de exoesqueleto para mano e implementar algoritmos de control que permitan ejecutar los movimientos útiles para terapias y rutinas de rehabilitación asistida.

#### **1.5.2. Objetivos específicos**

Para el logro del objetivo general se formulan los siguientes objetivos específicos:

- Rediseño y manufactura de componentes del exoesqueleto que sean necesarios para la integración del sistema.
- Validar el modelo cinemático del mecanismo del prototipo.
- Instrumentar el exoesqueleto de rehabilitación de mano utilizando sensores de fuerza y electromiografía para obtener las medidas de las variables necesarias para el sistema de control.
- Implementar algoritmos de control para el movimiento del prototipo.
- Desarrollar una interfaz humano-robot, para el control y la adquisición de datos.
- Evaluar el funcionamiento del sistema diseñado, mediante la implementación de movimientos básicos utilizados en rutinas de rehabilitación.

## 1.6. Delimitaciones

Las delimitaciones del proyecto de tesis estarán sujetas al prototipo de exoesqueleto con el que se cuenta y del cual se realizará la instrumentación y el control, así como algunas limitaciones asociadas al hardware y software que se va a implementar.

- Las pruebas se realizarán con movimientos básicos utilizados en una rutina de rehabilitación asistida.
- La adquisición y procesamiento de señales se realizará con una computadora.
- Las pruebas serán realizadas en personas sanas.

## 1.7. Metodología de desarrollo

El proyecto se enfocará en la aplicación de un modelo virtual mediante herramientas de informática de diseño mecánico avanzado (HI-DMA), desarrollo del sistema de instrumentación y control, con base en datos reportados en [36] para el exoesqueleto a trabajar. En la Figura 9 se muestran las principales etapas de la metodología a utilizar durante el desarrollo del trabajo de tesis. Se realizará el rediseño de algunas piezas del exoesqueleto tomando en consideración los sensores y cableado para la instalación de los sistemas, permitiendo la adaptación de los sistemas de control e instrumentación al prototipo existente, se realizará una validación cinemática del sistema. Para los sistemas de instrumentación y control, primero se obtendrán los requerimientos en función de la tarea que se va a reproducir, lo que permitirá realizar el diseño de cada uno de los sistemas, posteriormente la construcción de estos y finalmente la integración de todos los sistemas obteniendo como producto final del proyecto, un prototipo de exoesqueleto con un sistema de instrumentación y control que permita realizar movimientos de asistencia al usuario para posteriormente realizar terapias de rehabilitación.

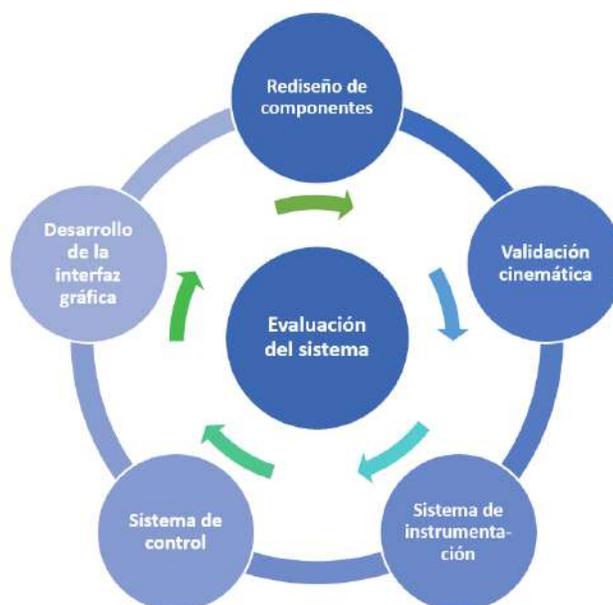


Figura 1.11: Metodología a desarrollar durante el proyecto de tesis.

Para lograr esto, durante todo el desarrollo del proyecto se estará realizando una evaluación del sistema para detectar errores o mejoras en algún momento del trabajo.

Se obtendrán los requerimientos del sistema para el diseño y construcción del sistema de instrumentación y control, que estarán en función del tipo de terapia a realizar por el exoesqueleto y de las propias restricciones del sistema.

## 1.8. Estructura de la tesis

El trabajo de tesis consta de seis capítulos los cuales se detallan a continuación. El Capítulo 1 presenta una breve introducción acerca de los sistemas de rehabilitación y las leyes de control implementadas en algunos de estos dispositivos, el planteamiento del problema, la justificación, los objetivos y la metodología que se siguió durante el desarrollo del trabajo. En el Capítulo 2 se mencionan brevemente algunos conceptos generales sobre la anatomía de la mano, conceptos de los distintos paradigmas de rehabilitación que existen y su relación con el desarrollo de dispositivos robóticos de asistencia, conceptos básicos de electromiografía y por último, se presenta de manera breve la teoría de la ley de control por lógica difusa que se desarrollará a lo largo del proyecto. En el Capítulo 3 se presentan las modificaciones que se realizaron al prototipo de exoesqueleto para la unión con el sistema de instrumentación y de control, así como la validación cinemática del prototipo. En el Capítulo 4 se muestra el proceso de diseño y de implementación del sistema de instrumentación del exoesqueleto, basado en señales electromiográficas. En el Capítulo 5 se describe el proceso de desarrollo del sistema de control del dispositivo con base en las señales obtenidas del sistema de instrumentación. En el Capítulo 6 se presentan las conclusiones y se plantean los trabajos futuros. Por último se presentan los apéndices y la bibliografía que se utilizó en este trabajo de investigación.

# Capítulo 2

## Marco teórico

### 2.1. Anatomía de la mano

La mano, el extremo distal de la extremidad superior, es un órgano de precisión muy especial tanto por sus habilidades motoras como por su discriminación sensitiva. Los movimientos realizados por el hombro, el codo y la muñeca tienen como fin proporcionar a la mano la posición, la orientación y la estabilidad necesarias para llevar a cabo una actividad [37]. Para el funcionamiento de la mano se requiere la interacción de los músculos, tendones, huesos, articulaciones y nervios. La mano proporciona una amplia gama de funciones como la manipulación, el sentido del tacto, la comunicación y la fuerza de presión [38].

#### 2.1.1. Huesos y articulaciones de la mano

El esqueleto de la mano se compone de 27 huesos, divididos en tres grupos: el carpo, los metacarpianos y las falanges. En su conjunto, adoptan la forma de arcos, dos transversales (a nivel del carpo y de los metacarpianos) y los arcos longitudinales digitales, [39]. La constitución de la mano incluye 29 articulaciones, 27 huesos y más de 30 músculos y tendones que trabajan juntos para el rango de movimiento, la percepción y la producción de fuerza [40].

Los huesos de los dedos de la mano, huesos tubulares pequeños, dispuestos uno a continuación del otro se denominan falanges. Cada dedo contiene tres falanges: primera falange o falange proximal; segunda falange o falange media y tercera falange o falange distal. Constituye una excepción el dedo pulgar, que solo tiene dos falanges, la proximal y la distal. En todos los animales este dedo presenta el desarrollo más débil, alcanzando el mayor desarrollo en los hombres [41].

La conformación ósea y de los ligamentos son los adecuados para evitar traducciones laterales tanto dorsal como palmar, pero lo suficiente para permitir la participación de los principales movimientos de la muñeca. En cada uno de los dedos, el diseño anatómico es esencialmente el mismo, con excepciones en el pulgar. Metacarpianos II al V están articulados tan estrechamente con los huesos del carpo adyacentes de la fila distal que a pesar de que son capaces de flexión y extensión, la independencia de movimiento es muy limitada. Los ejes metacarpianos son arqueados para formar la palma de la mano, y los extremos distales son casi semiesféricos para permitir la curvatura cóncava de los extremos proximales de las primeras

falanges [42].

Las articulaciones metacarpofalángica (MCF) de los cuatro dedos son articulaciones diartrodias unicondilares, permitiendo el movimiento en tres planos: flexión - extensión (plano sagital), abducción - aducción (plano frontal), y ligera pronación-supinación (plano transversal). El rango de flexión MCF desde la posición cero es aproximadamente  $90^\circ$ , pero este valor difiere entre los dedos. El meñique demuestra la mayor flexión ( $95^\circ$ ) y el índice aproximadamente  $70^\circ$ . Las articulaciones interfalángicas proximal y distal (IFP e IFD respectivamente) de los cuatro dedos son articulaciones bicondilares en bisagra. El rango de movimiento de las articulaciones MCF, IFP e IFD se describe a menudo individualmente para cada una de las tres articulaciones como se observa en la Figura 2.1, la flexión de la articulación MCF con un rango de movimiento de  $70^\circ$  a  $90^\circ$ , flexión de la articulación IFP promedio de  $100^\circ$  o más y la flexión de la articulación IFD promedio de  $90^\circ$  [43].

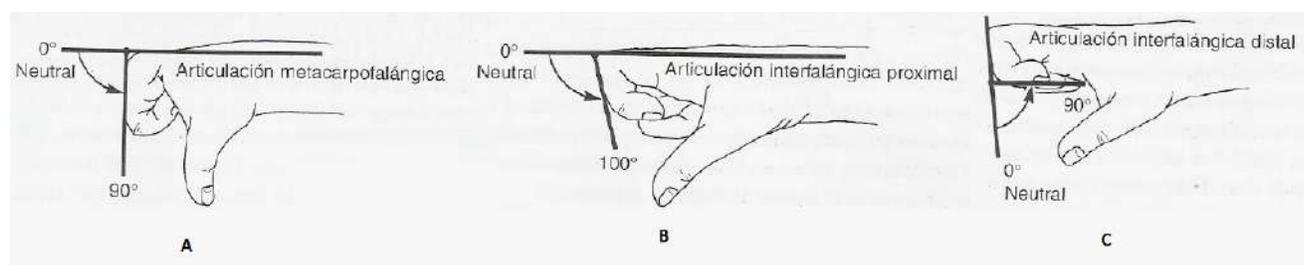


Figura 2.1: Flexión de las tres articulaciones de los dedos. A) Flexión de la articulación MCF. B) Flexión de la articulación IFP. C) Flexión de la articulación IFD [43].

Las articulaciones metacarpofalángica exhiben un patrón que se observa también en las articulaciones interfalángicas, el centro virtual de giro se encuentra aproximadamente en el centro de curvatura del extremo distal del miembro proximal. Las caras laterales de las superficies articulares son estrechas y cercanas con los ligamentos, por lo que la rotación lateral es pequeña en las articulaciones metacarpofalángicas y carente por completo en las articulaciones de las falanges. Por lo tanto, estos últimos son articulaciones típicas de las bisagras. El pulgar se diferencia de los otros dedos porque la segunda falange ha desaparecido, en segundo lugar, porque existe una mayor movilidad en la articulación carpometacarpiana, [42]. En la Figura 2.2 se muestra los huesos y ligamentos de la mano.

### 2.1.2. Tipos de agarre de la mano.

La variedad y patrones de movimientos de la mano se ven ligadas al número de músculos y articulaciones que tiene. En la determinación de la complejidad y destreza del movimiento, son las grandes áreas de la corteza cerebral las que dan la coordinación del movimiento y la sensación en la mano. En la corteza motora del área dedicada a las manos es aproximadamente igual a la superficie total dedicada a los brazos, el tronco, y las piernas. Esta circunstancia asegura una gran potencialidad para el movimiento coordinado y para el aprendizaje de nuevas actividades. Del mismo modo, las áreas sensoriales son grandes, para que determinen tales funciones avanzadas como estereognosis, la capacidad de reconocer la forma de un objeto simplemente lo sostenga en la mano. La gran sensibilidad táctil de la mano es, en gran parte

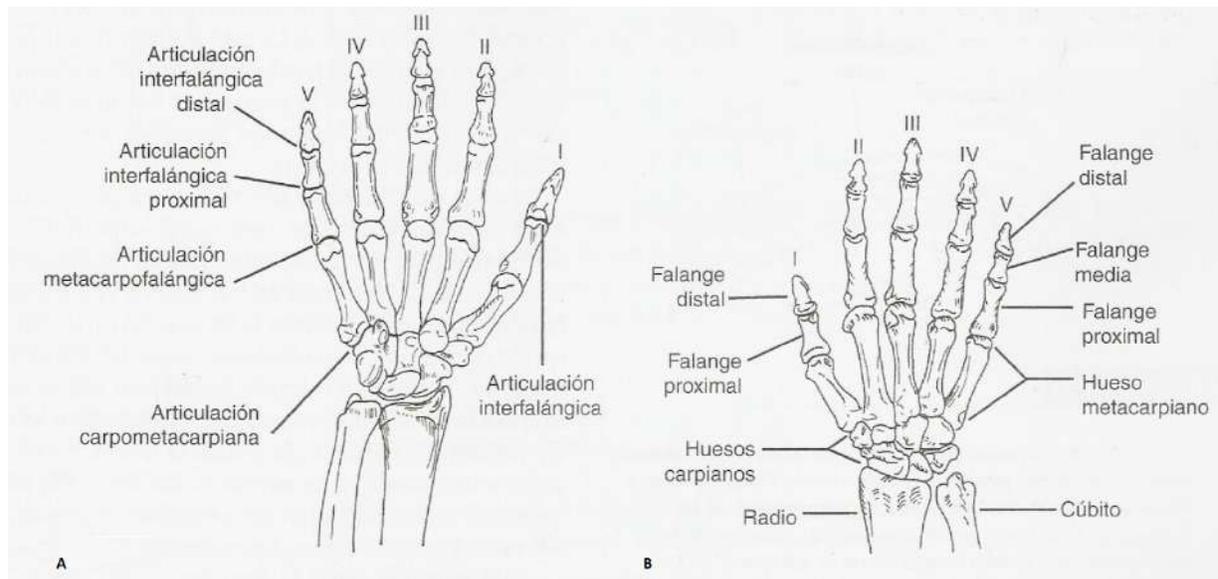


Figura 2.2: Dibujo esquemático del esqueleto de la mano. A) Vista palmar (anterior) de la mano derecha. B) Vista dorsal (posterior) de la mano derecha [43].

debido a los órganos de los sentidos en la mano [42]. Schlesinger en 1919 desarrolló una clasificación de la taxonomía para el estudio de la destreza de las manos humanas y agrupó en seis categorías, las estrategias de agarres de la mano humana: agarre cilíndrico, de punta, de gancho, de palma (palmar), esférico y de lado (lateral), como se observa en la Figura 2.3.



Figura 2.3: Seis tipos básicos de agarre propuestos por Schlesinger [42].

## 2.2. Rehabilitación

La rehabilitación tiene como objetivo incrementar y mantener el funcionamiento, la movilidad, así como evitar la deformidad y proporcionar las vías para adquirir una vida independiente para una plena integración en la sociedad. Tiene que haber un equipo multidisciplinario

con médicos, enfermeras, terapeutas, consejeros sociales y psicólogos que orienten y proporcionen un tratamiento rehabilitador personalizado. El estiramiento, la corrección quirúrgica de las deformidades de la columna y de las contracturas, la utilización de soportes en las piernas, puede mejorar o prolongar las posibilidades de caminar y el uso funcional de las extremidades. El debilitamiento de la musculatura esquelética es el principal problema de los afectados, por lo que se han realizado estudios para analizar los efectos del ejercicio en la enfermedad y se ha comprobado que en los desórdenes neuromusculares de progresión lenta, un programa de ejercicios de resistencia moderada, donde se utiliza el 30 % de la máxima fuerza isométrica, durante 12 semanas, se consigue un aumento del 4 al 20 % de la fuerza sin ningún efecto adverso. En un programa de elevada resistencia no aumentaba de manera proporcional el beneficio y en algunas personas incluso se observó un debilitamiento por el exceso [44].

Después de un daño cerebral no fatal, el cerebro tiene la capacidad de modificar funciones para compensarlo, a esta cualidad del sistema nervioso se le conoce como plasticidad cerebral, la cual es la suficiencia del sistema nervioso para minimizar los efectos de lesiones estructurales o fisiológicas a través de modificar su propia organización estructural y funcional [45]. La plasticidad cerebral está relacionada con la recuperación a largo plazo de los pacientes, mediante la reorganización cerebral que puede ser inducida por técnicas de rehabilitación, donde las neuronas sanas aprenden las funciones de las neuronas dañadas, existen estudios de neuroimagen funcional que muestran cambios evolutivos en la actividad cerebral en ambos hemisferios en pacientes que mejoran sus habilidades funcionales a través del entrenamiento [46], [47].

La evaluación del deterioro de una mano, consiste en la ejecución de tres ejercicios básicos, uno de ellos consiste en cerrar lentamente la mano hasta donde sea posible por el paciente y hacer el puño. Otra prueba es hacer que el paciente haga un pellizco de punta a punta del dedo índice y del pulgar, se trata de una presión de precisión que se usa para manipular objetos. El último movimiento consiste en una presión de pistola, donde todos los dedos deben flexionarse de manera adecuada para formar un gancho, el pulgar debe colocarse en abducción para un área contra la cual se ejerza fuerza. Dichos movimientos evidencian que todas las articulaciones se muevan y que existan todos los controles nerviosos requeridos, [48]. Los movimientos mencionados anteriormente se muestran en la Figura 2.4.

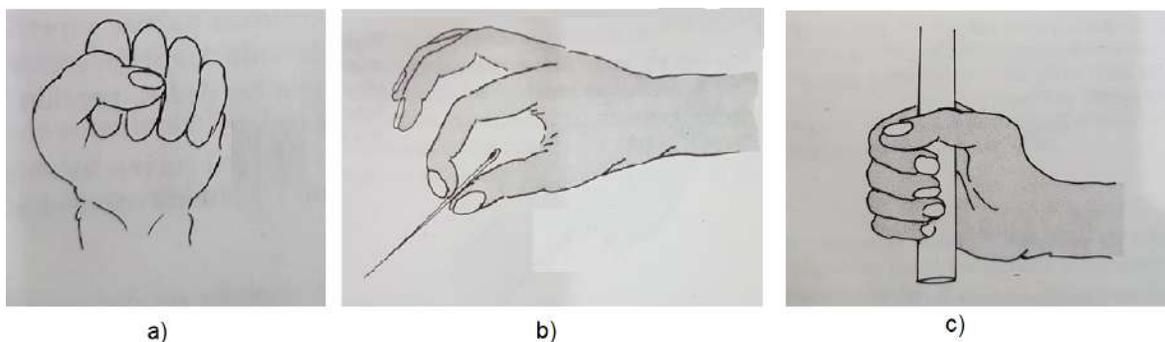


Figura 2.4: Ejercicios comunes en una terapia de rehabilitación. a) Puño. b) Oposición punta a punta. c) Presión de fuerza. [48].

### 2.2.1. Clasificación de los robots de rehabilitación.

Los robots para rehabilitación de mano se clasifican, ya sea por la cantidad de GDL, número de conexiones mecánicas, arquitectura del mecanismo o por el enfoque que el dispositivo tendrá. La clasificación basada en el enfoque y propuesta en [49], puede ser por planteamiento en el diseño mecánico o en la rehabilitación, aunque estas dependen una de la otra, un sistema de hardware depende de las capacidades básicas de la robótica de rehabilitación, considerando rangos de movimientos y retroalimentación de la información, mientras que los paradigmas de entrenamiento se encausa en la aplicación de teorías de rehabilitación específicas para el sistema. La descripción general de esta clasificación se muestra en la Figura 2.5.

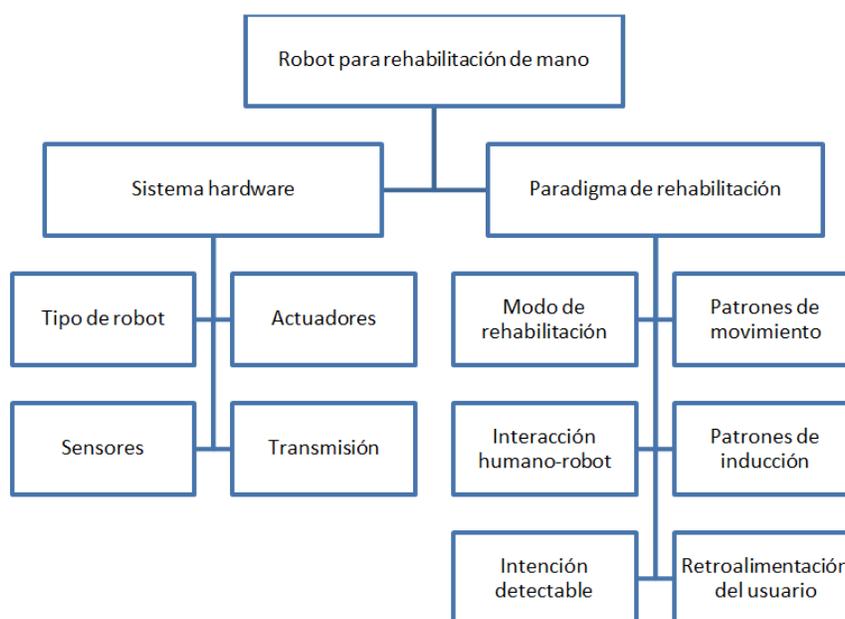


Figura 2.5: Clasificación de robots rehabilitadores de mano [49].

En el diseño enfocado en el hardware son considerados detalles mecánicos como el tipo de actuadores, sistema de transmisión y los sensores a utilizar, lo cual está relacionado con el enfoque a la rehabilitación donde para desarrollar el dispositivo es necesario considerar ambos.

### 2.2.2. Paradigma de rehabilitación

Para diseñar una mejor aplicación del sistema de robot de rehabilitación, los paradigmas de entrenamiento se clasifican según varios aspectos, donde el punto principal es la interacción humano-robot. La clasificación de las modalidades de entrenamiento se relaciona con los modos de terapia convencionales utilizados en la práctica clínica y se refiere al estado de un paciente durante la interacción y las propiedades de la fuerza aplicada a la mano por el dispositivo [50]. Esta clasificación se muestra en la Tabla 2.1, donde (+) significa la asistencia del robot durante el movimiento, (-) significa que el robot opone resistencia al movimiento y (0) significa que el robot no opone resistencia ni ayuda durante el movimiento al paciente [49].

Tabla 2.1: Diferentes demandas en la robótica de rehabilitación de mano con diferentes modalidades de entrenamiento.

Modo de terapia	Paciente		Fuerza del Robot
	Voluntad	Fuerza	
Pasiva	0	0	+ + +
Asistida	+	+	+
Activa-Asistida	++	+ / 0	+ / 0
Activa	++	+	0
Resistiva	++	++	-

En la Tabla 2.1 se muestra la modalidad de la terapia y la fuerza que realiza el paciente y el robot en cada una de estas, donde las modalidades de la terapia de acuerdo con [50] y [49] son:

- Pasiva: El robot realiza el movimiento sin tener en cuenta la actividad del sujeto, el movimiento de la mano del paciente depende totalmente de la fuerza ofrecida por el robot. De acuerdo con el concepto de movimiento pasivo continuo (CPM, por sus siglas en inglés), el entrenamiento pasivo realizado de manera constante ayuda a la recuperación motora. Estos robots generalmente están controlados por una posición para hacer un entrenamiento repetitivo, mueven la mano desde la posición inicial hasta la posición final y luego retroceden.
- Asistida: La actividad voluntaria del paciente es requerida durante todo el movimiento. Los robots pueden ayudar a proporcionar soporte de peso o fuerza con el objetivo de completar la tarea.
- Activa-Asistida: El paciente completa el movimiento con la ayuda de robots, el robot no ofrecerá fuerza a la mano hasta que el paciente no pueda moverse por sí mismo. La modalidad de entrenamiento de asistencia activa es la más utilizada en el sistema de robot de rehabilitación.
- Activa: El robot se está utilizando como un dispositivo de medición, sin proporcionar fuerza a la extremidad del sujeto, el paciente ejecuta el movimiento en su propia habilidad.
- Resistiva: El paciente completa el movimiento bajo la fuerza de resistencia ofrecida por el robot. Este tipo de entrenamiento se centra principalmente en fuerza, por lo que el robot generalmente adopta los esquemas de control de impedancia

### 2.2.3. Rehabilitación bilateral

Los humanos realizan movimientos coordinados todos los días, tanto de las extremidades superiores e inferiores en tareas como conducir un automóvil, buscar un objeto y abrir un

contenedor, pero para una persona que ha sufrido un derrame cerebral que resulta en parálisis parcial o debilidad en un lado del cuerpo o como resultado de algún accidente, las tareas sencillas de coordinación entre miembros, como lavarse, vestirse y comer, resultan difíciles de realizar, por lo que una de las técnicas de rehabilitación prominente actualmente es el entrenamiento de movimiento bilateral, la cual, es una estrategia de rehabilitación basada en la coordinación natural entre los miembros mejorando la eficiencia de los movimientos de agarre en el lado afectado [51], con cambios acompañados por una reorganización de las asignaciones cerebrales en el hemisferio afectado, esto se debe a que los movimientos voluntarios de la extremidad sana pueden facilitar los movimientos voluntarios en la extremidad que se encuentra con daño. La activación de la corteza motora primaria y del área motora suplementaria para la extremidad sana aumenta la probabilidad de contracciones musculares voluntarias en la extremidad dañada cuando se ejecutan movimientos simétricos [52].

Las tareas bimanuales requieren operar las dos manos juntas, de modo que cooperen para lograr la función deseada. Un ejemplo de la aplicación de este tipo de rehabilitación se muestra en la Figura 2.6, a la cual se le conoce como de espejo.



Figura 2.6: Rehabilitación de espejo de tipo bilateral [53].

En la literatura se ha encontrado estudios que indican que el movimiento simultáneo de ambas extremidades ayuda al sistema neuromuscular a recuperar estabilidad y mejora el uso de la extremidad dañada, incluyendo también en las tareas de agarre. En un estudio realizado en [54] con doce pacientes sobrevivientes de un accidente cerebrovascular, fueron asignados aleatoriamente a una intervención de seis sesiones que involucraba la realización de una tablilla de madera en un estante ya sea con el brazo lesionado o con los dos brazos moviéndose simultáneamente. Los pacientes que recibieron entrenamiento bilateral mostraron una disminución significativa en el tiempo para completar el movimiento y un aumento en la capacidad funcional de la extremidad afectada y por el contrario, los pacientes que recibieron entrenamiento unilateral no mostraron mejoría en la cinemática del movimiento o las medidas de la capacidad funcional.

## 2.3. Instrumentación de dispositivos

EL uso de sensores es indispensable en la automatización de industrias de proceso y manufacturados, incluida la robótica, en ingeniería experimental, en automóviles y electrodomésticos. La medida de un fenómeno físico, como la temperatura de una habitación, la intensidad de una fuente de luz o la fuerza aplicada a un objeto, comienza con un sensor. También llamado transductor, convierte un fenómeno físico en una señal eléctrica que se puede medir. Dependiendo del tipo de sensor, su salida eléctrica puede ser un voltaje, corriente, resistencia u otro atributo eléctrico que varía con el tiempo. Algunos sensores pueden requerir componentes adicionales y circuitos para producir correctamente una señal que puede ser leída con precisión y con toda seguridad por un Dispositivo de Adquisición de Datos (DAQ), [55].

### 2.3.1. Adquisición de datos

La adquisición de datos (DAQ) y el acondicionamiento de señales, cuando se mide un fenómeno físico del mundo real, es un requisito previo para procesar correctamente las señales eléctricas del sensor y mejorar la calidad general de los datos. Así como el trigo cultivado en el campo requiere mucho procesamiento antes de aparecer como bolsas de harina en el supermercado, las señales crudas deben ser limpiadas, transformadas y adecuadamente ajustadas para convertirse en una salida utilizable que humanos y máquinas puedan entender, [55].

Las soluciones con lectura y registro de datos desde la PC cada día son mas requeridas por la industria y la academia. La adquisición de datos (DAQ) es el proceso de medir con una PC un fenómeno eléctrico o físico como voltaje, corriente, temperatura, presión o sonido. Un sistema DAQ consiste de sensores, hardware de medidas DAQ y una PC con software programable. Comparados con los sistemas de medidas tradicionales, los sistemas DAQ basados en PC aprovechan la potencia del procesamiento, la productividad, la visualización y las habilidades de conectividad de las PCs estándares en la industria proporcionando una solución de medidas más potente, flexible y rentable, en la Figura 2.7 se observan las partes de un sistema de adquisición de datos, [55].



Figura 2.7: Partes de un sistema DAQ [55].

Componentes clave de medidas para un dispositivo DAQ, [55].

- Acondicionamiento de señales: Las señales de los sensores o del mundo exterior pueden ser ruidosas o demasiado peligrosas para medirse directamente. El circuito de acondicionamiento de señales manipula una señal de tal forma que es apropiado para entrada

a un ADC. Este circuito puede incluir amplificación, atenuación, filtrado y aislamiento. Algunos dispositivos DAQ incluyen acondicionamiento de señales integrado diseñado para medir tipos específicos de sensores.

- **Convertidor Analógico Digital (ADC):** Las señales analógicas de los sensores deben ser convertidas en digitales antes de ser manipuladas por el equipo digital como una PC. Un ADC es un chip que proporciona una representación digital de una señal analógica en un instante de tiempo. En la práctica, las señales analógicas varían continuamente con el tiempo y un ADC realiza muestras periódicas de la señal a una razón predefinida. Estas muestras son transferidas a una PC a través de un bus, donde la señal original es reconstruida desde las muestras en software.
- **Bus de la PC:** Los dispositivos DAQ se conectan a una PC a través de una ranura o puerto. El bus de la PC sirve como la interfaz de comunicación entre el dispositivo DAQ y la PC para pasar instrucciones y datos medidos. Los dispositivos DAQ se ofrecen en los buses de PC más comunes, incluyendo USB, PCI, PCI Express y Ethernet. Recientemente, los dispositivos DAQ han llegado a estar disponibles para 802.11 Wi-Fi para comunicación inalámbrica. Hay varios tipos de buses y cada uno de ellos ofrece ventajas para diferentes tipos de aplicaciones.

### 2.3.2. Procesamiento digital de señales

Las señales se representan matemáticamente como funciones de una o más variables independientes. La variable independiente más común es el tiempo, aunque existen señales que son funciones de dos variables independientes, como las imágenes que contienen información del color en un plano  $x, y$ . Esta información puede representarse, transformarse y manipularse mediante el procesamiento de señales, ya sea de forma analógica o digital [56].

El Procesamiento Digital de Señales (DSP, por sus siglas en inglés) es un área dentro de las ciencias de la computación, trabaja con un tipo de datos denominado como señal, estas son originadas como datos provenientes del mundo real como puede ser: vibraciones, imágenes visuales, ondas de sonido, etc. DSP se centra en el desarrollo de técnicas y algoritmos para la manipulación de este tipo de señales para diferentes fines como: reconocimiento y generación de voz, compresión de datos para almacenamiento, entre otras [57]. Este tipo de procesamiento tiene algunas ventajas sobre el Procesamiento Análogo de Señales (ASP, por sus siglas en inglés), como son [58]:

- El envejecimiento de los componentes y las derivas térmicas no afectan al resultado del proceso.
- Todos los dispositivos fabricados se comportan en forma idéntica.
- Se puede reconfigurar un dispositivo modificando los valores de algunos coeficientes.
- En el caso del procesamiento digital no existen limitaciones para trabajar con señales de baja frecuencia.

- Las señales digitales pueden almacenarse en unidades de memoria para un procesamiento fuera de línea.
- En algunos casos el DSP puede ser menos costoso que el ASP.

### 2.3.3. Filtrado de señales

Los filtros digitales son una parte muy importante de DSP, consiste en un proceso computacional en el cual una secuencia numérica de entrada se transforma en otra de salida con características predeterminadas [59]. Un filtro puede representarse de forma matemática mediante una ecuación diferencial y tienen dos usos:

- Separación de señal, necesaria cuando ha sido contaminada con interferencia, ruido u otras señales.
- Restauración de señal, se utiliza cuando una señal ha sido distorsionada de alguna manera.

Las aplicaciones de estos sistemas son numerosas y muy variadas, dependiendo de la respuesta que se desee se pueden clasificar de la siguiente forma[59]:

- Pasa bajas, dejan pasar las frecuencias que están por debajo de una determinada frecuencia
- Pasa altas, dejan pasar las frecuencias que están por encima de una determinada frecuencia.
- Pasa bandas, dejan pasar las frecuencias que están situadas en una determinada banda de frecuencia, es decir, entre dos determinadas frecuencias.
- Rechaza bandas, dejan pasar todas las frecuencias excepto las que están situadas en una determinada banda de frecuencia.

Un filtro digital es un sistema lineal e invariante en el tiempo (LTI) que modifica el espectro en frecuencia de la señal de entrada y pueden clasificarse como [57], [60]

- Filtro de respuesta infinita al impulso (IIR por sus siglas en inglés) presentan recursividad, la señal de salida del filtro se reinyecta a la entrada del mismo, constituyendo un circuito recursivo, lo que permite implementar filtros con respuesta más compleja y con menos datos.
- Filtros de respuesta finita al impulso (FIR por sus siglas en inglés), una muestra de la salida se puede definir como una combinación lineal de muestras de la entrada presentes y pasadas.

## 2.4. Electromiografía

Una de las principales técnicas para adquirir información del cuerpo es la electromiografía. La Electromiografía (EMG) es una técnica que permite registrar los potenciales eléctricos desarrollados a los músculos como respuesta a un estímulo eléctrico o también conocido como potencial de acción, que les llega a través de los nervios motores. Este potencial de acción puede ser generado voluntariamente por el sujeto o bien, por un electroestimulador nervioso [61]. Se pueden distinguir dos tipos principales de EMG:

- ✓ Clínica: También llamada diagnóstica, que estudia las características del potencial de la unidad motora, de interés especialmente en el diagnóstico de la patología neuromuscular, [62].
- ✓ Cinesiológica: Estudia la función muscular y la coordinación, con ella se puede analizar el tiempo de activación de los músculos durante la tarea estudiada, es decir, cuándo comienzan a actuar y cuándo dejan de hacerlo, la actividad que se produce durante su actuación (patrón de actividad muscular durante dicha tarea) y cuándo se produce el pico máximo de actividad, [62].

Existen dos técnicas principales para la adquisición de los potenciales generados por las unidades motoras, estas técnicas son:

- Electromiografía invasiva: Para medir los potenciales generados por las unidades motoras, se usan electrodos de aguja de metal, la cual es insertada en el músculo directamente y la amplitud de la señal registrada por los electrodos de aguja depende del área de registro que ocupan, así como también de la distancia del electrodo a la fuente de la señal, [63].
- Electromiografía de superficie: Los electrodos son colocados directamente sobre la piel del músculo del cual se quiere obtener información, utilizada para el estudio del comportamiento promedio de la actividad eléctrica de un músculo o grupo de músculos, [64].

La EMG cinesiológica es el único medio de definir con fiabilidad la actividad muscular. Permite evaluar el papel de los músculos durante posturas o movimientos por lo que es de utilidad en todas las disciplinas relacionadas con el movimiento. Entre ellas, se encuentra la rehabilitación, la ortopedia, el estudio de las circunstancias de trabajo y ergonomía y el deporte, en estas áreas, la EMG se utiliza principalmente para conocer la actividad de músculos individuales y su contribución al gesto, muchas veces alterada o con patrones de coordinación patológicos y asociados con reacciones a la fatiga.

## 2.5. Aprendizaje automático

Desde la creación de la primera computadora en el siglo XX, dio origen a una nueva era conocida como revolución informática, caracterizada por la expansión y acumulación de datos

dando paso al desarrollo de nuevas tecnologías. En la actualidad el uso de computadoras en actividades en la vida diaria y procesos industriales ha creado nuevos retos para el desarrollo de sistemas de procesamiento. Uno de estos retos es crear sistemas inteligentes donde a partir de la experiencia, los sistemas informáticos mejoren de forma automática sin necesidad de programar el nuevo conocimiento adquirido. A esta rama se le llama aprendizaje automático con una gran cantidad de aplicaciones en la actualidad como la detección de transacciones fraudulentas con tarjetas de crédito, a sistemas de filtrado de información que aprenden las preferencias de lectura de los usuarios, hasta vehículos autónomos que aprenden a conducir en autopistas públicas [65].

De acuerdo con [66], el aprendizaje automático es “Un programa de computadora que aprende del aprendizaje  $E$  respecto a alguna clase de tarea  $T$  y una medida de desempeño  $D$ , si su desempeño en la tarea  $T$ , medido a través de  $D$ , aumenta con la experiencia  $E$ ”. Donde es necesario identificar tres parámetros importantes para plantear de manera efectiva un problema de aprendizaje. La solución de estos problemas de aprendizaje se puede plantear desde diferentes perspectivas, entre las que se encuentran:

- Aprendizaje supervisado  
En este tipo de aprendizaje los algoritmos requieren el suministro de un conjunto de ejemplos llamados muestras de capacitación que son datos pares de entrada-salida con un conjunto de atributos y una etiqueta asociada correspondiente a alguna propiedad asociada o importante relativa a ella. La idea fundamental de este aprendizaje es que el algoritmo genere predicciones precisas de las etiquetas de futuros datos de entrada [67].
- Aprendizaje no supervisado  
En este tipo, los ejemplos no tienen etiquetas, el aprendizaje se basa en el razonamiento que si existe alguna estructura en los objetos de entrenamiento, es posible aprovechar esta redundancia y encontrar una breve descripción de los datos. Una de las formas más generales de representar datos es especificar una similitud entre cualquier par de objetos ya que si dos objetos comparten mucha estructura es posible deducir que son del mismo tipo [67].

### 2.5.1. Algoritmos de clasificación

Uno de los problemas más importante del aprendizaje automático es la clasificación, donde a partir de objetos de clasificación descritos por un vector de valores y una etiqueta de clase, se crea un modelo matemático capaz de predecir las etiquetas de cada clase de nuevos objetos que necesitan ser clasificados. En la actualidad existen variedad de soluciones para el problema de clasificación como son: redes bayesianas, árboles de decisión, redes neuronales, máquinas de soporte vectorial, algoritmos basados en instancias, algoritmos de votación, clasificadores basados en patrones, clasificadores basados en conjuntos de representaciones.

Los algoritmos basados en instancias son ampliamente usados para solucionar problemas de clasificación de manera concisa, estos algoritmos son conocidos como lazy learning o memory learning [68]. La idea general de su funcionamiento es a través de almacenar objetos de entrenamiento que cuando es necesario clasificar un nuevo objeto, se recuperan los objetos de entrenamiento más cercanos, posteriormente se extrae los que son más parecidos al nuevo

objeto para poder asignar una clase y clasificarlo. Entre de estos algoritmos se encuentra la clasificación por K-vecinos más cercanos (KNN, por sus siglas en inglés K nearest Neighbors) [65].

### 2.5.1.1. Algoritmo KNN

La idea de este algoritmo es almacenar el conjunto de entrenamiento, en el momento de realizar la clasificación de una nueva instancia asumiendo que corresponden a puntos en el espacio dimensional  $R_n$ , se busca en los ejemplos almacenados casos similares y se asigna la clase más probable en éstos. Los vecinos más cercanos a una instancia se definen en términos de la distancia euclidiana estándar, es decir, permita que una instancia arbitraria  $x$  sea descrita por el vector de características (2.1).

$$[a_1(x), a_2(x), a_n(x)] \quad (2.1)$$

donde  $a_r(x)$  denota el valor del atributo  $r_{th}$  de la instancia  $x$ . Entonces la distancia entre dos instancias  $x_i$  y  $x_j$  se define como  $d(x_i, x_j)$ , donde

$$d(x_i, x_j) = \sqrt{\sum_n^{r=1} (a_r(x_i) - a_r(x_j))^2} \quad (2.2)$$

Para evitar el ruido que se puede presentar en los K-vecinos más cercanos, al momento de clasificar nuevos objetos es necesario aumentar el número de vecinos (K), para asociar de manera más rápida el nuevo dato a los elementos más representativos o con mayor presencia en el espacio. A continuación se describe el algoritmo básico para el entrenamiento y clasificación por KNN [66].

Entrenamiento:

- Para cada ejemplo  $(x, f(x))$ , se agrega a la lista de entrenamiento

Clasificación:

- Dada una instancia de prueba  $x_i$  a ser clasificada
- Sean  $x_i \dots x_k$  los ejemplos de la lista de entrenamiento que son más cercanos a  $x_q$
- Regresar el argumento máximo de la evaluación

$$\hat{f}(x_q) \leftarrow \operatorname{argmax}_{u \in V} \sum_k^{i=1} \delta(u, f(x_i))$$

donde  $\delta(a, b) = 1$  si  $a = b$  y  $\delta(a, b) = 0$  en otro caso.

## 2.5.2. Lógica difusa

La lógica difusa es una extensión de la lógica tradicional, que permite reproducir los patrones del pensamiento humano y representar de manera matemática la incertidumbre, esta idea fue introducida por Zadeh en el año de 1965 cuando planteó la teoría de conjuntos difusos, mostrando la lógica difusa como una forma de procesamiento de información en la que los elementos o datos tienen un grado de pertenencia parcial a conjuntos [69]. El aporte de Zadeh tuvo una gran influencia en el pensamiento de la incertidumbre, ya que se desafía los fundamentos de la lógica clásica binaria, influencia que continúa hasta nuestros días donde los conceptos de conjuntos difusos y lógica difusa se utilizan en los sistemas inteligentes, constituyendo actualmente una de las técnicas de la Inteligencia Artificial más utilizadas en aplicaciones prácticas, dentro del ámbito industrial y empresarial, en tareas como clasificación, reconocimiento de patrones o control.

## 2.6. Sistemas de control

Los sistemas de control son objeto de estudio de la ingeniería de control, donde, a través de modelos matemáticos, se analizan, estiman previsiones y se verifica el funcionamiento de una planta o un proceso, con el que las salidas cumplan los requerimientos deseados. Los modelos matemáticos generados responden a ecuaciones diferenciales lineales y no lineales; para facilitar su optimización se recurre a software matemático, técnicas de linealización y metodologías como la sustitución de Laplace o sistemas PID (Proporcionales, Integrales, Derivativos) [70].

Un controlador automático compara el valor real de la salida de una planta con la entrada deseada, determina la desviación y produce una señal de control que reduce la desviación a cero o a un valor pequeño. La manera en la cual el controlador automático produce la señal de control se denomina acción de control. La Figura 2.8 es un diagrama de bloques de un sistema de control que consiste en un controlador automático, un actuador, una planta y un sensor (elemento de medición) [71].

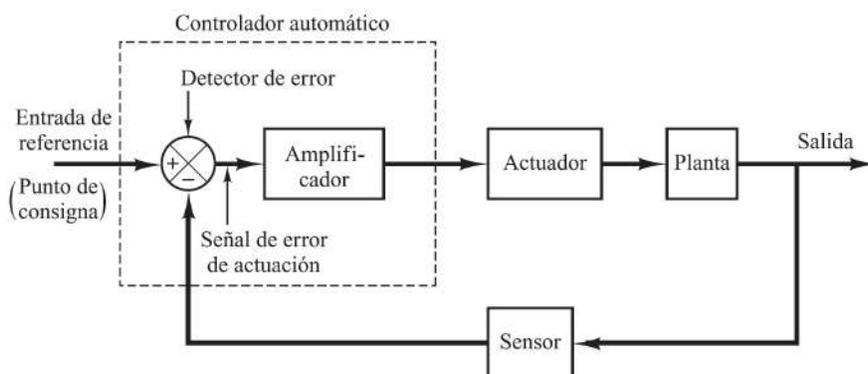


Figura 2.8: Diagrama de bloques de un sistema de control [71].

### 2.6.1. Control basado en lógica difusa

La aplicación de la lógica difusa en el diseño e implementación de sistemas de control es una de las áreas de mayor aplicación, ya que el uso de sistemas difusos brinda diferentes ventajas en su aplicación, el principal beneficio de la teoría de sistemas difusos es aproximar el comportamiento del sistema donde no existen funciones analíticas o relaciones numéricas, teniendo la capacidad de comprender sistemas relacionados con los sistemas biológicos, médicos, sociales, económicos o políticos. Por otra parte, los sistemas difusos pueden ser útiles para evaluar algunos sistemas no complejos, donde una solución aproximada y rápida pueda ser útil con el fin de ahorrar recursos en el sistema [72].

La tarea de modelar y simular sistemas complejos del mundo real para el desarrollo de sistemas de control ha motivado la investigación y aplicación de controles difusos, ya que incluso cuando es posible deducir un modelo relativamente preciso de un sistema dinámico, a menudo es demasiado complejo para usarlo en la aplicación de controladores, especialmente para muchos procedimientos de diseño de control convencionales, que requieren supuestos restrictivos para la planta, como la linealidad. En la práctica los controladores convencionales a menudo se plantean a través de modelos simples de la planta que satisfacen los supuestos necesarios, y a través del ajuste de controladores lineales o no lineales relativamente simples, donde la heurística se vuelve parte de este proceso para el ajuste de ganancias del controlador. Es en este punto donde el control difuso proporciona una metodología formal para representar, manipular e implementar el conocimiento heurístico de un ser humano sobre cómo controlar un sistema [73].

El diagrama de bloques del controlador difuso se muestra en la Figura 2.9, en el esquema se observa un controlador difuso integrado en un sistema de control de circuito cerrado. Las salidas de la planta se indican con  $y(t)$ , sus entradas se indican con  $u(t)$ , y la entrada de referencia al controlador difuso se denota como  $r(t)$ .

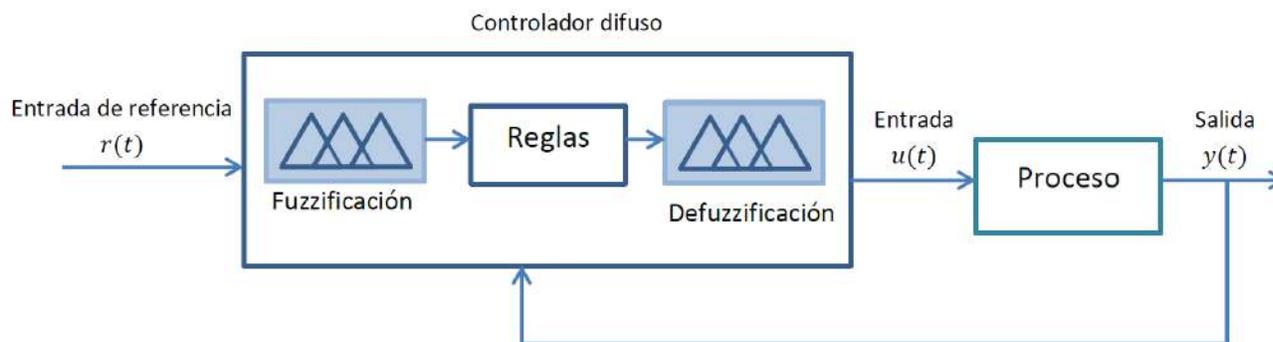


Figura 2.9: Diagrama a bloques de un controlador difuso.

El controlador difuso se compone de los siguientes elementos:

- Una interfaz de fuzzificación, que convierte las entradas del controlador en información que el mecanismo de inferencia puede usar fácilmente para activar y aplicar reglas. En esta etapa se toman los datos de entrada y se determina el grado en que pertenecen a cada uno de los conjuntos borrosos a través de las funciones de pertenencia [74].

- Una base de reglas: Un conjunto de reglas *SI-ENTONCES*, que contiene una cuantificación lógica difusa de la descripción lingüística del experto de cómo lograr un buen control. El formato de estas reglas es:  $SI U_1 \text{ es } A_1 Y U_2 \text{ es } A_2 Y \dots Y U_n \text{ es } A_n \text{ ENTONCES } v \text{ es } B$  donde los  $u_i$  y  $v$  son variables lingüísticas, y los  $A_i$  y  $B$  representan los valores lingüísticos que dichas variables pueden asumir. Cada una de las reglas tiene un lado izquierdo (LI) formado por uno o varios antecedentes y un lado derecho (LD) (consecuente), por ejemplo "SI la temperatura es alta y la presión es media ENTONCES bajar un poco el caudal de combustible" [74].
- Una interfaz de defuzzificación, que convierte las conclusiones del mecanismo de inferencia en entradas reales para el proceso. La selección del método de defuzzificación puede jugar un papel decisivo en la síntesis de modelos difusos para muchas áreas de aplicación. Particularmente dentro del área de control difuso, su influencia puede ser determinante en el comportamiento y la robustez del controlador [74].

La base de la lógica difusa se basa en hacer las fronteras que existen entre conjuntos de forma difusa, es decir, catalogar las entradas en conjuntos difusos que se superponen unos a otros. Por ejemplo, para la temperatura del agua la lógica booleana establece tres conjuntos: "Fría", "Templada" y "Caliente", que se encuentran definidos mediante rangos de valores. El agua se encuentra "Fría", en un rango de temperatura de  $0^\circ$ - $24.9^\circ$ , "Templada" de  $25^\circ$ - $34.9^\circ$  y "caliente" de  $35^\circ$ - $50^\circ$ , pero a una temperatura de  $24.9^\circ$ , el valor de esta se vuelven indistinguible para el ser humano y se puede decir que el agua tiene poco de "Fría" y mucho de "Templada", es aquí donde las fronteras desaparecen permitiendo sobreponerse a las incertidumbres de cualquier sistema físico y obtener un comportamiento inteligente. Por lo tanto, el primer paso en la lógica difusa consiste en catalogar la entrada en una serie de etiquetas lingüísticas como "Fría", "Templada" mediante funciones de pertenencia [75]. Así que los pasos que deben seguirse para el diseño de un control difuso son los siguientes [69].

- Identificar las variables (entradas, estados y salidas) de la planta.
- Particionar el universo de discurso o el intervalo que abarca cada variable en una serie de subconjuntos difusos, asignando a cada uno una etiqueta lingüística.
- Asignar las relaciones difusas entre los subconjuntos difusos de las entradas o de los estados, por una parte, y los subconjuntos difusos de las salidas, por otra, formando así la base de reglas.
- Elegir los factores de escala apropiados para las variables de entrada y salida para normalizar las variables al intervalo  $[0, 1]$  o  $[-1, 1]$ .
- Fuzzificar las entradas al controlador.
- Inferir el resultado de cada regla.
- Aplicar defuzzificación para formar una salida nítida.

# Capítulo 3

## Rediseño de prototipo y validación cinemática

En este capítulo se describe el proceso de rediseño del exoesqueleto presentado en [26], ya que esta propuesta presenta desgaste en las piezas durante el funcionamiento, las dimensiones finales del prototipo lo hacen lucir poco estético y grande en comparación con otros dispositivos reportados en la literatura. La ubicación del dedo pulgar fue obtenida con base en mediciones realizadas a manos sanas, pero al momento de realizar el ajuste del prototipo a la mano, los puntos de rotación para el cual fue diseñado no coinciden. Los cambios realizados al modelo anterior están relacionados con la geometría de cada una de las piezas, para aumentar su resistencia y su funcionalidad, conservando el mecanismo de accionamiento principal a través de engrane-cremallera. Para corroborar la posición de los motores en relación con los centros de rotación de los dedos, se usó un sistema de captura de movimiento. Se muestra el proceso de maquinado y los materiales utilizados para cada una de las piezas del sistema mecánico, obteniendo como resultado una nueva versión del prototipo del exoesqueleto, donde las consideraciones cinemáticas permanecen sin alterarse en comparación con el modelo anterior. La mayor parte de este prototipo fue elaborado a base de impresión 3D por deposición fundida de material (FDM), permitiendo la reproducción y rediseño de componentes para futuros usuarios o investigación.

### 3.1. Obtención de requerimientos para el nuevo diseño

En la Figura 3.1 a) se muestra la vista renderizada del prototipo de exoesqueleto a rediseñar y la construcción de éste en la Figura 3.1 b). Las dimensiones de este exoesqueleto lo hacen lucir grande en comparación con el tamaño de la mano, cada uno de los bloques que se encarga del movimiento de las falanges tiene una altura de 45mm, es posible reducir estas medidas si se logran encontrar relaciones de engranes que satisfagan las relaciones entre el centro de rotación de la falange y el dedo. Este diseño no considera los encoder que a instalar en cada uno de los motores, por lo que es necesario cambiar algunas geometrías de las bases principales, así como del soporte de los dedos que se coloca sobre la mano para tener salidas de los cables provenientes del circuito de adquisición de datos.

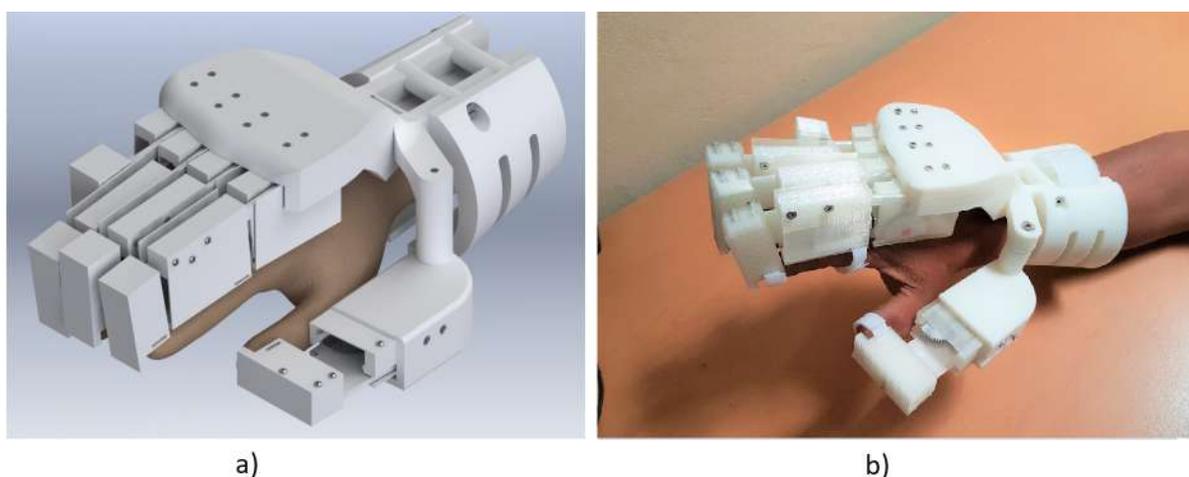


Figura 3.1: Prototipo de exoesqueleto a rediseñar. a) Simulación en Solidworks . b)Ensamble final del prototipo [26].

El sistema para ajustar el exoesqueleto al largo de los dedos no es totalmente eficiente y este en algún momento llega a afectar el movimiento de la falange, es preciso cambiar la forma en que los dos bloques se acoplan. La manera en que los bloques de movimiento se adhieren al dedo se realiza con una tira de tela velcro, este material resulta incómodo para el usuario en un uso prolongado con la piel, este material es sustituido por tiras elásticas que se vean discretas y eviten el deslizamiento del mecanismo al colocarlo. Con lo anterior se concluye que es indispensable tener en cuenta como puntos claves para el rediseño lo siguiente:

- Encontrar nuevas relaciones entre los juegos de engranes para reducir el tamaño del bloque que los contiene.
- Considerar las dimensiones de los componentes electrónicos.
- Colocar cubiertas para evitar el contacto del usuario con el mecanismo principal.
- Cambiar la forma de sujetar el mecanismo al dedo.
- Mejorar la forma en que los componentes se ajustan al largo de los dedos.

## 3.2. Especificaciones de diseño del pulgar

Para lograr los movimientos de agarre y sujeción de la mano, el pulgar es un factor clave en esta tarea. Los movimientos de abducción y flexión del pulgar son de vital importancia para lograr de forma correcta los movimientos naturales de una mano sana [76]. En la actualidad existen diversos estudios reportados en la literatura con información anatómica y el papel dinámico que el pulgar tiene durante el desarrollo de movimientos con la mano[43], [77]. Con la información obtenida de las pruebas realizadas con captura de movimiento para los agarres que se desea que el mecanismo realice y su comparación con información de la literatura, se realizan las modificaciones del mecanismo para controlar el movimiento del pulgar.

### 3.2.1. Obtención de trayectoria del pulgar con captura de movimiento

De acuerdo con [78], la captura de movimiento es un proceso en el que se utiliza un equipo capaz de capturar y determinar la ubicación de puntos en el cuerpo de un usuario en un espacio de tiempo determinado. Los puntos que detecta la cámara son marcas reflexivas que posteriormente son procesadas y es posible obtener información cinemática de ellas. Para este trabajo se utilizó el sistema Vicon<sup>TM</sup> que se encuentra en Universidad Estatal de Idaho (ISU, por sus siglas en inglés). El sistema Vicon<sup>TM</sup> cuenta con 8 cámaras infrarojas llamadas Bonita, que están conectadas directamente al sistema mediante cables LAN, y éste a su vez con la computadora. El software para realizar la captura de movimiento y la recolección de datos se llama Nexus<sup>®</sup>. Los datos son exportados fuera del software como un archivo de Excel que posteriormente puede ser manipulado por cualquier otro software numérico. El sistema se aprecia en la Figura 3.2.



Figura 3.2: Sistema Vicon para captura de movimiento.

Se realizaron distintas pruebas para cada uno de los movimientos que se desea que el exoesqueleto realice. En la Figura 3.3 se muestran la posición de las marcas en la mano para dos diferentes movimientos. Las marcas son unas pequeñas esferas que se colocaron en cada una de las articulaciones interfalángica distal, interfalángica proximal, metacarpofalángica y carpometacarpiana del dedo índice. Para el pulgar las marcas se colocaron en la articulación interfalángica distal, metacarpofalángica y carpometacarpiana.

Los movimientos para los que se realizaron las pruebas fueron cierre de puño, agarre de punta y pinza fina. Para la obtención correcta de datos es indispensable hacer una buena calibración, ya que permitirá obtener un menor error a nivel global del sistema. Hay que considerar la posición de las cámaras para no perder de vista las marcas, ya que, si no se alinean correctamente se obtienen errores en las mediciones. En el software es posible apreciar en tiempo real el seguimiento de las marcas que el usuario porta. En la Figura 3.4 se muestra una vista general del software Nexus al momento de realizar el agarre de punta. Cuando se esté creando la sesión para realizar una nueva prueba, hay que tener en cuenta como es el

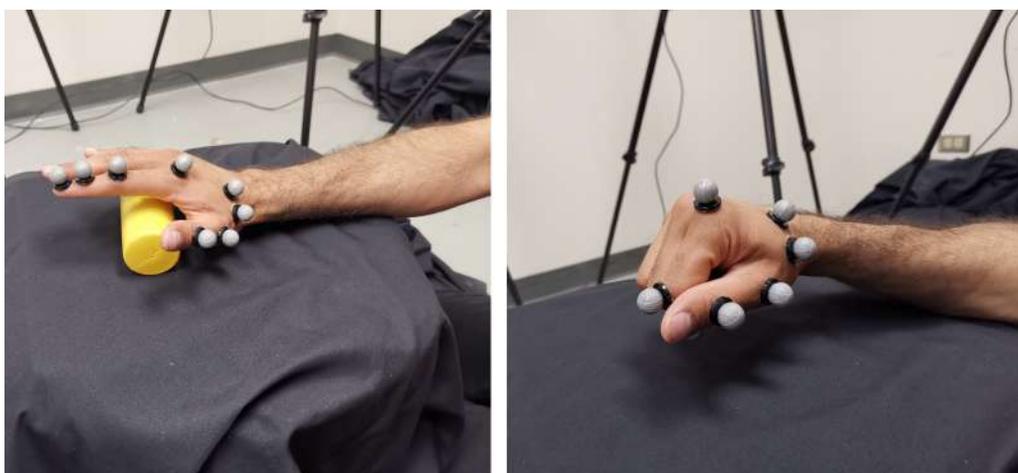


Figura 3.3: Colocación de las marcas para el sistema de captura de movimiento. a) Agarre de punta. b) Cierre de puño.

movimiento de las cadenas cinemáticas que están entre las marcas, en el software hacer la unión de ellas mediante eslabones permite seguir sus trayectorias.

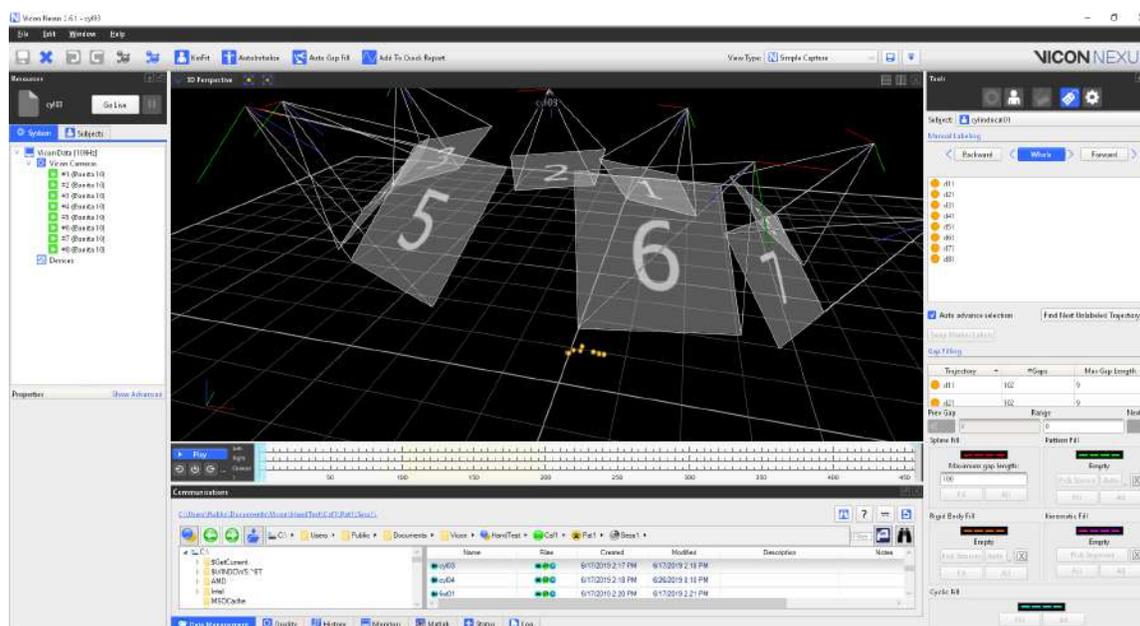


Figura 3.4: Vista del movimiento de las marcas con en el software Nexus.

Después de unir las cadenas cinemáticas es posible realizar una prueba sin perder la relación entre las marcas, éstas quedan almacenadas dentro del software y son accesibles para su exportación o su análisis dentro del mismo entorno.

### 3.2.2. Análisis de datos y obtención de requerimientos de diseño

Para cada prueba se generan distintos archivos exportables a otros sistemas, entre ellos se encuentran archivos de tipos Excel. Para cada una de las pruebas se obtuvo un archivo diferente que fue importado a Matlab<sup>®</sup> para su análisis y la obtención de movimientos angulares de cada una de las marcas, que contiene información en los ejes XYZ para cada uno de los lapsos de tiempo en que la cámara realizó una captura, con información para reconstruir y describir la trayectoria de cada una de las marcas. En la Figura 3.5 se muestra la posición de las diferentes marcas en el espacio XYZ para el archivo correspondiente al movimiento del cierre del puño. En este punto hay que tener en cuenta que es necesario considerar una de las marcas como fija, que sirve como pivote y con la cual serán relacionados todos los movimientos, así como la perspectiva de las cámaras y el error global que estas infieren en las mediciones.

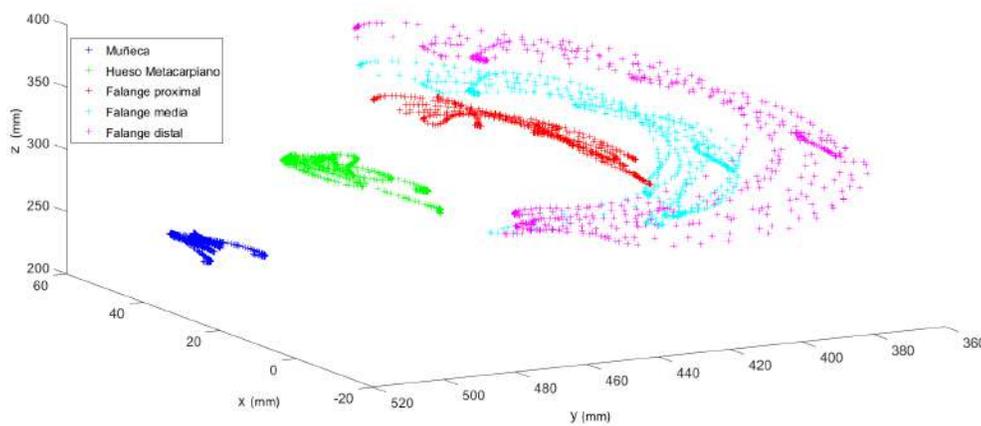


Figura 3.5: Posicionamiento de las marcas en el espacio XYZ para el agarre de punta.

Para realizar el análisis del desplazamiento angular de cada marca y obtener los rangos de movimiento para el dedo pulgar y para el dedo índice, se aplican matrices de rotación y traslación en el conjunto de puntos. Para realizarlo, se consideró como punto fijo a la articulación carpometacarpiana, que sirve de base para estabilizar las marcas correspondientes a las articulaciones. Las matrices de rotación ocupadas están descritas por las ecuaciones (3.1),(3.2),(3.3).

$$R_x(\theta) = \begin{pmatrix} 1 & 0 & 0 \\ 0 & \cos\theta & -\text{sen}\theta \\ 0 & \text{sen}\theta & \cos\theta \end{pmatrix} \quad (3.1)$$

$$R_y(\theta) = \begin{pmatrix} \cos\theta & 0 & \text{sen}\theta \\ 0 & 1 & 0 \\ -\text{sen}\theta & 0 & \cos\theta \end{pmatrix} \quad (3.2)$$

$$R_z(\theta) = \begin{pmatrix} \cos\theta & -\text{sen}\theta & 0 \\ \text{sen}\theta & \cos\theta & 0 \\ 0 & 0 & 1 \end{pmatrix} \quad (3.3)$$

Para corroborar que las rotaciones fueron correctas, se obtuvieron los valores angulares para cada falange del dedo índice durante el movimiento de agarre de punta, los resultados fueron comparados con los reportados en [79], estos son obtenidos a partir de la realización de varias pruebas de agarre a diferentes objetos cilíndricos utilizando un guante equipado con 10 sensores de presión. El estudio se realizó para observar la movilidad del dedo índice y los rangos de movimiento que tiene, también se reportan los rangos de movimiento para cada falange de los cinco dedos de la mano. En la Figura 3.6 se muestra en el plano XZ las trayectorias que cada una de las falanges del dedo índice siguen.

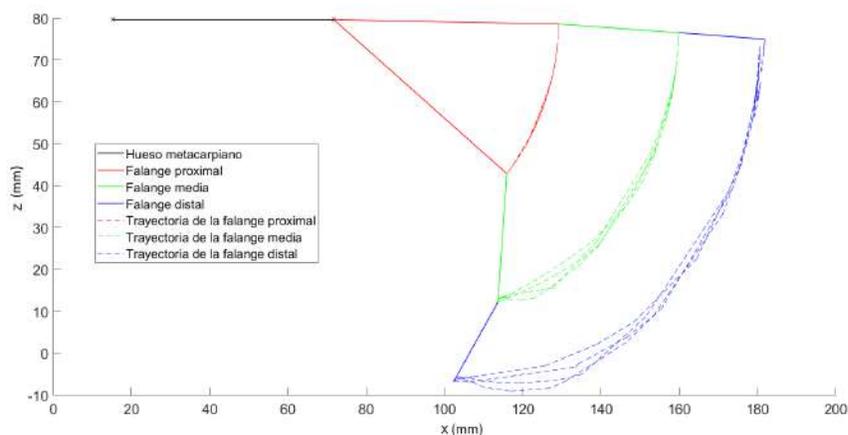


Figura 3.6: Rango de movimiento de cada falange del dedo índice.

Los resultados obtenidos con el sistema de captura de movimiento se encuentran en el rango de los datos obtenidos [79]. La comparación de los resultados se muestra más adelante en el Capítulo 5, ya que sirven como requerimientos para el sistema de control. El análisis para el dedo pulgar consiste básicamente en observar el movimiento angular de la falange distal, analizando el plano XY. En la Figura 3.7 se muestran las gráficas correspondientes.

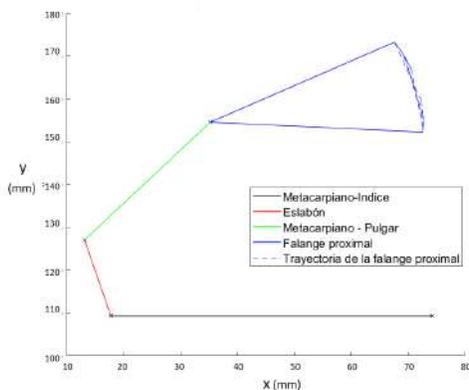


Figura 3.7: Movimiento del dedo pulgar en el plano XY.

### 3.3. Rediseño de componentes

Se conservó el funcionamiento del exoesqueleto utilizando juegos de engrane-cremallera para la transmisión de potencia. El rediseño se realizó con el dimensionamiento de algunas piezas para tener una mayor funcionalidad y estética. Las medidas antropométricas para el dimensionamiento de cada uno de los componentes fueron obtenidas de [25] y [80]. En la Tabla 3.1 se muestran los valores de las distancias máximas, radios y desplazamientos del mecanismo que posteriormente funcionaron como parámetros de diseño para cada uno de los componentes de las falanges proximal y media. Con los valores de radio y relación de distancia, fue posible reducir las dimensiones de los componentes de altura y largo para los bloques que se encargan del movimiento de las falanges.

Tabla 3.1: Valores de los parámetros optimizados para el diseño del exoesqueleto.

Articulación	$\theta$	d (mm)	r (mm)	$S_{max}$ (mm)	$S_{1max}$ (mm)
MCP	0 – 90°	22 – 24,5	13.95	21.91	2.59
PIP	0 – 110°	16.5-18	10.35	19.87	5.83

En la Figura 3.8 se muestra una vista lateral del nuevo diseño de un dedo del mecanismo. En comparación con la versión anterior del prototipo, el nuevo diseño permite cubrir completamente los componentes internos del mecanismo principal durante el movimiento, evitando contacto por parte del usuario hacia los elementos que transmiten la potencia. El bloque encargado del movimiento de la falange proximal y distal contienen el mismo número de piezas y las mismas medidas de los componentes.

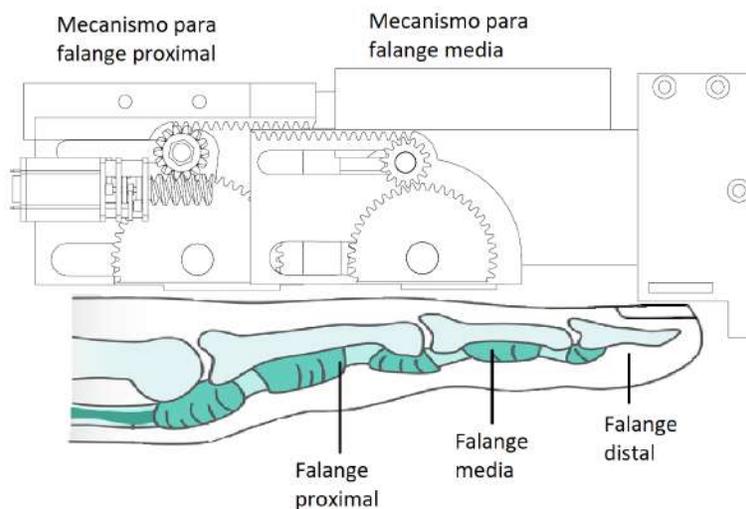


Figura 3.8: Vista lateral de un dedo de exoesqueleto.

Las diferencias de peso, dimensión total, número de piezas y de tornillos con respecto a las dos versiones del prototipo se muestran en la Tabla 3.2. El número de piezas entre una

versión y otra solo aumentó en uno, esta es la pieza correspondiente a una cubierta para el mecanismo encargado del movimiento angular de la falange. Las dimensiones totales para cada uno de los bloques que en su interior contienen el mecanismo principal, se redujeron en un 30 % con respecto al diseño anterior.

Tabla 3.2: Principales características comparativas de las dos versiones del exoesqueleto de mano.

Característica	Diseño anterior	Nuevo diseño
Peso total	703.2g	730g
Número de piezas	15	16
Número de tornillos	4	11
Tipo de actuación	Motor CD	Motor CD
Tiempo de impresión	45 horas	52 horas
Dimensión total de caja	45x21.1(mm)	34.6x19.90(mm)

En la Tabla 3.3 se muestran los valores de los elementos para la transmisión de potencia, los cambios de algunos engranes con respecto a las dos versiones del prototipo están relacionadas con el dimensionamiento de otros componentes, permitiendo la reducción de radio de los engranes y obteniendo de forma final, dimensiones menores en todo el bloque encargado del funcionamiento de cada falange. Para la transmisión de la potencia proveniente del motor hacia el resto del mecanismo se sigue conservado un tornillo sin fin y una corona, pero con mejor ajuste y soporte en la base del motor.

Los bloques que corresponden al movimiento de una falange conservan casi en su totalidad los mismos elementos al diseño anterior, una diferencia notable del nuevo diseño es que los bloques de movimiento son los mismos para cada una de las falanges, en la versión anterior el motor para el accionamiento de la falange proximal se encontraba del lado derecho y para la falange media en el izquierdo, actualmente es un mismo diseño para ambas falanges y el sistema de sin fin- corona se encuentra únicamente del lado derecho. La ubicación de los bloques depende completamente de la pieza “cubierta”, ya que es la encargada de unir a las dos falanges y su diseño es diferente para cada una de ellas. En la Figura 3.9 se observar una vista explosionada de un bloque correspondiente al movimiento de la falange proximal. La forma de la base principal se modificó con la intención de crear una cubierta para el mecanismo de los engranes, para permanecer fuera del posible contacto con el usuario. El número de piezas se conserva, se hizo la fusión de dos de ellas para crear un engrane con una base que soporte a la cremallera movable, en el ensamble se identifica con el número 4, lo que conlleva a cambiar el material original de este engrane de aluminio 1060 a PLA.

El número de tornillos se ha incrementado, pero se han aprovechado como parte integral del funcionamiento. El eje sobre el que el sistema rota se realiza sobre un tornillo que mantiene

Tabla 3.3: Parámetro dimensionales de engranes.

Elemento	m (mm)	$N$	$dp$ (mm)	$p$ (mm)	Barreno	Ancho de cara (mm)	Otro
Engrane A	0.6	52	31.2	1.88	6.2	1.9	-
Engrane B	0.6	14	8.4	1.88	4	4	-
Engrane C	0.6	14	8.4	1.88	4	1.9	-
Engrane D	0.6	45	27	1.88	6	1.9	-
Corona	0.64	14	9	2.03	3.5	5.83	$\alpha = 10^\circ$
Tornillo sinfín	0.64	-	-	2.03	2	$L_r =$ 11.45mm	$\alpha = 10^\circ$
Cremallera M	0.6	-	-	1.88	-	-	$L = 24.5$ mm
Cremallera N	0.6	-	-	1.88	-	-	$L = 25.7$ mm

unidas a las piezas internas a la base del motor y la base principal. Para evitar que el movimiento de los engranes separen la base inferior y superior del motor, dos tornillos M2.5x0.45 - 4 son usados.

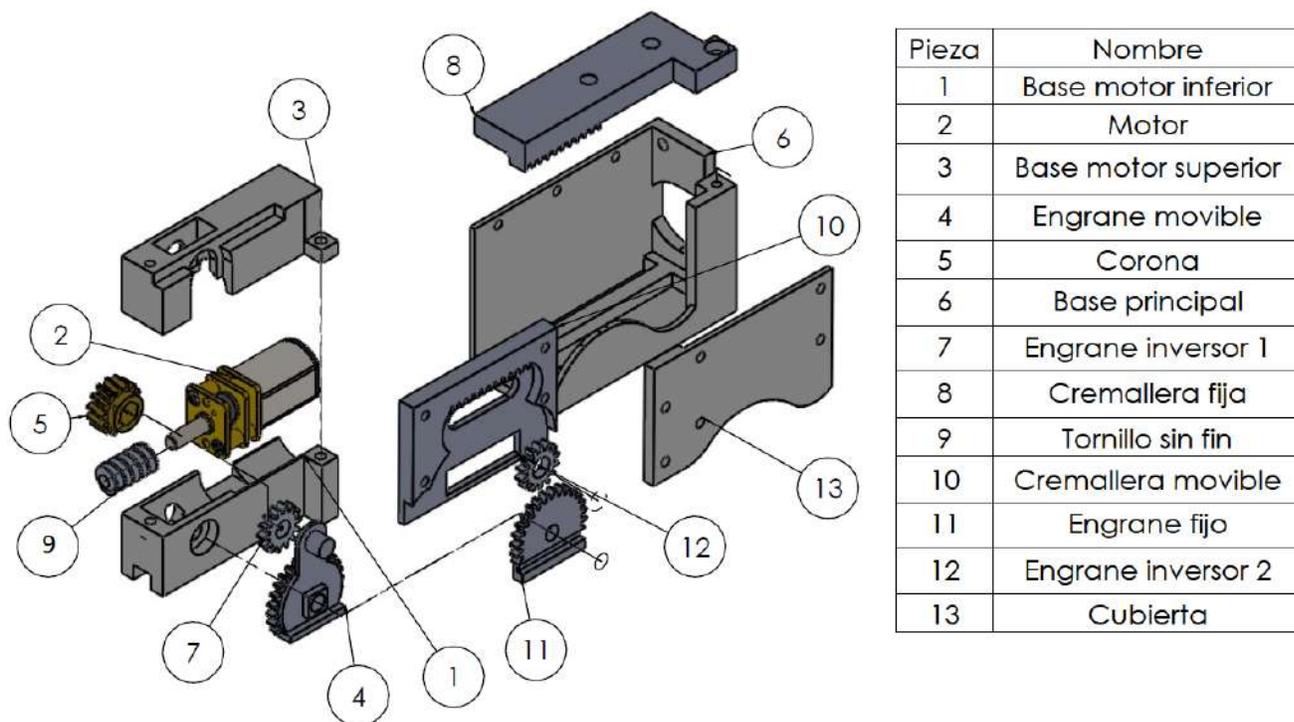


Figura 3.9: Vista explosionada del bloque encargado del movimiento de la falange proximal.

Como en el diseño anterior, ambos bloques consisten en las mismas piezas y en la misma distribución, la única variación es el desplazamiento lineal que permiten las cremalleras para lograr cubrir los rangos de movimiento. En la Figura 3.10 se muestran dos posiciones diferentes del exoesqueleto, el dedo se representa como una cadena de eslabones.

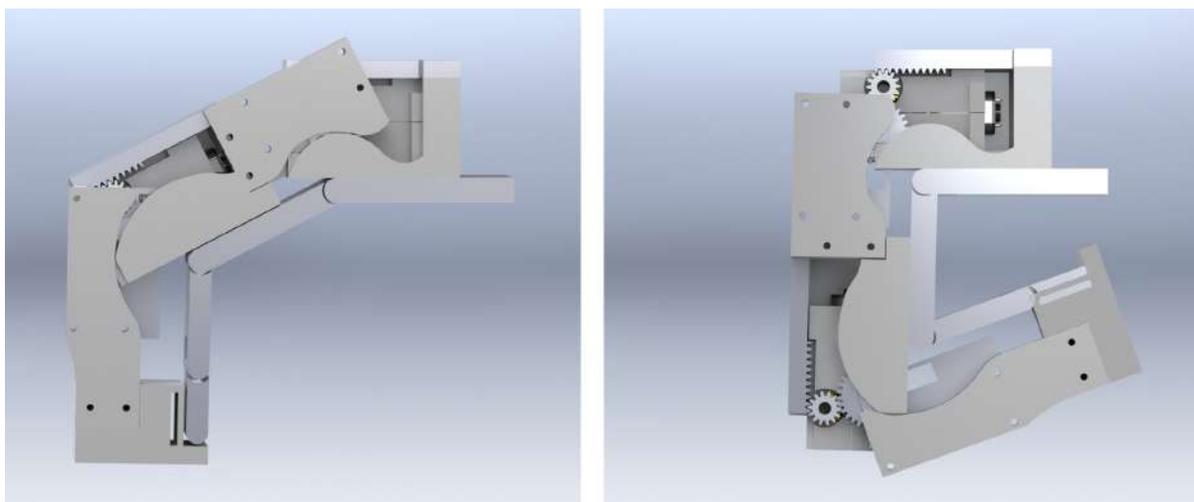


Figura 3.10: Vista lateral de un dedo del exoesqueleto para dos distintos movimientos. a) Falange proximal y falange distal a  $45^\circ$ . b) Falange proximal  $90^\circ$  y falange distal  $110^\circ$ .

### 3.4. Validación cinemática

Para la validación cinemática del prototipo se usó el análisis cinemático reportado en [26] y las simulaciones realizadas en SolidWorks. Con la simulación de SolidWorks se le aplicó un par específico a cada uno de los dos motores, con el movimiento obtenido de todo el dedo, se obtuvieron las coordenadas en el plano y, z (vista lateral) como se muestra en la Figura 3.11. El análisis se realizó únicamente con un dedo del exoesqueleto, correspondiente al dedo índice, para el resto de los dedos la cinemática y las dimensiones de las piezas son las mismas.

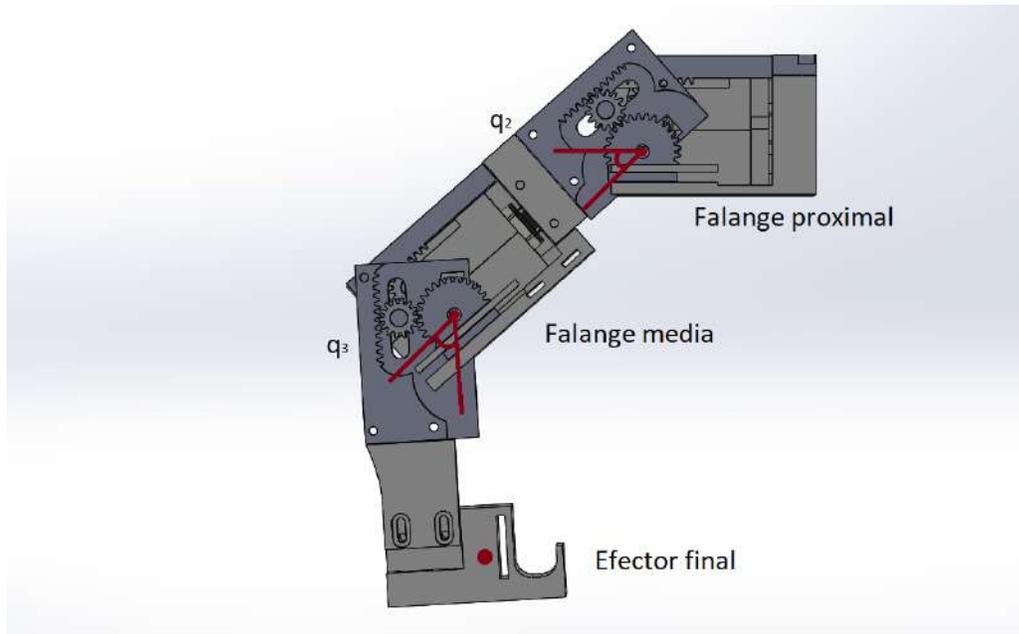


Figura 3.11: Vista lateral de un dedo del exoesqueleto con la medición de la punta del dedo para calcular el espacio de trabajo.

Para obtener un grupo de coordenadas que describan el espacio de trabajo de un dedo del exoesqueleto en el entorno virtual de SolidWorks, se realizaron variaciones en cada uno de los actuadores, hasta cubrir el rango máximo para el cual fue diseñado, para la falange proximal de  $0-90^\circ$  y la falange media de  $0-110^\circ$ . Obtenidos los datos, se graficaron en Matlab<sup>®</sup> y se obtuvo la Figura 3.12.

Con el espacio de trabajo obtenido en la simulación de SolidWorks, se calcula la correlación que existe entre éste con el resultante de la cinemática directa, se tienen dos diferentes curvas por lo que el coeficiente de correlación ayudará a verificar si ambas son proporcionales. Para calcular esto se obtuvieron 16 curvas diferentes, cada una de ella representa una variación en cada uno de los actuadores, las dimensiones correspondientes a cada uno de los parámetros de Denavit-Hartenberg fueron actualizados a las nuevas dimensiones de cada uno de los componentes. Con la ecuación (3.4) se obtiene este coeficiente [81]

$$E = \sum_{m=0}^{N-1} x(m)^2 \quad (3.4)$$

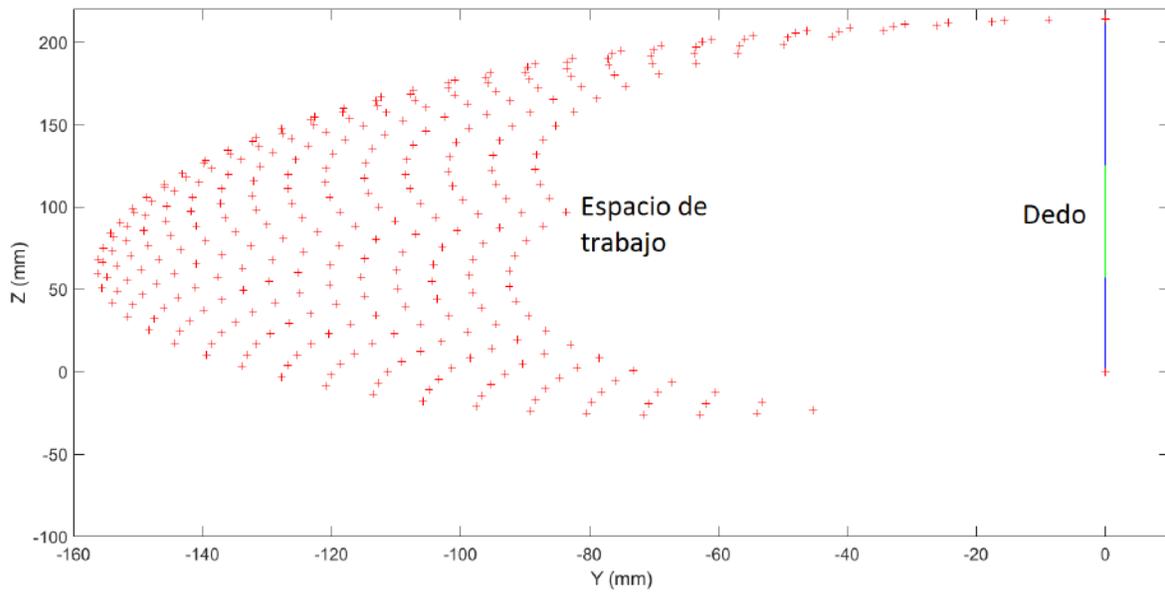


Figura 3.12: Espacio de trabajo obtenido de la simulación de SolidWorks

Donde:

$Cov_{xy}$  es la covarianza entre el valor de  $x$  y  $y$

$\sigma_x$  es la desviación típica de  $x$

$\sigma_y$  es la desviación típica de  $y$

En Matlab<sup>®</sup> se calcula este valor con la función  $\text{corr2}(x,y)$ , donde el resultado para todas las curvas fue de 0.98, que muestra una proporcionalidad directa entre ambas curvas, por lo cual las curvas son prácticamente iguales.

### 3.5. Construcción de prototipo

Para el ensamble de las piezas se usaron elementos para la fijación de las mismas, los tornillos seleccionados se encuentran de manera comercial en el área de relojería por las dimensiones que tienen. En la Tabla 3.4 se detallan los tornillos usados en el ensamble. En promedio para cada bloque se utilizan 12 tornillos.

Tabla 3.4: Elementos de fijación para el prototipo.

Elemento				Cantidad
Tornillo M2.5x0.45 - 4mm	avellanado	cabeza plana		48
Tornillo M2.5x0.45 - 10mm	avellanado	cabeza plana		16
Tornillo - 5mm	avellanado	cabeza plana	M2x0.4	24
Tornillo - 25mm	avellanado	cabeza plana	M3x0.5	8

El ensamble de un bloque se muestra en la Figura 3.13, la cubierta final no se encuentra ensamblada para observar el mecanismo interno principal. La posición mostrada es con una referencia de 90°, las únicas piezas metálicas son los engranes inversores, los engranes base fueron sustituidos por PLA. El color de las piezas es diferente con el objetivo de hacer un ensamble intuitivo y para facilitar la visibilidad del movimiento de las piezas, las bases de los motores son color gris, el engrane encargado del movimiento lineal y que acopla con la cremallera movable es color verde, el resto de los elementos que se encuentran a la vista del usuario son de color blanco.

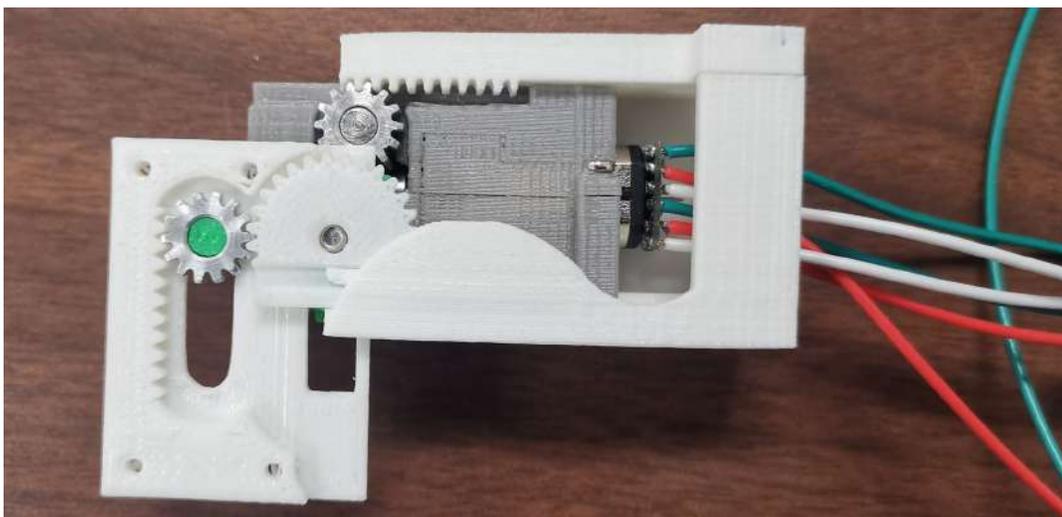


Figura 3.13: Vista lateral del mecanismo interno del bloque encargado del movimiento de la falange proximal.

Todas las piezas fueron impresas de nuevo, ya que las dimensiones de éstas cambiaron en su totalidad, las únicas piezas que se conservaron del diseño inicial son los engranes inversores de aluminio. La base principal que se coloca sobre la mano y sujeta los dedos, fue rediseñada con salida para los cables provenientes de los encoder, de manera similar las bases fueron recubiertas con una tela acolchada para evitar molestias al usuario. En la Figura 3.14 se muestra el ensamble final del exoesqueleto, se observan los cables para las conexiones entre el sistema de accionamiento y de adquisición de señales. El peso total del prototipo es de 730 gramos, incluyendo los actuadores, las piezas mecánicas y eléctricas dentro del exoesqueleto, las tarjetas de adquisición de datos no están contempladas.

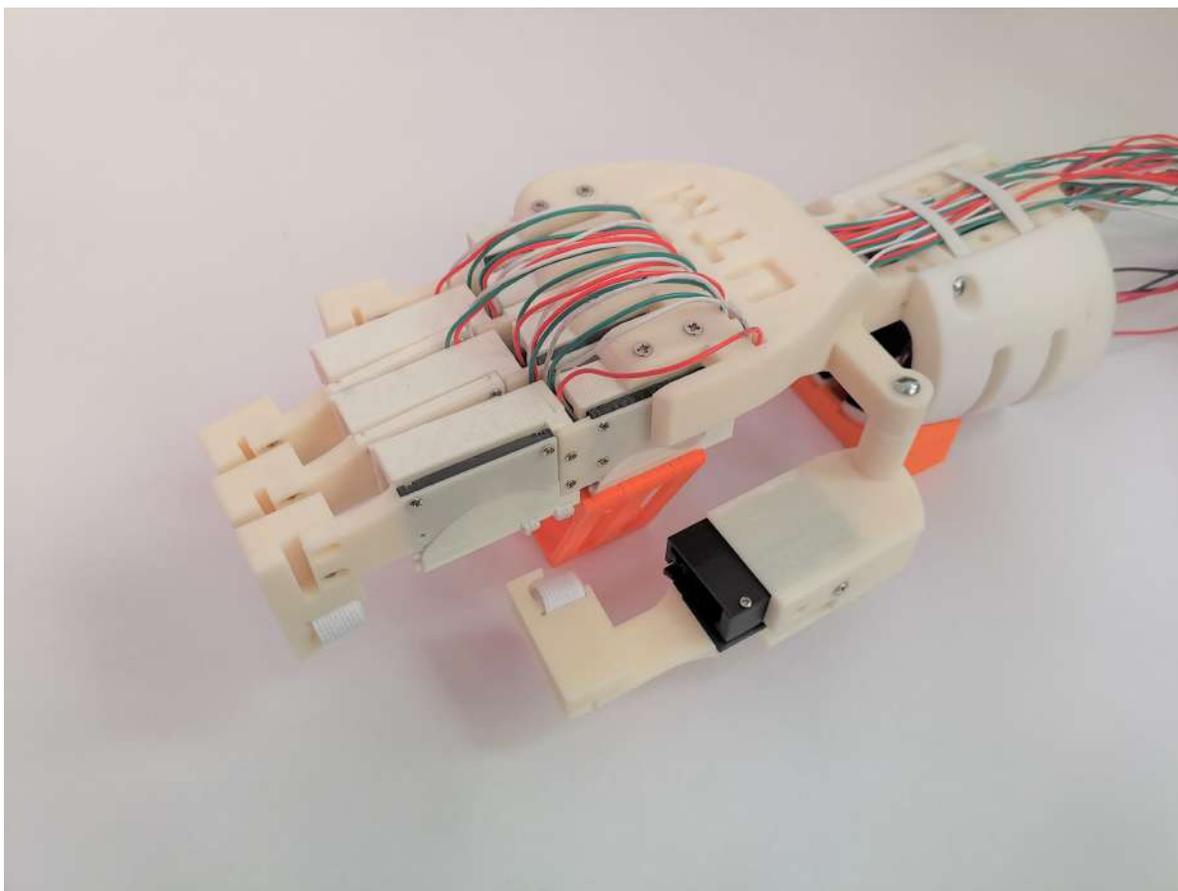


Figura 3.14: Ensamble final del prototipo.

La conexión entre el sistema mecánico, de instrumentación y de control se muestra en la Figura 3.15. Los motores internos del exoesqueleto se conectan a una tarjeta de adquisición de datos para la transmisión de datos a un ordenador para el procesamiento de las señales. La computadora recibe las señales del brazalete MYO y son reconocidas en la interfaz gráfica de usuario que se aprecia en la pantalla de la Figura 3.15, así como todos los elementos necesarios con los cuales se desarrolla una prueba, abarcando:

- Ordenador para procesamiento de las señales
- Sistema mecánico

- Sistema de control y de adquisición de datos
- Sistema de instrumentación
- Elementos de agarre para las pruebas (cilindro de plástico y plumón)



Figura 3.15: Vista general del sistema completo.



# Capítulo 4

## Diseño e implementación del sistema de instrumentación

En ésta sección se describe el desarrollo del sistema de instrumentación. La necesidad de obtener señales provenientes de los actuadores del exoesqueleto y del brazo del usuario dividen el sistema en dos partes. La primer parte consiste en obtener la posición actual y manipulación de los actuadores, para hacer regulación de voltaje y cubrir los rangos de movimiento en velocidades específicas.

La segunda parte del sistema debe de encargarse de la adquisición de señales EMG del usuario y posteriormente su clasificación para la detección del movimiento realizado. Se utiliza el brazalete MYO para la adquisición de señales, que posteriormente son rectificadas y acondicionadas para la extracción de características que permitan entrenar un algoritmo de clasificación para detectar en línea el movimiento que el usuario realiza. Se describen cada uno de los algoritmos desarrollados y aplicados para el reconocimiento de los gestos de la mano, así como los resultados obtenidos. En la Figura 4.1 se muestra el diagrama a bloques del proceso realizado fuera de línea para el entrenamiento del algoritmo de clasificación.



Figura 4.1: Diagrama esquemático de las conexiones realizadas entre la tarjeta Arduino, controlador y encoder del motor.

## 4.1. Instrumentación de actuadores

Los dedos índice, medio y anular del exoesqueleto necesitan dos motor reductores de CD como sistema de actuación, mientras que los dedos pulgar y meñique requieren sólo de un motor cada uno. Las pruebas realizadas en cada una de las fases fueron desarrolladas con un dedo del exoesqueleto. Las características relevantes de los actuadores se muestran en la Tabla 4.1.

Tabla 4.1: Especificaciones generales del micromotorreductor [82].

Característica	Valor
Dimensiones	24 x 10 x 12 <i>mm</i>
Velocidad	200 <i>RPM</i>
Corriente sin carga	120 <i>mA</i>
Corriente máxima	1600 <i>mA</i>
Par máximo	2.9 <i>kg</i>
Relación de engranes	150:1
Eje extendido	Si

Independientemente de la tarjeta de adquisición de datos con que se desee trabajar, un controlador para la regulación de velocidad de los motores es imprescindible, ya que algunas tarjetas como Arduino poseen pines específicos de entrada y salida, pero ellos no proporcionan los voltajes necesarios para que los actuadores alcancen la velocidad deseada. Para este proyecto se utilizaron controladores Toshiba TB6612FNG, este driver permite el control de dos motores de manera independiente y bidireccional, mediante la generación de una señal PWM (modulación de ancho de pulso, por sus siglas en inglés) [83]. En la Tabla 4.2 se muestran sus características más relevantes.

Tabla 4.2: Especificaciones generales del controlador Toshiba TB6612FNG [83].

Característica	Valor
Dimensiones	16 x 20 <i>mm</i>
Número de canales para motor	2
Voltaje recomendado del motor	4.5-13.5 V
Corriente de salida máxima	3 A por canal
Corriente de salida continua	1 A por canal
Frecuencia máxima de PWM	100 kHz

Para obtener la posición y la velocidad estimada de cada uno de los actuadores, se utilizan sensores (encoders). Cada motor cuenta con un encoder magnético (Pololu R), el cual se conecta de acuerdo a su hoja de datos [84]. La ventaja de usar encoders magnéticos es su tamaño reducido y su principio de detección sin contacto.

Para el movimiento del exoesqueleto es necesario el uso de ocho motores de CD, 4 drivers y 8 encoders magnéticos. La tarjeta para adquisición de datos debe cubrir la demanda de leer

todas las señales provenientes de los encoders, así como la capacidad de generar ocho señales de PWM. En la Figura 4.2 se muestra un esquema de la conexión entre los motores necesarios para el movimiento de un dedo y el sistema de instrumentación.

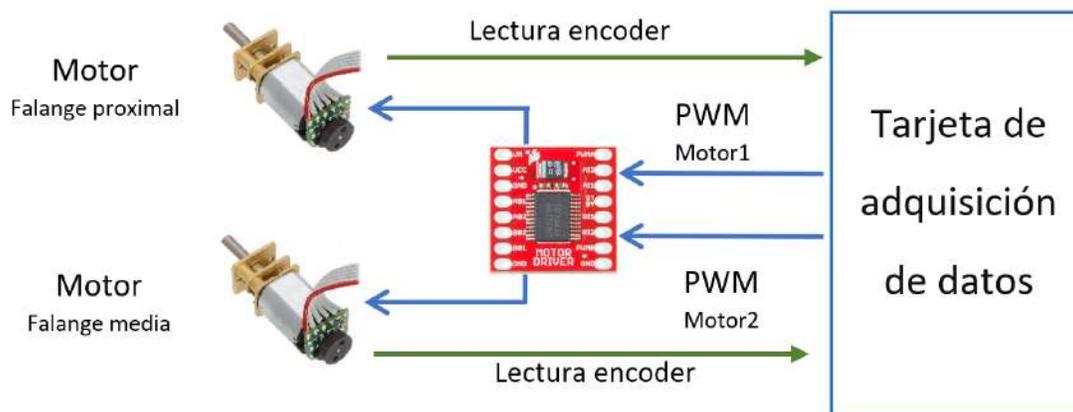


Figura 4.2: Diagrama esquemático de las conexiones realizadas entre la tarjeta Arduino, controlador y encoder del motor.

El esquema mostrado en la Figura 4.2 permite visualizar los requerimientos y considerar al momento del desarrollo del sistema de control.

## 4.2. Obtención de señales EMG

El desarrollo del control del exoesqueleto está orientado hacia una terapia de rehabilitación asistida, es necesario conocer la intención del usuario al momento de realizar los movimientos, lográndolo con el uso de señales electromiográficas. Una manera práctica de obtener señales EMG es con el brazalete MYO desarrollado por ThalmicLabs, usa la actividad eléctrica de los músculos del brazo para controlar dispositivos electrónicos, sus 8 sensores miden la energía que corre por el brazo y detecta los gestos que se hacen, posteriormente son comunicados a la computadora por medio de tecnología bluetooth. La ventaja que representa el brazalete es su peso de 97g y la posibilidad de obtener señales ya procesadas, permitiendo centrarse en la aplicación. El brazalete cuenta con su propio software para desarrolladores, para realizar la conexión con la computadora de donde posteriormente se accede a estos datos de las señales para trabajar con ellas en algún otro software como es Matlab<sup>®</sup>, [85]. En la Figura 4.3 se muestra el brazalete MYO el cual es ajustable al brazo del usuario.

Para la adquisición de las señales en Matlab<sup>®</sup> es necesario cumplir con la instalación de los siguientes programas y librerías.

- Matlab<sup>®</sup>
- Myo Connect
- Windows SDK 0.9.0



Figura 4.3: Brazalete MYO [85].

- Visual Studio
- Myo SDK MATLAB MEX Wrapper para Matlab®
- MinGW-w64 compiler 4.9.2 para Matlab®

Para observar el funcionamiento del brazalete y corroborar que la instalación del software fue correcta, se realizan pruebas para obtener la adquisición de señales provenientes del brazalete, el acceso a la IMU también se encuentra disponible, ya sea que se desee medir velocidad, aceleración y el movimiento rotacional, así como las lecturas directas de los sensores EMG. En la Figura 4.4 se muestran una prueba realizada, al fondo se observan las mediciones que se registran en Matlab® en tiempo real.



Figura 4.4: Prueba de funcionamiento y conexión con la interfaz de Matlab®.

Los movimientos básicos con los cuales se trabajaron durante el desarrollo de la tesis son: cierre de puño, agarre de punta y pinza fina. Estos movimientos son los realizados por una mano sana y los cuales son primordiales durante el desarrollo de diferentes actividades en la vida cotidiana de las personas. La selección de movimientos se hizo con base en lo reportado en la literatura para una terapia de rehabilitación [48], así como se muestra en la Figura 2.4. Para los ejercicios de agarre de punta el usuario toma un cilindro de plástico con un diámetro de

8cm y una longitud de 12 cm aproximadamente y para el desarrollo de la pinza fina el usuario toma un marcador comercial con la yema del dedo pulgar e índice. Para el cierre del puño el usuario solamente cierra el puño y lo mantiene en esa posición hasta que la prueba termina. En la Figura 4.5 se muestran los tres movimientos básicos que se utilizaron para el banco de datos.

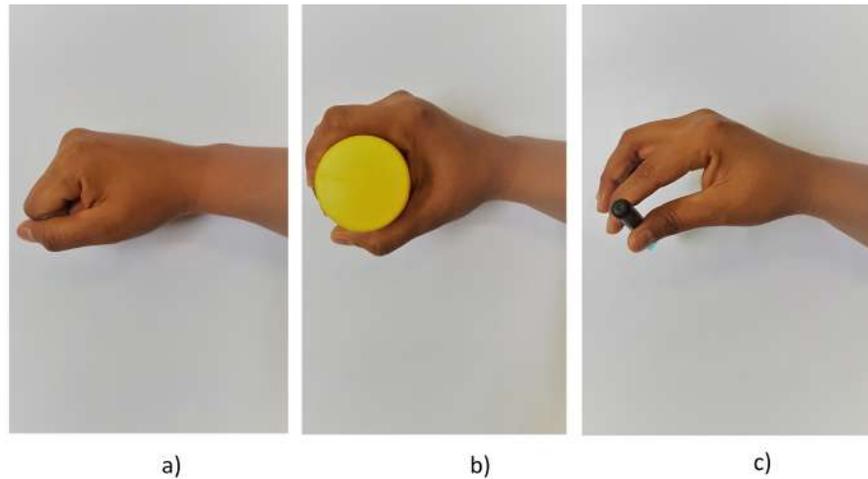


Figura 4.5: Movimientos de la mano a desarrollar por el usuario. a) Cierre de puño, b)Agarre de punta,c)Pinza fina.

Cada una de las pruebas que fueron realizadas tuvo una duración de 5 segundos, tiempo suficiente para obtener las señales EMG concernientes a cada movimiento de la mano sin fatigar al usuario. En la Figura 4.6, se muestran las señales adquiridas durante la apertura y el cierre de la mano, en la Figura 4.6 a) se muestran los resultados del sensor EMG, en la Figura 4.6 b) se muestra los resultados en la IMU relacionados al giro de movimiento.

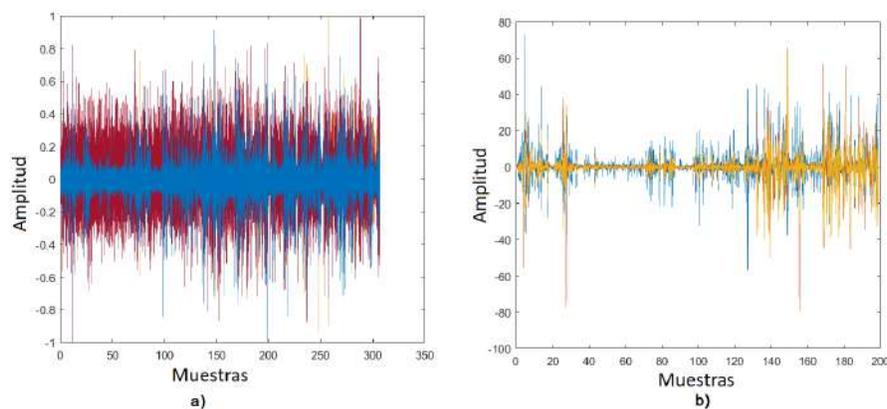


Figura 4.6: Señales obtenidas en Matlab<sup>®</sup> durante la apertura y el cierre de la mano utilizando el brazalete MYO.

En la Figura 4.7 se muestra cada uno de los canales de la señal EMG al realizar el movimiento de cerrar el puño.

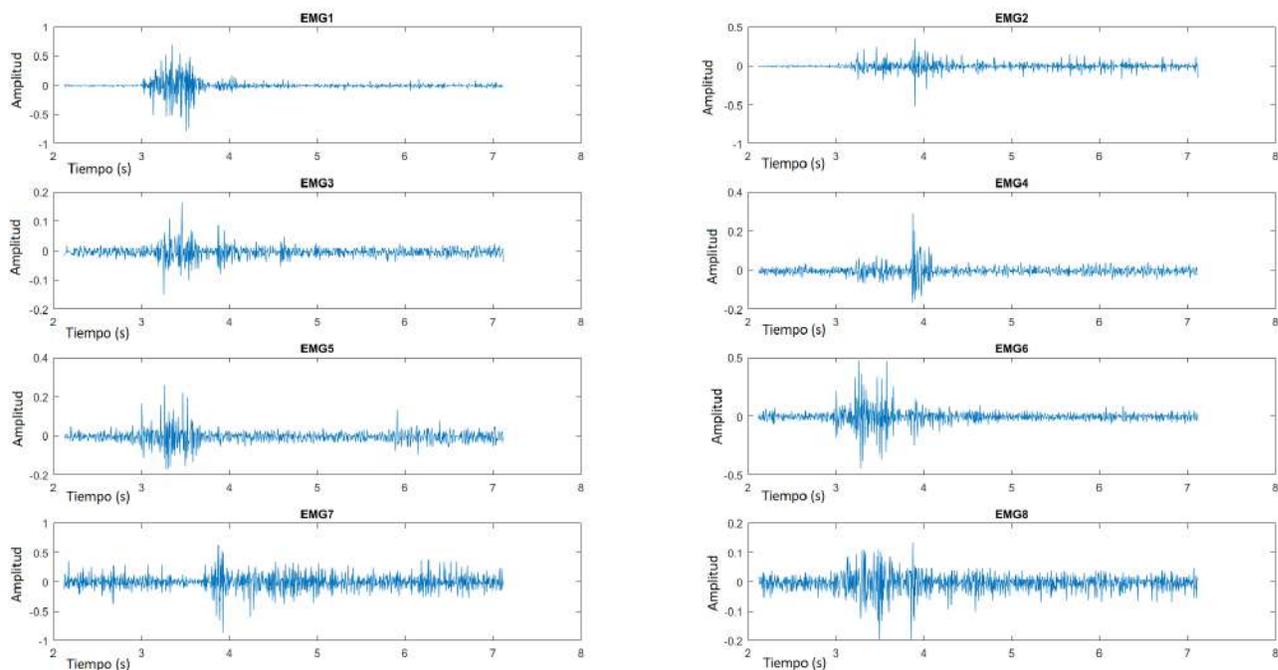


Figura 4.7: Señal EMG obtenida por el Myo Armband al momento de realizar el cierre del puño.

Para realizar la extracción de características primero hay que realizar el procesamiento de la señal para recuperar la señal base que brinde información de la actividad muscular.

### 4.3. Creación de banco de datos

Para entrenar el algoritmo de clasificación se crea un banco de datos que funcionen como base para el entrenamiento. El tiempo total muestreado fue de 5 segundos para cada uno de los movimientos. Se obtuvieron 225 muestras para cada movimiento y un total de 675 muestras para los tres movimientos. Se realizaron dos pruebas para cada movimiento con cada uno de los participantes pero se seleccionó solamente una de ellas, todas las pruebas fueron realizadas en el brazo izquierdo, con un rango de edad de 20-45 tanto de género masculino como femenino. En la Tabla 4.3 se presenta la cantidad de muestras obtenidas por género y movimiento de la mano.

Cada una de las muestras se almacenó en un archivo extensión .txt independiente con la información de 1000 datos por cada una de las 8 señales electromiográficas correspondientes a los sensores del brazalete MYO. La base de datos se encuentra disponible en [86], para su descarga gratuita.

Tabla 4.3: Cantidad de muestras recolectadas por género y movimiento de la mano.

Movimiento	Género	Muestras recolectada
Cierre de puño	Masculino	146
	Femenino	79
Agarre de punta	Masculino	146
	Femenino	79
Pinza fina	Masculino	146
	Femenino	79
Total		675

## 4.4. Preprocesamiento de señales EMG

La señal proveniente del brazalete de forma cruda contiene ruido y una mezcla con señales no deseadas, para la extracción de características la señal debe ser filtrada para eliminar información no deseada.

### 4.4.1. Detección de activación muscular

Durante el tiempo de muestreo en cada prueba (5 segundos) que se realiza existe un tiempo de retardo por parte del usuario al momento de realizar algún movimiento, ya sea cerrar el puño, realizar agarre de punta, esférico o de pinza fina. Por lo cual, es conveniente detectar en qué momento el usuario comienza a realizar el movimiento de la mano para recortar el tiempo de muestreo a solo un intervalo de tiempo que contenga la información de la activación muscular. Para la detección de la actividad muscular se usó un algoritmo para detección de voz, la idea general es tratar a la señal EMG como una señal de audio para realizar una detección de la activación del músculo al momento de realizar algún movimiento equivalente a una detección de activación de voz. Cuando el usuario se encuentra en reposo, las señales EMG se mantienen oscilando en valores cercanos a cero, al momento de que el usuario comienza a realizar algún movimiento con la mano las señales comienzan a presentar cambios en su amplitud, para detectar los cambios iniciales en las señales se implementó el algoritmo de Coper. El algoritmo de Coper propuesto en [87] y denominado así como un juego de palabras de sus creadores Cotrina-Peralta, es utilizado para la detección de actividad de voz o para detección automática de extremos determinando los instantes de inicio y final de una pronunciación con el fin de entregar al sistema de reconocimiento únicamente el segmento de señal de voz comprendida en dichos instantes. El algoritmo se basa en la evolución de la señal en el tiempo utilizando una combinación de energía y cruces por cero.

La energía de una señal discreta se define como:

$$E = \sum_{m=0}^{N-1} x(m)^2 \quad (4.1)$$

La fórmula de la densidad de cruces por cero es:

$$z = \sum_{m=0}^{N-1} |\text{sign}[x(m)] - \text{sign}[x(m-1)]| \quad (4.2)$$

donde:

$x(m)$  es la amplitud de la señal

$N$  es el número de muestras

$\text{Sign}$  es la función signo

Para el algoritmo de Coper las funciones  $\text{sign}$  son multiplicadas por la energía de la muestra analizada, para que la densidad acumulada dependa del signo de la muestra y de la amplitud, logrando que el ruido no tenga acumulación de densidad por poseer una pequeña amplitud en comparación con el resto de la señal, como se muestra en la ecuación (4.3).

$$\text{Coper} = \sum_{m=0}^N |x(m) \cdot |x(m)| - x(m-1) \cdot |x(m-1)|| \quad (4.3)$$

La aplicación del algoritmo con las señales EMG permitió detectar la muestra en que se realiza la activación muscular, que posteriormente mediante el cálculo de la media aritmética de las ocho señales provenientes del sensor se obtiene el punto de corte inicial de la señal, el segundo punto de corte de la señal se realiza 200 muestras posteriores al punto de corte resultante por el algoritmo de Coper, resultando en señales recortadas a 2 segundos que describen la activación del músculo para cada movimiento que se realiza. En la Figura 4.8 se muestra la aplicación de este algoritmo para una señal EMG adquirida del brazalete MYO para el cierre de puño.

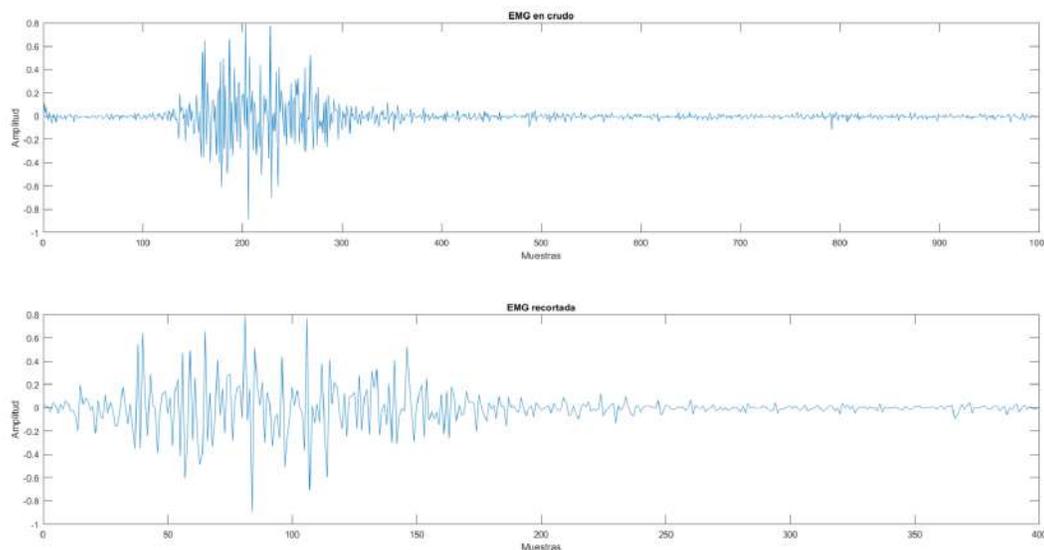


Figura 4.8: Señal EMG antes y después del recorte de activación muscular.

Como se puede observar en la Figura 4.8, las muestras concernientes al estado de reposo por parte de la mano del usuario son eliminadas, lo que permitirá una reducción de costo computacional para su futuro procesamiento y la reducción de error para su futura clasificación.

#### 4.4.2. Filtrado y normalización de señales

El modelo matemático de la señal EMG establece que está compuesta por una señal portadora de media cero modulada por una señal que contiene actividad muscular, por lo tanto cada una de señales EMG proveniente de los ocho canales del Myo Armband, es rectificadas y filtradas aplicando un filtro pasa bajo de Butterworth con los datos que se muestran en la Tabla 4.4.

Tabla 4.4: Parámetros del filtrado de la señal EMG.

Parámetro	
Filtro	Butterworth
Tipo	Pasa bajas
Orden	Cuarto orden
Frecuencia de corte	5 Hz

El primer procesamiento fue la rectificación de la señal EMG utilizando la función de valor absoluto para cada uno de los canales de la señal, con el fin de eliminar la componente gaussiana de ruido blanco y trabajar únicamente con las señales modulantes de muy baja frecuencia que contiene la información de la actividad muscular. La señal procesada se muestra en la Figura 4.9.

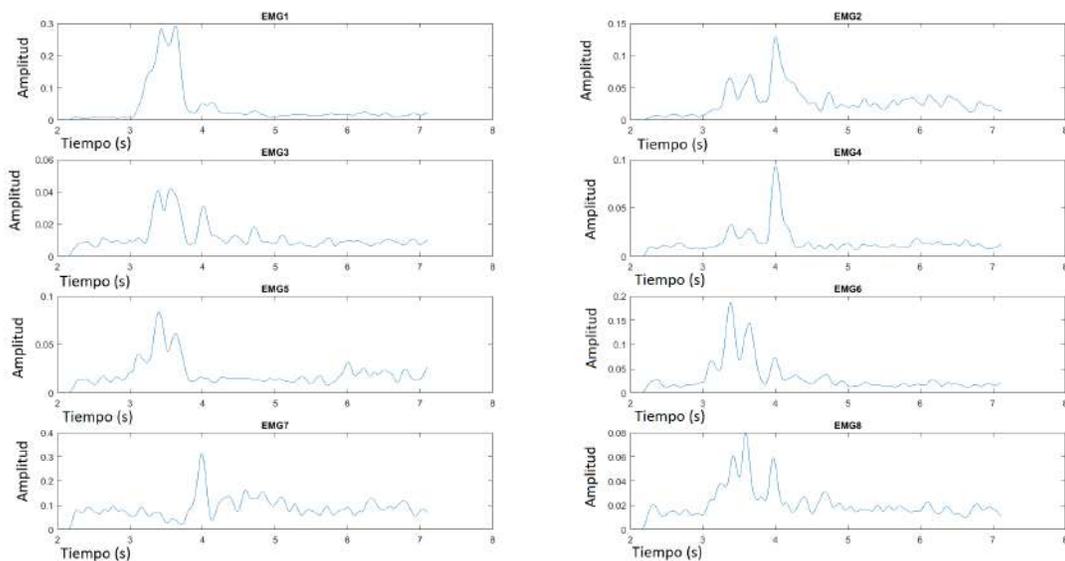


Figura 4.9: Señal EMG después del procesamiento para la acción de cierre de puño.

En este punto, la señal fue adquirida y procesada para recuperar la señal base que describe el movimiento realizado por la mano, el siguiente paso consiste en buscar las características que describan a cada una de las señales.

### 4.4.3. Extracción de características

La extracción de características de una señal consiste en la selección de valores representativos que posee en el dominio del tiempo o frecuencia. El número de características extraídas suele variar dependiendo del tipo de señal a analizar. Las características paramétricas tradicionales de las señales de EMG incluyen la raíz cuadrada media, los cruces por cero, los cambios de signo de pendiente, la longitud de la forma de onda y el parámetro de amplitud de Willison. Sin embargo, la complejidad computacional que llegue a tener es la principal consideración, para que todas las características lleguen a ser procesadas por una computadora en tiempo real. Las características extraídas para cada una de las señales EMG de cada sensor recolectada se mencionan a continuación, donde para cada característica,  $X_i$  denota los  $i$  datos de muestreo en la ventana y  $N$  es la longitud de la ventana para calcular las características.

#### 4.4.3.1. Longitud de onda.

La longitud de onda (WL, por sus siglas en inglés Waveform Length), es una variación acumulativa de la EMG que puede indicar el grado de variación sobre la señal de la EMG [88]. Se define por:

$$WL = \frac{1}{N} \sum_{i=1}^N |X_i|^2 \quad (4.4)$$

donde  $X_i$  denota los  $i$  datos de muestreo en un vector de longitud  $N$ .

#### 4.4.3.2. Valor absoluto medio.

Se calcula utilizando la media de la señal EMG rectificada de onda completa. En otras palabras, se calcula tomando el promedio del valor absoluto de la señal EMG. Es una forma fácil de detectar los niveles de contracción muscular y es una característica popular utilizada en la aplicación de control mioeléctrico [89]. Se define como:

$$VAM = \frac{1}{N} \sum_{i=1}^N |X_i| \quad (4.5)$$

donde  $X_i$  denota los  $i$  datos de muestreo en un vector de longitud  $N$ .

#### 4.4.3.3. Raíz cuadrada media.

La raíz cuadrada media refleja el valor promedio de la señal [89]. Definida por:

$$RMS = \sqrt{\frac{1}{N} \sum_{i=1}^N |X_i|^2} \quad (4.6)$$

donde  $X_i$  denota los  $i$  datos de muestreo en un vector de longitud  $N$ .

#### 4.4.3.4. Curtosis

La curtosis indica el punto máximo de la distribución de probabilidad asociada a las amplitudes instantáneas de las mediciones de series de tiempo, siendo una característica de la frecuencia-probabilidad de la señal [90]. La curtosis se define como:

$$Curtosis(x) = \frac{E((x - \mu)^4)}{\sigma^4} \quad (4.7)$$

donde  $\mu$  denota el momento centrado de la señal y  $\sigma$  la desviación estándar.

#### 4.4.3.5. Cruce por cero.

Los cruces por cero (ZC, por sus siglas en inglés Zero Crossing), es el número de veces que la señal pasa por el eje de amplitud cero. Este parámetro se usa para obtener la propiedad aproximada en el dominio de la frecuencia [91]. Dadas dos señales EMG contiguas,  $emg_k$  y  $emg_{k+1}$ , la ZC puede calcularse por la ecuación (4.8).

$$ZC = \sum_{k=1}^N sgn(f(x)) \quad (4.8)$$

$$sgn(x) = \begin{cases} 1 & \text{if } (X_i > 0 \text{ y } X_{i+1} < 0) \parallel (X_i < 0 \text{ y } X_{i+1} > 0) \\ 0, & \text{Cualquier otro caso} \end{cases}$$

donde  $X_i$  denota los  $i$  datos de muestreo en un vector de longitud  $N$ .

#### 4.4.3.6. Amplitud de Wilson.

La amplitud de Wilson (WAMP, por sus siglas en inglés Willison Amplitude), es un parámetro que se utiliza para contar el número de veces que la amplitud de la señal supera un umbral predefinido, es un indicador del nivel de contracción muscular [91]. Se define por la ecuación (4.9).

$$WAMP = \sum_{k=1}^{N-1} f(|X_k - X_{k+1}|) \quad (4.9)$$

$$f(x) = \begin{cases} 1, & \text{if } x > \text{umbral} \\ 0, & \text{Cualquier otro caso} \end{cases}$$

donde  $X_i$  denota los  $i$  datos de muestreo en un vector de longitud  $N$ .

#### 4.4.3.7. Cambio en el signo de la pendiente.

El cambio en el signo de la pendiente(SSC, por sus siglas en inglés Slope Sign Changes), cuenta el número de veces que la pendiente de la señal cambia de signo. Dadas las tres señales EMG contiguas emg k-1, emg k y emg k + 1, el número de cambios de signo de pendiente se calcula de la siguiente manera [88].

$$SSC = \sum_{k=1}^{N-1} f(x) \quad (4.10)$$

$$f(x) = \begin{cases} 1 & \text{if, } (X_k < X_{k+1}) \text{ y } (X_k < X_{k-1}) \parallel (X_k > X_{k+1}) \text{ y } (X_k > X_{k-1}) \\ 0, & \text{Cualquier otro caso} \end{cases}$$

#### 4.4.3.8. Entropía

La entropía mide la incertidumbre de una variable aleatoria discreta  $X$ , la cual está definida por:

$$H(X) = \sum_{x_i \in X} P(x_i) \log(P(x_i)) \quad (4.11)$$

donde  $P(x_i)$  es la probabilidad de la muestra  $i$ .

Las ecuaciones anteriormente descritas fueron desarrolladas y calculadas en Matlab® para cada una de las señales EMG. Después de que las señales fueron filtradas y normalizadas, se calculan cada una de las características que describen a cada señal. En total por cada muestra tomada se obtienen 88 características, esto da como resultado de 11 características por cada uno de los 8 sensores EMG. Para el banco de datos se cuentan con 675 pruebas para las cuales se crea un proceso iterativo que extraiga las 88 características de cada una de ellas, creando así un arreglo de 675x88, donde cada renglón representa las características de una prueba y cada columna un valor para una característica en específico. En la Figura 4.10 se ilustra cómo es que la matriz de características totales queda ordenada.

	Característica 1								• • •	
Muestra 1	EMG 1	EMG 2	EMG 3	EMG 4	EMG 5	EMG 6	EMG 7	EMG 8	EMG 1	EMG 2
Muestra 2										
Muestra 3										
•										
•										
•										
Muestra 844										

Figura 4.10: Matriz de características.

La matriz de características es almacenado en un archivo con extensión .txt, con la finalidad de manipular los datos en un futuro, donde el siguiente paso es determinar del total de características las más relevantes para cada movimiento.

## 4.5. Selección de características

Con el archivo que contiene las  $N$  características de las 675 pruebas, se crea un subconjunto  $M$  del conjunto original. Para que el conjunto original se reduzca de manera óptima, se realiza bajo un cierto criterio de evaluación, que se le conoce como selección de características. El desarrollo de éste tipo de algoritmos es un campo activo de investigación en informática desde 1970 y el cual resulta ser un problema fundamental para distintas áreas [92], ya que la selección brinda diferentes ventajas con las bases de datos, como son: facilitar la visualización de datos y su comprensión, reducir los requisitos de medición y almacenamiento, reducir tiempos de capacitación y utilización, mejorar el rendimiento de predicción [93]. Para crear estos subconjuntos existen diferentes algoritmos los cuales se clasifican en filtros, envoltorios y enfoques integrados [92]. Para la base de datos que se tiene se aplicaron 5 diferentes algoritmos para determinar las mejores características, los algoritmos empleados se describen a continuación.

### 4.5.1. Dependencia estadística

El objetivo del método de dependencia estadística (SD, por sus siglas en inglés Statistical dependency) mide la dependencia de los valores de una entidad en relación con las etiquetas de las clases asociadas, o si las dos simplemente ocurren de forma simultánea [94]. Dadas dos variables aleatorias  $x$  e  $y$ , su información mutua se define en términos de sus funciones de densidad probabilística  $p(z)$ ,  $p(y)$  y  $p(y, z)$ .

$$SD = \sum_{y \in Y} \sum_{z \in Z} P(y, z) \frac{P(y, z)}{P(y)P(z)} \quad (4.12)$$

### 4.5.2. Información mutua.

La información mutua (MI, por sus siglas en inglés Mutual Information) es una de las medidas más utilizadas para definir la dependencia de las variables [95]. Dadas dos variables aleatorias  $x$  e  $y$ , su información mutua se define en términos de sus funciones de densidad probabilística  $p(z)$ ,  $p(y)$  y  $p(y, z)$ :

$$MI = \sum_{y \in Y} \sum_{z \in Z} P(y, z) \log \left( \frac{P(y, z)}{P(y)P(z)} \right) \quad (4.13)$$

### 4.5.3. Selección secuencial hacia adelante

Selección secuencial hacia adelante (SFS, por sus siglas en inglés Sequential forward selection), es uno de los métodos más conocidos y ampliamente usados en la práctica para la selección de características. El algoritmo SFS comienza a trabajar a partir de un conjunto vacío, el conjunto de características se actualiza iterativamente al incluir, en cada paso, la característica  $f$  que da como resultado la puntuación máxima  $G(S, D, M)$ , implementando la Ecuación (4.14).

$$S_d = S_{d-1} \cup \operatorname{argmax}G(S_{d-1} \cup f, D, M) \quad (4.14)$$

donde  $S$  es el subconjunto de características,  $D$  es el conjunto de datos utilizado y  $M$  denota el modelo de clasificación. El subconjunto de características se elige usando el método de  $K$  vecino más cercano (KNN, por sus siglas en inglés K-Nearest Neighbors), obteniéndolo a partir de la ecuación (4.15).

$$S = \operatorname{argmax}G(S, D, M) \quad (4.15)$$

### 4.5.4. Selección secuencial flotante hacia adelante

Selección secuencial flotante hacia adelante (SFFS, por sus siglas en inglés Sequential Forward Floating Selection) es un algoritmo de selección de características que es parte de la familia de algoritmos SFS. Este algoritmo funciona como un procedimiento de búsqueda ascendente que incluye nuevas características mediante la aplicación del procedimiento SFS básico a partir del conjunto de características actual, seguido de una serie de exclusión condicional sucesiva de la peor característica al conjunto recién actualizado, mientras esté susceptible a mejoras adicionales, se hará a los conjuntos anteriores [96].

### 4.5.5. Selección de características de subconjunto aleatorio

La selección de características de subconjunto aleatorio (RSFS, por sus siglas en inglés Random Subset Feature Selection), es un algoritmo que selecciona un lote que funcionan mejor que una característica promedio del conjunto original disponible, el cual se obtiene al evaluar repetida y aleatoriamente pequeños grupos de características que son evaluados con KNN para determinar su rendimiento. Durante cada iteración, la relevancia de cada característica se ajusta de acuerdo con el rendimiento de la clasificación del subconjunto, cada una se evalúa en términos de su utilidad promedio en el contexto de muchas otras combinaciones [94].

Los algoritmos SD, MD, SFS, SFFS y RSFS fueron programados en Matlab<sup>®</sup>, con base en las implementaciones disponibles en [97], pero adaptados a las necesidades de la base de datos. El resultado de la implementación se muestra en la Tabla 4.5 con la selección de las diferentes características más relevantes para cada algoritmo, en el Apéndice B se muestra con mayor detalle las características que fueron relevantes para todos los algoritmos.

Tabla 4.5: Características más relevantes obtenidas por diferentes algoritmos.

Método de selección	Características relevantes
Dependencia estadística	Ordenamiento por relevancia Todas posibles
Información mutua.	Ordenamiento por relevancia Todas posibles
Selección de características de subconjunto aleatorio	30
Selección secuencial flotante hacia adelante	16
Selección secuencial hacia adelante	9

Como se observa en la Tabla 4.5 y en Apéndice B, varias de las características de los algoritmos convergen entre sí, de la tabla anterior lleva a concluir y a tomar la decisión, para el entrenamiento de la red neuronal serán tomadas únicamente 30 de las 88 características, tal como fue el resultado del algoritmo RSFS.

## 4.6. Clasificación de señales EMG

Las técnicas de aprendizaje automático utilizadas en la clasificación de los árboles neuronales son máquinas de soporte vectorial, k vecinos más cercanos, regresión logística multinomial y red neuronal de propagación hacia atrás. Utilizando el toolbox de MatLab<sup>®</sup> llamado Classification Learner, es posible aplicar y comparar diferentes algoritmos de aprendizaje de manera sencilla, el único requisito es tener una matriz de características y un vector de salidas correspondiente a cada una de las entradas. Para el desarrollo de este proyecto el algoritmo de clasificación con mayor rendimiento fue la técnica de los k vecinos más cercanos [98].

El clasificador de k vecinos más cercano (KNN, por su siglas en inglés k Nearest Neighbors), es un método no paramétrico de aprendizaje computacional no supervisado, que memoriza todos los casos posibles de entrenamiento, ya que no hay un modelo asociado al concepto aprendido. El clasificador asume que las señales de entrenamiento y las señales desconocidas se encuentran en espacio paramétrico de características, por lo que el algoritmo necesita de una función de distancias para comparar las señales de entrenamiento con las nuevas. Los datos del clasificador se muestran en la Tabla 4.6.

Tabla 4.6: Parámetros del clasificador KNN

Parámetro	Valor
Número de vecinos	1
Función de distancia	Euclidiana
Peso en la distancia	Igual
Tipo de validación	Cruzada - 50 divisiones

La matriz de confusión resultante de la implementación del método de k vecinos más cercanos implementada se muestra en la Figura 4.11.

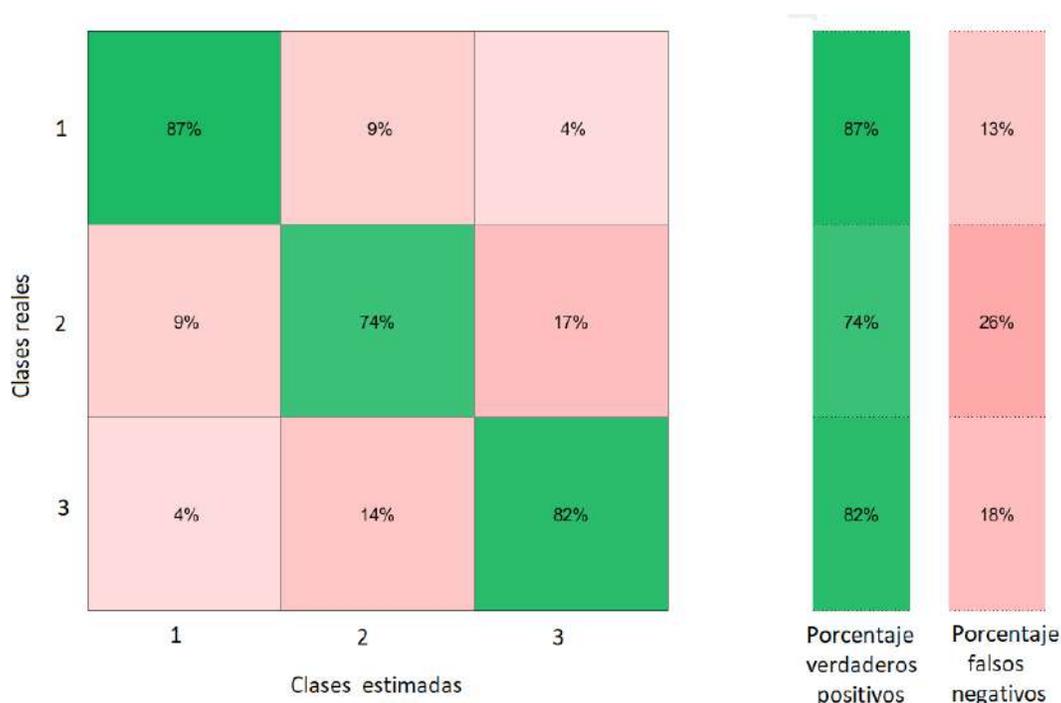


Figura 4.11: Matriz de confusión resultante.

De la matriz de confusión se observa que el porcentaje de verdaderos positivos es mayor en la detección del puño, para el agarre de punta se presentó un porcentaje de falsos negativos alta con el movimiento tres que es la pinza fina, así como viceversa. La gráfica de la curva ROC (Característica Operativa del Receptor) para el agarre de punta se muestra en la Figura 4.12. En esta representación de la eficiencia del algoritmo, cuanto mayor precisión se tenga en éste, la curva tendera a la esquina superior izquierda en la coordenada (0,1). Por el contrario, si el resultado de clasificación es malo, la curva tenderá hacia la esquina inferior derecha en la coordenada (1,0). El indicador AUC, es el área bajo la curva y puede ser interpretado como la exactitud global de la prueba.

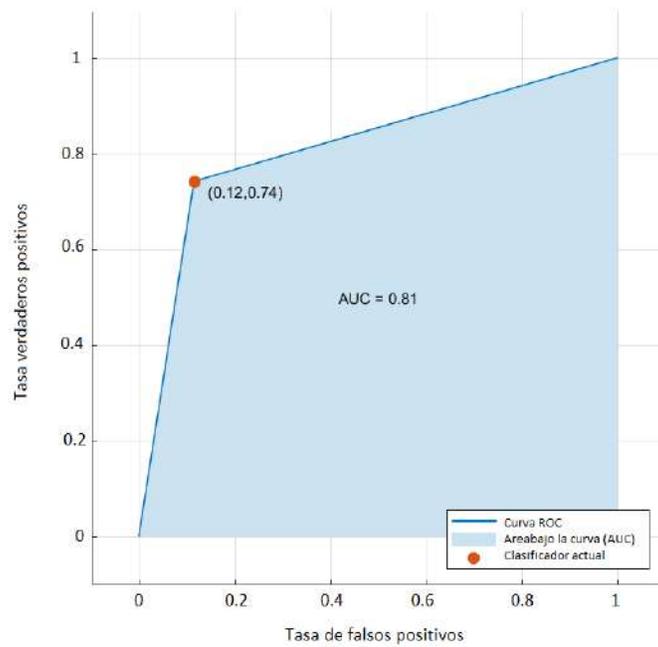


Figura 4.12: Curva ROC para el agarre de punta.

La gráfica de la curva ROC para el agarre de pinza fina se muestra en la Figura 4.13.

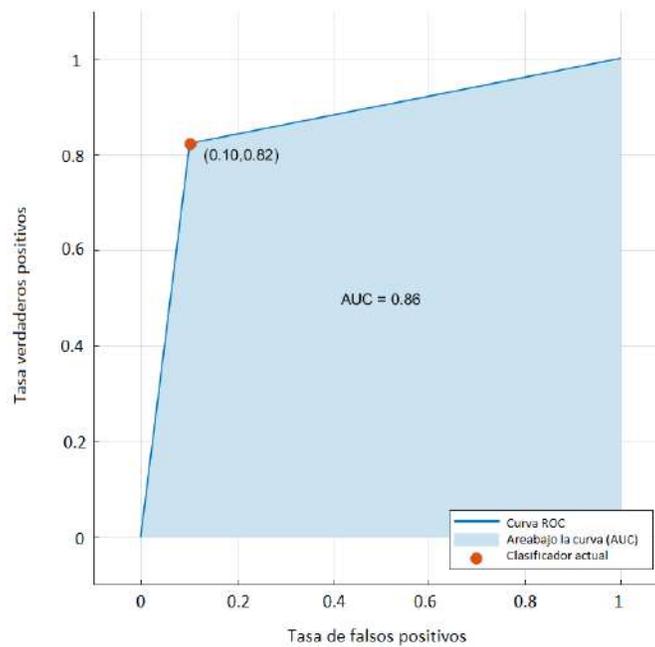


Figura 4.13: Curva ROC para la pinza fina.

La gráfica de la curva ROC para el cierre de puño se muestra en la Figura 4.14.

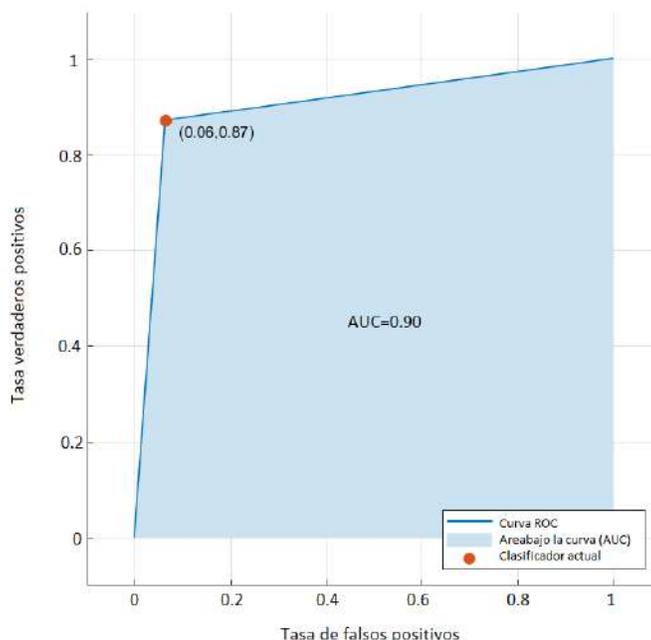


Figura 4.14: Curva ROC para el cierre de puño.

El desempeño total del algoritmo de clasificación es de 81.2%, debido a las confusiones que tuvo el sistema con los movimientos de agarre de punta y cierre de puño. En la literatura se encuentran reportados resultados mayores a un 90% de clasificación, pero en la gran mayoría de casos no se detallan las características de la base de datos o es el resultado de un gran número de muestras de una sola persona. Uno de los objetivos es crear una base de datos que proporcione robustez al sistema. Se busca la detección del movimiento independientemente del usuario y de la colocación exacta del brazaletes. Se reportó el número de muestras recolectadas por género y la edad de los participantes, la selección y extracción de características que permite el ahorro de tiempo de cómputo, así como la mejora del desempeño de la red. Se observó que el desempeño de la red se ve influenciado por la cantidad de características que son ingresadas a ellas, un mayor número de características no garantiza un mayor desempeño, por lo cual es conveniente la utilización de algoritmos para la selección de éstas. El preprocesamiento de las señales es un punto clave en el desempeño del algoritmo de aprendizaje, ya que se trabaja con la señal base obtenida de filtrar y normalizar mejora el desempeño final.

# Capítulo 5

## Diseño del sistema de control

En este capítulo se detalla el diseño del controlador difuso y los requerimientos que fueron obtenidos por medio de análisis de video y por captura de movimiento. Los requerimientos son planteados como posiciones angulares del rango de movimiento total que cada una de las falanges de los dedos tiene para cierre de puño, agarre de punta o pinza fina. Se describe la integración de los diferentes subsistemas con el sistema de control y la conexión entre estos. El controlador difuso se implementó en la tarjeta de desarrollo Arduino DUE, que a su vez se comunica con el exoesqueleto para conocer las posiciones angulares del motor y generar las señales de PWM pertinentes como salida del controlador difuso. Para interpretar las señales recibidas de los encoder, se caracterizaron con respecto al movimiento angular que presenta una falange y la señal leída en términos de revoluciones del encoder.

### 5.1. Descripción general del sistema

El sistema completo se compone de 5 subsistemas principales: detección del movimiento, clasificación del movimiento, asignación de referencia deseada, controlador y el exoesqueleto. El sistema comienza con la utilización del brazalete MYO en el brazo izquierdo del usuario, para detectar cuando se realiza algún movimiento, mediante la adquisición de señales EMG. Previamente se realiza un banco de datos para el entrenamiento del algoritmo de aprendizaje que determina qué tipo de movimiento está realizando el usuario en tiempo real. Ya que el movimiento es identificado, se asigna un valor a las posiciones deseadas de cada motor, previamente fuera de línea mediante captura de movimiento es posible determinar la trayectoria que cada una de las falanges necesita recorrer para realizar cierta acción. En el momento en que la trayectoria es asignada, los motores de CD que se encargan del movimiento del dedo del exoesqueleto se accionan, en conjunto con el sistema de control se debe seguir la referencia. En la Figura 5.1 se muestra el diagrama a bloques general de todo el sistema.

Como se observa en el diagrama, la función del sistema de control es primordial en el desarrollo del proyecto, ya que de tener un controlador que no cubra los requisitos del sistema es posible lesionar al usuario.

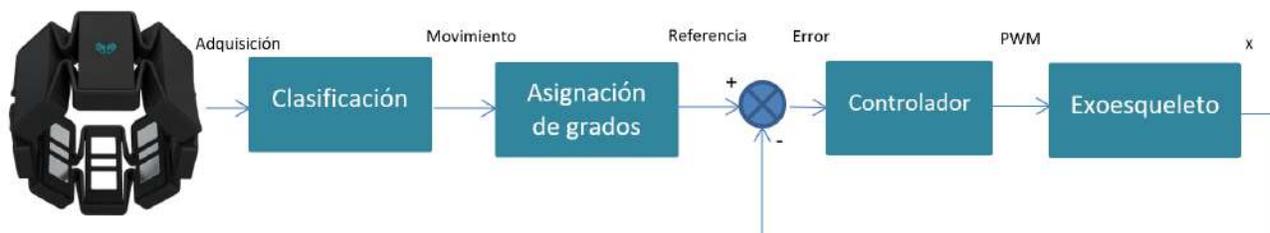


Figura 5.1: Diagrama abloques del sistema.

## 5.2. Obtención de requerimientos

Para la réplica de los movimientos básicos de la mano a desarrollar por el exoesqueleto, primero se caracterizaron los movimientos, existen diversos métodos para cuantificar los movimientos de las extremidades humanas, como es la captura de movimiento y análisis de video. Para adquirir la trayectoria que la falange proximal y media realizan durante cada uno de los movimientos, se utiliza análisis de video. Los movimientos predefinidos que el sistema debe realizar son cierre de puño, agarre de punta y pinza fina, estos movimientos son los básicos que se encuentran durante el desarrollo de terapias de rehabilitación y en la vida cotidiana de los seres humanos, desde tomar una pizca de sal, tomar una botella de agua o sostener una bolsa del supermercado. Los ángulos de movimiento para las falanges fueron obtenidos usando el software Kinovea, que es una herramienta para análisis de imagen y video ocupado comúnmente en la industria deportiva, ya que es posible determinar distancias, ángulos y velocidades en los análisis de video [99]. En la Figura 5.2 se muestra la ventana principal del software para el análisis de dos movimientos.

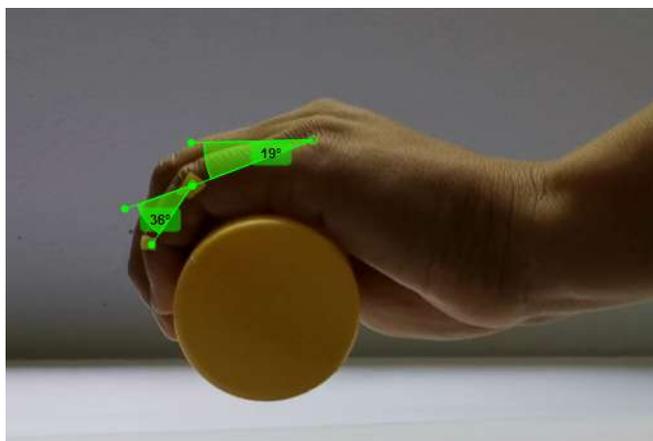


Figura 5.2: Vista principal del software Kinovea para la obtención de rangos de movimiento en cada uno de las falanges.

Los valores obtenidos para la falange proximal y media para cada uno de los movimientos se muestran a continuación en la Tabla 5.1. Estos valores serán la base para la generación de las trayectorias a seguir por el exoesqueleto.

Tabla 5.1: Rangos de movimientos para diferentes movimientos básicos de la mano.

Movimiento	Falange	Rango (grados)
Cierre de puño	Proximal	75
	Media	45
Agarre de punta	Proximal	30
	Media	735
Pinza fina	Proximal	40
	Media	36

Con los datos obtenidos de los ángulos final para cada una las falanges, se determinan las trayectorias que el exoesqueleto debe de seguir. Para la generación de la trayectoria existen diferentes métodos, como: curvas paramétricas, técnicas de interpolación, polinomios cúbicos, funciones spline, polinomios de Bézier, polinomios polares y concatenación de rectas con arcos y círculos. Uno de los métodos más utilizado para la unión de puntos, segmentos y generación de trayectorias son las curvas de Bézier, que son curvas expresadas con polinomios de Bernstein, implementadas en diferentes aplicaciones robóticas dentro de las cuales se encuentra robots móviles [100], robots manipuladores [101] y sistemas de suspensión [102]. Es posible obtener una trayectoria con los puntos intermedios en los que la falange se mueve, pero ésta puede presentar picos que ocasionarán cambios en la velocidad y posible vibración del motor, por lo cual se utiliza una trayectoria suave. Una trayectoria suave ayudará a no lesionar al usuario y cumplir con la posición deseada de una manera segura para el paciente y el dispositivo [103]. La curva de Bézier puede ser obtenida mediante la ecuación 5.1, [104].

$$B(t) = \sum_{i=0}^n \binom{n}{i} P_i (1-t)^{n-i} t^i = \binom{n}{0} P_0 (1-t)^n t^0 + \binom{n}{n-1} P_{n-1} (1-t)^1 t^{n-1} + \dots + \binom{n}{n} P_n (1-t)^0 t^n \quad (5.1)$$

donde:

$P_0, P_n$  son los puntos iniciales y finales de la curva de Bézier

$P_1, P_{n-1}$  son los puntos de control de la curva de Bézier

$t$  es el parámetro que influye en la distribución de la interpolación de los puntos.

Para la obtención de la trayectoria, el polinomio de Bézier fue implementado en Simulink, donde los parámetros a introducir para generar una trayectoria diferente son posición inicial, final y tiempo inicial y final en el que se desee que la trayectoria entre estos dos puntos se genere. En la Figura 5.3 se muestra la trayectoria obtenida para la Falange proximal en el agarre de punta, así como la velocidad, aceleración y el jerck (derivada de la aceleración).

Ya que la trayectoria ha sido generada es posible exportarla para su posterior uso en el sistema.

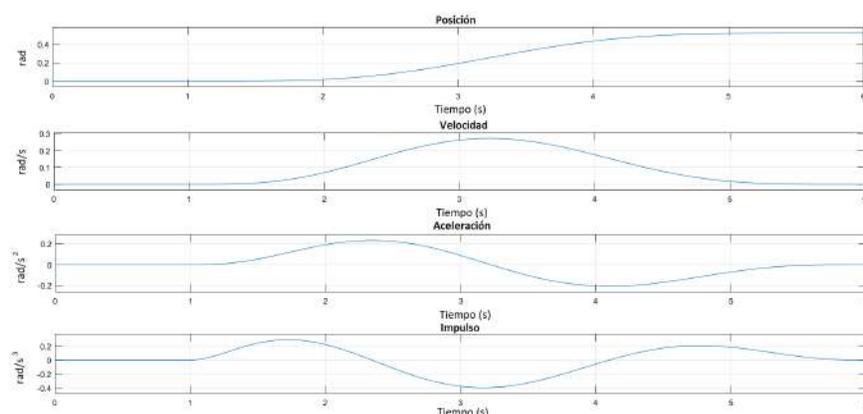


Figura 5.3: Trayectoria de la falange proximal para el agarre de punta generada con el polinomio de Bézier.

### 5.2.1. Adquisición de datos provenientes del encoder

Para comunicar la computadora con el exoesqueleto, se conecta una tarjeta que adquiera los pulsos por revolución del encoder para ser procesadas en la computadora y la señal de PWM correcta para llegar a la referencia. Para la selección de esta tarjeta se consideran varios aspectos como son: número de pines disponibles, números de PWM, capacidad de procesamiento y lenguaje de programación. Otro factor muy importante que hay que tener en cuenta es la memoria que ésta posee, ya que por la cantidad de motores a manejar y la recepción de señales a leer provenientes de los encoders, el programa final para realizarlo necesitará un espacio de almacenamiento mayor al que ofrecen algunas tarjetas de desarrollo básicas, para las capacidades que tienen algunas tarjetas. Se realizó una búsqueda con las tarjetas disponibles en el mercado, para determinar la tarjeta a usar durante el desarrollo del proyecto. Las tarjetas que fueron consideradas para esta tarea se muestran en la Tabla 5.2.

Tabla 5.2: Tabla comparativa de algunas tarjetas en el mercado.

Tarjeta	Raspberry Pi 3	Arduino Mega	Arduino Due
Procesador	ARM Cortex-A53	Atmega 2560	AT91SAM3X8E
RAM	1 GB	8 KB	96 KB
Frecuencia de reloj	1,2 GHz	16 MHz	84 MHz
Pines GPIO	40	54	54
Pines PWM	4	14	14
Peso	45 g	37 g	36 g
Alimentación	5 V	7-12V	7-12V

Por la cantidad de pines PWM que tiene con respecto a la cantidad que es necesario usar, se optó por una tarjeta Arduino, desafortunadamente la potencia de una versión Mega no llega a ser suficiente para la creación de proyectos ambiciosos. La opción de la tarjeta

Arduino DUE, proporciona la versatilidad de trabajar en un entorno de desarrollo sencillo, con las mismas capacidades de pines que un Arduino Mega pero con una memoria superior que permite la creación de proyectos mas grandes.

### 5.2.2. Diseño de la tarjeta de circuito impreso

Para la lectura de cada uno de los encoder que tienen los motores, se realiza la lectura de ellos con dos pines, un total de 16 pines de la tarjeta de desarrollo Arduino deben ser dedicados a esta tarea. Los pines para la generación de pulsos y señales que permitan trabajar con los cuatro controladores ocasionan problemas por la gran cantidad de cables que es necesario manejar. Para evitar tener un gran número de cables que comuniquen a los controladores con la tarjeta, se diseñó una tarjeta que contenga los controladores y se pueda montar sobre el Arduino. La cual fue desarrollada con el software Eagle. En la Figura 5.4 se muestra el diseño de la tarjeta de circuito impreso (PCB, por sus siglas en inglés).

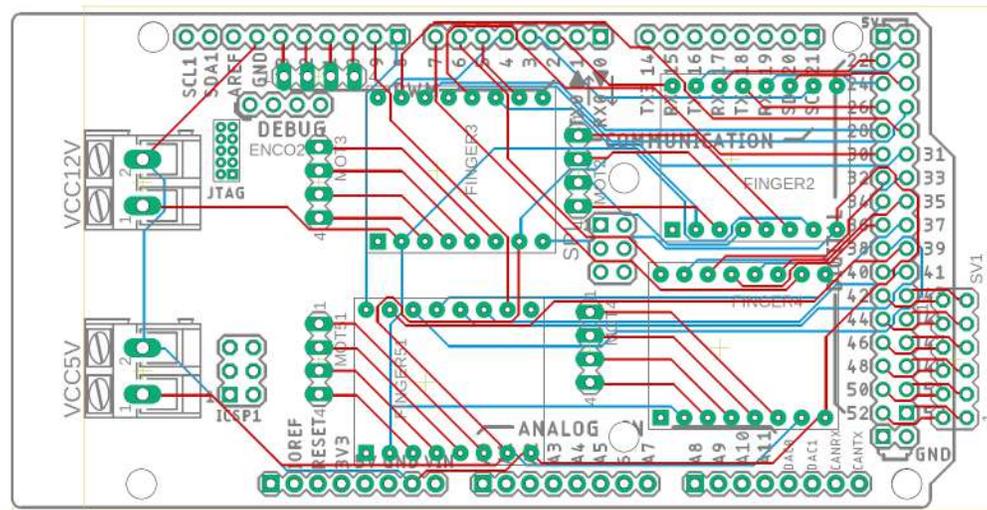


Figura 5.4: PCB diseñada en Eagle.

En la Figura 5.4 es posible observar que es una tarjeta de doble cara, con pintas en la capa superior e inferior. El diseño de la tarjeta es para ensamblarla directamente al Arduino sin necesidad de ocupar cables que hagan la comunicación a cada uno de los controladores. La PCB diseñada tiene pines de alimentación de 12V y 5V, que son para el voltaje de alimentación de los motores, encoder y controladores. Algunas tiras de pines están disponibles para la adquisición de las señales provenientes del encoder.

### 5.2.3. Caracterización de motores

Arduino es la tarjeta de desarrollo que adquiere las señales provenientes de los actuadores y al mismo tiempo se encarga de la generación de señales PWM. Estas señales deben generarse por un lapso diferente para cada uno de los actuadores, logrando así diferentes posiciones. Para determinar el lapso de tiempo que cada señal debe generarse, se adquiere la respuesta del motor, que está en relación de las vueltas generadas por el motor y el movimiento angular del mecanismo. La caracterización que se realizó está dada en función de la cantidad de vueltas leídas por el encoder y el movimiento angular que tiene el mecanismo para cada falange. Con la tarjeta Arduino fue posible adquirir el valor del encoder para algunos ángulos, los resultados se muestran en la Tabla 5.3.

Tabla 5.3: Datos obtenidos para la caracterización de los motores de CD sin carga.

Medición	Vueltas	Movimiento angular
1	534	6
2	1004	14
3	2046	18
4	3037	23
5	3539	25
6	4036	28
7	5047	34
8	5537	37
9	6034	43
10	6540	46
11	7033	50
12	8041	59
13	8843	64
14	9822	71
15	10523	75
16	10788	77
17	10800	79
18	11895	85
19	12759	90
20	13722	98

Algunos de los valores deseados se encuentran entre puntos intermedios de los datos de prueba obtenidos, para solucionar esta problemática, mediante interpolación lineal se generó una curva que describa la posición del mecanismo. El método utilizado para la interpolación fue Akima [105], el algoritmo proporciona buenos ajustes a los grupos de datos a través de la construcción de polinomios de tercer orden por partes, utilizando los valores vecinos para determinar los coeficientes del polinomio de interpolación. La curva que se muestra en la Figura 5.5 fue generada con los datos de la Tabla 5.3.

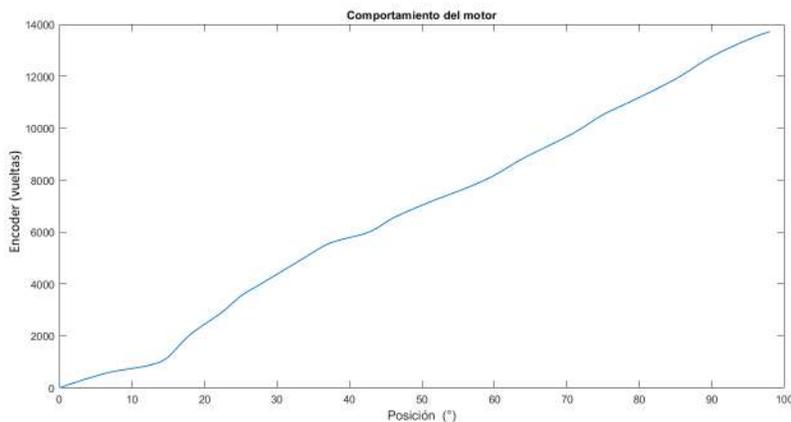


Figura 5.5: Interpolación del comportamiento del motor de CD sin carga.

Las siguientes pruebas realizadas fueron con el mecanismo y el dedo de un usuario promedio. La curva generada se muestra en la Figura 5.6.

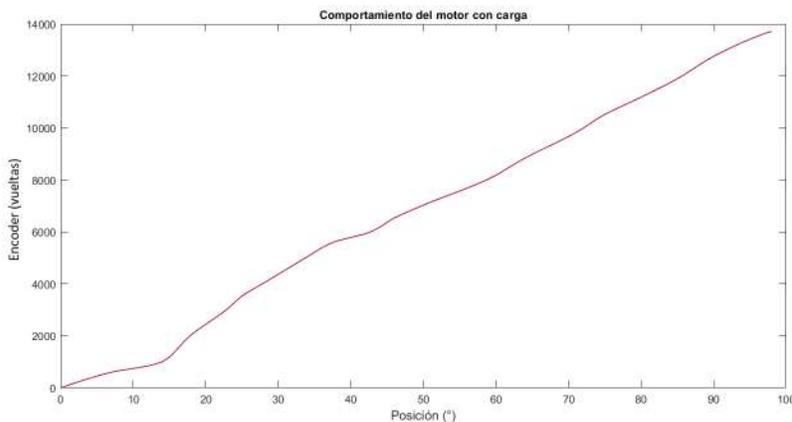


Figura 5.6: Interpolación del comportamiento del motor de CD con carga.

Los datos entre las dos pruebas tienen pequeñas variaciones. Con estos resultados se introduce el polinomio que describe el movimiento angular del mecanismo a la etapa de control, ya que de aquí se obtendrán la referencia para el controlador.

### 5.3. Selección de controlador

En la literatura se pueden encontrar una variedad de técnicas de control, algunas clásicas como el control Proporcional Derivativo (PD), Proporcional Integral Derivativo (PID) o algunas técnicas mas modernas que permiten mejorar la respuesta del sistema como el controlador Regulador Cuadrático Lineal (LQR, por sus siglas en inglés) [106] o el Modos Deslizantes (MD) [107]. Para realizar simulaciones, un requisito fundamental es conocer la dinámica del sistema.

#### 5.3.1. Modelo dinámico y linealización

Considerando el diseño mecánico del exoesqueleto y como consecuencia de que el sistema es totalmente actuado, para cada grado de libertad se tiene un actuador [108], cada dedeo se puede modelar como un manipulador planar de 2 grados de libertad (GDL). En la Figura 5.7 se muestra el modelo para un dedo del exoesqueleto, el resto de los dedos se modelan de igual forma.

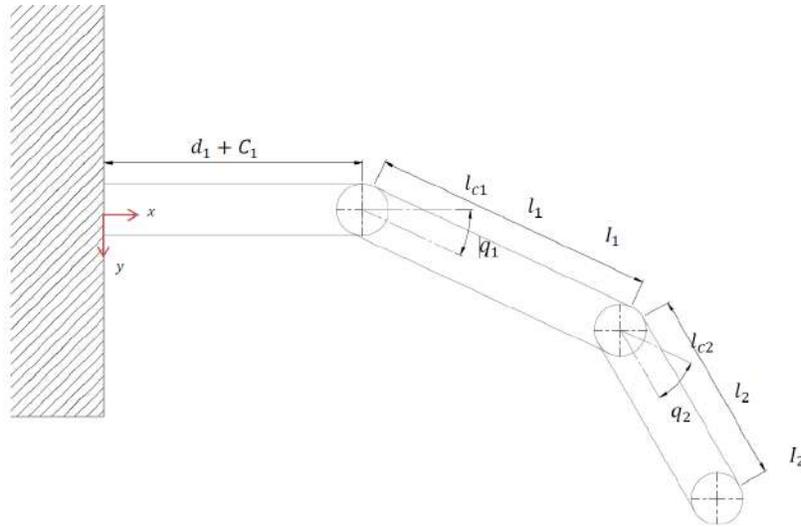


Figura 5.7: Modelo de un brazo de 2 grados de libertad.

Con el modelo del sistema se deducen las ecuaciones dinámicas que rigen al mecanismo. El análisis se detalla en el Apéndice D. Las ecuaciones (5.2) y (5.3) describen el modelo en su forma general.

$$\tau = M(q)\ddot{q} + C(q)\dot{q} + g(q) \quad (5.2)$$

O bien:

$$\begin{pmatrix} \tau_2 \\ \tau_3 \end{pmatrix} = \begin{bmatrix} M_{11}(q) & M_{12}(q) \\ M_{21}(q) & M_{22}(q) \end{bmatrix} \begin{bmatrix} \ddot{q}_2 \\ \ddot{q}_3 \end{bmatrix} + \begin{bmatrix} C_{11}(q, \dot{q}) & C_{12}(q, \dot{q}) \\ C_{21}(q, \dot{q}) & C_{22}(q, \dot{q}) \end{bmatrix} \begin{bmatrix} \dot{q}_2 \\ \dot{q}_3 \end{bmatrix} + \begin{bmatrix} g_2(q) \\ g_3(q) \end{bmatrix} \quad (5.3)$$

Como se observa en los parámetros de la ecuaciones dinamicas en el Apéndice D, el modelo resultante es de un sistema no lineal, se puede linealizar el sistema [109]. Linealizando por series de Taylor [110], las matrices del sistema en su modelo en variables de estado están definidas por:

$$A = \begin{pmatrix} \frac{\partial \dot{x}_1}{\partial x_1} & \frac{\partial \dot{x}_1}{\partial x_2} & \frac{\partial \dot{x}_1}{\partial x_3} & \frac{\partial \dot{x}_1}{\partial x_4} \\ \frac{\partial \dot{x}_2}{\partial x_1} & \frac{\partial \dot{x}_2}{\partial x_2} & \frac{\partial \dot{x}_2}{\partial x_3} & \frac{\partial \dot{x}_2}{\partial x_4} \\ \frac{\partial \dot{x}_3}{\partial x_1} & \frac{\partial \dot{x}_3}{\partial x_2} & \frac{\partial \dot{x}_3}{\partial x_3} & \frac{\partial \dot{x}_3}{\partial x_4} \\ \frac{\partial \dot{x}_4}{\partial x_1} & \frac{\partial \dot{x}_4}{\partial x_2} & \frac{\partial \dot{x}_4}{\partial x_3} & \frac{\partial \dot{x}_4}{\partial x_4} \end{pmatrix} \quad (5.4)$$

$$B = \begin{pmatrix} \frac{\partial \dot{x}_1}{\partial \tau_1} & \frac{\partial \dot{x}_1}{\partial \tau_2} \\ \frac{\partial \dot{x}_2}{\partial \tau_1} & \frac{\partial \dot{x}_2}{\partial \tau_2} \\ \frac{\partial \dot{x}_3}{\partial \tau_1} & \frac{\partial \dot{x}_3}{\partial \tau_2} \\ \frac{\partial \dot{x}_4}{\partial \tau_1} & \frac{\partial \dot{x}_4}{\partial \tau_2} \end{pmatrix} \quad (5.5)$$

$$C = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{pmatrix} \quad (5.6)$$

Sustituyendo los valores de los parámetros del sistema y la evaluación de los puntos de equilibrio, las matrices que describen al sistema de forma linealizada y en espacio de estados son:

$$A = \begin{pmatrix} 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 1 \\ -0,4568 & -0,6196 & 0 & 0 \\ 0,2485 & -6,6174 & 0 & 0 \end{pmatrix} \quad (5.7)$$

$$B = \begin{pmatrix} 0 & 0 \\ 0 & 0 \\ 0,7870 & 0 \\ 0 & 0,1349 \end{pmatrix} \quad (5.8)$$

para los puntos de equilibrio  $q_1 = \frac{\pi}{2}$ ,  $q_2 = \frac{\pi}{2}$ ,  $\dot{q}_1 = 0$  y  $\dot{q}_2 = 0$

El control PID es uno de los mas usados y reportados en la literatura para sistemas de rehabilitación, su simplicidad y desempeño lo vuelven atractivo para su implementación. Con el desarrollo de la inteligencia artificial se han desarrollado controladores PID con diferentes técnicas como es el caso de algoritmos genéticos o por enjambre de partículas [111].

### 5.3.2. Diseño de un controlador PID

Un controlador PID recibe su nombre de los elementos que lo componen, una acción Proporcional-Integral-Derivativa. El controlador calcula la diferencia entre la salida del sistema contra el valor deseado. El parámetro Proporcional (P) del controlador responde a los cambios presentes en la entrada y generara inmediatamente y proporcionalmente cambios en la salida. El Integral (I) disminuye y elimina el error en estado estacionario. El Derivativo (D) emite una acción predictiva, prevé el error e inicia una acción oportuna, reacciona a qué tan rápido cambia la entrada respecto al tiempo, alterando la señal de salida en proporción con la tasa de cambio de entrada [112]. La ecuación (5.9) describe el controlador PID.

$$U = K_p e(t) + K_i \int_0^t e(t) dt + K_d \frac{d}{dx} e(t) \quad (5.9)$$

donde las variables  $K_p$ ,  $K_i$  y  $K_d$  son las ganancias proporcional, integral y derivativa, respectivamente. Para el correcto funcionamiento del controlador se sintonizan las ganancias, ya que su valor influye directamente en la respuesta del sistema.

Una forma rápida de encontrar los valores para cada uno de los parámetros del controlador es realizar una simulación en Simulink de Matlab®, En la Figura 5.8 se muestra el diagrama a bloques simulado. Los valores de referencia son curvas de Bézier para cada uno de los motores denotadas con  $q_{1d}$  y  $q_{2d}$  para las posiciones finales obtenidas con el sistema de captura de movimiento, las salidas del sistema del controlador son  $q_1$  y  $q_2$ .

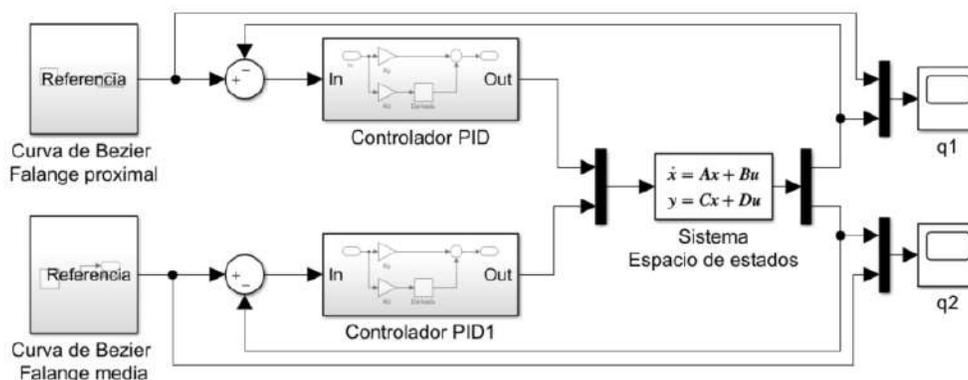


Figura 5.8: Diagrama a bloques del controlador PID en Simulink.

La respuesta del sistema se muestra en la Figura 5.9. La gráfica superior corresponde a la falange proximal para un ángulo final de movimiento de  $54.81^\circ$ , la gráfica inferior corresponde a la falange media con una referencia final de  $45.19^\circ$ . La curva de la respuesta y la de referencia se muestran en diferente color.

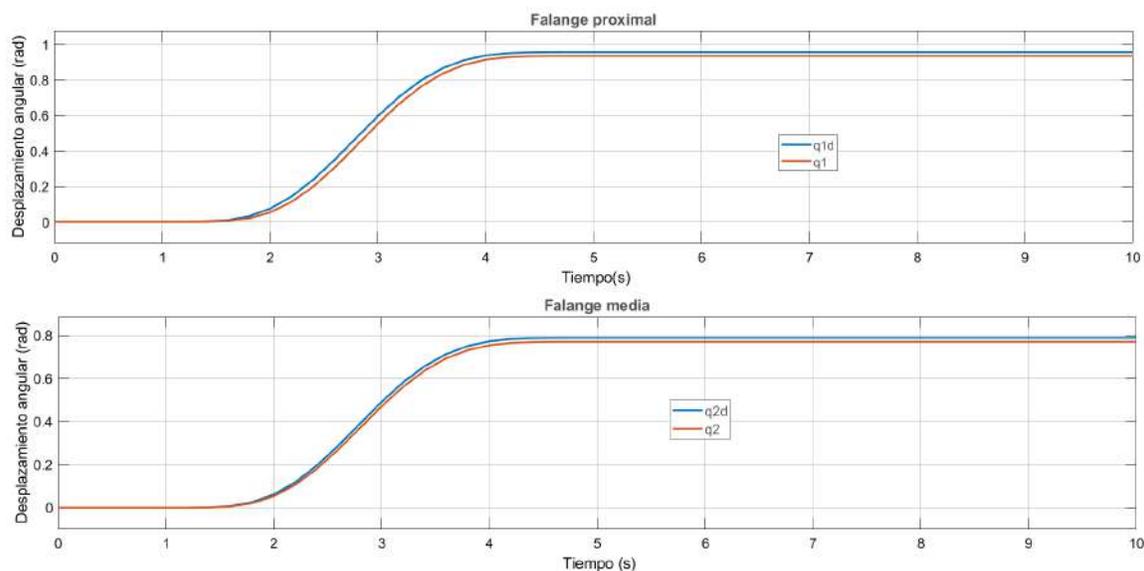


Figura 5.9: Respuesta en simulación del controlador PID para un agarre de punta.

Para las pruebas realizadas el error generado corresponde a un 2.2 % para la falange proximal y un 2.3 % para la falange media, su equivalente en grados es  $1.20^\circ$  y  $1.03^\circ$  correspondientemente.

Se ha realizado la aplicación de controladores PID en diferentes dispositivos enfocados a rehabilitación, la gran mayoría para el miembro superior sin abarcar la mano. En [113] se comparan distintos controladores para un dispositivo de rehabilitación de mano, uno de los que obtuvo mejor respuesta fue el PID. Actualmente se encuentra en desarrollo el diseño y la aplicación de técnicas de control basadas en inteligencia artificial, como en [114] que presenta los resultados de un controlador PID difuso para un rehabilitador brazo de 5 GDL y en [115] un controlador PD difuso para uno de 2 GDL, obteniendo errores menores al 1% y la ventaja de tener ganancias adaptables sin necesidad de volver a sincronizarlas. Resultados similares fueron reportados en [116] para un controlador difuso PD comparado con un PID clásico, razón por la cual se decidió implementar un control difuso PD en el exoesqueleto, además de que pocos sistemas para mano han implementados técnicas de control basadas en inteligencia artificial.

## 5.4. Diseño del controlador difuso

El control por lógica difusa se ha basado en las pautas marcadas por [72], en las que se parte de un control derivativo proporcional (PD) clásico, ya que el controlador combina el error y el cambio en él, obteniendo una versión difusa de un controlador de derivada proporcional. Un controlador PD es necesariamente de lazo cerrado para realizar la comparación de la entrada de referencia con la salida obtenida por el controlador, además de ser de algunos de dos tipos: de múltiples entradas-salida (MIMO) o de una sola entrada-salida (SISO). El controlador SISO típico regula una señal de control de acuerdo con una señal de error. El sistema del exoesqueleto contiene un motor para el movimiento de cada falange, lo que resulta

en tener un control descentralizado tipo SISO para el movimiento de cada una de los motores. Para la implementación del control difuso, las entradas del sistema son los valores de referencia en grados para el movimiento final que tiene que llegar cada una de las falanges. La posición angular de la salida del sistema de control se resta con la posición angular de referencia en la entrada, obteniendo así un error del cual se obtiene su variación en el tiempo [117].

### 5.4.1. Etapa de fuzificación.

Los controladores de tipo difuso tienen entradas de conjuntos de funciones de pertenencia, por lo que las variables de entrada son de tipo lingüística, para este controlador las variables lingüísticas son Error y Cambio, en este punto del desarrollo del controlador es donde las variables de entrada quedan como funciones de membresía. Los términos lingüísticos que se usaron para cada una de las variables lingüísticas se muestran en la Tabla 5.4 en su forma abreviada.

Tabla 5.4: Términos lingüísticos implementados en el controlador difuso.

Variable lingüística	Abreviatura	Término lingüístico
Error	NE	Error negativo
	ZE	Error cero
	PE	Error positivo
Cambio	NC	Cambio negativo
	ZC	Cambio cero
	PC	Cambio positivo
Voltaje	NV	Voltaje negativo
	ZV	Voltaje cero
	PV	Voltaje positivo

Las entradas del controlador difuso son el Error ( $e$ ) y el Cambio del error ( $de/dt$ ). El error está definido como la diferencia entre el ángulo deseado y el ángulo real, los valores de los ángulos están en relación con los pulsos detectados por el encoder y la interpolación mostrada en la Figura 5.6. El rango de valores entre los que está definido el error son  $[-1,1]$ , en cada lectura del encoder el error es actualizado y normalizado como se describe en la ecuación (5.10). Un error con un valor de 1 quiere decir que la falange está en  $0^\circ$ .

$$e = (Referencia - Actual) / Referencia \quad (5.10)$$

El cambio del error es obtenido mediante la ecuación (5.11). El rango de valores para esta variable lingüística se encuentra entre  $[-1,1]$ .

$$de = e_{actual} - e_{anterior} \quad (5.11)$$

La Figura 5.10 muestra las funciones de pertenencia que han sido implementadas en el controlador, las cuales son definidas mediante funciones trapezoidales. Estas funciones son de manera simétrica, el comportamiento del motor debe ser de la misma forma en ambos sentidos.

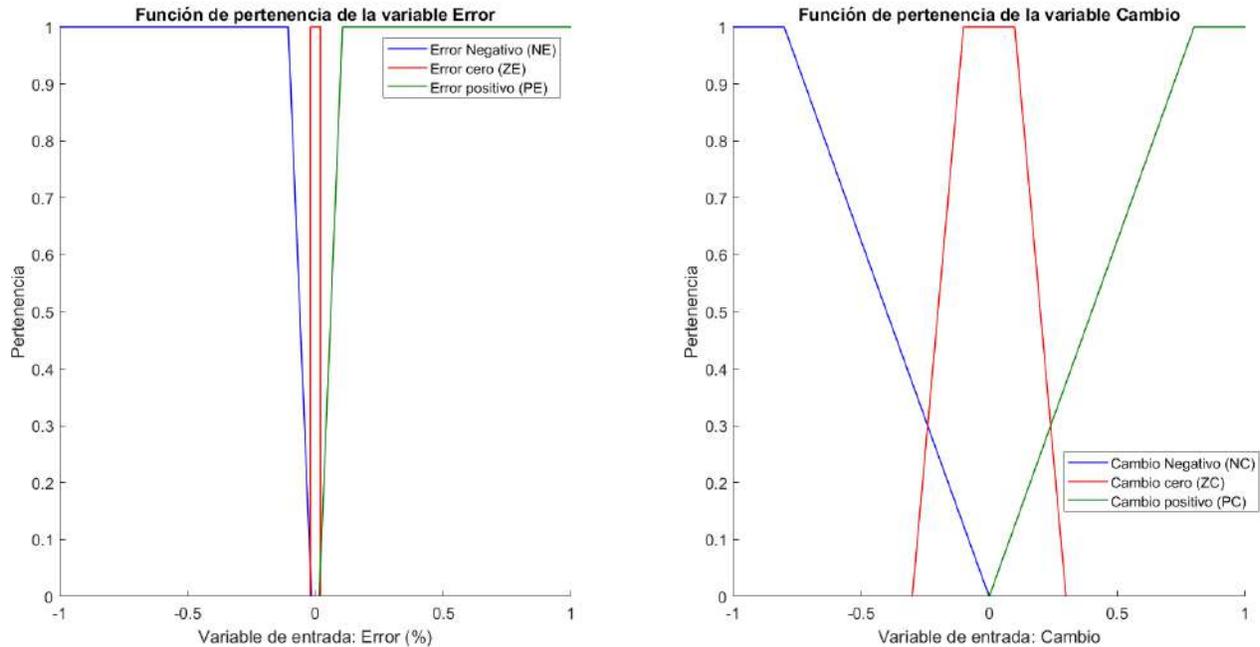


Figura 5.10: Función de membresía de la variable lingüística de entrada Error y Cambio.

La salida del controlador es un valor correspondiente al ciclo alto del modulador de ancho de pulso (PWM, por sus siglas en inglés), que controla la velocidad del actuador. Las funciones que modelan los términos lingüísticos son de tipo trapezoidal. El controlador es implementado en la tarjeta de desarrollo Arduino, para la generación de la señal de PWM trabaja con una resolución de 8 bits [0-255], lo que equivale al 0-100 % del ciclo de trabajo. Para la realización de pruebas en este proyecto el máximo ciclo de trabajo a generar por la tarjeta es de 21 %, ya que a esta velocidad el sistema de captura movimiento puede adquirir datos consecutivos de las marcas sin presentar confusiones o lecturas muy alejadas unas de otras. Escalando los valores máximos a su equivalente en 8 bits, se encuentra en un rango de [-53, 53]. La función de membresía se muestra en la Figura 5.11.

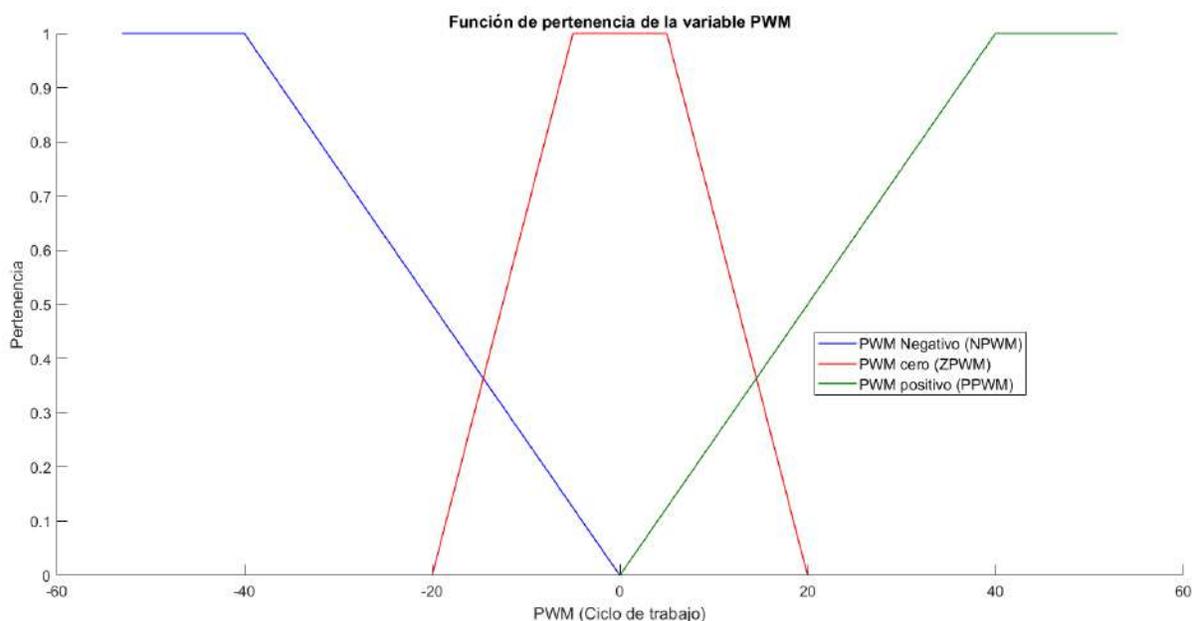


Figura 5.11: Función de membresía de la salida de un controlador difuso PD.

### 5.4.2. Reglas difusas.

Para relacionar las entradas con las salidas se establecen reglas difusas haciendo combinaciones de los grupos de entrada, dado que se tienen dos entradas con tres términos lingüísticos cada una, el número de reglas que se van a obtener son 9. Las reglas se observan en la Tabla 5.5.

Tabla 5.5: Reglas del controlador difuso.

No.	Reglas					
1	Si	(NE)	y	(NC)	Entonces	(NV)
2	Si	(NE)	y	(ZC)	Entonces	(NV)
3	Si	(NE)	y	(PC)	Entonces	(NV)
4	Si	(ZE)	y	(NC)	Entonces	(NV)
5	Si	(ZE)	y	(ZC)	Entonces	(ZV)
6	Si	(ZE)	y	(PC)	Entonces	(PV)
7	Si	(PE)	y	(NC)	Entonces	(PV)
8	Si	(PE)	y	(ZC)	Entonces	(PV)
9	Si	(PE)	y	(PC)	Entonces	(PV)

La Tabla 5.5. debe interpretarse como: “ Si el error es positivo y la derivada del error es negativa, entonces PWM positivo”. Esto de manera física en la falange, se traduce como: “ Para un ángulo de movimiento inferior al deseado y el ángulo medido va en aumento, entonces generar PWM positivo”.

### 5.4.3. Máquina de inferencia.

El método de inferencia que se empleará para la intersección difusa es el método de Mandani. Para obtener esta fase se hace uso de la herramienta de software MatLab®, con la ayuda de la herramienta de lógica difusa. Introduciendo las variables de entrada y salida se obtiene el sistema que se muestra en la Figura 5.12, se puede observar la pantalla principal de la herramienta para la creación del sistema difuso.

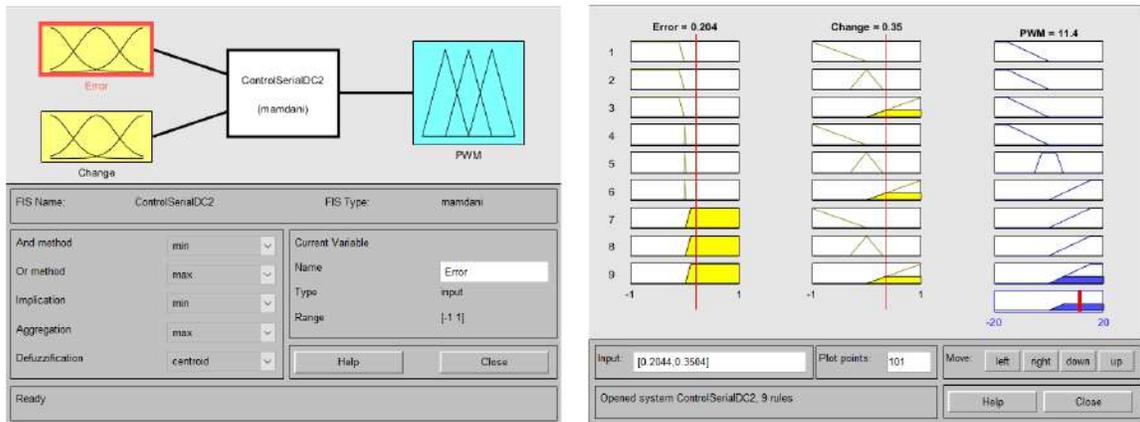


Figura 5.12: Pantalla principal de la herramienta de Matlab® para la creación del sistema difuso.

La gráfica de la Figura 5.13 muestra el comportamiento de la variable de salida del controlador difuso, señal de PWM en el controlador, en función de las variaciones en las señales de entrada.

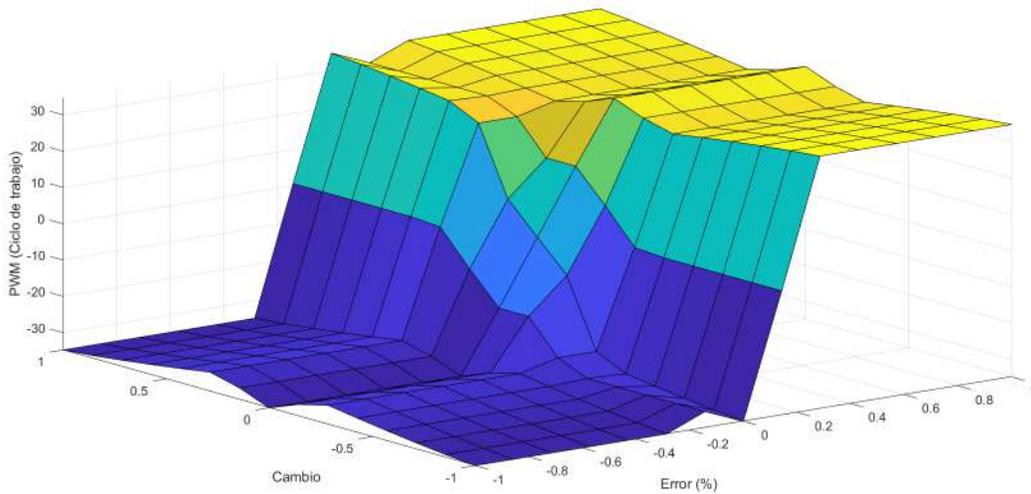


Figura 5.13: Relación de variables de entrada y salida del sistema difuso.

Matlab® es una herramienta útil para la calibración de un sistema difuso, gracias a su interfaz gráfica y la posibilidad de evaluar los resultados con la superficie de deslizamiento,

la ubicación de los parámetros de las funciones de membresía se vuelve una tarea menos complicada, de lo contrario el ajuste de estos parámetros deben hacerse a prueba y error con el sistema real.

## 5.5. Implementación del controlador difuso en Arduino

El control de los motores de CD se realiza en la tarjeta de adquisición de datos Arduino, para esto se crea el sistema difuso en este mismo lenguaje, con los parámetros obtenidos de Matlab® y las librerías correspondientes es diseñado. Para la implementación del código se cuenta con las siguientes librerías de Arduino: Biblioteca para el controlador TB6612FNG y la biblioteca de lógica difusa. El funcionamiento general de la implementación del código de Arduino se muestra en el diagrama de la Figura 5.14.

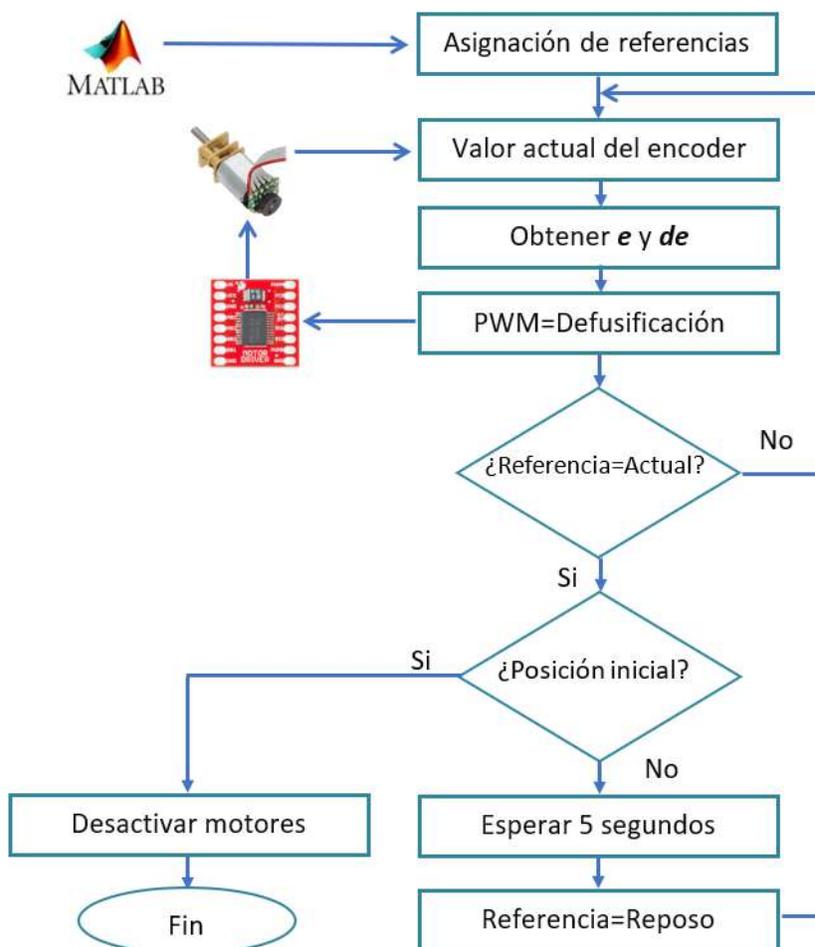


Figura 5.14: Diagrama del controlador implementado en Arduino

Los valores a los que cada uno de los motores debe llegar y los cuales son adquiridos con el encoder, son recibidos de Matlab® mediante comunicación serial, al momento de recibir los

datos, estos son asignados como los valores de referencia. Durante cada iteración la posición actual del motor es actualizado, con el cual se obtienen los valores del error y del cambio del error, para deducir la señal de PWM a generar por el Arduino, los valores se actualizan en cada iteración por medio de la defuzzificación del sistema difuso. Cuando el controlador difuso alcanza los valores deseados, el sistema hace una pausa de 6 segundos para reanudar el movimiento de los motores, pero ahora en forma inversa y retornar las falanges a la posición inicial. En las siguientes subsecciones se explica las principales líneas de código de Arduino relacionadas con cada una de las librerías usadas.

### 5.5.1. Biblioteca para el controlador TB6612FNG

La librería está disponible en [118] de forma gratuita. La primer parte es la declaración de los pines de Arduino conectados a cada uno de los pines del encoder. A continuación se muestran las líneas de código implementadas para la asignación de los pines de la tarjeta de adquisición de datos con los pines de los controladores.

```
1 //Motor 2 Falange Proximal
2 int pinAIN1 = 22; //Direction
3 int pinAIN2 = 23; //Direction
4 int pinPWMA = 2; //Speed
5 //Motor 2 Falange Media
6 int pinBIN1 = 24; //Direction
7 int pinBIN2 = 25; //Direction
8 int pinPWMB = 3; //Speed
9 //Standby Dedo 2
10 int pinSTBY2 = 26;
```

Ya que se han asignado los pines es necesario declarar si son de salida o de entrada, en este caso para el controlador todos los pines son declarados de salida, como se muestra a continuación.

```
1 pinMode(pinPWMA, OUTPUT); //
   Motor 2 Proximal
2 pinMode(pinAIN1, OUTPUT);
3 pinMode(pinAIN2, OUTPUT);
```

La declaración de los pines para cada uno de los Encoders se realiza de la siguiente manera.

```
1 Encoder myEnc21(42, 43);
```

Estos pines son encargados de adquirir los datos provenientes de cada uno de los encoders del motor, la declaración anterior corresponde para el motor encargado del movimiento de la falange proximal del dedo índice. Para la adquisición de los datos durante la ejecución del programa:

```
1 count21=myEnc21.read();
```

Cuando los datos son adquiridos la siguiente fase es su introducción al sistema de control y la obtención de una salida para la regulación de la salida PWM hacia los motores.

### 5.5.2. Biblioteca de lógica difusa

Esta librería se encuentra disponible en [119], no tiene un límite para el número de entradas, salidas y reglas difusas. Para construir cada una de las funciones difusas es necesario asignar cuatro parámetros(A, B, C, D), ya que por defecto éstas son de forma trapezoidal, si se desea construir una forma triangular o singleton solo es necesario ajustar los valores de los parámetros acorde a la forma que se desea obtener. En la Figura 5.15 se muestra la forma de construir estas funciones.

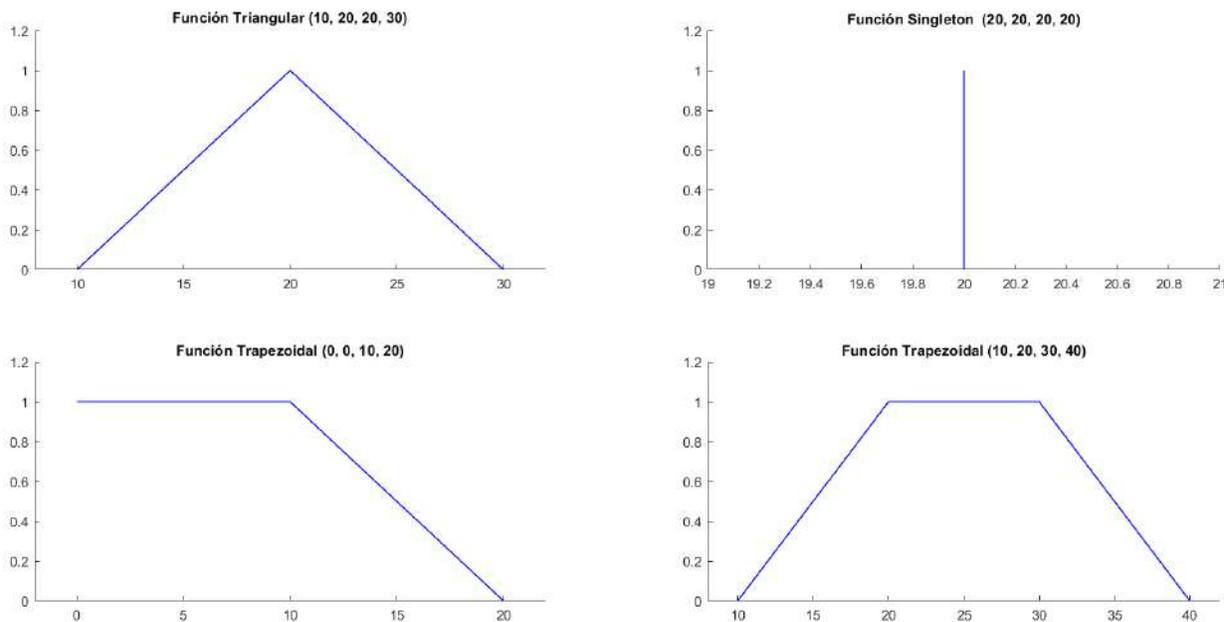


Figura 5.15: Formas de declarar una función difusa en Arduino

La declaración de las funciones difusas es el primer paso para la construcción del sistema. Con la siguiente sentencia se crea una función trapezoidal para la variable error negativo.

```
1 FuzzySet *NE = new FuzzySet (-1.72, -1.08, -0.106, -0.015);
```

En la Tabla 5.6 se muestran los valores para cada uno de los cuatro argumentos que son necesarios para crear cada una de las variables del sistema difuso.

Para crear cada uno de los objetos de entrada y salida equivalentes a las variables, se debe asociar cada uno de ellos con las correspondientes funciones difusas. Es decir, de los términos lingüísticos que fueron declarados con anterioridad, son agrupados de acuerdo a cada variable

Tabla 5.6: Términos lingüísticos implementados en el controlador difuso.

Término lingüístico	Abreviatura	Valores			
		A	B	C	D
Error negativo	NE	-1.72	-1.08	-0.106	-0.015
Error cero	ZE	-0.02	-0.02	0.02	0.02
Error positivo	PE	0.015	0.106	1.08	1.72
Cambio negativo	NC	-1	-1	-1	0
Cambio cero	ZC	-0.3	0	0	0.3
Cambio positivo	PC	0	1	1	1
Voltaje negativo	NV	-62	-40	-29	0
Voltaje cero	ZV	-10	-5	5	10
Voltaje positivo	PV	0	29	40	62

lingüística. En las siguientes líneas de código se agrupan los termino de “Error Negativo”, “Error Cero” y “Error Positivo” con la variable “Error”.

```

1 FuzzyInput *Error = new FuzzyInput(1);
2 Error->addFuzzySet(NE);
3 Error->addFuzzySet(ZE);
4 Error->addFuzzySet(PE);
5 fuzzy->addFuzzyInput(Error);

```

Como el sistema solo contiene una salida, que es el valor de PWM para los motores, no se hacen mas asociaciones. Como ya se han declarado las relaciones que existen entre las variables lingüísticas, se crean las reglas difusas. Para declarar una regla difusa en Arduino, hay que dividirla en tres partes. La primer parte es la declaración de los antecedentes, en esta parte se asocian las variables de entrada que están involucradas en una regla. La segunda parte es la declaración de las consecuencias, se declaran cuales son variables lingüísticas de salida asociadas como resultado del cumplimiento de cada regla. La tercer parte es la agrupación de la causa con cada una de las consecuencias. Como se muestra en las siguientes líneas de código.

```

1 FuzzyRuleAntecedent *ErrorNEAndChangeNC = new
  FuzzyRuleAntecedent();
2 ErrorNEAndChangeNC->joinWithAND(NE, NC);
3 FuzzyRuleConsequent *thenNPWM= new FuzzyRuleConsequent();
4 thenNPWM->addOutput(NPWM);
5 FuzzyRule *fuzzyRule1 = new FuzzyRule(1,
  ErrorNEAndChangeNC, thenNPWM);
6 fuzzy->addFuzzyRule(fuzzyRule1);

```

Las líneas anteriores de código corresponden a la primer regla del sistema, cuando hay un “Cambio Negativo” y un “Error Negativo” entonces la salida es un voltaje de PWM negativo. El último paso se encuentra dentro del código principal, la defuzzificación, es necesario introducir los valores correspondientes a cada una de las variables de entrada.

```
1 fuzzy->setInput (1, e21);  
2 fuzzy->setInput (2, de21);  
3 fuzzy->fuzzify();  
4 int Accion21=int(fuzzy->defuzzify(1));
```

Se introducen los valores correspondientes al error y la derivada del error del motor que se encarga del movimiento de la falange proximal del dedo índice. El código completo se encuentra disponible en el Apéndice E.

## 5.6. Diseño de la interfaz gráfica de usuario

Los usuarios potenciales para el sistema necesitan tener acceso al exoesqueleto de una manera amigable y sencilla, con el desarrollo de una interfaz gráfica de usuario (GUI, por sus siglas en inglés) la interacción entre ambos lados se vuelve más amigable. De acuerdo con [120], se define como parte de una computadora y su software que el usuario tenga la capacidad de oír, hablar, tocar o entender de forma directa. La interfaz por desarrollar debe recibir las señales EMG y enviar las referencias para cada falange del exoesqueleto, sin necesidad de que el usuario interfiera en este proceso. El usuario debe ser capaz de interactuar con el software después de una instrucción básica, tener nociones de cómo funciona de manera interna y las configuraciones posibles, el acceso que tiene a al sistema interno es mínimo, solo permite agregar nuevas pruebas a la base de datos. Para lograr esto se considera durante el diseño del GUI:

- Textos claros y concisos
- Gráficos sencillos y entendibles
- Selección de colores comúnmente usados en el sector salud
- Menús intuitivos
- Integración de todos los subsistemas
- Presentación de los elementos de manera accesible

Los requerimientos de este tipo de sistemas deben ser propuestos con base en conceptos de usabilidad, para que la aplicación sea fácil de usar y de aprender, para esto hay que considerar tres conceptos durante el diseño: que satisfaga los requerimientos, que los realice de manera efectiva y que el usuario se sienta cómodo mientras interactúa con ella.

De acuerdo con los requerimientos establecidos, la interfaz gráfica se desarrolló en Matlab<sup>®</sup>, debido a que el procesamiento y clasificación de las señales son realizadas en este software. La

herramienta GUIDE permite crear interfaces de usuario de manera gráfica y con la ventaja de utilizar los recursos de Matlab®. La pantalla principal del GUI desarrollado se muestra en la Figura 5.16 y se divide en cuatro secciones principales:

- Ejecución: en esta sección se encuentran las opciones de Iniciar y Parar, cuando el usuario presiona el botón de Iniciar, el sistema comienza la ejecución con la adquisición de las señales EMG y el movimiento del exosqueleto
- Notificaciones: Durante la ejecución de la aplicación se muestran diferentes mensajes que permiten al usuario si es necesario realizar algún movimiento o para notificarle en qué parte de la prueba se encuentra.
- Sistema: Es posible agregar una nueva prueba a la base de datos y posteriormente recalibrar el sistema para agregar robustez.
- Gráficos: En esta sección se muestran los tres movimientos que el sistema detecta, cambian de color dependiendo del movimiento que es detectado, cuando se reciben las señales EMG se muestran aquí también.

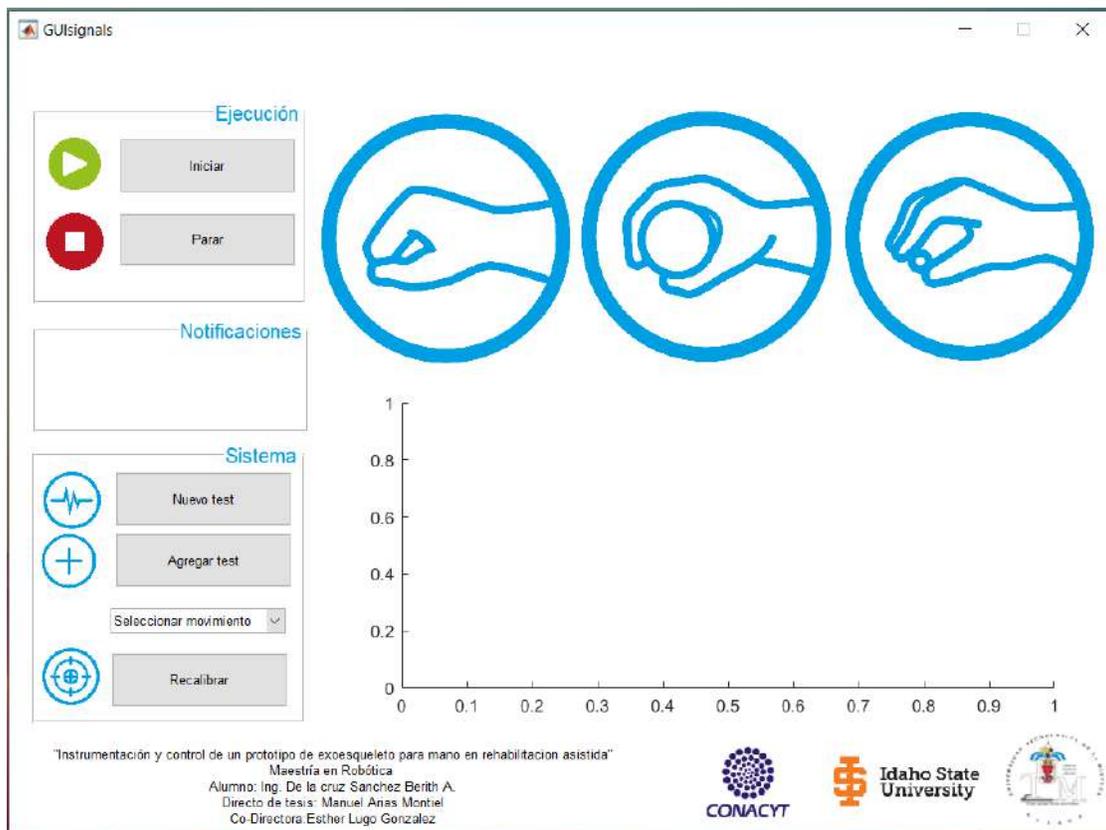


Figura 5.16: Pantalla inicial de la Interfaz Grafica de Usuario

En la parte inferior del GUI se puede encontrar información de contacto y los datos del proyecto durante el cual fue desarrollado.

### 5.6.1. Realización de una nueva prueba

Para iniciar una nueva prueba solo se pulsar el botón de “Iniciar”, el sistema mostrará diferentes leyendas en la sección de notificaciones, los mensajes en orden de aparición se muestran en la Figura 5.17.

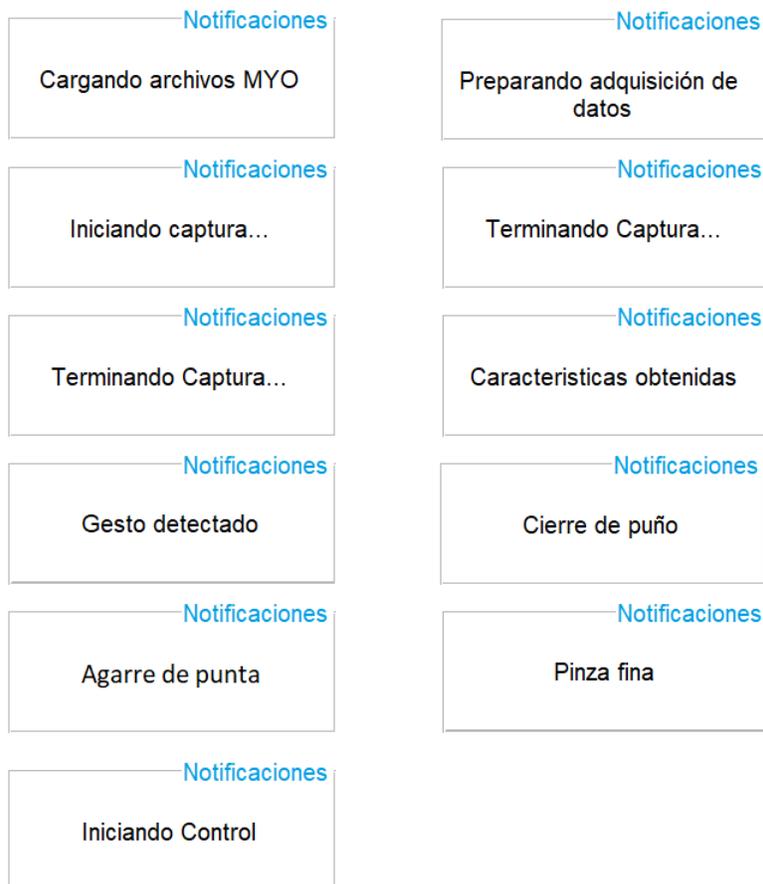


Figura 5.17: Notificaciones posibles durante la realización de una nueva prueba.

Como las leyendas muestran, el sistema comienza cargando los datos del brazalete MYO, cuando se encuentran listos para comenzar la adquisición de datos, se muestra una leyenda para que el usuario realice un movimiento con la mano, el tiempo de adquisición de los datos es de 5 segundos, una vez adquiridos, el proceso de acondicionamiento de la muestra comienza. La señal es recortada para detectar la activación muscular, filtrada, normalizada y su extracción de características con las cuales se evaluará su pertenencia a cierto movimiento. Cuando el vector de características que describen a la muestra es creado, se evalúa con el algoritmo de clasificación previamente entrenado. En el momento en que el sistema tiene un resultado, la sección de gráficos cambian con la inversión de los colores a la imagen correspondiente al movimiento detectado. En la Figura 5.18 se muestra la pantalla principal del programa cuando un movimiento de agarre de punta es detectado. En la parte inferior de los gráficos, se muestran las señales adquiridas de cada uno de los sensores del brazalete, cada uno de ellos

es representado con un color diferente.

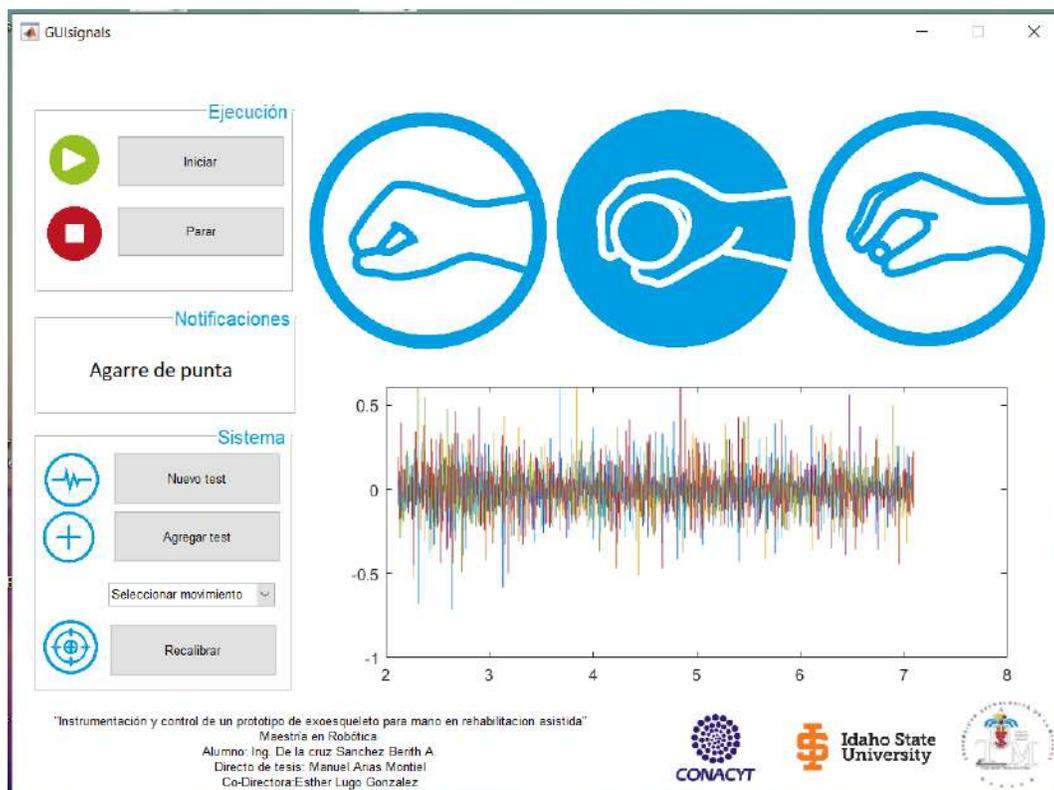


Figura 5.18: Vista de la pantalla principal del GUI al detectar movimiento de agarre de punta.

Detectado el movimiento, el sistema envía al Arduino las posiciones finales a las que cada una de las falanges debe llegar para reproducir el movimiento. Una vez enviada la información para las posiciones deseadas, el exoesqueleto comienza el movimiento hasta cubrir el rango de movimiento predeterminado. El sistema tiene un botón de “Parar”, con la posibilidad de presionarse en cualquier momento y el sistema se detiene, la aplicación se cierra y en caso en que el exoesqueleto se encuentre en movimiento se detiene.

### 5.6.2. Agregar prueba a la base de datos y recalibración

Otra de las aplicaciones del GUI, es agregar nuevas muestras a la base de datos para recalibrar al sistema. Sólo es necesario presionar el botón de “Nuevo test”, inmediatamente comenzará el sistema a cargar los datos y librerías del brazalete MYO, en cuando estén listas para usarse, se mostrarán indicaciones para que el usuario realice el movimiento con la mano, el proceso es similar al descrito para iniciar una prueba. Los posibles mensajes que se mostraran en el sistema durante el proceso, se muestran en la Figura 5.19, los cuales guían al usuario en qué parte se encuentra.

Cuando se termina de hacer la adquisición de la muestra, el siguiente paso es seleccionar del menú “Seleccionar movimiento”, el que corresponda con la prueba realizada, la figura correspondiente cambiará de color para notificar la selección. Una vez seleccionado, se presiona



Figura 5.19: Notificaciones posibles durante la recalibración del sistema.

el botón “Agregar test” para que la prueba se almacene en la base de datos del sistema. Hasta aquí sólo se agregó un nuevo archivo a la base de datos, sin hacer el entrenamiento del algoritmo de control, para actualizarlo, es necesario reentrenar el sistema con el botón “Recalibrar”. Al recalibrar el sistema puede tardar algunos segundos e inclusive puede llegar a tomar algunos minutos, ya que el sistema reentrena el algoritmo  $n$  veces hasta obtener un resultado mayor o próximo al valor de 82.2% de desempeño. La iteración que obtuvo mejor resultado es almacenada y sustituye a la clasificación previa que existe en el sistema.

## 5.7. Resultados del controlador

Para evaluar el desempeño del exoesqueleto, se realizaron pruebas con el sistema de captura de movimiento. Se colocaron marcas a cada una de las falanges del exoesqueleto, por su tamaño existe la posibilidad de confundir al sistema y recolectar datos erróneos, se hicieron pruebas de cada uno de los dedos por separado. En la Figura 5.20 se muestra el prototipo de exoesqueleto con las marcas en dos de los dedos durante una de las pruebas realizadas con el sistema de captura de movimiento. El exoesqueleto se colocó sobre la mano de una persona sana y las mediciones fueron obtenidas con referencia a una marca ubicada en la muñeca del usuario.

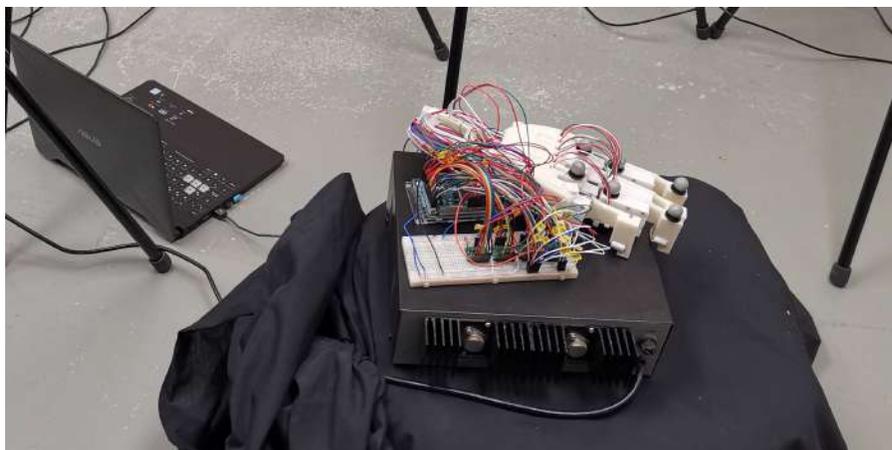


Figura 5.20: Pruebas al prototipo con el sistema de captura de movimiento.

Durante el desarrollo de las pruebas, la marca ubicada en la muñeca del usuario sirvió como referencia y base para la calibración del sistema. Después de recolectar las muestras para cada uno de los dedos de los diferentes movimientos, el análisis de las trayectorias se realizó en Matlab<sup>®</sup>, los datos fueron acondicionados con matrices de rotación para comparar los resultados obtenidos del movimiento del exoesqueleto con los del rango de una mano sana. Este proceso se describió en el Capítulo 3. Una forma de visualizar el rango que cubre el exoesqueleto de cada uno de los movimientos, es graficar ambas trayectorias, como se muestra en la Figura 5.21. En la gráfica se observa la trayectoria que debería seguir el dedo índice del exoesqueleto para un agarre de punta. Las curvas para cada una de las falanges se muestran en diferente color.

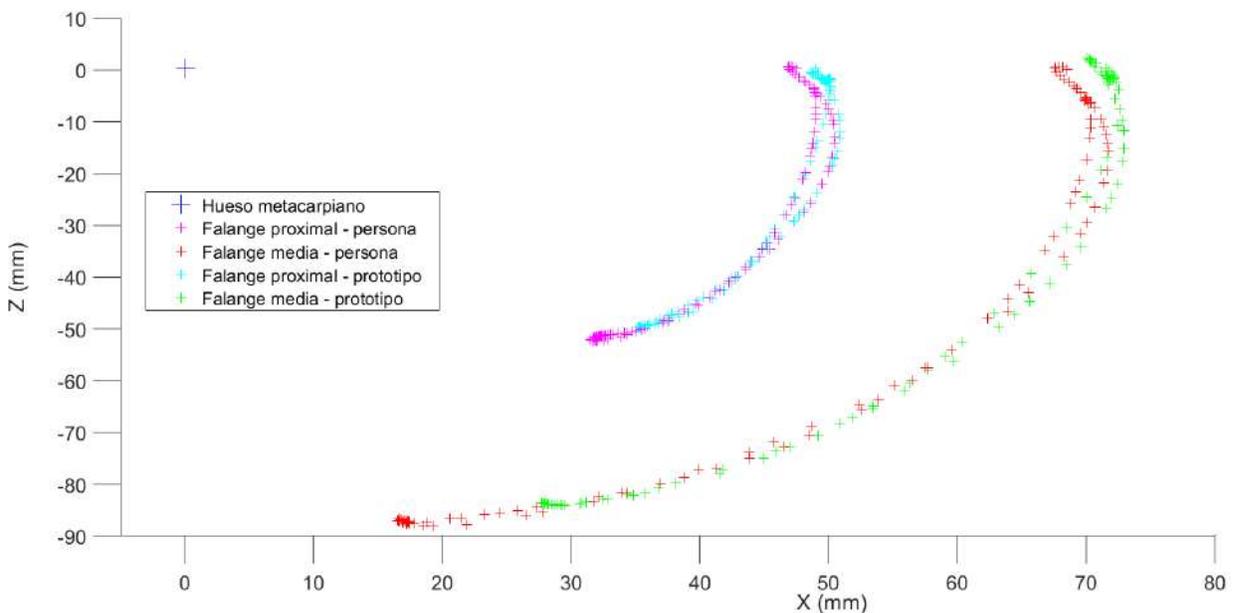


Figura 5.21: Resultados obtenidos con el sistema de captura de movimiento para el dedo índice en el agarre de punta, para el exoesqueleto y una mano sana.

En la Figura 5.22 se muestran las trayectorias obtenidas para el dedo índice en un movimiento de cierre de puño y pinza fina. En ambas gráficas se compara la trayectoria de referencia que debería seguir el exoesqueleto con respecto a los datos adquiridos para una mano sana. Las curvas de movimiento del exoesqueleto son muy parecidas con respecto a la del dedo, que sirven como referencia en el controlador. Para calcular el error que existe entre la referencia y la respuesta del mecanismo, se calculan con base en las mediciones obtenidas con el sistema de captura de movimiento, considerando como referencia la posición final a la que el exoesqueleto debería llegar y que se definen por los datos de la Tabla 5.1.

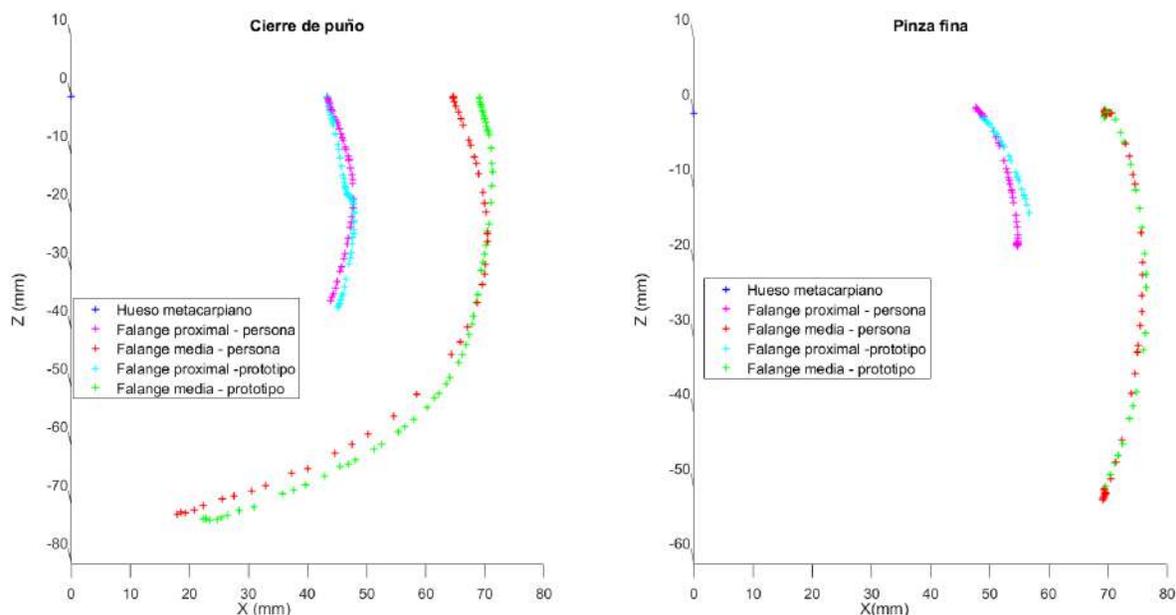


Figura 5.22: Resultados obtenidos con el sistema de captura de movimiento para el dedo índice en el de cierre de puño y pinza fina, para el exoesqueleto y una mano sana.

El error para cada una de las falanges de los dedos al realizar alguno de los movimientos propuestos, se muestra en la Tabla 5.7. Los datos están reportados en porcentaje de error, donde el menor equivale a mayor acercamiento al valor de referencia.

Tabla 5.7: Error obtenido por falange para cada uno de los movimientos con el uso del exoesqueleto.

Movimiento	Flange	Dedo					Total
		Pulgar	Indice	Medio	Anular	Meñique	
Cierre de puño	Proximal	7.57	7.89	7.03	9.49	7.59	7.914
	Media	9.42	6.89	7.07	7.2	8.42	7.8
Agarre de punta	Proximal	7.32	6.93	5.96	6.55	6.5	6.652
	Media	6.67	7.07	7.06	6.24	6.25	6.658
Pinza fina	Proximal	8.17	10.63	3.46	6.56	3.58	5.764
	Media	7.89	7.4	5.71	3.76	5.02	5.956

Para obtener el error de cada una de las falanges, se realizaron varias pruebas a cada uno de los dedos, el promedio para las mediciones obtenidas son las reportadas en este documento. Los archivos con los datos recolectados del sistema Vicon<sup>TM</sup> se encuentran disponibles en [121] para su libre descarga. El error máximo observado es de 10.63 % y el mínimo de 3.46 %, en términos de movimiento angular equivale a una diferencia de 1.89° y 0.07°.

Para las pruebas se involucraron las dos manos del usuario, el brazalete fue colocado en el brazo izquierdo y con el lado derecho se realizó el movimiento del mano auxiliado por el exoesqueleto, como se muestra en la Figura 5.23.

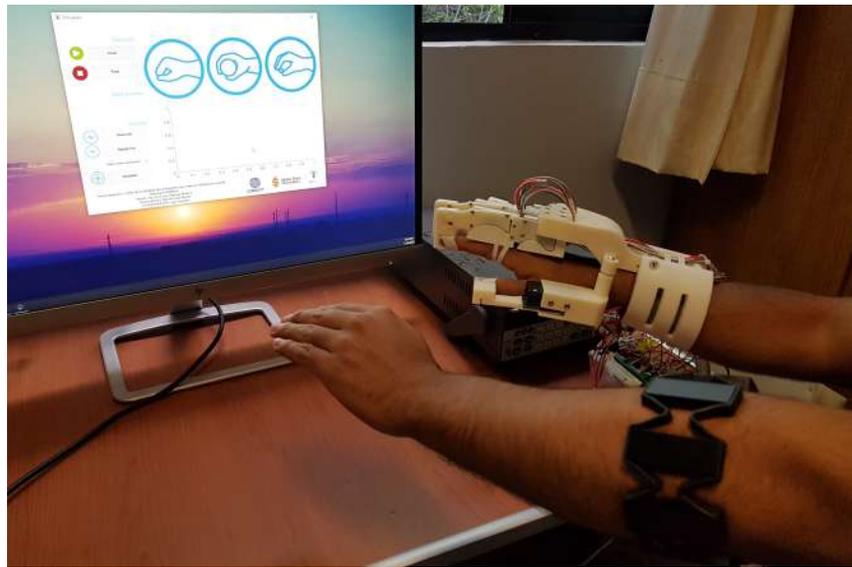


Figura 5.23: Vista general de una prueba realizada

En la Figura 5.24 se muestra el movimiento del exoesqueleto ajustado sobre la mano de un usuario para el movimiento de agarre de punta, para diferentes posiciones: en reposo, durante el desarrollo del movimiento y cuando finalmente se toma el cilindro.

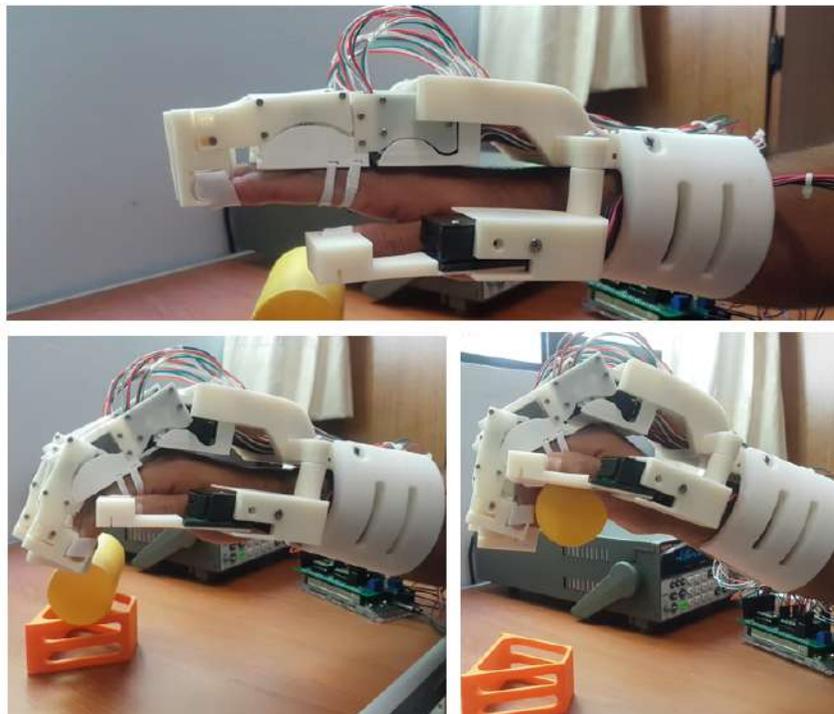


Figura 5.24: Posiciones del prototipo de exoesqueleto durante el desarrollo de una prueba de agarre de punta.



# Capítulo 6

## Conclusiones y trabajos futuros

En este capítulo se presentan las principales conclusiones obtenidas del desarrollo de esta tesis, posteriormente se muestran posibles mejoras y nuevos estudios para darle seguimiento a este proyecto.

### 6.1. Conclusiones

En este proyecto de tesis se presentó el diseño y la implementación de un sistema de instrumentación y control para un prototipo de exoesqueleto de mano enfocado a rehabilitación asistida. El diseño mecánico del exoesqueleto fue retomado del proyecto de tesis desarrollado en la Universidad Tecnológica de la Mixteca, Titulado “Diseño y construcción de un prototipo de exoesqueleto para rehabilitación de mano en accidentes cerebrovasculares”, del cual se rediseñaron algunos componentes para reducir dimensiones y aumentar la funcionalidad y resistencia. Para la construcción de prototipos, el uso de impresión 3D trae consigo ventajas en comparación con el uso de otro tipo de materiales, donde es necesario maquinarse con equipo de Control Numérico Computarizado (CNC) para lograr las geometrías complicadas de algunas piezas, además de la posibilidad de rediseñarlas en tiempos cortos, permitiendo la realización de pruebas preliminares. Considerando durante el desarrollo del proyecto los rangos de movimiento máximos para cada falange, las relaciones de engranes fueron redimensionados para conseguir reducir la altura de cada bloque de movimiento.

Como parte del desarrollo del proyecto, se creó una base de datos con tres movimientos diferentes de la mano, la cual se encuentra disponible en un repositorio y puede ser descargada de manera gratuita. Puede ser utilizado por terceros que trabajan en líneas de investigación similares o para carácter académico, ya que es una de las bases de datos más grandes reportadas donde cada una de las muestras son provenientes de una persona diferente. Recolectar una gran cantidad de muestras para cada movimiento, permitió al algoritmo de clasificación la posibilidad de reconocer los movimientos sin necesidad de colocar el brazalete Myo en la posición exacta del brazo, solo es necesario colocarlo en una posición aproximada.

En el documento se reporta la extracción de 11 diferentes características para las señales EMG. Los mejores resultados del clasificador se obtuvieron después de realizar una selección de características, ya que no todas aportan información relevante, algunas generan ruido

que ocasionan ajustes erróneos en los algoritmos de clasificación. Para descartar las características irrelevantes para cada uno de los sensores, se realizó un proceso de selección de características, los algoritmos desarrollados durante el proyecto fueron adaptados a las señales electromiográficas. Para la familia de algoritmos SFS es necesario obtener diferentes subconjuntos de características, para después evaluar su desempeño en el algoritmo de clasificación. Los resultados reportados en el documento son los mejores subconjuntos obtenidos después de diferentes evaluaciones.

El algoritmo de clasificación KNN con el que se realizó la clasificación de las señales EMG, fue el que obtuvo el mejor desempeño comparado con otros algoritmos, con un resultado del 81.2%. Cuando el mismo algoritmo fue probado con una base de datos de 7 pruebas para un solo usuario, se obtuvieron resultados superiores al 95%. Se hicieron evaluaciones con máquinas de soporte vectorial y árboles de decisión, pero su desempeño no superaba el 78% de clasificación, por lo cual fueron descartados para su implementación.

El desempeño del algoritmo de control permitió seguir de cerca la posición deseada, el error máximo obtenido es de un 10.63%, para la falange proximal del dedo índice en un movimiento de pinza fina, físicamente representa una diferencia de 1.89° menos en comparación al ángulo deseado. En termino de movimiento angular, el error más grande obtenido es de 5.7° para la falange media del dedo índice en el agarre de punta, aunque su porcentaje de error representa el 6.89%. Los errores más grandes fueron obtenidos durante los movimientos pequeños, los resultados obtenidos se ven fuertemente influenciados por la parte mecánica, el control por si solo en simulaciones alcanza la referencia. Esta diferencia es ocasionada debido al margen de error que existe en el control con respecto al valor deseado en que los motores reciben una señal de PWM de cero, si este margen se reduce, debido al tiempo de respuesta y procesamiento de la tarjeta de desarrollo Arduino, los motores comienzan a oscilar sobre el valor deseado sin lograr establecerse en él. Para evitarlo, el intervalo en que la señal de PWM con un ciclo de trabajo de 0% se amplía. Actualmente de manera visual y para una terapia este margen de error es imperceptible pero puede reducirse con la aplicación del sistema de control en otra tarjeta de desarrollo de mayor capacidad de procesamiento y en la mejora de los actuadores. Las mediciones adquiridas por el sistema de captura de movimiento para cada uno de los dedos se encuentran disponibles en un repositorio para su libre descarga.

## 6.2. Trabajos a futuro

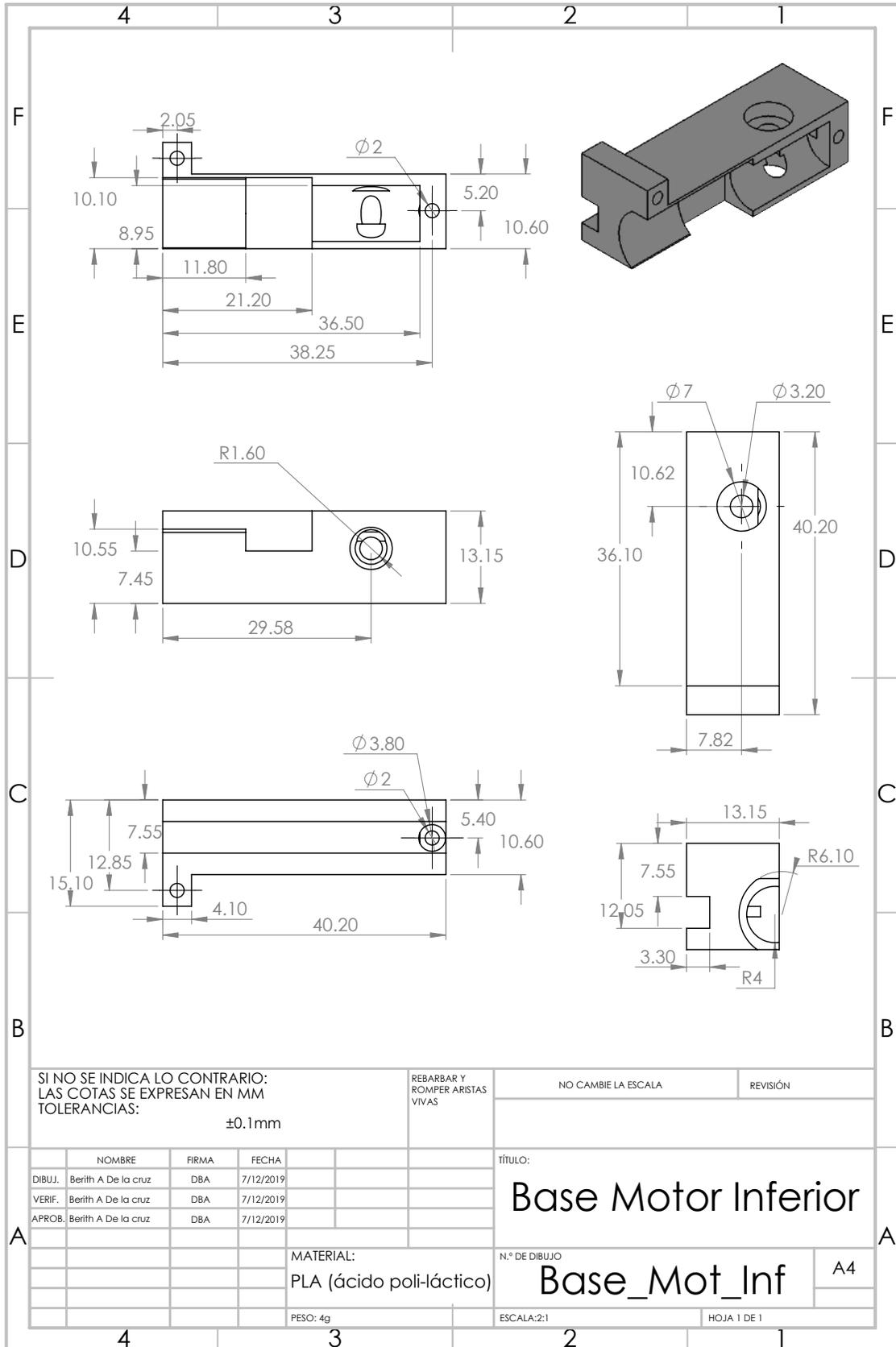
El control e instrumentación desarrollado durante este proyecto de tesis, es una primera aproximación a un sistema funcional que posteriormente pueda ser probados con otro usuarios o pacientes, por lo tanto, es susceptible a mejoras y cambios en los algoritmo implementados para la clasificación de señales y para el algoritmo de control implementado. Se proponen algunos posibles puntos a mejorar y desarrollar en trabajos futuros:

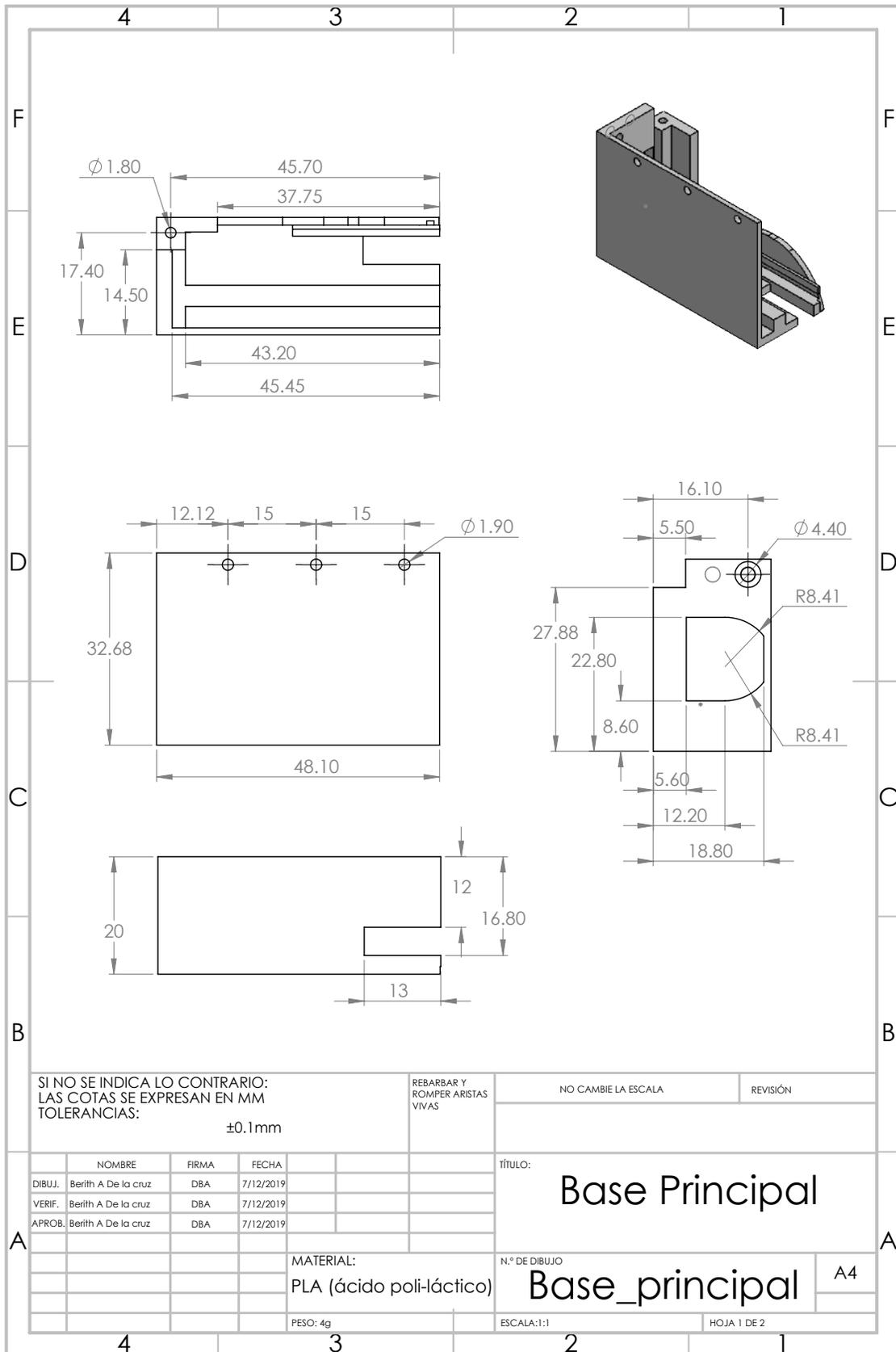
- Hacer pruebas con un muestreo de ventanas en las señales EMG y observar el desempeño que se obtiene al evaluar sus características en un algoritmo de clasificación.
- Obtener un permiso medico para la continuación del proyecto, permitiendo la realización de pruebas futuras con pacientes.
- Crear una base de datos con señales electromiográficas obtenidas de personas que padezcan alguna discapacidad en la mano.
- Migrar el sistema a una tarjeta electrónica, de tal forma que el prototipo no dependa de una conexión a la computadora para el procesamiento de los datos.
- Diseñar un controlador neuro difuso que permita reducir el tiempo de cálculo de la señal de control.

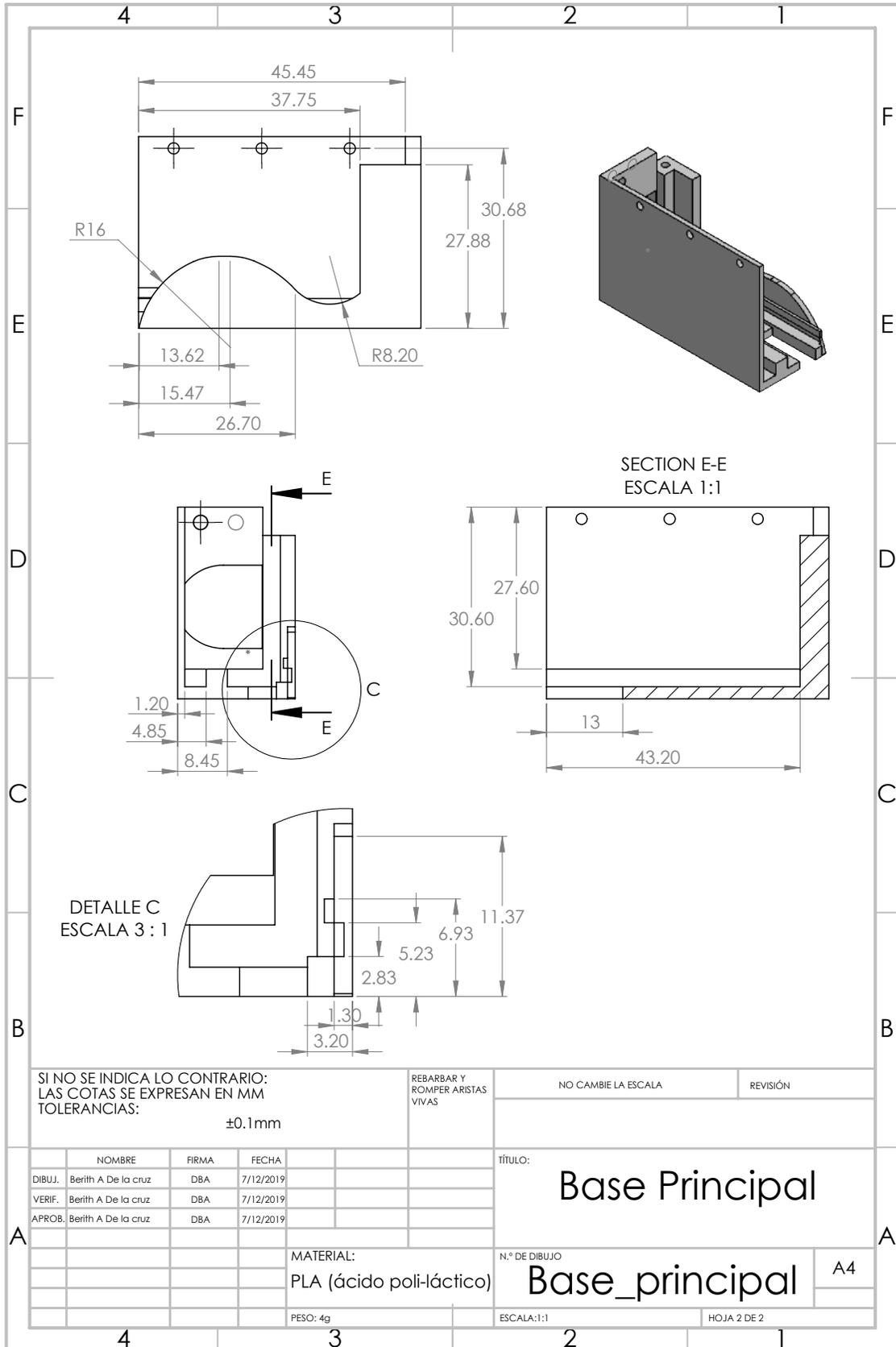


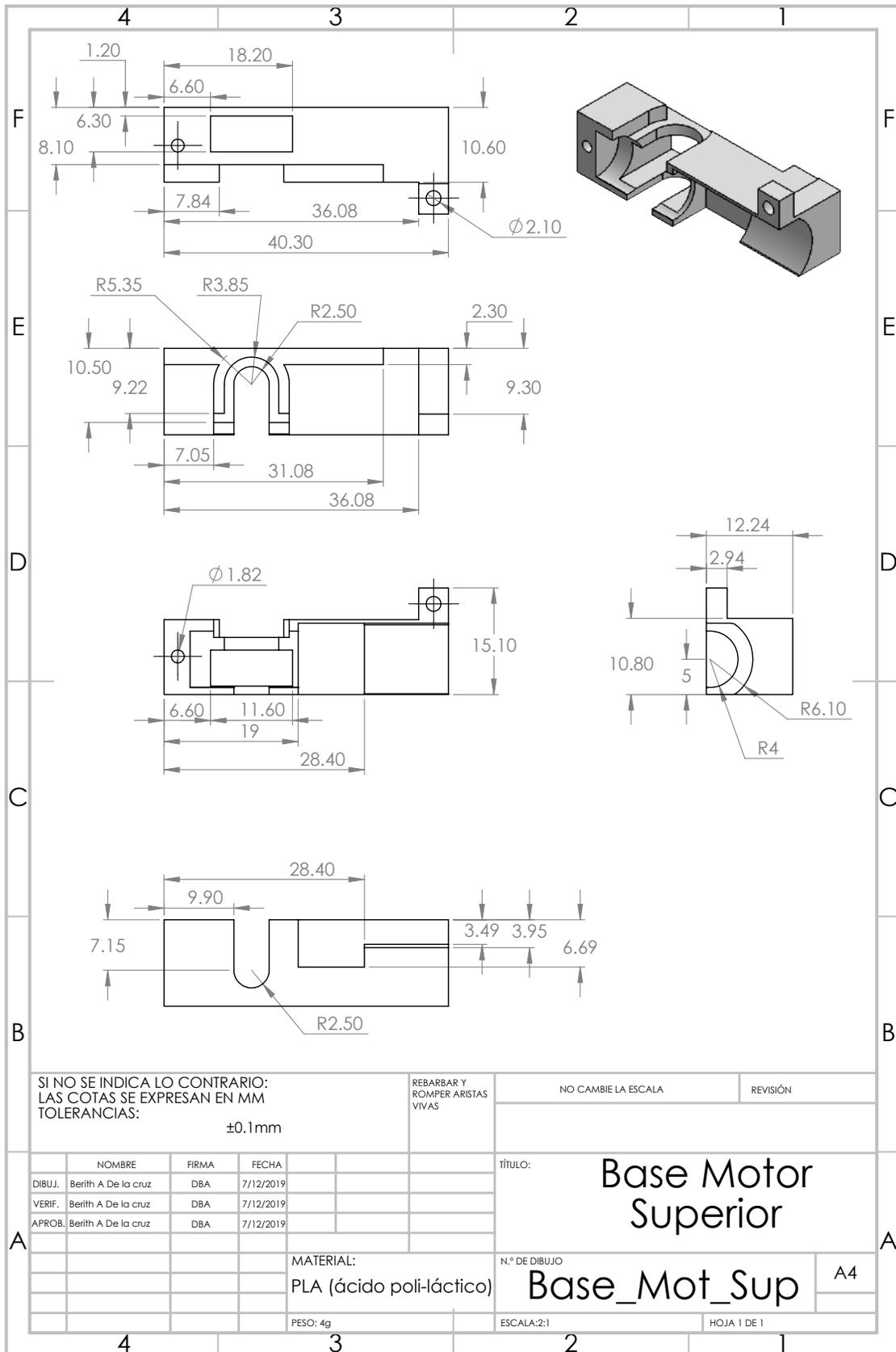
# Apéndice A

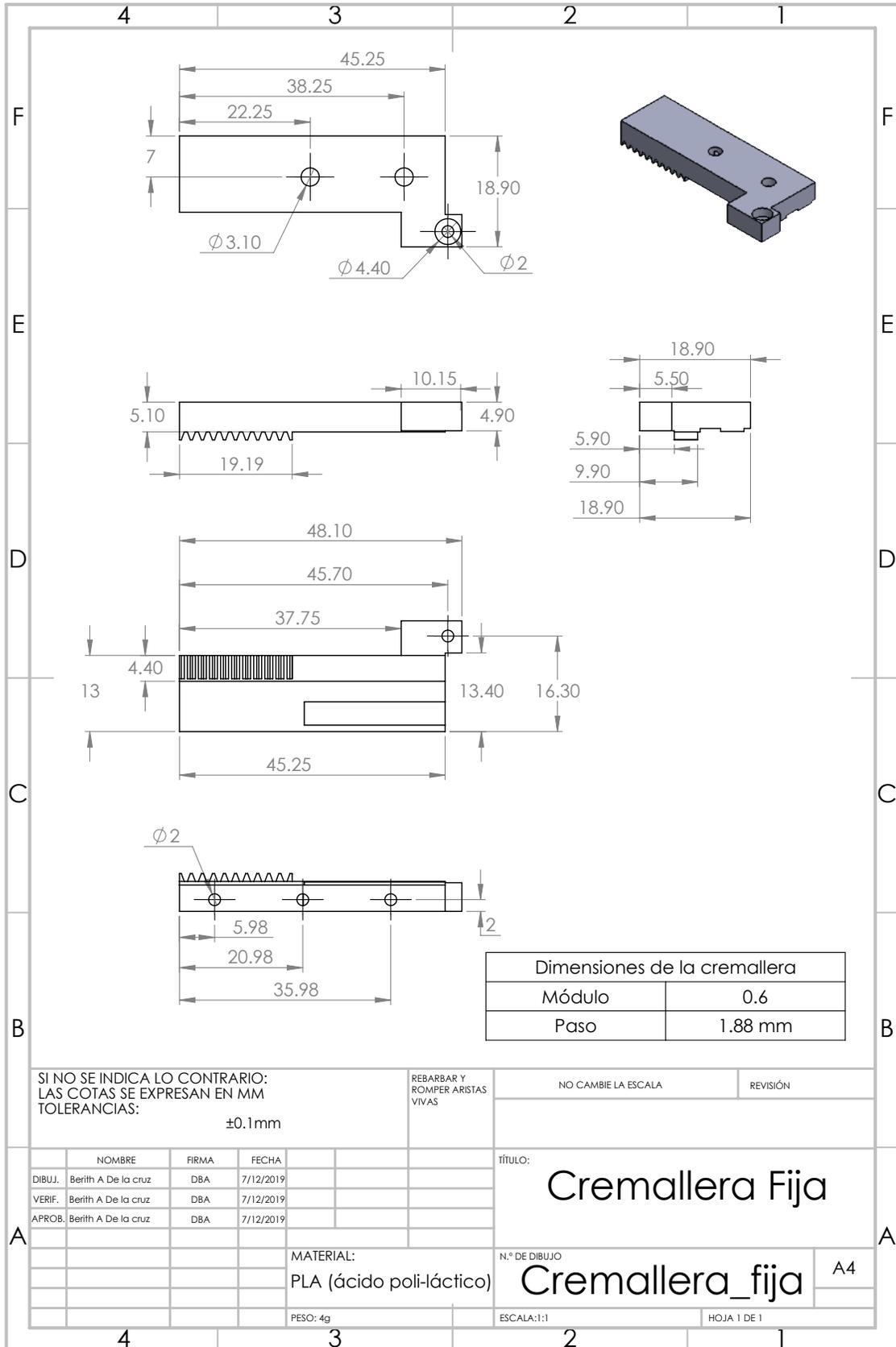
Planos constructivos.

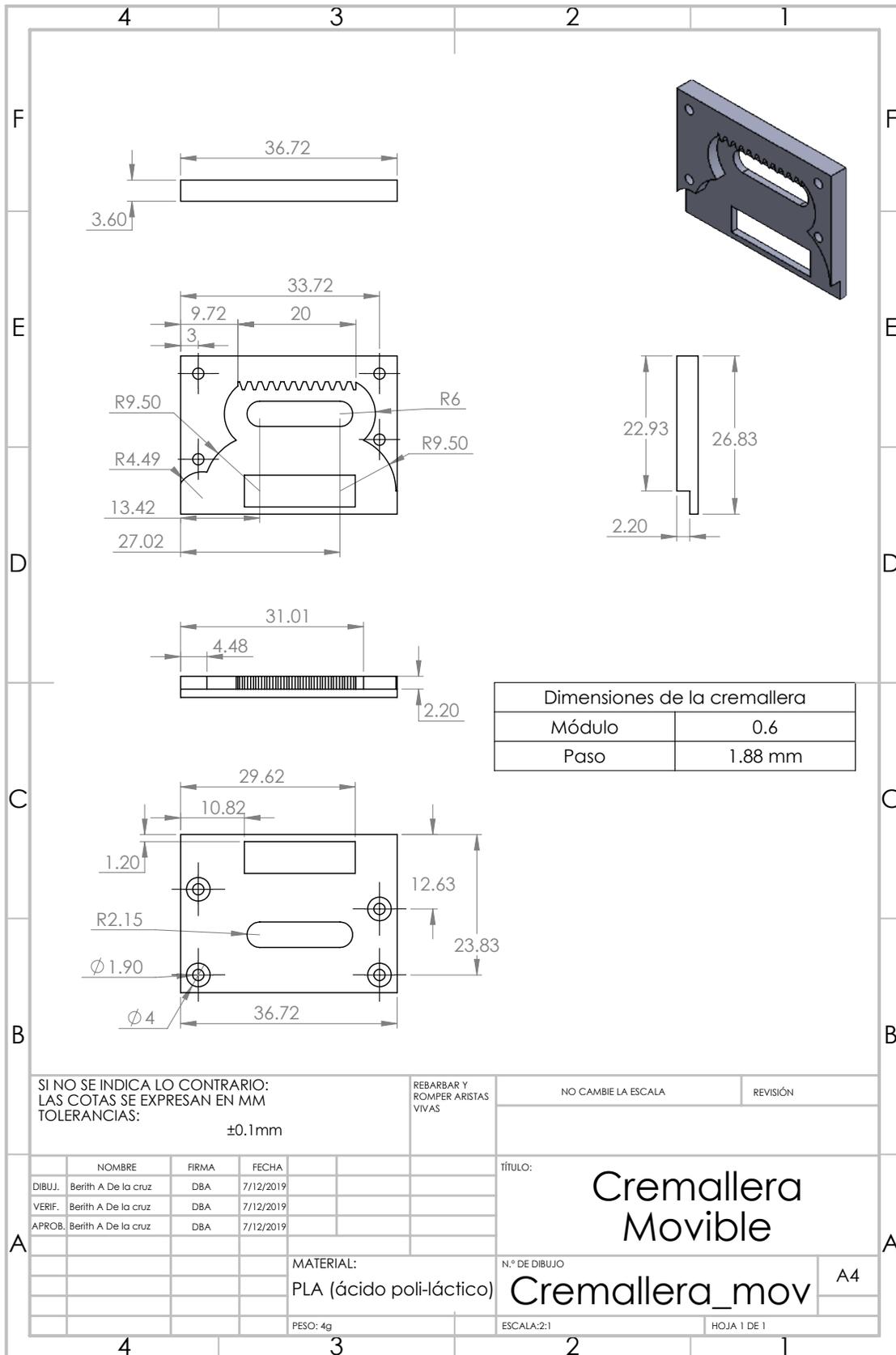


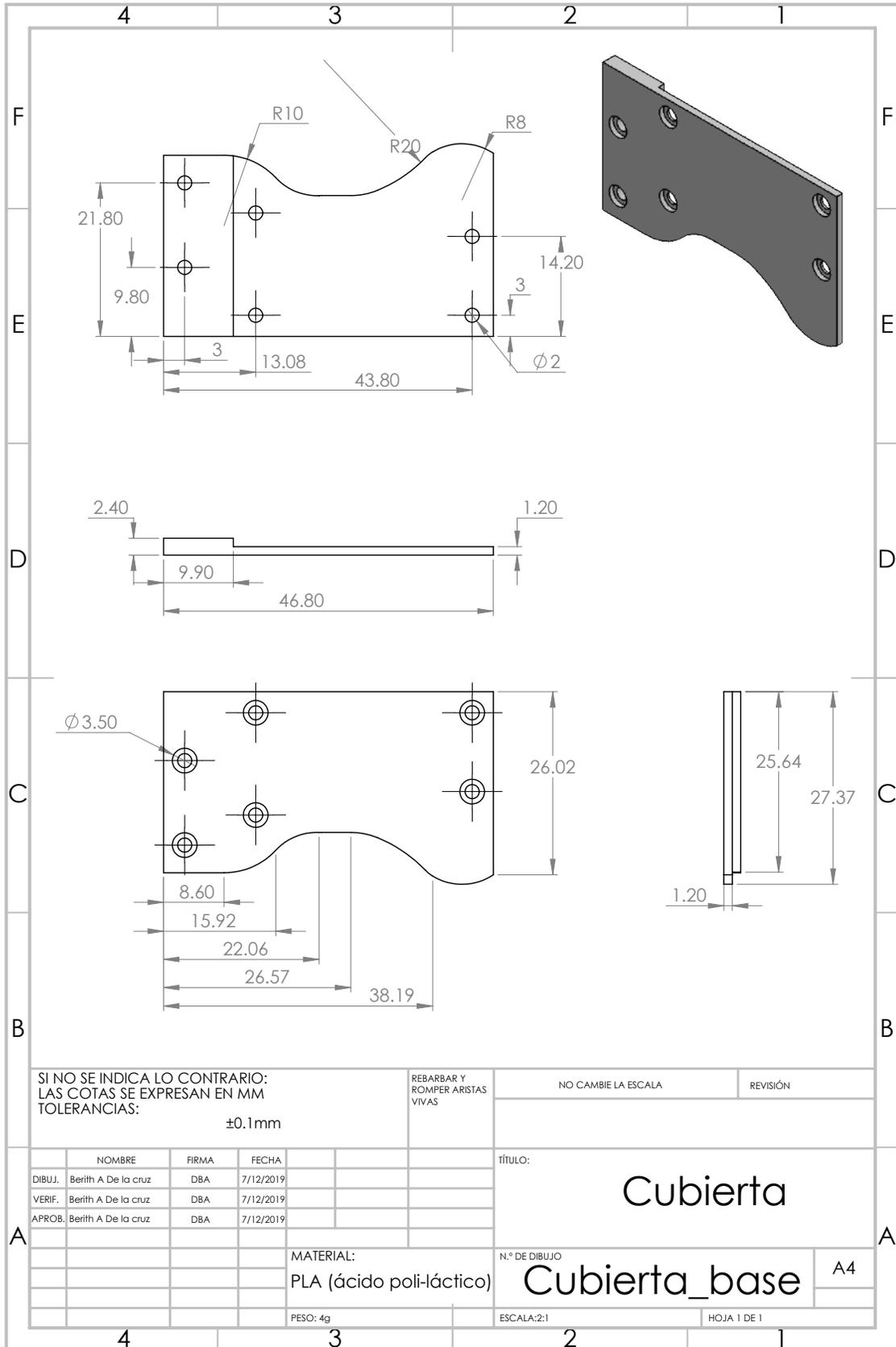












SI NO SE INDICA LO CONTRARIO:  
 LAS COTAS SE EXPRESAN EN MM  
 TOLERANCIAS:  
 ±0.1mm

REBARBAR Y  
 ROMPER ARISTAS  
 VIVAS

NO CAMBIE LA ESCALA

REVISIÓN

	NOMBRE	FIRMA	FECHA
DIBUJ.	Berith A De la cruz	DBA	7/12/2019
VERIF.	Berith A De la cruz	DBA	7/12/2019
APROB.	Berith A De la cruz	DBA	7/12/2019

TÍTULO:  
**Cubierta**

MATERIAL:  
**PLA (ácido poli-láctico)**

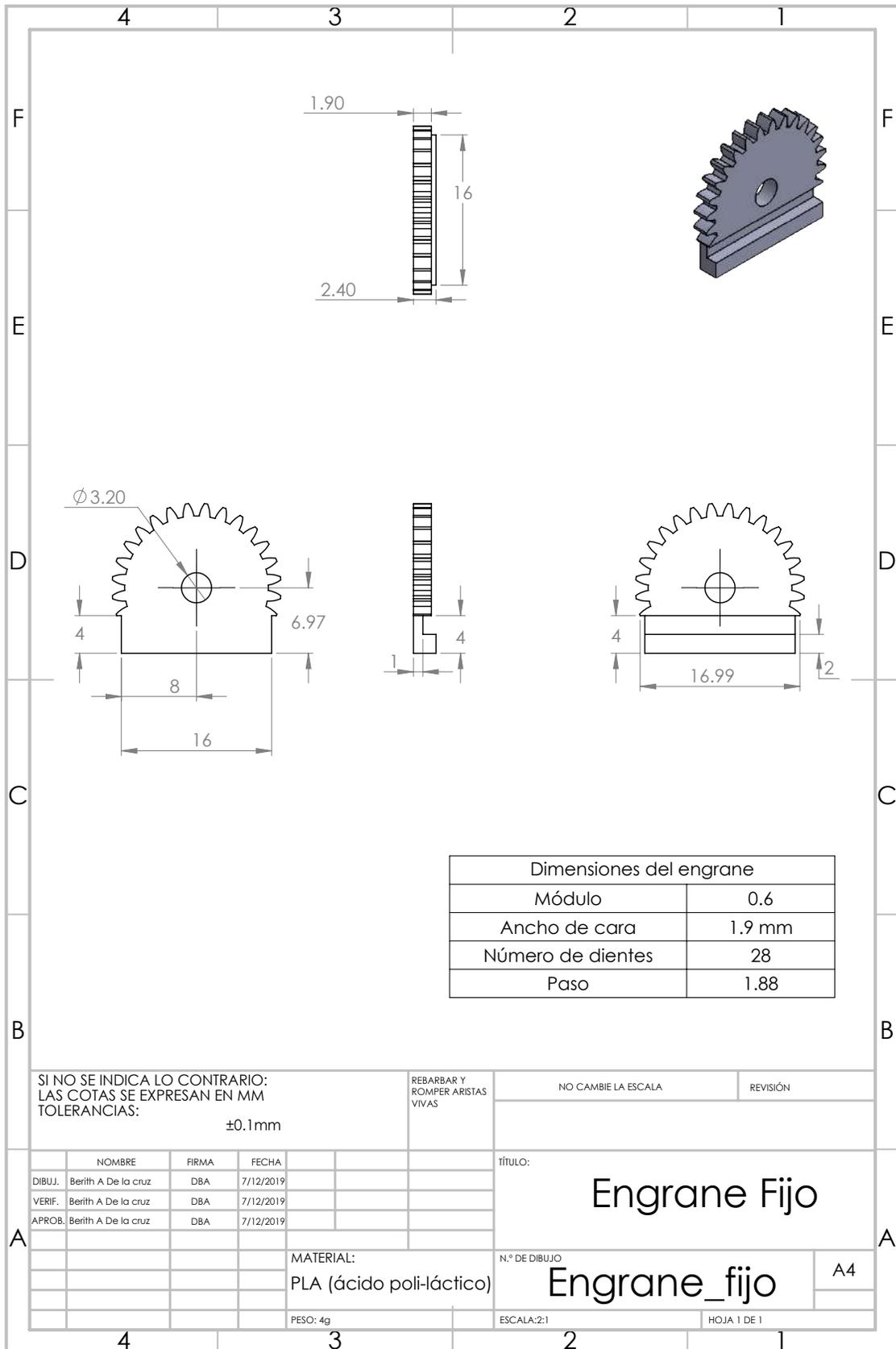
PESO: 4g

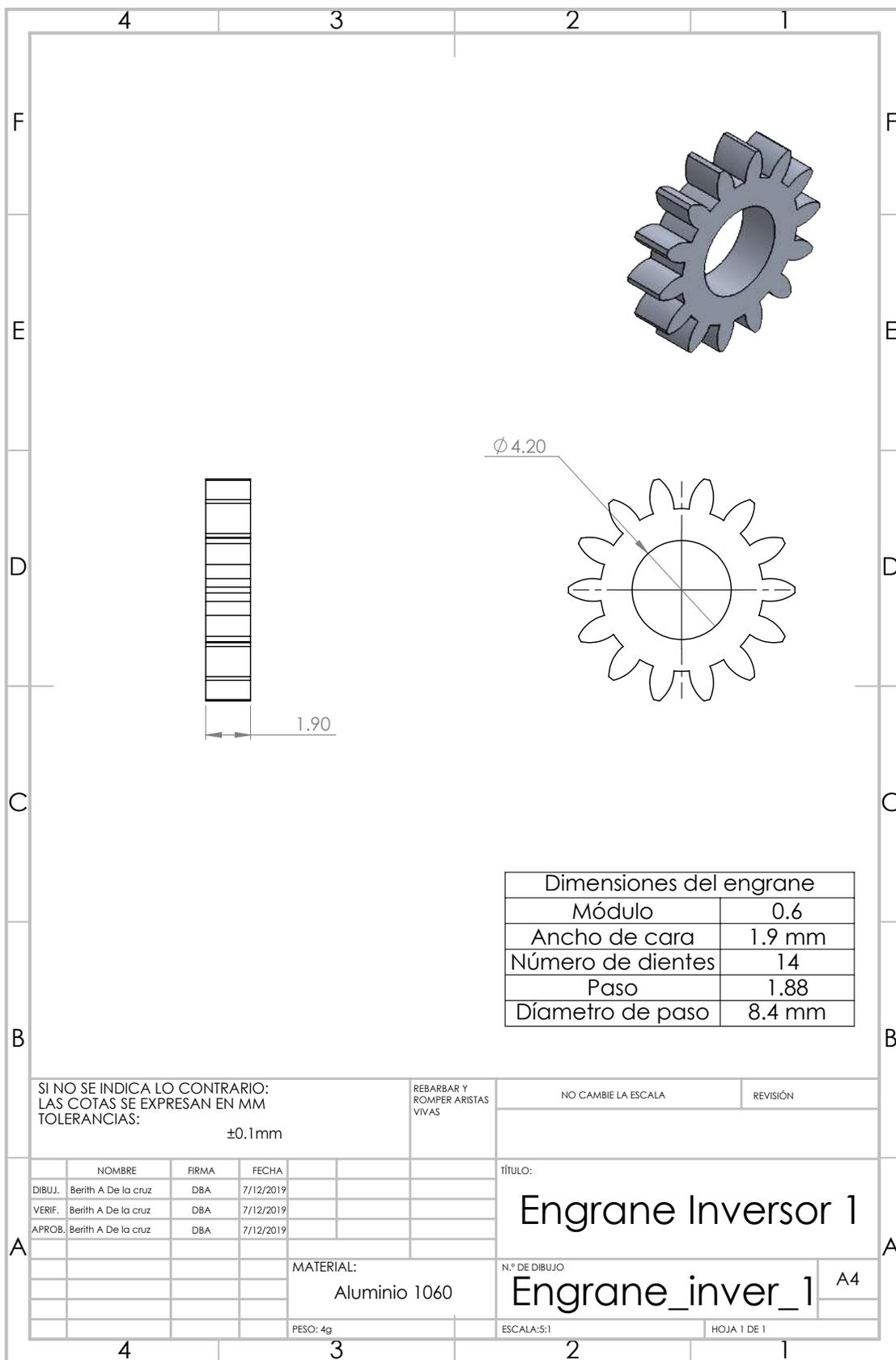
N.º DE DIBUJO  
**Cubierta\_base**

ESCALA: 2:1

HOJA 1 DE 1

A4





Dimensiones del engrane	
Módulo	0.6
Ancho de cara	1.9 mm
Número de dientes	14
Paso	1.88
Díametro de paso	8.4 mm

SI NO SE INDICA LO CONTRARIO:  
LAS COTAS SE EXPRESAN EN MM  
TOLERANCIAS:  
±0.1mm

REBARBAR Y  
ROMPER ARISTAS  
VIVAS

NO CAMBIE LA ESCALA

REVISIÓN

	NOMBRE	FIRMA	FECHA
DIBUJ.	Berith A De la cruz	DBA	7/12/2019
VERIF.	Berith A De la cruz	DBA	7/12/2019
APROB.	Berith A De la cruz	DBA	7/12/2019

TÍTULO:  
**Engrane Inversor 1**

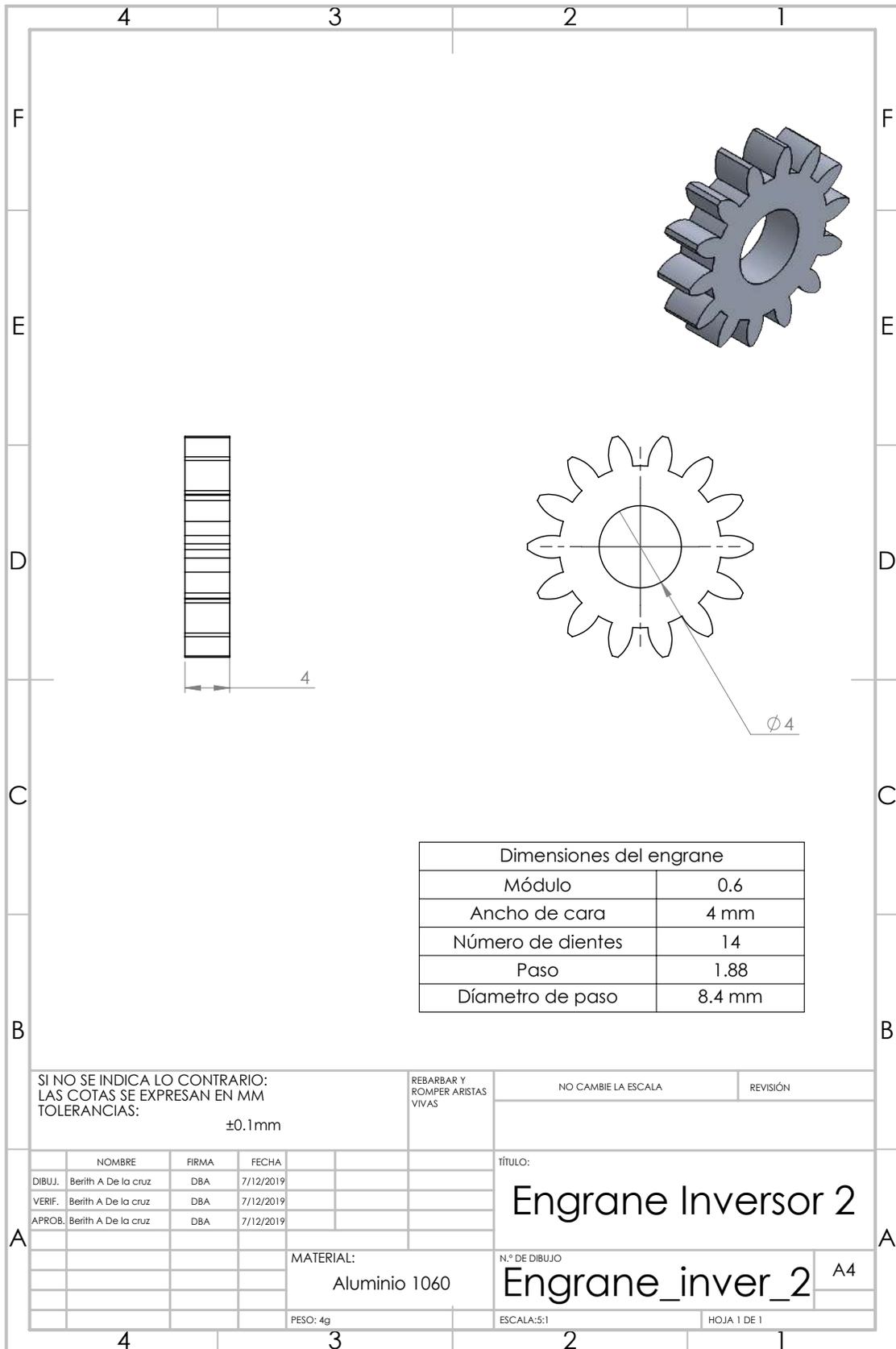
MATERIAL:  
**Aluminio 1060**

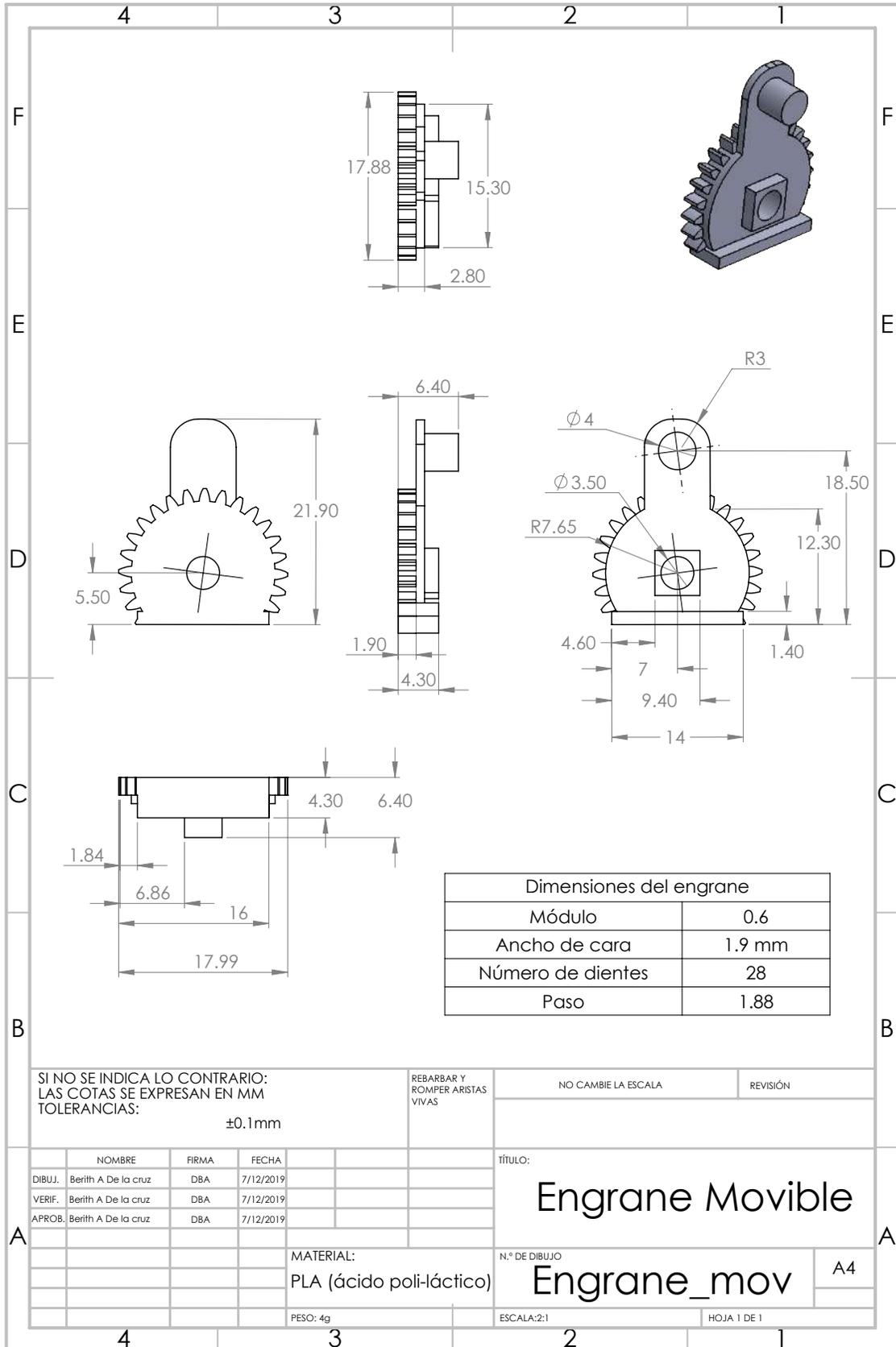
PESO: 4g

N.º DE DIBUJO  
**Engrane\_inver\_1** A4

ESCALA: 5:1

HOJA 1 DE 1





## Apéndice B

Características más relevantes del banco de datos.

Tabla B.1: Cantidad de muestras recolectadas por género y movimiento de la mano.

Característica general	Sensor sEMG	Nombre de la característica
1	1	Valor absoluto medio
2	2	Valor absoluto medio
3	3	Valor absoluto medio
6	6	Valor absoluto medio
7	7	Valor absoluto medio
9	1	Raíz cuadrada media
10	2	Raíz cuadrada media
13	5	Raíz cuadrada media
14	6	Raíz cuadrada media
15	7	Raíz cuadrada media
17	1	Varianza
22	6	Varianza
23	7	Varianza
45	5	Cruce por cero
50	2	Amplitud de Wilson
51	3	Amplitud de Wilson
54	6	Amplitud de Wilson
55	7	Amplitud de Wilson
57	1	Cambio en el signo de la pendiente
65	1	Longitud de onda
66	2	Longitud de onda
67	3	Longitud de onda
70	6	Longitud de onda
71	7	Longitud de onda
73	1	Potencia promedio de la señal
79	7	Potencia promedio de la señal
81	1	Entropía
82	2	Entropía
86	6	Entropía
87	7	Entropía

Tabla B.2: Cantidad de muestras recolectadas por género y movimiento de la mano.

Característica general	Sensor sEMG	Nombre de la característica
1	1	Valor absoluto medio
2	2	Valor absoluto medio
3	3	Valor absoluto medio
6	6	Valor absoluto medio
7	7	Valor absoluto medio
9	1	Raíz cuadrada media
10	2	Raíz cuadrada media
13	5	Raíz cuadrada media
14	6	Raíz cuadrada media
15	7	Raíz cuadrada media
17	1	Varianza
22	6	Varianza
23	7	Varianza
45	5	Cruce por cero
50	2	Amplitud de Wilson
51	3	Amplitud de Wilson
54	6	Amplitud de Wilson

Tabla B.3: Cantidad de muestras recolectadas por género y movimiento de la mano.

Característica general	Sensor sEMG	Nombre de la característica
1	1	Valor absoluto medio
2	2	Valor absoluto medio
3	3	Valor absoluto medio
6	6	Valor absoluto medio
7	7	Valor absoluto medio
9	1	Raíz cuadrada media
10	2	Raíz cuadrada media
13	5	Raíz cuadrada media
14	6	Raíz cuadrada media
15	7	Raíz cuadrada media
17	1	Varianza
22	6	Varianza
23	7	Varianza
45	5	Cruce por cero
50	2	Amplitud de Wilson
51	3	Amplitud de Wilson
54	6	Amplitud de Wilson



## Apéndice C

Rangos de movimiento por falange para diferentes movimientos de la mano.

Tabla C.1: Rangos de movimiento de las falanges para cierre de puño.

Dedo	Prueba 1			Prueba 2			Prueba 3		
	Proximal	Media	Distal	Proximal	Media	Distal	Proximal	Media	Distal
Pulgar	54.1086	-	62.9804	59.6022	-	62.7206	45.7411	-	82.3531-
Índice	38.6238	82.7016	81.3978	37.5407	85.758	80.8836	38.4293	82.9548	81.4159
Medio	60.2354	79.8522	76.847	62.6698	86.3061	72.7178	60.0759	86.6126	68.5651
Anular	33.1456	87.4303	81.3978	37.5407	85.758	80.8836	38.4293	82.9548	81.4159
Meñique	54.1086	62.9804	83.4123	59.6022	62.7206	83.3732	45.7411	82.3531	77.7764

Tabla C.2: Rangos de movimiento de las falanges para agarre de punta.

Dedo	Prueba 1			Prueba 2			Prueba 3		
	Proximal	Media	Distal	Proximal	Media	Distal	Proximal	Media	Distal
Pulgar	54.813	-	50.3636	52.6016	-	45.4198	53.3966	-	44.2991
Índice	54.813	45.1992	50.3636	52.6016	50.1377	45.4198	53.3966	54.5517	44.2991
Medio	61.4474	45.5139	52.9135	56.3975	52.1099	49.6339	58.6563	50.8405	50.84
Anular	61.4474	45.5139	52.9135	56.3975	52.1099	49.6339	58.6563	50.8405	50.84
Meñique	54.813	45.1992	50.3636	52.6016	50.1377	45.4198	53.3966	54.5517	44.2991





# Apéndice D

## Modelo dinámico

En este apéndice se encuentra el modelo matemático de un dedo del exoesqueleto de 2 GDL activos y 2 GDL pasivos, para ello se emplea el método de Euler-Lagrange. Donde las ecuaciones de movimiento que gobiernan a un robot rígido de n grados de libertad están determinadas por:

$$\frac{d}{dt} \left( \frac{\partial L(q, \dot{q})}{\dot{q}} \right) - \frac{\partial L(q, \dot{q})}{q} = \tau_1 \quad (\text{D.1})$$

$L(q, \dot{q})$  denota el lagrangiano del sistema, definido por la diferencia entre la energía cinética,  $K(q(t), \dot{q}(t))$ , y la energía potencial de un robot rígido de n grados de libertad,  $U(q(t))$ , es decir:

$$L(q, \dot{q}) = K(q(t), \dot{q}(t)) - U(q(t)) \quad (\text{D.2})$$

La energía cinética asociada al sistema está determinada por:

$$K(q, \dot{q}) = K_2(q, \dot{q}) + K_3(q, \dot{q}) \quad (\text{D.3})$$

$$K(q, \dot{q}) = \left[ \frac{1}{2} m_2 v_2^2 \right] + \left[ \frac{1}{2} I_2 \dot{q}_2^2 \right] + \left[ \frac{1}{2} m_3 v_3^2 \right] + \left[ \frac{1}{2} I_3 \dot{q}_3^2 \right] \quad (\text{D.4})$$

Donde:

$m_2, m_3$  Masas de los eslabones

$I_2, I_3$  Momentos de inercia

$v_2, v_3$  Rapidez del centro de masa de los eslabones

Como:

$$v_2 = (\dot{x}_2 \dot{y}_2)^T v_3 = (\dot{x}_3 \dot{y}_3)^T \quad (\text{D.5})$$

Las coordenadas del centro de masa del eslabón 2 en el plano X-Y son:

$$x_2 = (d_1 + C_1) + lc_2 \cos(q_2) \quad (D.6)$$

$$y_2 = -lc_2 \sin(q_2) \quad (D.7)$$

Las coordenadas del centro de masa del eslabón 3 en el plano X-Y son:

$$x_3 = (d_1 + C_1) + l_2 \cos(q_2) + lc_3 \cos(q_2 + q_3) \quad (D.8)$$

$$y_3 = -lc_2 \sin(q_2) - lc_3 \sin(q_2 + q_3) \quad (D.9)$$

El vector velocidad para los eslabones es:

$$v_2 = \begin{bmatrix} \dot{x}_2 \\ \dot{y}_2 \end{bmatrix} = \begin{bmatrix} -lc_2 \sin(q_2) \dot{q}_2 \\ -lc_2 \cos(q_2) \dot{q}_2 \end{bmatrix} \quad (D.10)$$

$$v_3 = \begin{bmatrix} \dot{x}_3 \\ \dot{y}_3 \end{bmatrix} = \begin{bmatrix} -l_2 \sin(q_2) \dot{q}_2 - lc_3 \sin(q_2 + q_3) (\dot{q}_2 + \dot{q}_3) \\ -l_2 \cos(q_2) \dot{q}_2 - lc_3 \cos(q_2 + q_3) (\dot{q}_2 + \dot{q}_3) \end{bmatrix} \quad (D.11)$$

La velocidad al cuadrado de cada eslabón resulta ser:

$$v_2^2 = \dot{x}_2^2 + \dot{y}_2^2 \quad (D.12)$$

$$v_2^2 = lc_2^2 \dot{q}_2^2 \quad (D.13)$$

$$v_3^2 = \dot{x}_3^2 + \dot{y}_3^2 \quad (D.14)$$

$$v_3^2 = l_2^2 \dot{q}_2^2 + 2 \cos(q_3) l_2 lc_3 \dot{q}_2^2 + 2 \cos(q_3) l_2 lc_3 \dot{q}_2 \dot{q}_3 + lc_3^2 \dot{q}_2^2 + 2 lc_3^2 \dot{q}_2 \dot{q}_3 + lc_3^2 \dot{q}_3^2 \quad (D.15)$$

De esta manera las energías cinéticas asociadas a cada eslabón del robot rígido están determinadas por:

$$K_2(q, \dot{q}) = \left[ \frac{1}{2} m_2 (lc_2^2 \dot{q}_2^2) \right] + \left[ \frac{1}{2} I_2 \dot{q}_2^2 \right] \quad (D.16)$$

$$K_3(q, \dot{q}) = \left[ \frac{1}{2} m_3 (l_2^2 \dot{q}_2^2 + 2 \cos(q_3) l_2 lc_3 \dot{q}_2^2 + 2 \cos(q_3) l_2 lc_3 \dot{q}_2 \dot{q}_3 + lc_3^2 \dot{q}_2^2 + 2 lc_3^2 \dot{q}_2 \dot{q}_3 + lc_3^2 \dot{q}_3^2) \right] + \left[ \frac{1}{2} I_3 \dot{q}_3^2 \right] \quad (D.17)$$

La energía potencial asociado a los eslabones están determinadas por:

$$U(q) = U_2(q) + U_3(q) = [-m_2 g l c_2 \sin(q_2)] + [-m_3 l_2 g \sin(q_2) - m_3 l c_3 g \sin(q_2 + q_3)] \quad (D.18)$$

Así, sustituyendo las Ecuaciones (D.16), (D.17) y (D.18), en la Ecuación (D.2) se obtiene el lagrangiano:

$$\begin{aligned}
 L = & \frac{1}{2}m_2(lc_2^2\dot{q}_2^2) + \frac{1}{2}I_2\dot{q}_2^2 + \frac{1}{2} + \frac{1}{2}I_3\dot{q}_3^2 + \\
 & \frac{1}{2}m_3(l_2^2\dot{q}_2^2 + 2\cos(q_3)l_2lc_3\dot{q}_2^2 + 2\cos(q_3)l_2lc_3\dot{q}_2\dot{q}_3 + lc_3^2\dot{q}_2^2 + 2lc_3^2\dot{q}_2\dot{q}_3 + lc_3^2\dot{q}_3^2) \\
 & m_2glc_2\sin(q_2) + m_3l_2g\sin(q_2) + m_3lc_3g\sin(q_2 + q_3)
 \end{aligned} \tag{D.19}$$

Las ecuaciones de movimiento del sistema están dadas por:

$$\frac{d}{dt} \left( \frac{\partial L(q, \dot{q})}{\dot{q}_2} \right) - \frac{\partial L(q, \dot{q})}{q_2} = \tau_1 \tag{D.20}$$

$$\frac{d}{dt} \left( \frac{\partial L(q, \dot{q})}{\dot{q}} \right) - \frac{\partial L(q, \dot{q})}{q_3} = \tau_1$$

Desarrollando las ecuaciones de movimiento para el eslabón 2 se obtiene:

$$\begin{aligned}
 \frac{\partial L(q, \dot{q})}{\dot{q}_2} = & I_2\dot{q}_2 + I_3\dot{q}_2 + I_3\dot{q}_3 + l_2^2m_3\dot{q}_2 + lc_2^2m_2\dot{q}_2 + lc_3^2m_3\dot{q}_2 + lc_3^2m_3\dot{q}_3 + \\
 & 2l_2lc_3m_3\dot{q}_2\cos(q_3) + l_2lc_3m_3\dot{q}_3\cos(q_3)
 \end{aligned} \tag{D.21}$$

$$\begin{aligned}
 \frac{d}{dt} \left( \frac{\partial L(q, \dot{q})}{\dot{q}_2} \right) = & \ddot{q}_2 [m_2lc_2^2 + m_3l_2^2 + m_3lc_3^2 + 2m_3l_2lc_3\cos(q_3)] + \ddot{q}_3 [m_3lc_3^2 + m_3l_2lc_3\cos(q_3)] \\
 & - 2m_3\sin(q_3)l_2lc_3\dot{q}_2\dot{q}_3 - m_3\sin(q_3)l_2lc_3\dot{q}_3^2 + I_2\ddot{q}_2 + I_3[\ddot{q}_2 + \ddot{q}_3]
 \end{aligned} \tag{D.22}$$

$$\frac{\partial L(q, \dot{q})}{q_2} = g(l_2m_3\cos(q_2) + lc_2m_2\cos(q_2) + lc_3m_3\cos(q_2 + q_3)) \tag{D.23}$$

Desarrollando las ecuaciones de movimiento para el eslabón 3 se obtiene:

$$\frac{\partial L(q, \dot{q})}{\dot{q}_3} = I_3(\dot{q}_2 + \dot{q}_3) + m_3(lc_3^2\dot{q}_2 + lc_3^2\dot{q}_3 + l_2lc_3\dot{q}_2\cos(q_3)) \tag{D.24}$$

$$\begin{aligned}
 \frac{d}{dt} \left( \frac{\partial L(q, \dot{q})}{\dot{q}_2} \right) = & \ddot{q}_2 [m_3lc_3^2 + m_3l_2lc_3\cos(q_3)] + \ddot{q}_3 [m_3lc_3^2] \\
 & - m_3\sin(q_3)l_2lc_3\dot{q}_2\dot{q}_3 + I_3[\ddot{q}_2 + \ddot{q}_3]
 \end{aligned} \tag{D.25}$$

$$\frac{\partial L(q, \dot{q})}{\partial \dot{q}_3} = glc_3m_3\cos(q_2 + q_3) - m_3(l_2lc_3\sin(q_3)\dot{q}_2^2 + l_2lc_3\dot{q}_3\sin(q_3)\dot{q}_2) \quad (D.26)$$

Las ecuaciones de movimiento para el sistema de un dedo del exoesqueleto, tras haber empleado el método de Euler-Lagrange, quedan determinadas por:

$$\begin{aligned} \tau_1 = \ddot{q}_2 [m_2lc_2^2 + m_3l_2^2 + m_3lc_3^2 + 2m_3l_2lc_3\cos(q_3)] + \ddot{q}_3 [m_3lc_3^2 + m_3l_2lc_3\cos(q_3)] \\ - 2m_3\sen(q_3)l_2lc_3\dot{q}_2\dot{q}_3 - m_3\sen(q_3)l_2lc_3\dot{q}_3^2 + I_2\ddot{q}_2 + I_3[\ddot{q}_2 + \ddot{q}_3] \\ - g(l_2m_3\cos(q_2) - lc_2m_2\cos(q_2) + lc_3m_3\cos(q_2 + q_3)) \end{aligned} \quad (D.27)$$

$$\begin{aligned} \tau_2 = \ddot{q}_2 [m_3lc_3^2 + m_3l_2lc_3\cos(q_3)] + \ddot{q}_3 [m_3lc_3^2] + I_3[\ddot{q}_2 + \ddot{q}_3] - glc_3m_3\cos(q_2 + q_3) \\ + m_3l_2lc_3\sin(q_3)\dot{q}_2^2 \end{aligned}$$

El modelo del sistema D.27 y D.28 puede escribirse en la forma general:

$$\tau = M(q)\ddot{q} + C(q)\dot{q} + g(q) \quad (D.28)$$

O bien:

$$\begin{pmatrix} \tau_2 \\ \tau_3 \end{pmatrix} = \begin{bmatrix} M_{11}(q) & M_{12}(q) \\ M_{21}(q) & M_{22}(q) \end{bmatrix} \begin{bmatrix} \ddot{q}_2 \\ \ddot{q}_3 \end{bmatrix} + \begin{bmatrix} C_{11}(q, \dot{q}) & C_{12}(q, \dot{q}) \\ C_{21}(q, \dot{q}) & C_{22}(q, \dot{q}) \end{bmatrix} \begin{bmatrix} \dot{q}_2 \\ \dot{q}_3 \end{bmatrix} + \begin{bmatrix} g_2(q) \\ g_3(q) \end{bmatrix} \quad (D.29)$$

Donde:

$$\begin{aligned} M_{11}(q) &= m_2lc_2^2 + m_3l_2^2 + m_3lc_3^2 + 2m_3l_2lc_3\cos(q_3) + I_2 + I_3 \\ M_{12}(q) &= m_3lc_3^2 + m_3l_2lc_3\cos(q_3) + I_3 \\ M_{21}(q) &= m_3lc_3^2 + m_3l_2lc_3\cos(q_3) + I_3 \\ M_{22}(q) &= m_3lc_3^2 + I_3 \\ C_{11}(q, \dot{q}) &= -2m_3\sen(q_3)l_2lc_3\dot{q}_3 \\ C_{12}(q, \dot{q}) &= -m_3\sen(q_3)l_2lc_3\dot{q}_3 \\ C_{21}(q, \dot{q}) &= m_3l_2lc_3\sin(q_3)\dot{q}_2 \\ C_{22}(q, \dot{q}) &= 0 \\ g_1(q) &= -g(l_2m_3\cos(q_2) - lc_2m_2\cos(q_2) + lc_3m_3\cos(q_2 + q_3)) \\ g_2(q) &= glc_3m_3\cos(q_2 + q_3) \end{aligned}$$

# Apéndice E

## Codigo Arduino

```
1 ////////////////////////////////////////////////////////////////////Librerias
2 #include <Fuzzy.h>
3 #include <Encoder.h>
4
5 ////////////////////////////////////////////////////////////////////Funciones
6     difusas
7 Fuzzy *fuzzy = new Fuzzy();
8
9 // FuzzyInput Error
10 FuzzySet *NE = new FuzzySet(-1.72, -1.08, -0.106, -0.015);
11 FuzzySet *ZE = new FuzzySet(-0.02, -0.02, 0.02, 0.02);
12 FuzzySet *PE = new FuzzySet(0.015, 0.106, 1.08, 1.72);
13
14 // FuzzyInput Cambio en el error
15 FuzzySet *NC = new FuzzySet(-1, -1, -1, 0);
16 FuzzySet *ZC = new FuzzySet(-0.3, 0, 0, 0.3);
17 FuzzySet *PC = new FuzzySet(0, 1, 1, 1);
18
19 // FuzzyOutput PWM
20 FuzzySet *NPWM= new FuzzySet(-62, -40, -29, 0);
21 FuzzySet *ZPWM = new FuzzySet(-10, -5, 5, 10);
22 FuzzySet *PPWM = new FuzzySet(0, 29, 40, 62);
23
24 ////////////////////////////////////////////////////////////////////Encoders
25 Encoder myEnc21(42, 43);
26 Encoder myEnc22(44, 45);
27 Encoder myEnc31(46, 47);
28 Encoder myEnc32(48, 49);
29 Encoder myEnc41(50, 51);
30 Encoder myEnc42(52, 53);
```

```
30 Encoder myEnc51(13, 12);
31
32 ////////////////////////////////////////////////////////////////////Motores
33 //Motor 2 Falange Proximal
34 int pinAIN1 = 22; //Direccion
35 int pinAIN2 = 23; //Direccion
36 int pinPWMA = 2; //PWM
37
38 //Motor 2 Falange Media
39 int pinBIN1 = 24; //Direccion
40 int pinBIN2 = 25; //Direccion
41 int pinPWMB = 3; //PWM
42
43 //Standby Dedo 2
44 int pinSTBY2 = 26;
45
46 ////////////////////////////////////////////////////////////////////
47 //Motor 3 Falange Proximal
48 int pinCIN1 = 28; //Direccion
49 int pinCIN2 = 29; //Direccion
50 int pinPWMC = 4; //PWM
51
52 //Motor 3 Falange Media
53 int pinDIN1 = 30; //Direccion
54 int pinDIN2 = 31; //Direccion
55 int pinPWMD = 5; //PWM
56
57 //Standby Dedo 3
58 int pinSTBY3 = 27;
59
60 ////////////////////////////////////////////////////////////////////
61 //Motor 4 Falange Proximal
62 int pinEIN1 = 32; //Direccion
63 int pinEIN2 = 33; //Direccion
64 int pinPWME = 6; //PWM
65
66 //Motor 4 Falange Media
67 int pinFIN1 = 34; //Direccion
68 int pinFIN2 = 35; //Direccion
69 int pinPWME = 7; //PWM
70
71 //Standby Dedo 4
72 int pinSTBY4 = 36;
73
```

```
74 ///////////////////////////////////////////////////////////////////
75 //Motor 5 Falange
76 int pinGIN1 = 38; //Direccion
77 int pinGIN2 = 39; //Direccion
78 int pinPWMG = 8; //PWM
79
80 //Standby Dedo 5
81 int pinSTBY5 = 37;
82
83 ///////////////////////////////////////////////////////////////////Otras variables
84 boolean pinIn11; //Pin de activación
85 boolean pinIn21;
86 boolean pinIn22;
87 boolean pinIn31;
88 boolean pinIn32;
89 boolean pinIn41;
90 boolean pinIn42;
91 boolean pinIn51;
92
93 int Referencia11=0; //Referencia para cada motor
94 int Referencia21=0;
95 int Referencia22=0;
96 int Referencia31=0;
97 int Referencia32=0;
98 int Referencia41=0;
99 int Referencia42=0;
100 int Referencia51=0;
101
102 int Referencia11P=0; //Copia de la variable
103 int Referencia21P=0; //referencia
104 int Referencia22P=0;
105 int Referencia31P=0;
106 int Referencia32P=0;
107 int Referencia41P=0;
108 int Referencia42P=0;
109 int Referencia51P=0;
110
111 float e11=0; //Variable error
112 float e21=0; //para cada encoder
113 float e22=0;
114 float e31=0;
115 float e32=0;
116 float e41=0;
117 float e42=0;
```

```
118 float e51=0;
119
120 float de11=0; //Variable para la derivada
121 float de21=0; //del error
122 float de22=0;
123 float de31=0;
124 float de32=0;
125 float de41=0;
126 float de42=0;
127 float de51=0;
128
129 float eant11=0; //Error anterior
130 float eant21=0;
131 float eant22=0;
132 float eant31=0;
133 float eant32=0;
134 float eant41=0;
135 float eant42=0;
136 float eant51=0;
137
138 float count11=0; //Variables para recibir valores
139 float count21=0; //de referencia de Matlab
140 float count22=0;
141 float count31=0;
142 float count32=0;
143 float count41=0;
144 float count42=0;
145 float count51=0;
146
147 int Meta11=0; //Variable auxilia para almacenar
148 int Meta21=0; //los valores obtenidos de los
149 int Meta22=0; //encoders
150 int Meta31=0;
151 int Meta32=0;
152 int Meta41=0;
153 int Meta42=0;
154 int Meta51=0;
155
156 int recibir=0;
157 int Regreso=0;
158 int Vez=0;
159 int Parar=0;
160
161 void setup() {
```

```
162 // Establecer velocidad del serial
163 Serial.begin(9600);
164
165 // FuzzyInput Error
166 FuzzyInput *Error = new FuzzyInput(1);
167
168 Error->addFuzzySet (NE);
169 Error->addFuzzySet (ZE);
170 Error->addFuzzySet (PE);
171 fuzzy->addFuzzyInput (Error);
172
173 // FuzzyInput Cambio
174 FuzzyInput *Change = new FuzzyInput(2);
175
176 Change->addFuzzySet (NC);
177 Change->addFuzzySet (ZC);
178 Change->addFuzzySet (PC);
179 fuzzy->addFuzzyInput (Change);
180
181 // FuzzyOutput PWM
182 FuzzyOutput *Voltaje = new FuzzyOutput(1);
183
184 Voltaje->addFuzzySet (NPWM);
185 Voltaje->addFuzzySet (ZPWM);
186 Voltaje->addFuzzySet (PPWM);
187 fuzzy->addFuzzyOutput (Voltaje);
188
189 ///////////////////////////////////////////////////Construyendo
    reglas difusas
190
191 //Declaración de los antecedentes para cada una de las reglas
192 FuzzyRuleAntecedent *ErrorNEAndChangeNC = new
    FuzzyRuleAntecedent ();
193 ErrorNEAndChangeNC->joinWithAND (NE, NC);
194
195 FuzzyRuleAntecedent *ErrorNEAndChangeZC = new
    FuzzyRuleAntecedent ();
196 ErrorNEAndChangeZC->joinWithAND (NE, ZC);
197
198 FuzzyRuleAntecedent *ErrorNEAndChangePC = new
    FuzzyRuleAntecedent ();
199 ErrorNEAndChangePC->joinWithAND (NE, PC);
200
```

```
201 FuzzyRuleAntecedent *ErrorZEAndChangeNC = new
      FuzzyRuleAntecedent ();
202 ErrorZEAndChangeNC->joinWithAND (ZE, NC);
203
204 FuzzyRuleAntecedent *ErrorZEAndChangeZC = new
      FuzzyRuleAntecedent ();
205 ErrorZEAndChangeZC->joinWithAND (ZE, ZC);
206
207 FuzzyRuleAntecedent *ErrorZEAndChangePC = new
      FuzzyRuleAntecedent ();
208 ErrorZEAndChangePC->joinWithAND (ZE, PC);
209
210 FuzzyRuleAntecedent *ErrorPEAndChangeNC = new
      FuzzyRuleAntecedent ();
211 ErrorPEAndChangeNC->joinWithAND (PE, NC);
212
213 FuzzyRuleAntecedent *ErrorPEAndChangeZC = new
      FuzzyRuleAntecedent ();
214 ErrorPEAndChangeZC->joinWithAND (PE, ZC);
215
216 FuzzyRuleAntecedent *ErrorPEAndChangePC = new
      FuzzyRuleAntecedent ();
217 ErrorPEAndChangePC->joinWithAND (PE, PC);
218
219 //
      ////////////////////////////////////////
      Consecuencias
220 //Declaración de las consecuencias para cada una de las reglas
221 FuzzyRuleConsequent *thenNPWM= new FuzzyRuleConsequent ();
222 thenNPWM->addOutput (NPWM);
223
224 FuzzyRuleConsequent *thenZPWM= new FuzzyRuleConsequent ();
225 thenZPWM->addOutput (ZPWM);
226
227 FuzzyRuleConsequent *thenPPWM= new FuzzyRuleConsequent ();
228 thenPPWM->addOutput (PPWM);
229
230 ////////////////////////////////////////
      Reglas
231 //Asociar los antecedentes con las consecuencias de acuerdo a
      cada una
232 // de la de las reglas
233
```



```
267
268  pinMode(pinPWMB, OUTPUT); //Motor 2 Media
269  pinMode(pinBIN1, OUTPUT);
270  pinMode(pinBIN2, OUTPUT);
271
272  pinMode(pinSTBY2, OUTPUT);
273
274  pinMode(pinPWMC, OUTPUT); //Motor 3 Proximal
275  pinMode(pinCIN1, OUTPUT);
276  pinMode(pinCIN2, OUTPUT);
277
278  pinMode(pinPWMD, OUTPUT); //Motor 3 Media
279  pinMode(pinDIN1, OUTPUT);
280  pinMode(pinDIN2, OUTPUT);
281
282  pinMode(pinSTBY3, OUTPUT);
283
284  pinMode(pinPWME, OUTPUT); //Motor 4 Proximal
285  pinMode(pinEIN1, OUTPUT);
286  pinMode(pinEIN2, OUTPUT);
287
288  pinMode(pinPWMF, OUTPUT); //Motor 4 Media
289  pinMode(pinFIN1, OUTPUT);
290  pinMode(pinFIN2, OUTPUT);
291
292  pinMode(pinSTBY4, OUTPUT);
293
294
295  pinMode(pinPWMG, OUTPUT); //Motor 5
296  pinMode(pinGIN1, OUTPUT);
297  pinMode(pinGIN2, OUTPUT);
298
299 //  pinMode(pinPWMD, OUTPUT); //Motor 1 Pulgar
300 //  pinMode(pinDIN1, OUTPUT);
301 //  pinMode(pinDIN2, OUTPUT);
302
303  pinMode(pinSTBY5, OUTPUT);
304
305 }
306
307 void loop() {
308  ////////////////////////////////////////Ida
309  if (recibir==0){
310      Serial.println(1000);
```

```
311
312     String str1 = Serial.readString();
313     Referencia21 = str1.toInt();
314     Referencia21P=Referencia21;
315     Serial.println(Referencia21);
316
317     String str2= Serial.readString();
318     Referencia22 = str2.toInt();
319     Referencia22P=Referencia22;
320     Serial.println(Referencia22);
321
322     String str3 = Serial.readString();
323     Referencia31 = str3.toInt();
324     Referencia31P=Referencia31;
325     Serial.println(Referencia31);
326
327     String str4 = Serial.readString();
328     Referencia32 = str4.toInt();
329     Referencia32P=Referencia32;
330     Serial.println(Referencia32);
331
332     String str5 = Serial.readString();
333     Referencia41 = str5.toInt();
334     Referencia41P=Referencia41;
335     Serial.println(Referencia41);
336
337     String str6 = Serial.readString();
338     Referencia42 = str6.toInt();
339     Referencia42P=Referencia42;
340     Serial.println(Referencia42);
341
342     String str7 = Serial.readString();
343     Referencia51 = str7.toInt();
344     Referencia51P=Referencia51;
345     Serial.println(Referencia51);
346
347     recibir=2;
348 }
349 ////////////////////////////////////////////////////Regreso
350 if ((Regreso==7) && (Vez==0)) {
351     delay(1000);
352     delay(1000);
353     delay(1000);
354     delay(1000);
```



```
399 e21=(Referencia21-count21)/Referencia21P;
400 de21=(e21-eant21);
401 eant21=e21;
402 fuzzy->setInput(1, e21);
403 fuzzy->setInput(2, de21);
404 fuzzy->fuzzify();
405 int Accion21=int(fuzzy->defuzzify(1));
406 if (Accion21>0){
407     pinIn21 = HIGH;
408 }
409 if (Accion21<0){
410     pinIn21 = LOW;
411     Accion21=-Accion21;
412 }
413 if ((Accion21==0) && (Meta21==0)) {
414     Regreso=Regreso+1;
415     Meta21=1;
416 }
417 ////////////////////////////////////////Motor 22
418 count22=-myEnc22.read();
419 e22=(Referencia22-count22)/Referencia22P;
420 de22=(e22-eant22);
421 eant22=e22;
422 fuzzy->setInput(1, e22);
423 fuzzy->setInput(2, de22);
424 fuzzy->fuzzify();
425 int Accion22=int(fuzzy->defuzzify(1));
426 if (Accion22>0){
427     pinIn22 = HIGH;
428 }
429 if (Accion22<0){
430     pinIn22 = LOW;
431     Accion22=-Accion22;
432 }
433 if ((Accion22==0) && (Meta22==0)) {
434     Regreso=Regreso+1;
435     Meta22=1;
436 }
437 ////////////////////////////////////////Motor 31
438 count31=-myEnc31.read();
439 e31=(Referencia31-count31)/Referencia31P;
440 de31=(e31-eant31);
441 eant31=e31;
442 fuzzy->setInput(1, e31);
```

```
443 fuzzy->setInput(2, de31);
444 fuzzy->fuzzify();
445 int Accion31=int(fuzzy->defuzzify(1));
446 if (Accion31>0){
447     pinIn31 = HIGH;
448 }
449 if (Accion31<0){
450     pinIn31 = LOW;
451     Accion31=-Accion31;
452 }
453 if ((Accion31==0)&&(Meta31==0)){
454     Regreso=Regreso+1;
455     Meta31=1;
456 }
457
458 ////////////////////////////////////////////////////Motor 32
459 count32=-myEnc32.read();
460 e32=(Referencia32-count32)/Referencia32P;
461 de32=(e32-eant32);
462 eant32=e32;
463 fuzzy->setInput(1, e32);
464 fuzzy->setInput(2, de32);
465 fuzzy->fuzzify();
466 int Accion32=int(fuzzy->defuzzify(1));
467 if (Accion32>0){
468     pinIn32 = HIGH;
469 }
470 if (Accion32<0){
471     pinIn32 = LOW;
472     Accion32=-Accion32;
473 }
474 if ((Accion32==0)&&(Meta32==0)){
475     Regreso=Regreso+1;
476     Meta32=1;
477 }
478
479 ////////////////////////////////////////////////////Motor 41
480 count41=-myEnc41.read();
481 e41=(Referencia41-count41)/Referencia41P;
482 de41=(e41-eant41);
483 eant41=e41;
484 fuzzy->setInput(1, e41);
485 fuzzy->setInput(2, de41);
486 fuzzy->fuzzify();
```

```
487 int Accion41=int(fuzzy->defuzzify(1));
488 if (Accion41>0){
489     pinIn41 = HIGH;
490 }
491 if (Accion41<0){
492     pinIn41 = LOW;
493     Accion41=-Accion41;
494 }
495 if ((Accion41==0)&&(Meta41==0)){
496     Regreso=Regreso+1;
497     Meta41=1;
498 }
499
500 ////////////////////////////////////////////////////////////////////Motor 42
501 count42=-myEnc42.read();
502 e42=(Referencia42-count42)/Referencia42P;
503 de42=(e42-eant42);
504 eant42=e42;
505 fuzzy->setInput(1, e42);
506 fuzzy->setInput(2, de42);
507 fuzzy->fuzzify();
508 int Accion42=int(fuzzy->defuzzify(1));
509 if (Accion42>0){
510     pinIn42 = HIGH;
511 }
512 if (Accion42<0){
513     pinIn42 = LOW;
514     Accion42=-Accion42;
515 }
516 if ((Accion42==0)&&(Meta42==0)){
517     Regreso=Regreso+1;
518     Meta42=1;
519 }
520
521 ////////////////////////////////////////////////////////////////////Motor 51
522 count51=-myEnc51.read();
523 e51=(Referencia51-count51)/Referencia51P;
524 de51=(e51-eant51);
525 eant51=e51;
526 fuzzy->setInput(1, e51);
527 fuzzy->setInput(2, de51);
528 fuzzy->fuzzify();
529 int Accion51=int(fuzzy->defuzzify(1));
```

```
531  if (Accion51>0){
532      pinIn51 = HIGH;
533  }
534  if (Accion51<0){
535      pinIn51 = LOW;
536      Accion51=-Accion51;
537  }
538  if ((Accion51==0) && (Meta51==0)) {
539      Regreso=Regreso+1;
540      Meta51=1;
541  }
542
543  ////////////////////////////////////////////Activaciones
544  if ((Parar==1) && (Regreso==14)) {
545
546      digitalWrite(pinSTBY2, LOW);
547
548      digitalWrite(pinSTBY3, LOW);
549
550      digitalWrite(pinSTBY4, LOW);
551
552      digitalWrite(pinSTBY5, LOW);
553
554  }
555  else{
556
557      digitalWrite(pinAIN1, pinIn21);
558      digitalWrite(pinAIN2, !pinIn21); //Motor 21
559      analogWrite(pinPWMA, Accion21);
560
561      digitalWrite(pinBIN1, pinIn22);
562      digitalWrite(pinBIN2, !pinIn22); //Motor 22
563      analogWrite(pinPWMB, Accion22);
564
565      digitalWrite(pinSTBY2, HIGH);
566
567      digitalWrite(pinCIN1, pinIn31);
568      digitalWrite(pinCIN2, !pinIn31); //Motor 31
569      analogWrite(pinPWMC, Accion31);
570
571      digitalWrite(pinDIN1, pinIn32);
572      digitalWrite(pinDIN2, !pinIn32); //Motor 32
573      analogWrite(pinPWMD, Accion32);
574
```

```
575     digitalWrite(pinSTBY3, HIGH);
576
577
578     digitalWrite(pinEIN1, pinIn41);
579     digitalWrite(pinEIN2, !pinIn41); //Motor 41
580     analogWrite(pinPWME, Accion41);
581
582     digitalWrite(pinFIN1, pinIn42);
583     digitalWrite(pinFIN2, !pinIn42); //Motor 42
584     analogWrite(pinPWF, Accion42);
585
586     digitalWrite(pinSTBY4, HIGH);
587
588     digitalWrite(pinGIN1, pinIn51);
589     digitalWrite(pinGIN2, !pinIn51); //Motor 51
590     analogWrite(pinPWGM, Accion51);
591
592     digitalWrite(pinSTBY5, HIGH);
593
594 }
595 }
```



# Apéndice F

## Publicaciones derivadas.

- F.1. 4th IFToMM Symposium on Mechanism Design for Robotics  
(MEDER 2018)



# Development of Hand Exoskeleton Prototype for Assisted Rehabilitation

B. A. De la Cruz-Sánchez<sup>1</sup>(✉), M. Arias-Montiel<sup>2</sup>, and E. Lugo-González<sup>2</sup>

<sup>1</sup> Division of Postgrade Studies, Universidad Tecnológica de la Mixteca,  
Huajuapán de León, Oaxaca, Mexico  
berithcruzs@gmail.com.mx

<sup>2</sup> Institute of Electronics and Mechatronics, Universidad Tecnológica de la Mixteca,  
Huajuapán de León, Oaxaca, Mexico  
{mam,elugog}@mixteco.utm.mx

**Abstract.** The present work describes the design and construction of an exoskeleton prototype for hand rehabilitation. The device is designed for independent movements on each finger, considering the anthropometry of Mexican people and the rehabilitation paradigm. The stages for the prototype development are the design, mechanical characterization, position and orientation control, and exoskeleton assembly. As a result, the first prototype of exoskeleton for hand rehabilitation was obtained, thus, indicating the materials used in the different pieces and performing some physical tests for the validation of the mechanism.

**Keywords:** Exoskeleton · Mechanical design · Rehabilitation

## 1 Introduction

The hand is one of the parts of the body with the highest use in daily life; therefore it is frequently exposed to injuries that can be caused by many reasons related to diseases and accidents at work or home. Numerous studies have shown that early intervention and the use of assistive devices with patients allow higher recovering of their abilities, stimulating in recent years the development of exoskeletal robotic systems with different approaches to the problem.

In the bibliography it is possible to observe a trend in the development of devices that during design and construction do not consider the type of therapy to be performed, [1] is emphasized to consider this throughout all the design process, allowing to obtain devices that are more secure. The training paradigms are classified according to several aspects, where the primary point represents the human-robot interaction. The classification of the training modalities is related to the conventional therapy modes used in clinical practice. The assisted rehabilitation, where the voluntary activity of the patient is required throughout the movement. The robot can help to provide support weight or strength in order to complete the task [2].

Hand rehabilitation devices have been developed with distinct kinds of actuation, either pneumatic [3], hydraulic [4] or by cables [5]. But these characteristics; like weight, degrees of freedom and limitation to only a certain type of therapy, make them less competitive devices for requirements of portability and use. One of the problems that needs to be solved with regards to the development of devices of rehabilitation, is the coincidence of the rotation center of the mechanism with that of human joints because the offset will cause limited movements and injuries in the patient's hand. The device developed in [6] solves this problem, by proposing a new articulation mechanism called indirect link, which adopts the mechanism of symmetrical rack and pinion (SPRM). Another device that solves this problem is the robot developed in [7], this device was designed for passive and active rehabilitation, it has a total weight of 731 g, being one of the lightest devices reported in literature. The selection of the material for the construction of these types of devices is very important, because it allows the acquisition of light and resistant devices. In [8] a rehabilitation hand mechanism of 1 DOF is proposed, built by using Acrylonitrile Butadiene Styrene (ABS), and a motion capture system (MoCap) for the design of kinematic chains of the exoskeleton, based on a Full-HD camera with slow motion rate and a 2D acquisition software.

Nowadays portability is one of the most important points in the development of devices thus, allowing its use in activities of daily living (ADL). One example of this, is the Mano device [9], designed to assist and restore hand functions of people with motor disabilities during ADL and in neurorehabilitation scenarios. The Mano system consists of two main parts, the chest pack, wearable by means of adjustable belts which, host the actuation control and the energy storage unit and the artificial tendons that consist of Bowden cables actuated by linear servomotors. Based on the studies, this paper proposes the design and construction of a prototype of hand exoskeleton printing with 3D technology, based on an SPRM mechanism with DC motor actuation focus on assistive rehabilitation, obtaining a device with a lightweight and the ability to have independent movement of the fingers.

## 2 Design

Based on the designs of [6,7] the concept is created as a combination of both mechanisms for the development of the exoskeleton. Figure 1 shows the base mechanism of the device reported in [6], which is made of gears and racks; it is possible to observe that to reduce the error between the instantaneous velocity centers of the phalanx ( $C$ ) and the mechanism ( $V$ ), it is necessary to consider a variable ( $x$ ) that allows relating the distance between these two centers ( $P$ ), making this distance as small as possible. The distance  $d$  between the center of rotation of the phalanx with the center of the gear of the mechanism can be related to the radius of the gear as  $r = xd$ . By performing some trigonometric and algebraic relationships, it is possible to obtain the displacement  $S_1$  that the mechanism will have in relation to  $\theta$  and which allows the acquisition of the value for  $x$  that satisfies the initial conditions, as shown in Eq. 1.

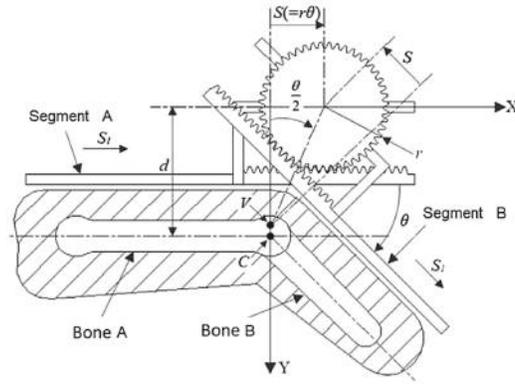


Fig. 1. Kinematic representation of the base mechanism [6].

$$S_1 = d \tan\left(\frac{\theta}{2}\right) - S = d \left( \tan\left(\frac{\theta}{2}\right) - x\theta \right) \quad (1)$$

The operation of the mechanism consists of the transmission of power from the motors of DC with a worm gearset to two sets of gear-rack, one for the rotational movement and the other for a prismatic movement. Figure 2 shows one of the three parts in which each of the exoskeleton fingers was divided, the first one is located on the metacarpal bone and secured to the base of the exoskeleton it is responsible for the movement of the proximal phalanx. The second block is located on the proximal phalanx and is responsible for the movement of the medial phalanx, the latter functions as a receptor of the movement of the block responsible for the movement of the medial phalanx and is located on the distal phalanx.

The design of the exoskeleton prototype for hand rehabilitation was obtained, selecting PLA as material for the supports and bases, making their

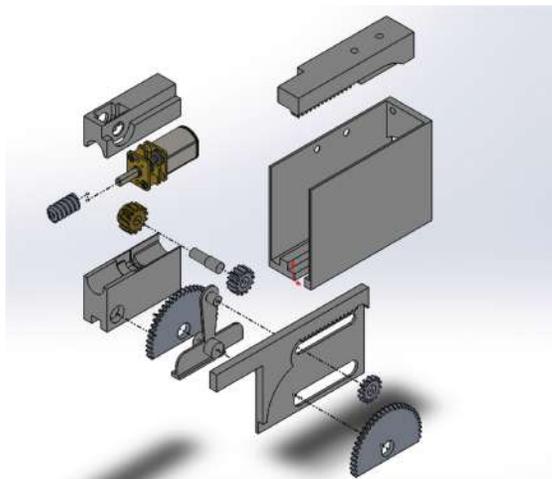
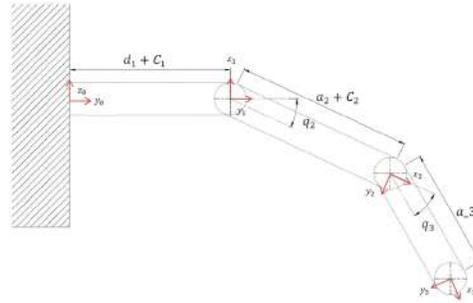


Fig. 2. Exploded view of each of the blocks that make up the exoskeleton.

manufacturing viable through 3D printing. For the elements that will be subject to greater wear such as gears, a 1060 aluminum alloy was proposed, because of the mechanical properties and commercial availability of this material. A more detailed description of the exoskeleton mechanism design is presented in [10].

### 3 Kinematic Characterization

Each finger of the exoskeleton has 2 rotational DOF that were analyzed independently. Figure 3 shows a schematic representation of the links of the finger that was used to develop the kinematic analysis.



**Fig. 3.** Links of a finger of the hand exoskeleton.

Table 1 shows the Denavit–Hartenberg used, where to allow for the inverse and forward models. Equation (2) shows the homogeneous transformation matrix for a finger of the exoskeleton.

$$T_3^0 = \begin{pmatrix} 0 & 0 & 1 & 0 \\ -\sin(q_2 + q_3) & -\cos(q_2 + q_3) & 0 & -a_3 \sin(q_2 + q_3) - a_2 \sin(q_2) - C_2 \sin(q_2) \\ \cos(q_2 + q_3) & -\sin(q_2 + q_3) & 0 & C_1 + d_1 + a_3 \cos(q_2 + q_3) + a_2 \cos(q_2) + c_2 \cos(q_2) \\ 0 & 0 & 0 & 1 \end{pmatrix} \quad (2)$$

The analytic Jacobian was obtained by using the geometric Jacobian given by the following standard formula [11]. The matrix  $T(\phi)$ , is obtained from the angles of Euler and that can be obtained by the Roll-Pitch-Yaw method (ZYX), from the angles  $\phi = [\varphi \vartheta \psi]^T$ , which represent the rotation with respect to a fixed point in the center of mass.

**Table 1.** Denavit- Hartenberg parameters

Join i	$\theta_i$	$d_i$	$a_i$	$\alpha_i$
1	$\frac{\pi}{2}$	$d_1 + C_1^*$	0	$\frac{\pi}{2}$
2	$q_2^* + \frac{\pi}{2}$	0	$a_2 + C_2^*$	0
3	$q_3^*$	0	$a_3$	0

$$J_A = \begin{bmatrix} I & 0 \\ 0 & T(\phi)^{-1} \end{bmatrix} J = \begin{bmatrix} I & 0 \\ 0 & T(\phi)^{-1} \end{bmatrix} \begin{bmatrix} J_v \\ J_w \end{bmatrix} \quad (3)$$

As each finger of the exoskeleton is of two degrees of freedom, Jacobian matrix of  $[6 \times 2]$  is obtained, in which the first three rows correspond to linear velocity and the three bottom rows correspond to rotational velocity:

$$J = \begin{bmatrix} 0 & 0 \\ 1 - d_1 - a_2 \cos(q_2) - c_2 \cos(q_2) - c_1 & -\cos(q_2)(a_2 + c_2) \\ -\sin(q_2)(a_2 + c_2) & -\sin(q_2)(a_2 + c_2) \\ 0 & \cos(th)/\cos(ga) \\ 0 & -\sin(th) \\ 1 & (\cos(th)\sin(ga))/\cos(ga) \end{bmatrix} \quad (4)$$

If we wish to know the joint velocities from the speed of the end effector, it is necessary to calculate the inverse of the Jacobian matrix, by means of the procedure of obtaining the Jacobian matrix pseudoinverse to the left, if the number of rows is greater than columns, or the Jacobian matrix pseudoinverse to the right if the number of columns is greater than rows.

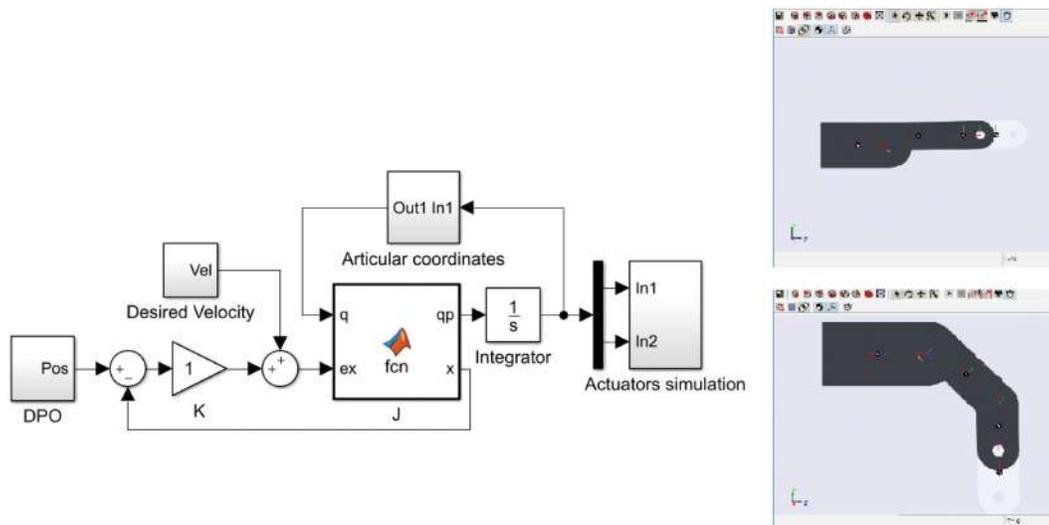
$$\dot{q} = J_a^{-1} \dot{x} \quad (5)$$

where  $\dot{x}$  is the vector of linear and rotational velocities of the final effector  $[6 \times 2]$ , in the Cartesian space (xyz).

## 4 Position and Orientation Control

The implementation of a position control is possible by using the inverse Jacobian matrix, Fig. 4 shows the Simulink block diagram for the implementation of a position control. Different position tests were performed. With the information obtained from the workspace a desired position and orientation of the end-effector can be located that we want to reach, that is, when the desired point  $x_d$  is introduced into the workspace, the finger performs a movement until it reaches the desired position. Based on the desired position and orientation it is possible to obtain the values of the joints of a finger of the exoskeleton, using the inverse kinematics and the Jacobian matrix.

By using Simulink it is possible to perform simulations in real time that allow the validation of the kinematics and the position of the finger, graphically visualizing the results obtained in the simulation. It is necessary to have the mechanism that will be linked assembled in Solidworks in order to assign actuators through a movement study. For the finger, being of 2 degrees of freedom, it has a rotary motor in each of its joints.



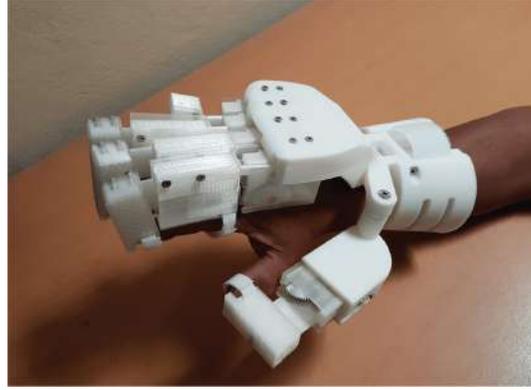
**Fig. 4.** Simulation performed in Simulink for a position control of a finger of the exoskeleton. The desired angle for each joint is  $45^\circ$ .

## 5 Exoskeleton Assembly

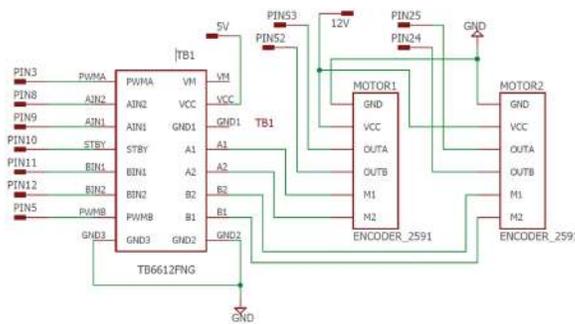
For the development of the prototype, it was decided to use 3D printing by Fused Deposition Modeling (FDM). For the movement of each phalange of the exoskeleton, 8 DC metal gearmotors with a maximum torque of 2.9 kg-cm and a weight of 10 g were used. ADAMS View software was used to calculate the required torque for each motor. A total of 40 gears were manufactured from aluminium alloy 1060, which went through different manufacturing processes to obtain the distinct sets of pinion-rack and crowns for power transmission. Figure 5 shows the actual assembly of the exoskeleton prototype in the hand of a healthy male person. The exoskeleton was tested with three different people with different thicknesses and sizes of hands, with measures of the length of hand of 16 cm, 16.7 cm and 17.8 cm. To observe the fit that the exoskeleton has and to avoid hurting the user, the inside of the base is covered with a sponge layer and a fabric layer.

All of the tests are done with one finger of the exoskeleton; each motor has a magnetic encoder from Pololu Company. For the bidirectional control of the motors, a Pololu brand card with a Toshiba TB6612FNG controller will be used [12], which allows the control of two motors by means of the generation of a pulse width modulation (PWM) signal, the connections with the motor and the microcontroller. Figure 6 shows the connection diagram of the Arduino development card with the driver, encoder, and motors.

For the manipulation of the motors it is necessary to develop a program that is loaded in the Arduino development card, with which it is possible to modify parameters of the PWM signal that will be introduced in the driver and which is seen again. In the same way, it is possible to alter the direction of rotation of the motors and to manage the working time of the motors. The test program



**Fig. 5.** Physical test of the prototype.



**Fig. 6.** Schematic diagram of the connections made between the Arduino board.

carried out allows obtaining the values read by the encoders, which are shown by the serial monitor, for the serial transmission a baud rate of 9600 is necessary.

## 6 Conclusions and Future Work

In this work, a proposal of hand exoskeleton design was presented with the capacity to reach the ranges of movement of the joints of a healthy hand, making its application viable in the generation of assisted rehabilitation routines. The design minimizes the error that exists between the instantaneous centers of the velocity of the mechanism with respect to the center of rotation of the phalanges. It has an overall weight of 702.9 g with a lower weight than those reported in literature, considering design for assemblage and manufacturing so that the device is reproducible and has an intuitive assemblage. It has 2 active DOF and 2 passive DOF in the index, middle and ring fingers, 1 active DOF and 1 passive DOF for the thumb and little finger. The geometry of the pieces was designed for 3D printing and the gears machined with aluminum. The exoskeleton satisfactorily covers the rank of movement of the phalanges, for the proximal phalanx a range of motion of 0–87.5° was obtained, for the middle phalanx a range of motion of 0–109° was obtained, the length available to be able to fit

the exoskeleton to larger hands, is 10 mm and 6 mm in each phalanx, which represents that 46.2% and 50% of the proposed values for the proximal and middle phalanx were covered respectively.

Currently, the first mechanical prototype is in the stage of instrumentation and control based on electromyographic signals, in order to implement control algorithms in the future to execute the movements useful for rehabilitation therapies and routines.

## References

1. Basteris, A., Nijenhuis, S.M., Stienen, A.H., Buurke, J.H., Prange, G.B., Amirabdollahian, F.: Training modalities in robot-mediated upper limb rehabilitation in stroke: a framework for classification based on a systematic review. *J. Neuroeng. Rehabil.* **11**(1), 111 (2014)
2. Yue, Z., Zhang, X., Wang, J.: Hand rehabilitation robotics on poststroke motor recovery. *Behav. Neurol.* **2017**, 1–20 (2017)
3. Polygerinos, P., Lyne, S., Wang, Z., Nicolini, L.F., Mosadegh, B., Whitesides, G.M., Walsh, C.J.: Towards a soft pneumatic glove for hand rehabilitation. In: 2013 IEEE/RSJ International Conference on Proceedings of Intelligent Robots and Systems (IROS), pp. 1512–1517. IEEE (2013)
4. Unluhisarcikli, O., Weinberg, B., Sivak, M., Mirelman, A., Bonato, P., Mavroidis, C.: A robotic hand rehabilitation system with interactive gaming using novel electro-rheological fluid based actuators. In: IEEE International Conference on Proceedings of Robotics and Automation (ICRA), pp. 1846–1851. IEEE (2010)
5. Delph, M.A., Fischer, S.A., Gauthier, P.W., Luna, C.H.M., Clancy, E.A., Fischer, G.S.: A soft robotic exomusculature glove with integrated sEMG sensing for hand rehabilitation. In: 2013 IEEE International Conference on Proceedings of Rehabilitation Robotics (ICORR), pp. 1–7. IEEE (2013)
6. Zhang, F., Hua, L., Fu, Y., Chen, H., Wang, S.: Design and development of a hand exoskeleton for rehabilitation of hand injuries. *Mech. Mach. Theory* **73**, 103–116 (2014)
7. Sandoval-Gonzalez, O., Jacinto-Villegas, J., Herrera-Aguilar, I., Portillo-Rodriguez, O., Tri-picchio, P., Hernandez-Ramos, M., Flores-Cuautle, A., Avizzano, C.: Design and development of a hand exoskeleton robot for active and passive rehabilitation. *Int. J. Adv. Robot. Syst.* **13**(2), 66 (2016)
8. Conti, R., Meli, E., Ridolfi, A.: A novel kinematic architecture for portable hand exoskeletons. *Mechatronics* **35**, 192–207 (2016)
9. Randazzo, L., Iturrate, I., Perdakis, S., Millán, J.D.R.: Mano: a wearable hand exoskeleton for activities of daily living and neurorehabilitation. *IEEE Robot. Autom. Lett.* **3**(1), 500–507 (2018)
10. De la Cruz Sánchez, B.A., Arias-Montiel, M., González-Lugo, E.: Design of a prototype exoskeleton for hand rehabilitation (in Spanish). *Pistas Educativas* **39**(125), 96–111 (2017)
11. Sciavicco, L., Siciliano, B.: Modelling and Control of Robot Manipulators. Springer-Verlag, London UK (2000)
12. Pololu. [https://www.pololu.com/docs/pdf/0J55/dua\\_mc33926motor\\_driver\\_shield.pdf](https://www.pololu.com/docs/pdf/0J55/dua_mc33926motor_driver_shield.pdf). Accessed 1 Feb 2018

## **F.2. Congreso Nacional de Ingeniería Biomédica (SOMIB 2018)**

## Diseño y Construcción de un Prototipo de Exoesqueleto para Rehabilitación de Mano

B. A. De La Cruz Sánchez<sup>1,\*</sup>, M. Arias Montiel<sup>1,\*\*</sup>, E. Lugo González<sup>1,\*\*\*</sup>

<sup>1</sup> Instituto de Electrónica y Mecatrónica, Universidad Tecnológica de la Mixteca, Huajuapán de León, Oaxaca, México

\* berithcruzs@gmail.com, \*\*mam@mixteco.utm.mx, \*\*\*elugog@mixteco.utm.mx

*Resumen*— El presente trabajo describe el diseño y la construcción de un prototipo de exoesqueleto para rehabilitación de mano. El dispositivo se diseña con capacidad de desarrollar movimientos independientes en cada uno de los dedos, tomando en cuenta la antropometría de personas mexicanas y considerando metodologías de diseño para el ensamble y la manufactura. Se presentan las etapas de diseño, caracterización cinemática del mecanismo, control de posición y orientación y el ensamble del prototipo. Como resultado, presenta un primer prototipo de exoesqueleto para rehabilitación de mano, indicando los materiales usados en las diferentes piezas, construcción y realización de algunas pruebas físicas para la validación del mecanismo.

*Palabras clave*—Exoesqueleto, Diseño mecánico, Rehabilitación

### I. INTRODUCCIÓN

La mano es una de las partes del cuerpo con mayor uso en la vida diaria, por lo que se encuentra expuesta a sufrir lesiones por enfermedades o accidentes durante las actividades en el trabajo o en casa, generando en algunas ocasiones discapacidades que imposibilitan la reintegración de los afectados a la vida cotidiana. La rehabilitación permite mejorar esas deficiencias buscando la recuperación de las capacidades de los pacientes, lo que ha llevado a distintas universidades, institutos y empresas privadas, a incursionar en el diseño y fabricación de dispositivos de rehabilitación. Este tipo de dispositivos conlleva ciertas ventajas, como ayudar al terapeuta con la carga de trabajo, ayudar en movimiento repetitivos, aumentar el número de terapias [1], acelerar el proceso de recuperación de los pacientes, de manera segura y confiable, incremento en la fuerza, mejoría de la coordinación, modificaciones al tono muscular y mayor independencia funcional [2].

En los últimos años, el desarrollo de sistemas mecatrónicos para el apoyo a la rehabilitación ha tenido un auge e incremento considerable, en el mercado se pueden encontrar opciones de este tipo de dispositivos para distintas partes del cuerpo, pero cabe mencionar que esta tecnología en México apenas se está desarrollando, siendo necesaria la importación de países como EUA, Alemania, Japón entre otros. La importancia de mejorar la función de la mano

después de sufrir alguna lesión ha estimulado el desarrollo de dispositivos exoesqueléticos con diferentes enfoques al problema para este miembro del cuerpo, diseñados con diferentes tipos de accionamiento, ya sea neumático [3], hidráulico [4] o por cables [5]. Pero características como el peso, los grados de libertad y la limitación al tipo de terapia de aplicación, los convierten en dispositivos poco competitivos para los requisitos de portabilidad y uso.

Uno de los problemas principales durante del desarrollo de dispositivos de rehabilitación, es la coincidencia del centro de rotación del mecanismo con el de las articulaciones humanas, ya que este error que puede existir en diferentes mecanismos puede provocar limitaciones para cubrir los rangos de movimiento o inclusive ocasionar lesiones al paciente. El dispositivo desarrollado en [6] resuelve este problema, al proponer un nuevo mecanismo de articulación llamado enlace indirecto, adoptando un mecanismo de cremallera y piñón simétricos (SPRM, por sus siglas en inglés). Otro dispositivo que resuelve este problema es el exoesqueleto desarrollado en [7] por el Instituto Tecnológico de Orizaba en conjunto con la Universidad Nacional Autónoma de México, este dispositivo fue diseñado para la rehabilitación pasiva y activa, el movimiento se logra con tornillos sin fin y engranes rectos para la transmisión de potencia, tiene un peso total de 731 g, siendo uno de los dispositivos más ligeros reportados en la literatura. La selección del material para la construcción de este tipo de dispositivos es muy importante, ya que permite la obtención de dispositivos ligeros y resistentes.

Con base en lo mencionado anteriormente, en este artículo se propone el desarrollo de un prototipo de exoesqueleto, basado en un mecanismo de SPRM con actuación mediante motores de DC, enfocado a la rehabilitación de mano de pacientes que sufrieron algún accidente cerebrovascular. El dispositivo se diseña para el movimiento independiente de los dedos y para mantener la coincidencia del centro de rotación de las falanges con el mecanismo del exoesqueleto. Además, las juntas del mecanismo son capaces de cubrir los ángulos de movimiento de los dedos, considerando factores de ergonomía, funcionalidad, diseño para el ensamble y para la

manufactura [8], [9]. El diseño es evaluado utilizando simulaciones numéricas en MATLAB® para el análisis cinemático del exoesqueleto. Se realizó la construcción de un primer prototipo del exoesqueleto propuesto y se llevaron a cabo algunas pruebas en personas sanas para observar el ajuste del dispositivo.

## II. METODOLOGÍA

### A. Diseño conceptual

El diseño conceptual para el exoesqueleto se realiza mediante el proceso frontal detallado en [8] y la metodología de diseño para el ensamble y la manufactura [9]. El concepto se crea como una combinación de los diseños presentados en [6,7]. El mecanismo base consiste en dos juegos de engranes-cremalleras como se muestra en la Fig. 1.

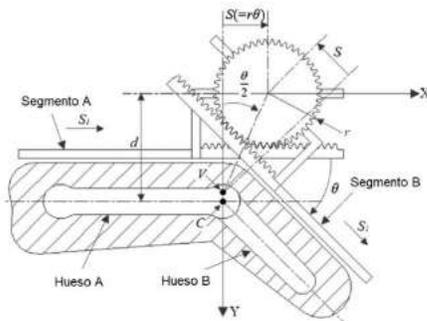


Fig. 1. Representación cinemática del mecanismo base [6].

Es posible observar que para reducir el error entre los centros de velocidad instantáneos de la falange (C) y el mecanismo (V), es necesario considerar una variable (x) que permita relacionar la distancia (P) entre estos dos centros, para hacer esta distancia tan pequeña como sea posible. La distancia (d) entre el centro de rotación de la falange y el centro del engranaje del mecanismo puede ser relacionada con el radio del engrane como  $r = xd$ . Al realizar algunas relaciones trigonométricas y algebraicas, es posible obtener el desplazamiento  $S_1$  que tendrá el mecanismo en relación con  $\theta$  mediante (1), permitiendo la estimación del valor de x que satisfice las condiciones iniciales. Una descripción detallada de la propuesta de diseño para los mecanismos del exoesqueleto se puede consultar en [10].

$$S_1 = d \cdot \tan(\theta/2) - S = d(\tan(\theta/2) - x\theta) \quad (1)$$

### B. Análisis cinemático

Cada dedo del exoesqueleto cuenta con 2 GDL rotacionales que se analizaron de forma independiente. En la Fig. 2 se muestra una representación esquemática de los eslabones del dedo que se utilizó para la obtención de la cinemática.

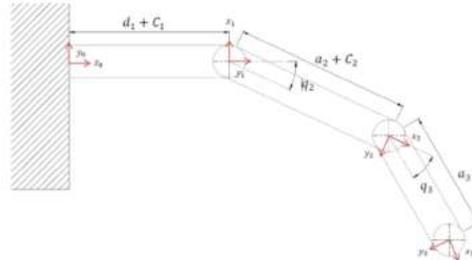


Fig. 2. Eslabones de un dedo del exoesqueleto de mano.

La Tabla 1 muestra los parámetros de Denavit-Hartenberg utilizados, permitiendo obtener la matriz de transformación homogénea, para un dedo del exoesqueleto.

TABLA I  
PARÁMETROS DE DENAVIT-HARTENBERG

Articulación	$\theta_i$	$d_i$	$a_i$	$\alpha_i$
1	$\pi/2$	$d_1 + c_1$ *	0	$\pi/2$
2	$q_2 + \pi/2$	0	$a_2 + c_2$ *	0
3	$q_3$ *	0	$a_3$	0

La relación de las velocidades de articulaciones y las velocidades lineales y angulares, en el efector final, pueden describirse por la matriz Jacobiana, también conocida simplemente como Jacobiano [11]. El Jacobiano analítico se obtiene utilizando el Jacobiano geométrico ( $J$ ) dado por la fórmula estándar (2), donde la matriz  $T(\phi)$  se obtiene de los ángulos de Euler a partir del método Roll-Pitch-Yaw para los ángulos  $\phi = [\varphi \ \theta \ \psi]^T$  que representan la rotación con respecto a un punto fijo colocado en el centro de masa del sistema.

$$J_A = \begin{bmatrix} I & 0 \\ 0 & T(\phi) \end{bmatrix} J \quad (2)$$

Como cada dedo del exoesqueleto tiene dos grados de libertad, se obtiene la matriz jacobiana de  $[6 \times 2]$ , las primeras tres filas corresponden a la velocidad lineal y las tres filas inferiores corresponden a la velocidad de rotación.

Si se desean conocer las velocidades articulares a partir de la velocidad del efector final, es necesario calcular la inversa de la matriz jacobiana, por medio del procedimiento de la obtención de la matriz jacobiana pseudoinversa a la izquierda, si es mayor el número de renglones que columnas, o la matriz jacobiana pseudoinversa a la derecha

si es mayor el número de columnas que renglones. En la ecuación (3) se muestra la relación de las velocidades de las articulaciones con la inversa de la matriz Jacobiana.

$$\dot{q} = J_A^{-1} \dot{x} \quad (3)$$

Donde  $\dot{x}$  es el vector de velocidades lineales y rotacionales del efector final [6x2], en el espacio cartesiano (x y z).

### C. Control de posición y orientación

Con la matriz Jacobiana Analítica Inversa de un dedo del Exoesqueleto de mano, es posible la implementación de un control proporcional de posición y orientación, para la validación del modelo cinemático de un dedo del exoesqueleto. Siguiendo el esquema de control propuesto por [11] se muestra la Fig. 3 la implementación en Simulink. Se realizaron diferentes pruebas de posición con la información obtenida de posición y orientación del espacio de trabajo al introducir un punto deseado  $x_d$ .

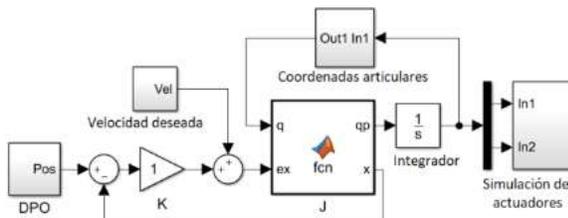


Fig. 3. Simulación del control de posición en Simulink.

En la Fig. 4 es posible observar gráficamente mediante una simulación de Simscape Multibody de Simulink/Matlab, la posición del mecanismo para una posición deseada, en este caso las posiciones mostradas son el reposo y 45° para cada falange.



Fig. 5. Simulación en Simscape Multibody de Simulink/Matlab.

### B. Ensamble del exoesqueleto

Para el desarrollo del prototipo, se decidió utilizar la impresión 3D mediante Modelado por Deposición Fundida (FDM). Para el movimiento de cada falange del exoesqueleto, se utilizó 1 motorreductor de CD con un par máximo de 2,9 kg-cm y un peso de 10 g. El software ADAMS View se usó para calcular el par requerido para

cada motor. Se fabricaron un total de 40 engranajes a partir de la aleación de aluminio 1060, que pasó por diferentes procesos de fabricación para obtener los distintos juegos de piñones y coronas para la transmisión de potencia. En la Fig. 4a) se muestra el ensamble correspondiente al movimiento de la falange proximal, mientras que en la Fig. 4b) se presenta el ensamble del motor y las bases para la transmisión de potencia. En la Fig. 5 muestra el acoplamiento del prototipo del exoesqueleto en la mano de una persona masculina sana. El exoesqueleto se probó con tres personas diferentes con diferentes espesores y tamaños de manos, con medidas de longitud de la mano de 16 cm, 16,7 cm y 17,8 cm. Para observar el ajuste que el exoesqueleto tiene y evitar lastimar al usuario, el interior de la base está cubierto con una capa de esponja y una capa de tela.

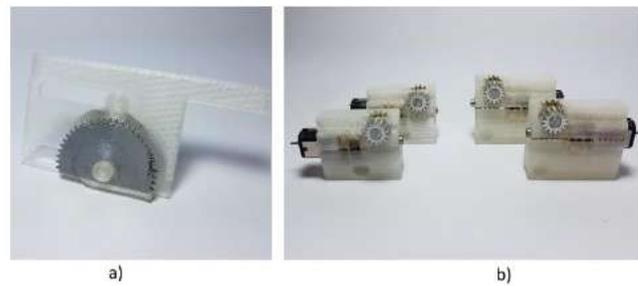


Fig. 4. Ensamble componentes.

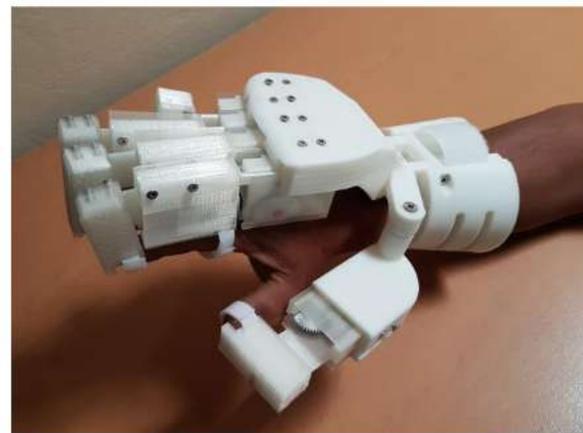


Fig. 5. Prueba física del prototipo.

Todas las pruebas se realizan con un dedo del exoesqueleto, donde cada motor tiene un codificador magnético. Para el control bidireccional de los motores, se utilizó una tarjeta de marca Polulo con un controlador Toshiba TB6612FNG [12], que permite el control de dos motores mediante la generación de una señal de modulación por ancho de pulso (PWM), estas pruebas fueron realizadas en lazo abierto para observar de manera física que el

mecanismo lograba los rangos de movimiento. Para la manipulación de los motores fue necesario desarrollar un programa que se carga en la tarjeta de desarrollo Arduino, con la cual es posible modificar los parámetros de la señal PWM que se introducirán en el controlador. De la misma manera, es posible alterar la dirección de rotación de los motores y gestionar el tiempo de trabajo de estos.

### III. RESULTADOS

El diseño mecánico propuesto del exoesqueleto tiene un peso total de 702.9 g, 2 GDL activos y 2 GDL pasivos en los dedos índice, medio y anular, 1 GDL activo y 1 GDL pasivo para el dedo pulgar y meñique. La geometría de las piezas fue diseñada para impresión 3D y un ensamble intuitivo. El exoesqueleto cubre satisfactoriamente el rango de movimiento de las falanges, para la falange proximal se obtuvo un rango de movimiento de  $0^{\circ}$ - $87.5^{\circ}$  para la falange media se obtuvo un rango de movimiento de  $0^{\circ}$ - $109^{\circ}$ , la longitud disponible para poder acoplar el exoesqueleto a manos más grandes, es de 10 mm y 6 mm en cada falange, lo que representa que se cubrió un 46.2% y un 50% de los valores propuestos para la falange proximal y media respectivamente.

### IV. DISCUSIÓN

El diseño propuesto del exoesqueleto propuesto tiene la capacidad de alcanzar los rangos de movimiento de las articulaciones de una mano sana, haciendo viable su aplicación en rutinas de rehabilitación. El prototipo de exoesqueleto construido tiene un peso de 702.9 g, lo cual lo convierte en uno de los más ligeros en comparación con los reportados en la literatura consultada. El dispositivo fue construido a base de piezas fabricadas mediante impresión 3D, engranes maquinados en aluminio y piezas disponibles comercialmente. Su diseño permite un ensamble fácil e intuitivo, permitiendo el reemplazo o la modificación de los componentes.

### V. CONCLUSIÓN

En este trabajo se presentó el diseño mecánico, el análisis cinemático, la manufactura y el ensamble de un primer prototipo de exoesqueleto para rehabilitación de mano, el cual fue diseñado considerando la antropometría mexicana, ofreciendo una ventaja en comparación a la gran mayoría de dispositivos presentados en el estado del arte, capaz de desarrollar movimiento independiente de cada uno de los dedos con la capacidad de alcanzar los rangos de movimiento de las articulaciones de una mano sana,

haciendo viable su aplicación en la generación de rutinas de rehabilitación.

El diseño está enfocado a minimizar el error que existe entre el centro instantáneo de velocidad del mecanismo con respecto al centro de rotación de las falanges, para lo cual fue necesario considerar la fisiología de la mano y la antropometría de la población mexicana. El diseño presentado, es una primera aproximación a un modelo que pueda ser funcional, por lo tanto es susceptible a mejoras y cambios en el diseño, implementar y desarrollar los sistemas de instrumentación y control de manera física que permitan la realización de pruebas con manos de usuarios.

### BIBLIOGRAFÍA

- [1] Tsoi YH, Xie SQ. Design and control of a parallel robot for ankle rehabilitation. *International Journal of Intelligent Systems Technologies and Applications*. 8(1-4):100-113, 2009.
- [2] Magaña PL. Introducción a la rehabilitación robótica para el tratamiento de la enfermedad vascular cerebral: revisión. *Revista Mexicana de Medicina Física y Rehabilitación*. 27(2):44-48, 2015.
- [3] Stilli A, Cremonesi A, Bianchi M, Ridolfi A, Gerii F, Vannetti F, Wurdemann HA, Allotta B, Althoefer K. AirExGlove—A novel pneumatic exoskeleton glove for adaptive hand rehabilitation in post-stroke patients. In: 2018 IEEE International Conference on Soft Robotics (RoboSoft). IEEE, 2018.
- [4] Yap HK, Khin PM, Koh TH, Sun Y, Liang X, Lim JH, Yeow CH. A fully fabric-based bidirectional soft robotic glove for assistance and rehabilitation of hand impaired patients. *IEEE Robotics and Automation Letters*. 2(3):1383-90, 2017.
- [5] Agarwal P, Deshpande AD. Subject-specific Assist-as-needed Controllers for a Hand Exoskeleton for Rehabilitation. *IEEE Robotics and Automation Letters*. 3(1):508-15, 2018.
- [6] Zhang F, Hua L, Fu Y, Chen H, Wang S. Design and development of a hand exoskeleton for rehabilitation of hand injuries. *Mechanism and Machine Theory*, 73:103-106, 2014.
- [7] Sandoval-Gonzalez O, Jacinto-Villegas J, Herrera-Aguilar I, Portillo-Rodriguez O, Tripicchio P, Hernandez-Ramos M, Flores-Cuautle A, Avizzano C. Design and development of a hand exoskeleton robot for active and passive rehabilitation. *International Journal of Advanced Robotic Systems*. 13(2):66, 2016.
- [8] Ulrich-Karl T, Eddinger SD. Diseño y desarrollo de productos-enfoque multidisciplinario. Madrid: Editorial McGraw Hill. 2004.
- [9] Ullman DG. The mechanical design process. New York: McGraw-Hill; 1992.
- [10] De La Cruz Sánchez BA. Diseño de un prototipo de exoesqueleto para rehabilitación de mano. Ingeniería en Mecatrónica, Tesis de Licenciatura. Universidad Tecnológica de la Mixteca. Huajuapán de León, Oaxaca. 2017.
- [11] Sciacivico L, Siciliano B. Modelling and control of robot manipulators. Springer Science & Business Media; 2012.
- [12] Pololu. <https://www.pololu.com/docs/pdf/0J55/dualmc33926motordrivershield.pdf>. Consulted [1/02/2018].

### **F.3. Congreso Mexicano de Robótica 2019 (XXI COMROB)**

# sEMG pattern recognition for assisted rehabilitation

Berith A. De La Cruz-Sánchez<sup>1</sup>, Marco P. Schoen<sup>2</sup>, Alba Perez-Gracia<sup>2</sup>,  
Manuel Arias-Montiel<sup>3</sup> and Esther Lugo-González<sup>3</sup>

**Abstract**—This paper presents a novel algorithm for classification of surface electromyographic signals (sEMG) obtained by the Myo™ armband. The proposed algorithm allows for the detection of movements by the user and is able to reproduce these movements through an exoskeleton in an assisted rehabilitation session. The classification algorithm is based on the k-nearest neighbors (kNN) method. The performance of this algorithm depends on the preprocessing of the sEMG signals and the features that are extracted. To select the optimal set of features, five different methods were employed, where the Sequential Forward Selection (SFS) algorithm yielded the best results. The optimal feature set selection resulted in a 81.2% success rate. Similar or better results can be found in the literature, however these are based on limited subject sets. For the present study, the proposed classification fails for users who are not in the training database. To obtain robustness for classification, a database with 225 different subjects was created, allowing the armband placement in an approximate position and not in and hence creating spatial robustness.

## I. INTRODUCTION

In recent years, the development of robotic systems in support of rehabilitation tasks has been increase considerably. Because of the continuous desire to improve the function of the hand, after suffering an injury, much effort has been placed in the development of exoskeletal devices. These devices have been utilized in different types of rehabilitation applications, with most common being active therapy or passive therapy. Bilateral rehabilitation is a rehabilitation type based on the natural coordination between the two hands, improving the efficiency of grip movements on the affected hand [1]. This simultaneous movement of both limbs helps the neuromuscular system to regain stability and improves the use of the affected limb [2] [3]. For this type of therapy, the use of surface electromyography signals (sEMG) can help to identify the movement of the healthy hand and reproduce it in the affected hand. To do this, it is necessary to recognize patterns of the measured signals.

Hand gestures recognition is one of the major areas of research for engineers, scientists, and bioinformatics to interpret the human behavior in different scenarios [4]. Recognition of gestures usually is a pattern classification problem, where the individual and distinct gestures are classes and

gestural instances containing gesture-data are instances [5]. The most common classifiers used for hand gesture recognition include Support Vector Machines, Artificial Neuronal Networks, Deep convolutional Neuronal Networks and k nearest neighbors [6].

For the acquisition of sEMG signals there are several options on the market. A practical way to obtain these signals is with the Myo™ armband developed by ThalmicLabs. It has eight electromyographic sensors, a weight of 97g, transmission via bluetooth and the portability option wich is its main advantage. Myo™ armband has been used in several applications such as sign language recognition [7], robotic arm control [8], vehicle control [9] and videogames [10].

For this project, the sEMG signals are obtained from the Myo™ armband, the signal processing and its classification will be carried out in Matlab™. The main idea is to obtain a classification that can be implemented in a handexoskeleton for assisted rehabilitation. To achieve this, it is necessary to create a database of different subjects in order to address robustness issues. The approach utilized in this work is to use the bracelet with new users who have no previous exposure to it, during the training of the algorithm.

## II. CREATION OF A NEW DATABASE

The basic movements used in this project are: fist, cylindrical grasp and tip pinch. These movements are performed by healthy hands and are essential during the development of different activities in people’s daily lives. For cylindrical grasp exercises, the user exercises with a plastic cylinder that has a diameter of 8cm and a length of approximately 12cm. For the development of the fine clamp the user takes a commercial marker with the tip of the thumb and index finger. For the fist, the user only closes the fist and keeps it in that position until the test ends. Figure 1 shows the three basic movements with which the database was created.

To train the classification algorithm it is necessary to create a large database. For each movement described above, data was recorded for 5 seconds. Usually the subject does not perform the movement in a time greater than 3 seconds, the remaining time serves as a margin for the subject to react and perform the movement. 225 tests were performed with different subjects for each movement, resulting in a total set of 675 tests for the three movements. All tests were performed with the arm corresponding to the non dominant hand. The test subjects’ age was between 20 to 45 years of both gender. The tests were obtained from volunteers, the

<sup>1</sup>Master’s student, Postgraduate Division, Technological University of the Mixteca, 69000, Acatlima, Oaxaca, México berithcruz@gmail.com

<sup>2</sup>Department of Mechanical Engineering, Idaho State University, 83209, Idaho, USA. schomarc@isu.edu, perealba@isu.edu

<sup>3</sup>Institute of Electronics and Mechatronics, Technological University of the Mixteca,69000, Acatlima, Oaxaca, México. mam@mixteco.utm.mx, elugog@mixteco.utm.mx

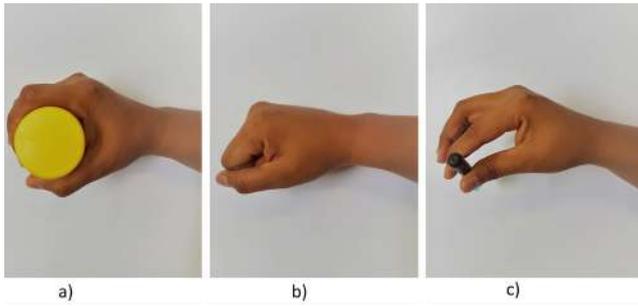


Fig. 1. Hand movements to be performed by the user. a) Cylindrical grasp, b) Fist, c) Tip pinch.

database is available in [11]. Table I shows the information regarding to the performed tests.

TABLE I

NUMBER OF SAMPLES COLLECTED BY GENDER AND HAND MOVEMENT.

Movement	Gender	Number of tests
Fist	Male	146
	Female	79
Cylindrical grasp	Male	146
	Female	79
Tip pinch	Male	146
	Female	79
Total		675

Each of the samples was stored in a separate .txt extension file with 1000 data points for each of the eight sEMG signals of the Myo<sup>TM</sup> armband.

### III. PRE-PROCESSING SIGNAL

The raw signals from the Myo<sup>TM</sup> armband contains noise and a mixture with unwanted signals. For the extraction of features, it is necessary to apply filters in order to eliminate unwanted signals components.

#### A. Detecion of muscle activation

In order to assure that the timing of the onset of a movement is captured correctly, i.e. without a delay, a data trimming process is used. Trimming the sampled data arranges the corresponding data containing the the muscular activation with actual movement hand movement class. For the detection of muscle activity, an algorithm that is mainly employed in voice detection, was used. The general idea of this concept is to consider the sEMG signal as audio. When the hand is on rest, the sEMG signals keep oscillating at values close to zero. When the user begins to make a hand movement, the signals change their amplitude. To detect these initial changes in the signals, Coper's algorithm was implemented. The algorithm proposed in [12] and named by the authors Cotrina-Peralta, is utilized for the detection of voice activity or for extremes detection, determining the start and the end moments of pronunciation with the aim to obtain a voice signal segment included in those moments. This algorithm is based on the evolution of the signal over time using a combination of energy and zero crossings concepts.

The energy of a discrete signal is defined as:

$$E = \sum_{m=0}^{N-1} x(m)^2 \quad (1)$$

Zero crossing rate of each frame is calculated as:

$$z = \sum_{m=0}^{N-1} |\text{sign}[x(m)] - \text{sign}[x(m-1)]| \quad (2)$$

For the Coper algorithm, the sign functions are multiplied by the energy of the analyzed sample, since the accumulated density depends on the sign and the amplitude of the sample. The noise has no accumulation of density because of it has a small amplitude compared to the rest of the samples, as shown in Equation (3).

$$\text{Coper} = \sum_{m=0}^N |x(m) \cdot |x(m)| - x(m-1) \cdot |x(m-1)|| \quad (3)$$

The application of the algorithm with the sEMG signals allowed for obtaining the starting point of the hand movement in the signal. By computing the arithmetic mean of the eight signals acquired from the Myo<sup>TM</sup> armband, the initial cut-off point of the signal is obtained. The second cut-off point of the signal is made 300 samples after the cut-off point obtained by the Coper algorithm. This allow for obtaining signals of 3 seconds that contain the movement information. Figure 2 shows the application of this algorithm for an sEMG signal obtained from the Myo<sup>TM</sup> armband.

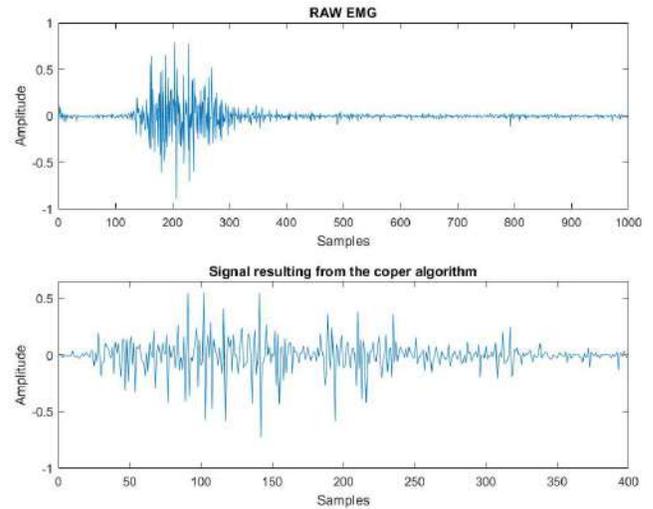


Fig. 2. sEMG signal before and after muscle activation trimming.

#### B. Filtering and normalization of the signal

The mathematical model of the sEMG signal establishes that it is composed of a mean zero carrier wave modulated by a signal that contains muscular activity. Therefore, to each of the sEMG signals from the eight channels of the Myo<sup>TM</sup> Armband, they are rectified and filtered by a Butterworth fourth order low pass filter, at a cutoff frequency of 5Hz. According to the standards for an EMG signal reported in

[13], the power density function of the surface EMG signals has negligible contributions outside the range of 5-10 Hz. The rectification of the sEMG signal was carried out with the absolute value function for each of the channels in order to eliminate the white Gaussian noise and work only with the very low frequency modulated signals that contained the muscular activity information. The processed signal is shown in Figure 3.

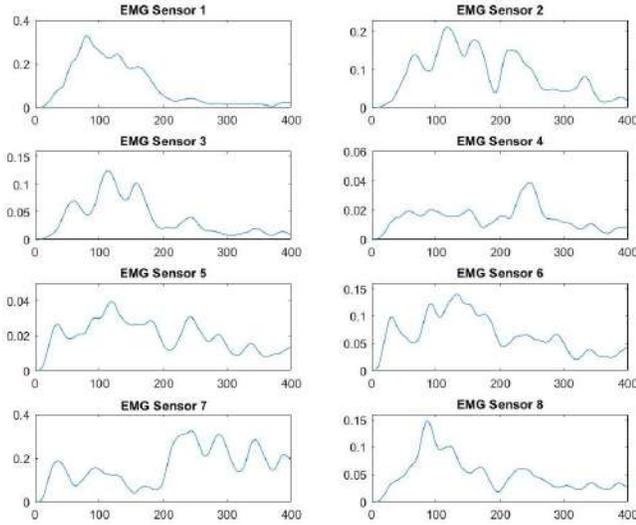


Fig. 3. sEMG signal processing for the fist movement.

### C. Feature extraction

The features extracted for the sEMG signals of each sensor are mentioned below, where for each characteristic,  $X(k)$  denotes the  $k$  sampling data in the vector and  $N$  is the length of the data vector.

1) *Waveform Length*: Waveform Length (WL) is a cumulative variation of the sEMG that can indicate the degree of variation on the sEMG signal [14]. It is defined by the Equation (4).

$$WL = \frac{1}{N} \sum_{i=1}^N |X_i|^2 \quad (4)$$

2) *Mean Absolute Value*: Mean Absolute Value (MAV) can be calculated using the average of the full-wave rectified sEMG signal, calculated by taking the average of the absolute value of the sEMG signal. It is an easy way to detect muscle contraction levels and is a popular feature used in the myoelectric control application [15]. It is defined as:

$$VAM = \frac{1}{N} \sum_{i=1}^N |X_i|^2 \quad (5)$$

3) *Root Mean Square*: Root Mean Square (RMS) is the average value of the signal [15]. It can be calculated by the Equation (6).

$$RMS = \sqrt{\frac{1}{N} \sum_{i=1}^N |X_i|^2} \quad (6)$$

4) *Kurtosis*: The kurtosis indicates the maximum point of the probability distribution associated with the instantaneous amplitudes of the time series measurements, it is a characteristic of the frequency-probability of the signal [16]. The definition is given in Equation (7):

$$Kurtosis = \frac{E((x - \mu)^4)}{\sigma^4} \quad (7)$$

5) *Zero Crossing*: Zero Crossing (ZC) is the number of times the signal passes through the zero amplitude axis. This parameter is used to obtain the approximate property in the frequency domain [17]. Given two contiguous sEMG signals,  $EMG(k)$  and  $EMG(k+1)$ , the ZC can be calculated by the Equation (8).

$$ZC = \sum_{k=1}^N \text{sgn}(f(x)) \quad (8)$$

$$\text{sgn}(x) = \begin{cases} 1 \text{ if,} & (X_k > 0 \text{ y } X_{k+1} < 0) \\ & \parallel (X_k < 0 \text{ y } X_{k+1} > 0) \\ 0, & \text{Otherwise} \end{cases}$$

6) *Willison Amplitude*: Willison Amplitude (WAMP) is a parameter that is used to count the number of times that the signal amplitude exceeds a predefined threshold, it is an indicator of the level of muscular contraction [17]. It can be written as

$$WAMP = \sum_{k=1}^{N-1} f(|X_i - X_{i+1}|) \quad (9)$$

$$f(x) = \begin{cases} 1, & \text{if } x > \text{threshold} \\ 0, & \text{Otherwise} \end{cases}$$

7) *Slope Sign Changes*: Slope Sign Changes (SSC) counts the number of times the slope of the signal changes sign. Given the three contiguous sEMG signals  $EMG(k-1)$ ,  $EMG(k)$  and  $EMG(k+1)$ , the number of slope sign changes [14]. It is given by

$$SSC = \sum_{k=1}^{N-1} f(x) \quad (10)$$

$$f(x) = \begin{cases} 1 \text{ if,} & (X_k < X_{k+1}) \text{ y } (X_k < X_{k-1}) \\ & \parallel (X_k > X_{k+1}) \text{ y } (X_k > X_{k-1}) \\ 0, & \text{Otherwise} \end{cases}$$

8) *Entropy*: Entropy (H) measures the uncertainty of a discrete random variable  $X$ , which is defined by

$$H(X) = \sum_{x \in X} P(x_i) \log(P(x_i)) \quad (11)$$

#### IV. FEATURE SELECTION

For each sampled signal it is possible to obtain 88 features, this is the result of 11 features for each of the 8 sEMG sensors. For the current database there are 675 different tests, hence an iterative process is created that extracts the 88 features of each sEMG test, thus creating an arrangement of 675x88, where each row represents a test and each column a feature. One of the purposes for creating such a large database is to reduce the influence of the random sEMG sensors alignment with regards to the muscles, as each subject and each experiment are different.

With the data containing the  $N$  features of the 675 tests, it is necessary to create a subset of  $M$  features from the original set, so that the space of features is optimally reduced. An evaluation criterion is applied in order to select the relevant features. Feature selection is an active field of research and development in computer science since 1970 and it is a fundamental problem for different areas [18]. The selection offers different advantages with the databases, such as: facilitating data visualization and understanding, reducing measurement and storage requirements, reducing training and usage times and improving prediction performance [19]. To create these subsets, there are different algorithms which can be classified into filters, wrappers and embedded methods [18]. For the database of this project, five different algorithms were applied to determine the best features. The algorithms used in this work are described below. Three algorithms of the SFS family were applied as well as the algorithms on which this family is based. SFS has proven effective in selecting features taking into account their mutual information. The application of these algorithms is useful in different areas, as in [20], where different algorithms were applied for face recognition. SFS yielded good results in databases where there were few photos of the same person.

##### A. Statistical dependency

Statistical dependency (SD) measures the dependence of the values of an entity with the associated class tags, or if the two occur simultaneously [21].

$$SD = \sum_{y \in Y} \sum_{z \in Z} P(y, z) \frac{P(y, z)}{P(y)P(z)} \quad (12)$$

##### B. Mutual Information

Mutual information (MI) is one of the most used measures to define the dependence of the variables [22]. Given two random variables  $x$  and  $y$ , their mutual information is defined in terms of their probabilistic density functions  $p(x)$ ,  $p(y)$  and  $p(x, y)$ :

$$MI = \sum_{y \in Y} \sum_{z \in Z} P(y, z) \log \left( \frac{P(y, z)}{P(y)P(z)} \right) \quad (13)$$

##### C. Sequential forward selection

Sequential forward selection (SFS), is one of the most known and widely used methods in practice for feature selection. The SFS algorithm starts working from an empty

set, the feature set is updated iteratively by including, in each step, the feature  $f$  with the maximum score  $G(S, D, M)$ . The model is expressed as the following form:

$$S_d = S_{d-1} \cup \text{argmax}G(S_{d-1} \cup f, D, M) \quad (14)$$

Where  $S$  is the subset of feature,  $D$  is the data set used and  $M$  is the classification model. The subset of features is chosen using the k Nearest Neighbor (kNN) method. This can be mathematically expressed as

$$S = \text{argmax}G(S, D, M) \quad (15)$$

##### D. Sequential Forward Floating Selection

Sequential Floating Forward Selection (SFFS) is a feature selection algorithm that is part of the SFS algorithms family. This algorithm works as an ascending search procedure that includes new features by applying the basic SFS procedure from the current feature set, followed by a series of successive conditional exclusion of the worst feature to the newly updated set, while a further improvement will be made to the previous sets [23].

##### E. Random Subset Feature Selection

The random subset feature selection (RSFS) selects a set of features that work better than an average feature of the available feature set. The most relevant feature set is obtained by repetitively choosing a random subset of features from the set of all possible features and then sorting the data with a kNN classifier. During each iteration, the relevance of each feature is adjusted according to the performance of the subset classification in which it participates, each characteristic is evaluated in terms of its average utility in the context of many other combinations of features [21].

##### F. Features selection results

The algorithms described above were implemented in Matlab<sup>TM</sup>, the result of the implementation is shown in Table II with the selection of the different most relevant features for each algorithm. The results of the algorithms coincide in the features that describe the signals. All the subsets obtained were implemented in tests using only half of the database, with the intention of evaluating the results. The best results were obtained with the SFS subset. Therefore, this algorithm was implemented for the complete database and for the final results. For the learning algorithm training, the results obtained by SFS were used, this result contains 30 of the most relevant features.

#### V. CLASSIFICATION

The machine learning techniques used in the classification of neuronal trees are vector support machines, k Nearest Neighbours, multinomial logistic regression and backward propagation in neuronal network. Using the Matlab<sup>TM</sup> toolbox called Classification Learner, it is possible to apply and compare different learning algorithms in a simple way, the only requirement is to have an array of characteristics and an output vector corresponding to each of the inputs.

TABLE II  
MOST RELEVANT CHARACTERISTICS OBTAINED BY DIFFERENT  
ALGORITHMS.

Selection algorithm	Relevant features
Statistical dependency	Relevance ordination All possible
Mutual information	Relevance ordination All possible
SFS	30
SFFS	16
RSFS	9

For the development of this project, the best performance classification algorithm was the algorithm of the k Nearest Neighbours [24].

The k Nearest Neighbors classifier (kNN) is a non-parametric method of unsupervised computer learning, in this classifier, all possible training cases are memorized, since there is no model associated with the concept learned [25]. The classifier assumes that training signals and unknown signals are in parametric space of characteristics, so the algorithm needs a distance function to compare the training signals with the new ones. The classifier input data is shown in Table III.

TABLE III  
KNN CLASSIFIER PARAMETERS

parameter	Value
Number of neighbor	1
Distance function	Euclidean
Distance	same
Validation	Cross Validation - 50 divisions

The confusion matrix resulting from the implementation k nearest neighbors classifier implemented is shown in Figure 4.

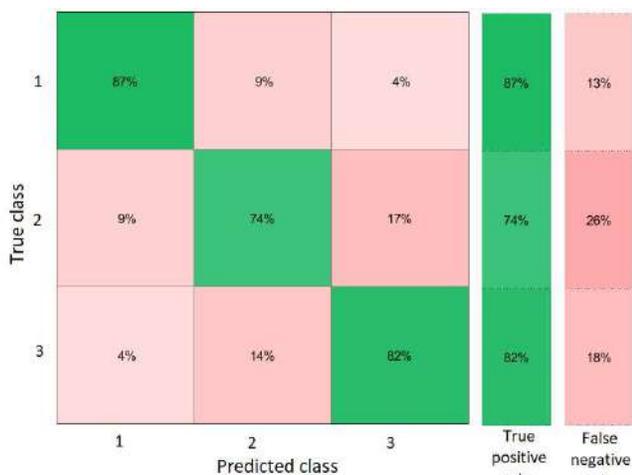


Fig. 4. confusion matrix obtained.

From the confusion matrix in Figure 4 it is possible to observe that the percentage of true positives is higher for the fist detection. For the cylindrical grip, a high percentage of

false negatives was presented with the tip pinch movement, and vice versa.

## VI. CONCLUSIONS

The total performance of the classification algorithm was 81.2 %, because the movements of fist closure and cylindrical grip are very similar. In the literature it is possible to find reported results that are above 90%, but in many cases these results do not detail the characteristics of the database, otherwise, they were obtained with the data of the same user or with a large number of tests from the same person. The goal of this work is to create a robust classification algorithm, for future implementation in an electromyographic based control. The training was carried out with an extensive database obtained from a single test per subject. Since a database has been created that allows obtaining robustness in the classification algorithm, the objective is the detection of the movement independently of the user and the exact positioning of the Myo™ armband. The number of tests obtained by gender and age of the participants were reported. The selection and extraction of features allow for saving computation time as well as the improvement of the algorithm performance. It was possible to observe that performance changes depending on the number of features, and that a large number of features do not guarantee a higher performance. Hence, it is convenient to use algorithms for the selection of a feature subset. The preprocessing of the signals is a key point in the performance of the learning algorithm, working with the base signal obtained from filtering and normalizing helps improve the classification performance. The results obtained from this work will be used to control an exoskeleton for assisted hand rehabilitation, whose mechanical design has been reported in [26] y [27].

## REFERENCES

- [1] D. Leonardis, M. Barsotti, C. Loconsole, M. Solazzi, M. Troncosi, C. Mazzotti, V. P. Castelli, C. Procopio, G. Lamola, C. Chisari *et al.*, "An emg-controlled robotic hand exoskeleton for bilateral rehabilitation," *IEEE Transactions on Haptics*, vol. 8, no. 2, pp. 140–151, 2015.
- [2] K. C. Stewart, H. J. Cauraugh, and J. J. Summers, "Bilateral movement training and stroke rehabilitation: a systematic review and meta-analysis," *Journal of the Neurological Sciences*, vol. 244, no. 1, pp. 89–95, 2006.
- [3] J. J. Summers, M. I. Garry, F. A. Kagerer, C. Y. Hiraga, and A. Loftus, "Bilateral training and recovery of upper arm function after stroke," in *Xth International Symposium on Motor Control, Sofia, Bulgaria*, 2004.
- [4] A. Chaudhary, *Robust Hand Gesture Recognition for Robotic Hand Control*. Springer, 2018.
- [5] A. Konar and S. Saha, "Gesture recognition," *Principles, Techniques and Applications.*, Cham: Springer International Publishing, 2018.
- [6] J. A. Hertz, *Introduction to the Theory of Neural Computation*. CRC Press, 2018.
- [7] M. J. Cheok, Z. Omar, and M. H. Jaward, "A review of hand gesture and sign language recognition techniques," *International Journal of Machine Learning and Cybernetics*, vol. 10, no. 1, pp. 131–153, 2019.
- [8] S. Tortora, M. Moro, and E. Menegatti, "Dual-myo real-time control of a humanoid arm for teleoperation," in *2019 14th ACM/IEEE International Conference on Human-Robot Interaction (HRI)*. IEEE, 2019, pp. 624–625.
- [9] S. Bisi, L. De Luca, B. Shrestha, Z. Yang, and V. Gandhi, "Development of an emg-controlled mobile robot," *Robotics*, vol. 7, no. 3, p. 36, 2018.

- [10] S. Gorzkowski and G. Sarwas, "Exploitation of emg signals for video game control," in *2019 20th International Carpathian Control Conference (ICCC)*. IEEE, 2019, pp. 1–6.
- [11] B. De la Cruz-Sánchez, M. Arias-Montiel, and E. Lugo-González, *sEMG database of the MYO bracelet for hand gestures*. Mendeley Data, 2019.
- [12] F. P. Reyes and A. C. Atencio, "Algoritmo coper para la detección de actividad de voz," *Electrónica-UNMSM*, no. 8, pp. 3–12, 2002.
- [13] R. Merletti and P. Di Torino, "Standards for reporting emg data," *J Electromyogr Kinesiol*, vol. 9, no. 1, pp. 3–4, 1999.
- [14] D. Joshi, K. Kandpal, and S. Anand, "Feature evaluation to reduce false triggering in threshold based emg prosthetic hand," in *4th Kuala Lumpur International Conference on Biomedical Engineering 2008. IFMBE Proceedings*, N. A. Abu Osman, F. Ibrahim, W. A. B. Wan Abas, H. S. Abdul Rahman, and H.-N. Ting, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 2008, pp. 769–772.
- [15] A. Phinyomark, C. Limsakul, and P. Phukpattaranont, "A novel feature extraction for robust emg pattern recognition," *Journal of Computing*, vol. 1, no. 1, pp. 71–80, 2009.
- [16] Y. Wang, J. Xiang, R. Markert, and M. Liang, "Spectral kurtosis for fault detection, diagnosis and prognostics of rotating machines: A review with applications," *Mechanical Systems and Signal Processing*, vol. 66, pp. 679–698, 2016.
- [17] H.-P. Huang and C.-Y. Chen, "Development of a myoelectric discrimination system for a multi-degree prosthetic hand," in *Robotics and Automation, 1999. Proceedings. of International Conference on Robotics and Automation*, vol. 3. IEEE, 1999, pp. 2392–2397.
- [18] J. Novaković, "Toward optimal feature selection using ranking methods and classification algorithms," *Yugoslav Journal of Operations Research*, vol. 21, no. 1, 2016.
- [19] I. Guyon and A. Elisseeff, "An introduction to variable and feature selection," *Journal of Machine Learning Research*, vol. 3, no. Mar, pp. 1157–1182, 2003.
- [20] N. Gu, M. Fan, L. Du, and D. Ren, "Efficient sequential feature selection based on adaptive eigenspace model," *Neurocomputing*, vol. 161, pp. 199–209, 2015.
- [21] J. Pohjalainen, O. Räsänen, and S. Kadioglu, "Feature selection methods and their combinations in high-dimensional classification of speaker likability, intelligibility and personality traits," *Computer Speech & Language*, vol. 29, no. 1, pp. 145–171, 2015.
- [22] H. Peng, F. Long, and C. Ding, "Feature selection based on mutual information criteria of max-dependency, max-relevance, and min-redundancy," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 27, no. 8, pp. 1226–1238, 2005.
- [23] P. Pudil, J. Novovičová, and J. Kittler, "Floating search methods in feature selection," *Pattern Recognition Letters*, vol. 15, no. 11, pp. 1119–1125, 1994.
- [24] S. Zhang, D. Cheng, Z. Deng, M. Zong, and X. Deng, "A novel knn algorithm with data-driven k parameter computation," *Pattern Recognition Letters*, vol. 109, pp. 44–54, 2018.
- [25] C. M. Bishop, *Pattern Recognition and Machine Learning*. springer, 2006.
- [26] B. De la Cruz-Sánchez, M. Arias-Montiel, and E. Lugo-González, "Development of hand exoskeleton prototype for assisted rehabilitation," in *IFTToMM Symposium on Mechanism Design for Robotics*. Springer, 2019, pp. 378–385.
- [27] B. De La Cruz-Sánchez, M. Arias-Montiel, and E. Lugo-González, "Diseño y construcción de un prototipo de exoesqueleto para rehabilitación de mano," in *Memorias del Congreso Nacional de Ingeniería Biomédica*, vol. 5, no. 1, 2018, pp. 398–401.



# Bibliografía

- [1] AER. <http://www.aeratp.com/aer-atp/robotica-industrial-y-de-servicio/>. Consultado [2/01/2018].
- [2] Comité de expertos de la OMS en rehabilitación medica. *Organización Mundial de la Salud Serie de Informes Tecnicos*. Organización Mundial de la Salud, 1969.
- [3] I. Devesa Gutiérrez, M. E. Mazadiego González, M. A. Baldomero Hernández Hernández, and H. A. Mancera Cruz. Rehabilitación del paciente con enfermedad vascular cerebral (evc). *Revista Mexicana de Medicina Física y Rehabilitación*, 26(3-4):94–108, 2014.
- [4] M. López. <http://www.conacytprensa.mx/index.php/tecnologia/tic/4556-reportaje-desarrollos-tecnologicos-hechos-en-mexico-para-personas-con-discapacidad>. CONACYT, December 2015. Consultado [25/12/2017].
- [5] PROMEXICO. [www.gob.mx/promexico/acciones-y-programas/dispositivos-medicos-26794](http://www.gob.mx/promexico/acciones-y-programas/dispositivos-medicos-26794). February 2016. Consultado[1/01/2018].
- [6] MMS México. <https://www.mms-mexico.com/art%C3%ADculos/la-manufactura-de-dispositivos-mdicos-en-mxico>. Consultado [30/12/2017].
- [7] 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. In *Proc. IEEE/RSJ Int. Conf. Intelligent Robots and Systems*, pages 1512–1517, November 2013.
- [8] Y. Narváez, O. A. Vivas, S. C. Enriquez, J. M. Sabater Navarro, N. García, and A. Martínez. Diseño de un dispositivo neumático para rehabilitación de mano mediante funda termoretráctil. *Actas de las XXXV Jornadas de Automática*, pages 424–429, sep 2014.
- [9] L. Connelly, Y. Jia, M. L. Toro, M. E. Stoykov, R. V. Kenyon, and D. G. Kamper. A pneumatic glove and immersive virtual reality environment for hand rehabilitative training after stroke. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 18(5):551–559, October 2010.
- [10] M. DiCicco, L. Lucas, and Y. Matsuoka. Comparison of control strategies for an EMG controlled orthotic exoskeleton for the hand. In *Proc. IEEE Int. Conf. Robotics and Automation ICRA '04*, volume 2, pages 1622–1627, April 2004.

- [11] O. Unluhisarcikli, B. Weinberg, M. Sivak, A. Mirelman, P. Bonato, and C. Mavroidis. A robotic hand rehabilitation system with interactive gaming using novel electro-rheological fluid based actuators. In *Proc. IEEE Int Robotics and Automation (ICRA) Conf*, pages 1846–1851, May 2010.
- [12] M. A. Delph, S. A. Fischer, P. W. Gauthier, C. H. M. Luna, E. A. Clancy, and G. S. Fischer. A soft robotic exomusculature glove with integrated semg sensing for hand rehabilitation. In *Proc. IEEE Int Rehabilitation Robotics (ICORR) Conf*, pages 1–7, June 2013.
- [13] Vitia. <http://www.vitia.es/>. 2015. Consultado [15/10/16].
- [14] JASE Systems. <http://www.jacesystems.com/>. Consultado [15/11/2016].
- [15] Festo. <https://www.festo.com/group/en/cms/10233.htm>. 2012. Consultado [21/10/16].
- [16] R. C. V. Loureiro and W. S. Harwin. Reach & grasp therapy: Design and control of a 9-dof robotic neuro-rehabilitation system. In *Proc. IEEE 10th Int. Conf. Rehabilitation Robotics*, pages 757–763, June 2007.
- [17] N. Irastorza Landa, A. Sarasola-Sanz, E. López-Larraz, C. Bibián, F. Shiman, N. Birbaumer, and A. Ramos Murguialday. Design of continuous emg classification approaches towards the control of a robotic exoskeleton in reaching movements. In *Proc. Rehabilitation Robotics (ICORR), 2017 International Conference on*, pages 128– 133. IEEE, July 2017.
- [18] F. Zhang, L. Hua, Y. Fu, H. Chen, and S. Wang. Design and development of a hand exoskeleton for rehabilitation of hand injuries. *Mechanism and Machine Theory*, 73:103–116, 2014.
- [19] Y. Fu, Q. Zhang, F. Zhang, and Z. Gan. Design and development of a hand rehabilitation robot for patient-cooperative therapy following stroke. In *Proc. Mechatronics and Automation (ICMA), 2011 International Conference on*, pages 112–117. IEEE, 2011.
- [20] A. Chiri, N. Vitiello, F. Giovacchini, S. Roccella, F. Vecchi, and M. C. Carrozza. Mechatronic design and characterization of the index finger module of a hand exoskeleton for post-stroke rehabilitation. *IEEE/ASME Transactions on Mechatronics*, 17(5):884–894, October 2012.
- [21] P. Agarwal and A. D. Deshpande. Subject-specific assist-as-needed controllers for a hand exoskeleton for rehabilitation. *IEEE Robotics and Automation Letters*, 3(1):508–515, 2018.
- [22] M. Fontana, F. Salsedo, S. Marcheschi, and M. Bergamasco. Haptic hand exoskeleton for precision grasp simulation. *Journal of Mechanisms and Robotics*, 5(4):041014, 2013.

- [23] N. Ho, K. Y. Tong, X. L. Hu, K. L. Fung, X. J. Wei, W. Rong, and E. A. Susanto. An emg-driven exoskeleton hand robotic training device on chronic stroke subjects: task training system for stroke rehabilitation. In *Proc. Rehabilitation Robotics (ICORR), 2011 IEEE International Conference on*, pages 1–5. IEEE, 2011.
- [24] R. Conti, E. Meli, and A. Ridolfi. A novel kinematic architecture for portable hand exoskeletons. *Mechatronics*, 35:192–207, 2016.
- [25] O. Sandoval Gonzalez, J. Jacinto Villegas, I. Herrera Aguilar, O. Portillo Rodriguez, P. Tripicchio, M. Hernandez Ramos, A. Flores Cuautle, and C. Avizzano. Design and development of a hand exoskeleton robot for active and passive rehabilitation. *International Journal of Advanced Robotic Systems*, 13(66):1–12, 2016.
- [26] B. A. de la Cruz Sánchez. *Diseño y construcción de un prototipo de exoesqueleto para rehabilitación de mano en accidentes cerebrovasculares*. Tesis de licenciatura, Universidad Tecnológica de la Mixteca, December 2016.
- [27] B. A. de la Cruz Sánchez, M. Arias Montiel, and E. Lugo González. Diseño de un prototipo de exoesqueleto para rehabilitación de mano. *Pistas Educativas*, 39(125):96–111, 2017.
- [28] S. Khan, K. Khurshid, and M. Zceshan. Emg data acquisition and flight control of quadcopter on different emg signals. In *Proc. 2019 14th Iberian Conference on Information Systems and Technologies (CISTI)*, pages 1–5. IEEE, 2019.
- [29] A. Krasoulis, S. Vijayakumar, and K. Nazarpour. Multi-grip classification-based prosthesis control with two emg-imu sensor. *IEEE Transactions on Neural Systems & Rehabilitation Engineering*, 2020.
- [30] Universidad Johns Hopkins. <http://www.jhuapl.edu/>. Consultado [12/01/2020].
- [31] itie Knowledge Solutions. <http://itie.in/>. Consultado [9/01/2020].
- [32] Biometrics Ltd. <http://www.biometricsltd.com/>. Consultado [20/12/2019].
- [33] DELSYS. <https://www.delsys.com/>. Consultado [26/12/2019].
- [34] N. M. Caro Sánchez. Gesture classification based on electromyography. Master’s thesis, Czech Technical University in Prague. Czech Republic, 2016.
- [35] Instituto Nacional de Estadística y Geografía. <http://www.inegi.org.mx>. 2016. Consultado [26/11/2016].
- [36] J. M. Sabater and J. Martínez Verdu. *Guía docente para el desarrollo de robots de servicio*. Primera edition, 2012.
- [37] R. C. Marrero Miralles and M. Cunillera Puig. *Biomecánica clínica del aparato locomotor*, chapter Mano, pages 140–170. Masson, 2000.

- [38] M. H. Schieber and M. Santello. Hand function: peripheral and central constraints on performance. *Journal of Applied Physiology*, 96(6):2293–2300, 2004.
- [39] C. I. Castro. *Lesiones de la mano y la muñeca*. Editorial Paidotribo, S.L., 2004.
- [40] S. Brorsson. Biomechanical studies on hand function in rehabilitation. In Tarun Goswami, editor, *Human Musculoskeletal Biomechanics*, pages 87–106. InTech, 2011.
- [41] M. Prives, N. Linsekov, and N. Bushkovich. *Anatomía humana I*, chapter Aparato de sostén y locomoción, pages 242–243. Quinta edition, 1984.
- [42] C. L. Taylor and R. J. Schwarz. Artificial limbs: A review of current developments. *Artificial limbs*, 2(2):46–62, 1955.
- [43] M. Nordin, V. H. Frankel, and K. Forssen. *Biomecánica básica del sistema musculoesquelético*. McGraw-Hill Interamericana de España, tercera edition, 2004.
- [44] INFOGEN. <http://infogen.org.mx/distrofia-muscular/>. Consultado [28/11/17].
- [45] F. Aguilar Rebolledo. Plasticidad cerebral. parte 1. *Revista Médica del Instituto Mexicano del Seguro Social*, 41(1):55–64, 2003.
- [46] A. Moyano. El accidente cerebrovascular desde la mirada del rehabilitador. *Servicio de Medicina Física y Rehabilitación del Hospital Clínico de la Universidad de Chile*, 21:348–55, 2010.
- [47] A. Arias Cuadrado. Rehabilitación del acv: evaluación, pronóstico y tratamiento. *Galicia Clínica*, 70(3):25–40, 2009.
- [48] M. D. Rene Cailliet. *Síndromes dolorosos: Mano*. El manual moderno, 4ta edition, 1994.
- [49] A. Basteris, S. M. Nijenhuis, A. H. A. Stienen, J. H. Buurke, G. B. Prange, and F. Amirabdollahian. Training modalities in robot-mediated upper limb rehabilitation in stroke: a framework for classification based on a systematic review. *Journal of neuroengineering and rehabilitation*, 11(1):111, 2014.
- [50] Z. Yue, X. Zhang, and J. Wang. Hand rehabilitation robotics on poststroke motor recovery. *Behavioural neurology*, 2017, 2017.
- [51] D. Leonardis, M. Barsotti, C. Loconsole, M. Solazzi, M. Troncossi, C. Mazzotti, V. P. Castelli, C. Procopio, G. Lamola, C. Chisari, et al. An emg-controlled robotic hand exoskeleton for bilateral rehabilitation. *IEEE transactions on haptics*, 8(2):140–151, 2015.
- [52] K. C. Stewart, H. J. Cauraugh, and J. J. Summers. Bilateral movement training and stroke rehabilitation: a systematic review and meta-analysis. *Journal of the neurological sciences*, 244(1):89–95, 2006.
- [53] P. G. Wittkopf and M. I. Johnson. Mirror therapy: A potential intervention for pain management. *Revista de Asociacion Medica Brasileña*, 63(11):1000–1005, 2017.

- [54] J. J. Summers, M. I. Garry, F. A. Kagerer, C. Y. Hiraga, and A. Loftus. Bilateral training and recovery of upper arm function after stroke. In *Proc. Xth International Symposium on Motor Control, Sofia, Bulgaria*, 2004.
- [55] National Instruments. <http://www.ni.com/data-acquisition/what-is/esa/>. 2017. Consultado[26/11/17].
- [56] S. W. Smith. *The scientist and engineer's guide to digital signal processing*. 1997.
- [57] L. Tan and J. Jiang. *Digital signal processing: fundamentals and applications*. Academic Press, 2018.
- [58] J. V. Barchiesi. *Introducción al procesamiento digital de señales*. Eds. Universitarias de Valparaíso, 2008.
- [59] J. O. Smith. *Introduction to digital filters: with audio applications*, volume 2. Julius Smith, 2007.
- [60] M. Mirghani. Implementation of matched filters using microcontrollers. In *Proc. 2016 Conference of Basic Sciences and Engineering Studies (SGCAC)*, pages 128–132. IEEE, 2016.
- [61] Carlos Del Águila. *Electromedicina*. Nueva Librería, 2da edition, 1994.
- [62] J. I. Ibarra Lúzar. Técnicas instrumentales de diagnóstico y evaluación en rehabilitación. *Rehabilitación*, 39(6):monográfico–monográfico, 2005.
- [63] R. Khandpur. *Biomedical instrumentation: Technology and applications*. McGraw-Hill Prof Med/Tech, 2004.
- [64] R. Merletti and P. A. Parker. *Electromyography: physiology, engineering, and non-invasive applications*, volume 11. John Wiley & Sons, 2004.
- [65] N. J. Nilsson. Introduction to machine learning: An early draft of a proposed textbook. pages 175-188. <http://robotics.stanford.edu/people/nilsson/mlbook.html>, 1996.
- [66] T. M. Mitchell. *Machine Learning*. McGraw-Hill, Inc., New York, NY, USA, 1 edition, 1997.
- [67] R. Herbrich. *Learning Kernel Classifiers: Theory and Algorithms*. MIT Press, Cambridge, MA, USA, 2001.
- [68] C. M. Bishop. *Pattern Recognition and Machine Learning (Information Science and Statistics)*. Springer-Verlag, Berlin, Heidelberg, 2006.
- [69] T. J. Ross. *Fuzzy logic with engineering applications*. John Wiley & Sons, 2005.
- [70] S. M. Kuo and D. Morgan. *Active noise control systems: algorithms and DSP implementations*. John Wiley & Sons, Inc., 1995.

- [71] K. Ogata. *Ingeniería de control moderna*. Pearson Educación, 2003.
- [72] C. W. De Silva. *Intelligent control: fuzzy logic applications*. CRC press, 2018.
- [73] K. M. Passino, S. Yurkovich, and M. Reinfrank. *Fuzzy control*, volume 20. Citeseer, 1998.
- [74] R. A. Maguiña Pérez. Sistemas de inferencia basados en lógica borrosa: Fundamentos y caso de estudio. *Revista de investigación de Sistemas e Informática*, 7(1):91–104, 2010.
- [75] D. J. Calvo Cerezo. *Desarrollo, implementación y evaluación de un algoritmo de lógica difusa para el control de actitud de nanosatélites*. PhD thesis, Universidad Politécnica de Madrid, 2017.
- [76] A. M. Wing and C. Fraser. The contribution of the thumb to reaching movements. *The Quarterly Journal of Experimental Psychology Section A*, 35(2):297–309, 1983.
- [77] Y. Y. Huang and K. H. Low. Initial analysis and design of an assistive rehabilitation hand device with free loading and fingers motion visible to subjects. In *Proc. 2008 IEEE International Conference on Systems, Man and Cybernetics*, pages 2584–2590. IEEE, 2008.
- [78] Motion capture technologyâbenefits and challenges. *International Journal of Innovative Research in Technology Science(IJIRTS)*.
- [79] J. Gülke, N. J Wachter, T. Geyer, H. Schöll, G. Apic, and M. Mentzel. Motion coordination patterns during cylinder grip analyzed with a sensor glove. *The Journal of Hand Surgery*, 35(5):797–806, 2010.
- [80] J. W. Garrett. The adult human hand: some anthropometric and biomechanical considerations. *Human Factors*, 13(2):117–131, 1971.
- [81] J. D. Lalinde, J. F. Castro, M. Elena Tarazona, J. E. Rodriguez, J. G. Rangel, C. A. Tolosa Sierra, M. K. . Torrado, S. M. Sierra, and V. J. Pirela. Sobre el uso adecuado del coeficiente de correlación de pearson: definición, propiedades y suposiciones. *Archivos Venezolanos de Farmacología y Terapéutica*, 37(5):587–595, 2018.
- [82] Pololu. [www.pololu.com](http://www.pololu.com). 2017. Consultado [20/08/2017].
- [83] Pololu Corporation. [www.pololu.com/product/713](http://www.pololu.com/product/713). Consultado [10/03/2018].
- [84] Pololu Corporation. [www.pololu.com/product/2598](http://www.pololu.com/product/2598). Consultado [10/03/2018].
- [85] MYO. [www.myo.com](http://www.myo.com). 2017. Consultado [18/02/2018].
- [86] B. A. De la Cruz-Sánchez, M Arias-Montiel, and E Lugo-González. *sEMG database of the MYO bracelet for hand gestures*. Mendeley Data, 2019.
- [87] F. P. Reyes and A. C. Atencio. Algoritmo coper para la detección de actividad de voz. *Electrónica-UNMSM*, (8):3–12, 2002.

- [88] D. Joshi, K. Kandpal, and S. Anand. Feature evaluation to reduce false triggering in threshold based emg prosthetic hand. In Noor Azuan Abu Osman, Fatimah Ibrahim, Wan Abu Bakar Wan Abas, Herman Shah Abdul Rahman, and Hua-Nong Ting, editors, *4th Kuala Lumpur International Conference on Biomedical Engineering 2008*, pages 769–772, Berlin, Heidelberg, 2008. Springer Berlin Heidelberg.
- [89] A. Phinyomark, C. Limsakul, and P. Phukpattaranont. A novel feature extraction for robust emg pattern recognition. *arXiv preprint arXiv:0912.3973*, 2009.
- [90] Y. Wang, J. Xiang, R. Markert, and M. Liang. Spectral kurtosis for fault detection, diagnosis and prognostics of rotating machines: A review with applications. *Mechanical Systems and Signal Processing*, 66:679–698, 2016.
- [91] H. P. Huang and C. Y. Chen. Development of a myoelectric discrimination system for a multi-degree prosthetic hand. In *Proc. Robotics and Automation, 1999. Proceedings. 1999 IEEE International Conference on*, volume 3, pages 2392–2397. IEEE, 1999.
- [92] J. Novaković. Toward optimal feature selection using ranking methods and classification algorithms. *Yugoslav Journal of Operations Research*, 21(1), 2016.
- [93] I. Guyon and A. Elisseeff. An introduction to variable and feature selection. *Journal of Machine Learning Research*, 3(Mar):1157–1182, 2003.
- [94] J. Pohjalainen, O. Räsänen, and S. Kadioglu. Feature selection methods and their combinations in high-dimensional classification of speaker likability, intelligibility and personality traits. *Computer Speech & Language*, 29(1):145–171, 2015.
- [95] H. Peng, F. Long, and C. Ding. Feature selection based on mutual information criteria of max-dependency, max-relevance, and min-redundancy. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 27(8):1226–1238, 2005.
- [96] P. Pudil, J. Novovičová, and J. Kittler. Floating search methods in feature selection. *Pattern recognition letters*, 15(11):1119–1125, 1994.
- [97] GitHub. <https://github.com/skadio/featureSelection>. 2015. Consultado [10/01/2019].
- [98] B. Wang, J. Gao, and Y. Qi. A theoretical framework for robustness of (deep) classifiers against adversarial examples. *arXiv preprint arXiv:1612.00334*, 2016.
- [99] Y. R. Hong and E. Moon. Reliability and validity of free software for the analysis of locomotor activity in mice. *Yeungnam University Journal of Medicine*, 35(1):63–69, 2018.
- [100] K. R. Simba, N. Uchiyama, and S. Sano. Real-time smooth trajectory generation for nonholonomic mobile robots using bézier curves. *Robotics and Computer-Integrated Manufacturing*, 41:31–42, 2016.

- [101] C. Wang, Y. Li, S. S. Ge, and T. H. Lee. Reference adaptation for robots in physical interactions with unknown environments. *IEEE Transactions on Cybernetics*, 47(11):3504–3515, 2017.
- [102] F. Beltran-Carbajal, A. Valderrabano-Gonzalez, A. R. Favela-Contreras, and J. C. Rosas-Caro. Active disturbance rejection control of a magnetic suspension system. *Asian Journal of Control*, 17(3):842–854, 2015.
- [103] Z. M. Ju and W. L Hong. Inserting control points for bezier curve approximation. In *Proceedings of the 31st Chinese Control Conference*, pages 2416–2420. IEEE, 2012.
- [104] Z. Xu, S. Wei, N. Wang, and X. Zhang. Trajectory planning with bezier curve in cartesian space for industrial gluing robot. In *Proc. International Conference on Intelligent Robotics and Applications*, pages 146–154. Springer, 2014.
- [105] H. Akima. A new method of interpolation and smooth curve fitting based on local procedures. *Journal of the ACM (JACM)*, 17(4):589–602, 1970.
- [106] Y. S. Wang, N. Matni, and J. C. Doyle. Localized lqr control with actuator regularization. In *Proc. American Control Conference (ACC), 2016*, pages 5205–5212. IEEE, 2016.
- [107] J. J. E Slotine and W. Li. *Applied nonlinear control*, volume 199. Prentice hall Englewood Cliffs, NJ, 1991.
- [108] A. D. Shakibjoo and M. D. Shakibjoo. 2-dof pid with reset controller for 4-dof robot arm manipulator. In *Proc. Advanced Robotics and Intelligent Systems (ARIS), 2015 International Conference on*, pages 1–6. IEEE, 2015.
- [109] H. K. Khalil and J. W. Grizzle. *Nonlinear systems*, volume 3. Prentice hall Upper Saddle River, NJ, 2002.
- [110] K. Komatsu and H. Takata. Nonlinear feedback control of stabilization problem via formal linearization using taylor expansion. In *Proc. Information Theory and Its Applications, 2008. ISITA 2008. International Symposium on*, pages 1–5. IEEE, 2008.
- [111] G. A. Salman, A. S. Jafar, and A. I. Ismael. Application of artificial intelligence techniques for lfc and avr systems using pid controller. *Int J Pow Elec & Dri Syst ISSN*, 10(3):1694–1704, 2019.
- [112] E. D. Sontag. *Mathematical control theory: deterministic finite dimensional systems*, volume 6. Springer Science & Business Media, 2013.
- [113] M. Hajiyan, S. Biglarbegan, M. and Mahmud, and H. A. Abdullah. Design and control of an exoskeleton based device for hand rehabilitation. In *Proc. 2017 IEEE International Symposium on Robotics and Intelligent Sensors (IRIS)*, pages 91–97. IEEE, 2017.

- [114] A. Abane, M. Guiatni, M. A. Alouane, I. Benyahia, M. Tair, and N. Ababou. Auto-tuning fuzzy force/position control of a 5 dof exoskeleton for upper limb rehabilitation. In *Proc. 2017 IEEE International Conference on Advanced Intelligent Mechatronics (AIM)*, pages 1731–1736. IEEE, 2017.
- [115] J. Tang, J. Zheng, and Y. Wang. Direct force control of upper-limb exoskeleton based on fuzzy adaptive algorithm. *Journal of Vibroengineering*, 20(1):636–650, 2018.
- [116] S. K. Ali and M. O. Tokhi. Control design of a de-weighting upper-limb exoskeleton: extended-based fuzzy. *Indonesian Journal of Electrical Engineering and Informatics (IJEI)*, 7(1):1–14, 2019.
- [117] Jan Jantzen. *Foundations of fuzzy control: a practical approach*. John Wiley & Sons, 2013.
- [118] SparkFun Electronics. <https://github.com/sparkfun/SparkFunTB6612FNGArduinoLibrary>. 2016. Consultado [21/02/19].
- [119] A. J. Alves. <https://github.com/zerokol/eFLL>. 2019. Consultado [01/06/19].
- [120] B. A. Myers. User interface software tools. *ACM Transactions on Computer-Human Interaction (TOCHI)*, 2(1):64–103, 1995.
- [121] B. A. De la Cruz-Sánchez, M Arias-Montiel, and E Lugo-González. *MOCAP data base for three hand movements*. Mendeley Data, 2020.