



FACULTAD DE INGENIERÍA
ESCUELA DE INGENIERÍA CIVIL MECATRÓNICA

DISEÑO MECÁNICO Y GENERACIÓN DE
TRAYECTORIA PARA UN ROBOT
ANTROPOMÓRFICO DE 6 GRADOS DE LIBERTAD

**Memoria para optar al título de
Ingeniero Civil Mecatrónico**

Profesor Guía:

Doc. Ing. Roberto Ramírez Alegría

CARLOS DAMIÁN DÍAZ BRAVO

CURICÓ – CHILE

2022

CONSTANCIA

La Dirección del Sistema de Bibliotecas a través de su encargado Biblioteca Campus Curicó certifica que el autor del siguiente trabajo de titulación ha firmado su autorización para la reproducción en forma total o parcial e ilimitada del mismo.



UNIVERSIDAD DE TALCA
DIRECCIÓN
SISTEMA DE BIBLIOTECAS

UNIVERSIDAD DE TALCA
SISTEMA DE BIBLIOTECAS
CAMPUS CURICO

Curicó, 2023

DISEÑO MECÁNICO Y GENERACIÓN DE
TRAYECTORIA PARA UN ROBOT
ANTROPOMÓRFICO DE 6 GRADOS DE
LIBERTAD

Carlos Damián Díaz Bravo

Julio 2022

© Carlos Damián Díaz Bravo

Resumen

El tema central de esta memoria de título es el diseño mecánico de un robot antropomórfico de 6 grados de libertad. El proceso de diseño comienza por conocer las partes que componen al manipulador como: grados de libertad, eslabones, articulaciones rotacionales, etc. Al conocer lo básico se procede a hacer un cálculo general y estático de torques para luego adaptarlo al diseño mecánico creado. El modelo se crea en el software Inventor Profesional 2020, este permite crear piezas individuales del robot para luego generar un ensamblaje final.

Luego se realiza una caracterización matemática que permite calcular las cinemáticas y dinámicas asociadas al robot. La cinemática directa permite estudiar la posición y orientación del manipulador respecto a un sistema base, esto se puede corroborar al medir las distancias desde el sistema base hasta el efector final tanto en el software de diseño como en el software de programación Matlab R2020b. La cinemática inversa encuentra los valores que permiten posicionar y orientar el manipulador según una determinada localización espacial. Las dinámicas del robot no se utilizarán en esta memoria para realizar algún tipo de trayectoria o control.

Finalmente, se presenta un tutorial para establecer la comunicación entre Matlab e Inventor. Además, se presentan las simulaciones y resultados obtenidos al realizar un generador de trayectoria para un robot antropomórfico de 6 grados de libertad.

...dedicado a mis padres...

Agradecimientos

En esta parte de mi memoria de título quiero mencionar y agradecer a todas las personas que han sido importantes en todo este proceso universitario, desde principio a fin. En primer lugar, me quiero auto felicitar por no rendirme cuando todo se veía más complicado y el objetivo final se alejaba, hoy estoy finalizando uno de los momentos más importantes de mi vida y se genera un nuevo comienzo lleno de nuevos objetivos. Durante mi proceso universitario quiero destacar a:

- A mis padres, Claudia Bravo y Juan Arias, por siempre apoyarme en mis estudios, por inculcarme buenos valores, por darme la posibilidad de educarme y ser un profesional. Siempre me dieron su confianza con la convicción de que podía lograr el objetivo de ser el primer profesional de mi familia.
- A mis amigos Felipe Rojas, Matías Peña, Hernán Peña, Fernando Urra, Alex Martínez, Bastián Álvarez, Dubán Padilla, Guillermo Cáceres les agradezco por cada buen momento vivido en mi vida universitaria, por ayudarme y apoyarme cuando más lo necesitaba, siempre me brindaron una mano, me dieron su cariño y siempre me aconsejaron de la mejor manera posible.
- A mis tíos Freddy Arias y Alejandra Arellano, por acompañarme durante mi proceso universitario, me cobijaron y apoyaron siempre que faltaba algo.
- A Rafael Villena por permitirme ser parte de la fundación de estudiantes de Curicó y posteriormente en Corporación Progressio, también agradecer por su preocupación constante, amistad y hermandad.
- A mi profesor guía Roberto Ramírez, por permitirme ser su memorista, por plantearme un tema que yo quería trabajar y estudiar, por la confianza y por sus buenos consejos respecto al trabajo realizado.
- Al señor Tomás Esparza quién fue mi profesor de Matemática cuando estaba en la educación media, él siempre me dijo que yo podía ser un profesional, me ayudó a elegir mi carrera, me aconsejó siempre de buena fe y me brindó su cariño como un padre lo hace con su hijo, sin compartir la misma sangre.
- A la fundación de estudiantes universitarios, por recibirme con los brazos abiertos cuando la economía no acompañó durante el proceso universitario, por enseñarme valores

diferentes y habilidades que se aprenden al vivir con 18 personas más en la misma casa y compartir pieza con 2 personas. La fundación me enseñó lo que es la hermandad, la amistad y el no sólo pensar en mí, sino que en cómo mis acciones pueden afectar a los demás.

- Al servicio de intercambio de la Universidad de Talca y al DAAD – Servicio alemán de intercambio académico, por permitirme vivir una de las mejores experiencias en mi vida, poder realizar un intercambio académico y conocer lugares que solo veía en la televisión.
- A la corporación Progressio que me cobijó durante mis últimos 3 años de estudios universitarios, pudiendo compartir casa con nuevas personas en un lugar cómodo y hogareño, pudiendo absorber más habilidades que me ayudarán en el futuro para desempeñarme como profesional.
- A María Jesús Ponce por ayudarme en mi proceso de tesis, a brindarme su apoyo cuando sentía que todo iba a salir mal, por escucharme cuando me quejaba que algo no me funcionaba, gracias por llegar a mi vida y gracias por estar siempre cuando te necesito.

TABLA DE CONTENIDOS

RESUMEN	III
AGRADECIMIENTOS.....	V
TABLA DE CONTENIDOS	V
LISTA DE TABLAS.....	IX
LISTA DE FIGURAS.....	X
ABREVIACIONES	X
CAPÍTULO 1. INTRODUCCIÓN	1
1.1. INTRODUCCIÓN GENERAL.....	1
1.2. ESTADO DEL ARTE.....	1
1.2.1. INTRODUCCIÓN A LA ROBÓTICA.....	1
1.2.2. CLASIFICACIÓN DE LOS ROBOTS.....	2
1.2.3. ROBOTS INDUSTRIALES:	3
1.2.4. APLICACIONES DE ROBOTS INDUSTRIALES	7
1.2.5. MORFOLOGÍA DE LOS ROBOTS INDUSTRIALES	7
1.2.6. ELEMENTO TERMINAL	9
1.2.7. GRADOS DE LIBERTAD	11
1.2.8. ROBOT ANTROPOMÓRFICO	11
1.2.9. ACTUADORES.....	16
1.2.10. TIPOS DE TRAYECTORIAS.....	19
1.2.11. GENERACIÓN DE TRAYECTORIA	21
1.2.12. INTERPOLACIÓN DE TRAYECTORIAS	21
1.2.13. DISCUSIÓN	24
1.3. DESCRIPCIÓN DE LA PROBLEMÁTICA	25
1.4. OBJETIVOS	26
1.4.1. OBJETIVO GENERAL	26
1.4.2. OBJETIVOS ESPECÍFICOS.....	26
1.5. ALCANCES	26
1.6. LIMITACIONES	26
1.7. METODOLOGÍA.....	27
1.7.1. ESTUDIO TEÓRICO DE LOS FUNDAMENTOS DE LA ROBÓTICA Y MECANISMOS	27
1.7.2. DISEÑO MECÁNICO Y SIMULACIÓN DE UN ROBOT ANTROPOMÓRFICO DE 6 GRADOS DE LIBERTAD.....	27
1.7.3. ESTUDIO TEÓRICO DE LA CINEMÁTICA Y DINÁMICA DEL MANIPULADOR	27
1.7.4. SIMULACIÓN DE LA CINEMÁTICA Y DINÁMICA DEL MANIPULADOR.....	28
1.7.5. SIMULACIÓN DEL MANIPULADOR EN SIMMECHANICS – MULTIBODY	28

1.7.6. ESTUDIO TEÓRICO PARA LA GENERACIÓN DE TRAYECTORIAS	28
1.7.7. SIMULACIÓN PARA LA GENERACIÓN DE TRAYECTORIAS EN UN ROBOT ANTROPOMÓRFICO DE 6 GDL	28
1.8. RESUMEN CAPÍTULO 1	29
CAPÍTULO 2. CARACTERIZACIÓN MATEMÁTICA DE UN ROBOT ANTROPOMÓRFICO DE 6 GRADOS DE LIBERTAD.....	30
2.1. INTRODUCCIÓN	30
2.2. CINEMÁTICA DIRECTA	30
2.2.1. ALGORITMO DE DENAVIT-HARTENBERG PARA LA OBTENCIÓN DEL MODELO CINEMÁTICO DIRECTO	31
2.2.2. ANÁLISIS DE CINEMÁTICA DIRECTA DE UN ROBOT ANTROPOMÓRFICO DE 6 GDL ..	32
2.3. CINEMÁTICA INVERSA	38
2.3.1. ANÁLISIS DE CINEMÁTICA INVERSA DE UN ROBOT ANTROPOMÓRFICO DE 6 GDL ..	38
2.4. DINÁMICA	48
2.4.1. ANÁLISIS DE LA DINÁMICA DE UN ROBOT ANTROPOMÓRFICO DE 6 GDL.....	48
2.4.2. ANÁLISIS DE LA DINÁMICA DE LOS ACTUADORES DE UN MANIPULADOR.....	50
2.5. RESUMEN CAPÍTULO 2	52
CAPÍTULO 3. DISEÑO DE UN ROBOT ANTROPOMÓRFICO DE 6 GRADOS DE LIBERTAD	53
3.1. INTRODUCCIÓN	53
3.2. CÁLCULO DE TORQUES	53
3.3. TRANSMISIONES DEL ROBOT.....	60
3.4. DISEÑO DE LA BASE.....	61
3.5. DISEÑO DE LA CADERA	63
3.6. DISEÑO DEL BRAZO.....	65
3.7. DISEÑO DEL ANTEBRAZO	67
3.8. DISEÑO MUÑECA.....	68
3.9. DISEÑO PINZA	68
3.10. ENSAMBLAJE FINAL	69
3.11. ANÁLISIS DE ESFUERZOS	70
3.12. COSTOS PARA IMPLEMENTAR EL MANIPULADOR	73
3.13. RESUMEN CAPÍTULO 3	76
CAPÍTULO 4. CONFIGURACIONES NECESARIAS PARA GENERAR UNA TRAYECTORIA EN UN ROBOT ANTROPOMÓRFICO DE 6 GRADOS DE LIBERTAD.....	77
4.1. INTRODUCCIÓN	77
4.2. COMUNICACIÓN DEL MODELO CAD CON MATLAB MEDIANTE SIMSCAPE MULTIBODY LINK.....	77

PASO 1: REALIZAR INSTALACIÓN DE LOS PROGRAMAS A UTILIZAR	78
PASO 2: DESCARGA DEL PLUGIN/ADDON	78
PASO 3: INSTALACIÓN DEL PLUGIN/ADDON EN MATLAB Y EN EL SOFTWARE DE CAD.	80
PASO 4: CAMBIAR “COMAS POR PUNTOS”	82
PASO 5: ABRIR EL DISEÑO MECÁNICO EN MATLAB.	82
4.3. LIBRERÍA DE PETER CORKE.....	83
4.4. INTEGRACIÓN DEL MODELO CAD DESDE INVENTOR HACIA MATLAB	84
4.4.1. CINEMÁTICA DIRECTA MEDIANTE LA LIBRERÍA DE PETER CORKE	89
4.4.2. CINEMÁTICA INVERSA MEDIANTE LA LIBRERÍA DE PETER CORKE	90
4.5. DINÁMICA	90
4.6. RESUMEN CAPÍTULO 4	91
CAPÍTULO 5. RESULTADOS SIMULACIONES	92
5.1. INTRODUCCIÓN	92
5.2. GENERACIÓN DE TRAYECTORIA.....	92
5.3. TORQUES	105
5.4. DISCUSIÓN	107
CAPÍTULO 6. CONCLUSIONES	109
6.1. INTRODUCCIÓN	109
6.2. CONCLUSIONES	109
6.3. TRABAJOS FUTUROS	111
BIBLIOGRAFÍA	112
ANEXOS	119
ANEXO A. PLANOS MECÁNICOS	119
ANEXO B. CÓDIGOS	135

Lista de tablas

TABLA 1.1 - TIPOS DE BRAZOS MANIPULADORES O INDUSTRIALES.	3
TABLA 2.1 – PARÁMETROS DE DENAVIT-HARTENBERG PARA UN MANIPULADOR DE 6 GDL	35
TABLA 3.1 – PARÁMETROS PARA LOS TORQUES.....	54
TABLA 3.2 – PARÁMETROS PARA LAS TRANSMISIONES	61
TABLA 3.3 – VALORES DE ACTUADORES Y TRANSMISIONES	73
TABLA 3.4 – COSTOS DE LAS PIEZAS DEL MANIPULADOR	73
TABLA 4.1 – CARACTERÍSTICAS SEGÚN CADA SOFTWARE CAD [40].....	78
TABLA 5.1 – VALORES INICIALES Y FINALES PARA LA SIMULACIÓN DE TRAYECTORIA	94

Lista de figuras

FIG. 1.1 – PORTADA DE LA OBRA DE TEATRO ROSSUM’S UNIVERSAL ROBOTS Y ESCENA DE UNA REPRESENTACIÓN [14].	2
FIG. 1.2 – MANIPULADOR CARTESIANO [14].	4
FIG. 1.3 – MANIPULADOR CILÍNDRICO [14].	5
FIG. 1.4 – MANIPULADOR ESFÉRICO [14].	5
FIG. 1.5 – MANIPULADOR SCARA [14].	6
FIG. 1.6 – MANIPULADOR ANTROPOMÓRFICO [14].	7
FIG. 1.7 – EJEMPLO DE DOS ESLABONES CON UNA ARTICULACIÓN [14].	8
FIG. 1.8 – ELEMENTOS DE UN ROBOT MANIPULADOR [14].	9
FIG. 1.9 – PINZA DE DOS DEDOS CON ACCIONAMIENTO NEUMÁTICO [18].	10
FIG. 1.10 – VENTOSA PARA MANIPULACIÓN POR VACÍO [18].	10
FIG. 1.11 – ANTORCHA DE SOLDADURA AL ARCO [18].	10
FIG. 1.12 – EJEMPLO DE SISTEMA CON 1 Y 2 GDL [14].	11
FIG. 1.13 – PRIMER ROBOT DE LA HISTORIA (UNIMATE) [14].	12
FIG. 1.14 – STANFORD ARM [14].	12
FIG. 1.15 – PRIMER ROBOT INDUSTRIAL DE KUKA, LLAMADO FAMULUS. [70]	13
FIG. 1.16 – BRAZO ROBÓTICO DE ACCIONAMIENTO DIRECTO [14].	13
FIG. 1.17 - ROBOT PUMA [18].	14
FIG. 1.18 - ROBOT IRB6 DE LA FIRMA SUECA ASEA [18].	15
FIG. 1.19 – MOTOR CC – 12 V.	17
FIG. 1.20 – MOTOR PASO A PASO NEMA 17.	18
FIG. 1.21 – SERVOMOTOR [45].	19
FIG. 1.22 – POSICIÓN, VELOCIDAD Y ACELERACIÓN PARA UN INTERPOLADOR LINEAL [18].	23
FIG. 2.1– CONCEPTO DE CINEMÁTICA DIRECTA E INVERSA [18].	31
FIG. 2.2 – CONFIGURACIÓN DE UN ROBOT ANTROPOMÓRFICO CON PINZA [41].	33
FIG. 2.3 – CONFIGURACIÓN DE ESLABONES Y ARTICULACIONES SEGÚN D-H.	33
FIG. 2.4 – UBICACIÓN DE EJES EN CADA ARTICULACIÓN DEL MANIPULADOR SEGÚN D-H.	34
FIG. 2.5 – CARACTERÍSTICAS PRINCIPALES DE UN MANIPULADOR DE 6 GDL SEGÚN D-H.	35
FIG. 2.6 – PROBLEMA CINEMÁTICO INVERSO PARA UN ROBOT ANTROPOMÓRFICO DE 6 GDL.	40
FIG. 2.7 –DESACOPAMIENTO CINEMÁTICO PARA UN ROBOT ANTROPOMÓRFICO DE 6 GDL.	40
FIG. 2.8 –PROYECCIÓN DEL PROBLEMA CINEMÁTICO SOBRE EL PLANO ZR.	41
FIG. 2.9 – RESULTADO DE LA PROYECCIÓN DEL PROBLEMA CINEMÁTICO SOBRE EL PLANO ZR.	41
FIG. 2.10 – TRAZO DE LÍNEA RECTA H.	42
FIG. 2.11 – TRIÁNGULO DEL CUAL SE OBTIENE EL VALOR DE θ_3 .	43
FIG. 2.12 – MOTOR DE CORRIENTE CONTINUA DE EXCITACIÓN SEPARADA.	51
FIG. 3.1 – PLANTEAMIENTO DE TORQUES QUE ACTÚAN EN EL ROBOT.	54
FIG. 3.2 – DISTANCIAS ENTRE TORQUES.	55
FIG. 3.3 – TORQUE EN EL PUNTO A.	56
FIG. 3.4 – TORQUE EN EL PUNTO B.	57

FIG. 3.5 – TORQUE EN EL PUNTO C	58
FIG. 3.6 – TORQUE EN EL PUNTO D.....	59
FIG. 3.7 – PLANTEAMIENTO PARA LAS RELACIONES DE TRANSMISIÓN.....	60
FIG. 3.8 – ENSAMBLAJE DE LA BASE.....	62
FIG. 3.9 – ENSAMBLAJE ACTUADOR A LA BASE.....	62
FIG. 3.10 – TRANSMISIÓN EN LA BASE DEL MANIPULADOR.....	63
FIG. 3.11 – ENSAMBLAJE DE CADERA VISTA PRINCIPAL.....	64
FIG. 3.12 – TRANSMISIÓN DE LA CADERA.....	64
FIG. 3.13 – ENSAMBLAJE DE CADERA VISTA FRONTAL.....	65
FIG. 3.14 – ENSAMBLAJE DEL BRAZO VISTA PRINCIPAL.....	65
FIG. 3.15 – ENSAMBLAJE DEL BRAZO CON SU TRANSMISIÓN.....	66
FIG. 3.16 – ENSAMBLAJE DE PERNOS EN EL BRAZO, PARTE A.....	66
FIG. 3.17 – ENSAMBLAJE DE PERNOS EN EL BRAZO, PARTE B.....	66
FIG. 3.18 – ENSAMBLAJE DEL ANTEBRAZO.....	67
FIG. 3.19 – ENSAMBLAJE DEL ANTEBRAZO CON TRANSMISIÓN.....	67
FIG. 3.20 – ENSAMBLAJE MUÑECA.....	68
FIG. 3.21 – ENSAMBLAJE PINZA.....	68
FIG. 3.22 – ENSAMBLAJE ROBOT ANTROPOMÓRFICO DE 6 GDL, VISTA FRONTAL.....	69
FIG. 3.23 – ENSAMBLAJE ROBOT ANTROPOMÓRFICO DE 6 GDL VISTA SUPERIOR.....	69
FIG. 3.24 – ENSAMBLAJE FINAL ROBOT ANTROPOMÓRFICO DE 6 GDL.....	70
.....	70
FIG. 3.25 – TENSIÓN DE VON MISES AL APLICAR CARGA MÁXIMA SOBRE LA PINZA.....	71
FIG. 3.26 – TENSIÓN DE VON MISES AL APLICAR CARGA MÁXIMA SOBRE LA MUÑECA.....	71
FIG. 3.26 - DESPLAZAMIENTO DEL MANIPULADOR AL APLICAR CARGA MÁXIMA SOBRE LA PINZA.....	72
FIG. 3.27 – DESPLAZAMIENTO DEL MANIPULADOR AL APLICAR CARGA MÁXIMA SOBRE LA MUÑECA.....	72
FIG. 3.29 – ESTIMACIÓN DE COSTOS DE LA CADERA EN EL SOFTWARE CURA.....	75
FIG. 3.30 – ESTIMACIÓN DE COSTOS UNA PARTE DE LA BASE EN EL SOFTWARE CURA (50% DE RELLENO).....	75
FIG. 4.1 – ESPACIO EN BLANCO PARA INTRODUCIR EL CORREO QUE TIENEN LICENCIA DE MATLAB [40].....	79
FIG. 4.2 – EJEMPLO DE CORREO INSTITUCIONAL QUE TIENE LICENCIA DE MATLAB [40].....	79
FIG. 4.3 – ARCHIVOS SELECCIONADOS PARA ESTE TUTORIAL [40].....	80
FIG. 4.4 – CARPETA DE UBICACIÓN DE ARCHIVOS.....	80
FIG. 4.5 – COMANDOS A EJECUTAR PARA LA INSTALACIÓN DEL PLUGIN.....	81
FIG. 4.6 – MENSAJE DE INSTALACIÓN COMPLETADA Y REVISIÓN DE DOCUMENTACIÓN.....	81
FIG. 4.7 – COMANDO FINAL PARA CONECTAR CON INVENTOR.....	81
FIG. 4.8 – COMPROBACIÓN DEL VÍNCULO DE COMUNICACIÓN.....	82
FIG. 4.9 – COMPLEMENTO PARA EXPORTAR EL DISEÑO CAD A MATLAB.....	83
FIG-4.10 – CURRENT FOLDER DONDE DEBEN ESTAR LOS ARCHIVOS CREADOS POR INVENTOR EN MATLAB.....	83
FIG. 4.11 – ARCHIVO PARA LA INSTALACIÓN DE LIBRERÍA DE PETER COORKE [64].....	83
FIG. 4.12 – DISEÑO MECÁNICO CREADO EN INVENTOR EXPORTADO A MATLAB.....	84
FIG. 4.13 – BLOQUES GENERADOS EN SIMSCAPE MULTIBODY EN MATLAB.....	84
FIG. 4.14 – ROBOT ANTROPOMÓRFICO DISEÑADO EN EL ENTORNO DE SIMULACIÓN.....	85
FIG. 4.15 – CONFIGURACIÓN DE LA ACTUACIÓN DE CADA ARTICULACIÓN.....	86

FIG. 4.16 – CONFIGURACIÓN PARA SENSAR LAS VARIABLES.....	86
FIG. 4.17 – CONFIGURACIÓN DEL CONVERTIDOR EN RADIANES.....	87
FIG. 4.18 – CONFIGURACIONES IMPORTANTES PARA QUE FUNCIONE LA SIMULACIÓN.....	88
FIG. 4.19 – TRANSFORMACIÓN DE GRADOS A RADIANES.....	88
FIG. 4.20 – ENTRADA DE DATOS AL MANIPULADOR.....	89
FIG. 4.21 – SIMULACIÓN CINEMÁTICA DIRECTA CON LIBRERÍA DE PETER CORKE.....	90
FIG. 5.1 – DIAGRAMA DE BLOQUES PARA UN GENERADOR DE TRAYECTORIA [18].....	93
FIG. 5.2 – MOVIMIENTO DE LA ARTICULACIÓN 1.....	95
FIG. 5.3 – MOVIMIENTO DE LA ARTICULACIÓN 2.....	95
FIG. 5.4 – MOVIMIENTO DE LA ARTICULACIÓN 3.....	96
FIG. 5.5 – MOVIMIENTO DE LA ARTICULACIÓN 4.....	96
FIG. 5.6 – MOVIMIENTO DE LA ARTICULACIÓN 5.....	97
FIG. 5.7 – TRAYECTORIA PICK AND PLACE ARTICULACIÓN 1.....	98
FIG. 5.8 – TRAYECTORIA PICK AND PLACE ARTICULACIÓN 2.....	98
FIG. 5.9 – TRAYECTORIA PICK AND PLACE ARTICULACIÓN 3.....	99
FIG. 5.10 – TRAYECTORIA PICK AND PLACE ARTICULACIÓN 4.....	99
FIG. 5.11 – TRAYECTORIA PICK AND PLACE ARTICULACIÓN 5.....	100
FIG. 5.12 – POSICIÓN INICIAL DEL MANIPULADOR.....	101
FIG. 5.13 – POSICIÓN DEL MANIPULADOR TRANSCURRIDOS 2.5 [S].....	101
FIG. 5.14 – POSICIÓN FINAL DEL MANIPULADOR.....	102
FIG. 5.15 – POSICIÓN DEL MANIPULADOR TRANSCURRIDOS 8 [S].....	102
FIG. 5.16 – POSICIÓN DEL MANIPULADOR TRANSCURRIDOS 10 [S] (REGRESO A POSICIÓN INICIAL).....	103
FIG. 5.17 – POSICIÓN DE TRAYECTORIA PICK AND PLACE ARTICULACIÓN 1 EN SIMSCAPE MULTIBODY.....	103
FIG. 5.18 – VELOCIDAD DE TRAYECTORIA PICK AND PLACE ARTICULACIÓN 1 EN SIMSCAPE MULTIBODY.....	104
FIG. 5.19 – ACELERACIÓN DE TRAYECTORIA PICK AND PLACE ARTICULACIÓN 1 EN SIMSCAPE MULTIBODY.....	104
.....	104
FIG. 5.20 – TORQUE PARA LA ARTICULACIÓN 2.....	105
FIG. 5.21 – TORQUE PARA LA ARTICULACIÓN 3.....	106
FIG. 5.22 – TORQUE PARA LA ARTICULACIÓN 4.....	106
FIG. 5.23 – TORQUE PARA LA ARTICULACIÓN 5.....	106

Abreviaciones

CC	Corriente Continua
GDL	Grados De Libertad
ABS	Acrlonitrilo Butadieno Estireno
CNC	Control Numérico por Computadora
kg	Kilogramos
g	Gramos
SCARA	Selective Compliance Assembly Robot Arm
C++	Lenguaje de programación
MatLab	Software matemático
PWM	Modulación por ancho de pulsos
CAD	Diseño asistido por computador
D-H	Denavit-Hartenberg
Simmechanics	Función de Matlab que permite realizar una interfaz gráfica para el control de robots

Capítulo 1. Introducción

1.1. Introducción general

Generalmente, las personas que ingresan al mundo de la robótica lo primero que investigan es que es la robótica y sus características principales. Según la DRAE, la robótica es la técnica que aplica la informática al diseño y empleo de aparatos que, en sustitución de personas, realizan operaciones o trabajos, por lo general en instalaciones industriales [12]. Otra definición muy genérica que nos puede servir para englobar ambas vertientes de la robótica es la siguiente: un robot es una máquina que puede sentir, pensar y actuar para conseguir un objetivo predefinido [15]. A pesar de obtener conceptos totalmente distintos, ambos son correctos. Además, existe una gran variedad de conceptos relacionados al área de la robótica, comprendiendo que su estudio es amplio y puede abarcar diferentes aspectos de la ciencia.

La robótica industrial constituye hoy, una de las más importantes áreas de investigación y desarrollo tecnológico [1]. Esto se profundiza en [2], donde se explica la naturaleza multidisciplinaria de la robótica que permite involucrar una gran cantidad de áreas de conocimiento tales como matemáticas, física, electrónica, computación, visión e inteligencia artificial. En particular, la robótica ha ingresado en diferentes áreas como lo son: la industria, la educación, la medicina y el área científica [5].

Esta memoria de título se centrará en obtener una validación del funcionamiento general de un robot antropomórfico de 6 GDL, creando un diseño mecánico basado en un robot industrial, obteniendo una memoria de cálculo que permite observar el comportamiento del manipulador a través de las cinemáticas y dinámicas de sus actuadores. De esta manera, esta memoria de título puede funcionar como una ante sala a una implementación de un robot antropomórfico en la Universidad de Talca, campus Los Niches.

1.2. Estado del arte

1.2.1. Introducción a la robótica

El término robot se dice que procede de la voz checa robota, que significa trabajos forzados y fue acuñado por el escritor checo Karel Čapek para su obra de teatro Rossum's Universal Robots en 1921, donde unas máquinas de material orgánico reemplazan al hombre en las tareas cotidianas

[14], la representación de la obra se puede apreciar en la Figura 1.1. En los años 40, Isaac Asimov, en sus historias de ciencia ficción se imaginaba al robot como ayudante de la humanidad y postulaba tres reglas básicas para robots. Estas 3 leyes, por lo general, se conocen como las leyes de la robótica.

Leyes de la robótica:

1. **Primera ley:** un robot no debe hacer daño a un ser humano o, por inacción, permitir que un ser humano sufra daño.
2. **Segunda ley:** un robot debe obedecer las órdenes dadas por los seres humanos, excepto si estas órdenes entrasen en conflicto con la primera ley.
3. **Tercera ley:** un robot debe proteger su propia existencia en la medida en que esta protección no entre en conflicto con la primera o la segunda ley.

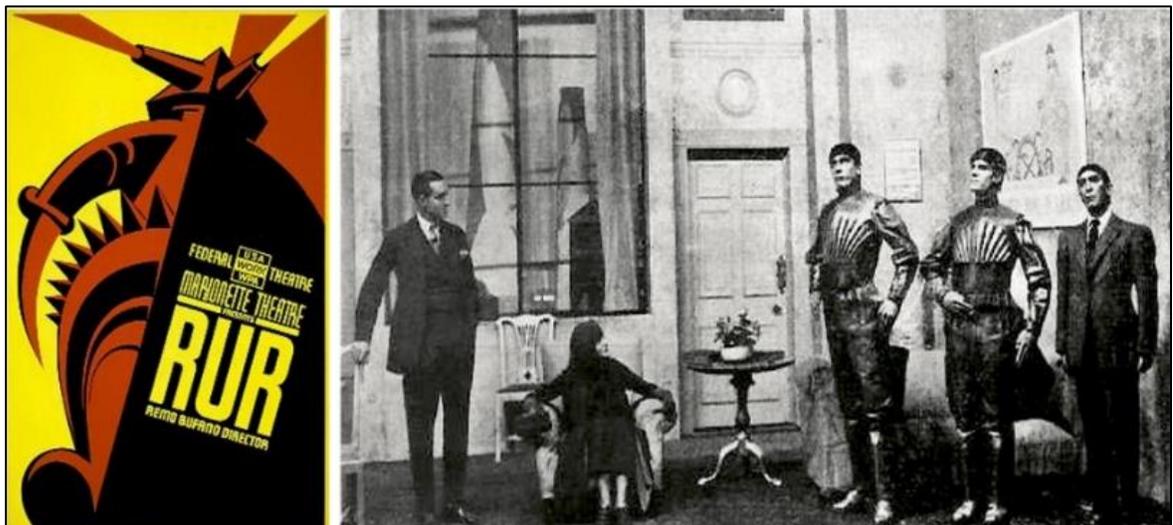


Fig. 1.1 – Portada de la obra de teatro Rossum's Universal Robots y escena de una representación [14].

1.2.2. Clasificación de los robots

Existen diversas clasificaciones respecto a los robots. Atendiendo al tema central de esta investigación, las clasificaciones más relevantes se presentan a continuación:

- Clasificación según sus capacidades:
 - Robot móvil: exploración y transporte.
 - Robot fijo: asistencia médica, automatización de procesos industriales.
 - Otros: control de prótesis, entretenimiento.

- Clasificación según su arquitectura: considera la potencia del software en el controlador, lo que es determinante de la utilidad y flexibilidad del robot.
 - Generación.
 - Nivel de inteligencia.
 - Nivel de control.
 - Lenguaje de programación.
- Clasificación según su forma:
 - Androides y humanoides: simular o realizar robots de forma humana.
 - Zoomórficos: imitan la forma de animales.
 - Robots móviles: tienen elementos que les permiten desplazarse como patas, ruedas u orugas.
 - Robots industriales: autómatas destinados a realizar procesos de fabricación o manipulación.
 - Robots teledirigidos: dirigir a un robot por conexión eléctricamente activa.
 - Robots híbridos: tienen varias configuraciones, básicamente combinando una forma con otra.

Esta tesis está centrada en el estudio de un robot antropomórfico, el cual pertenece a la clasificación de los robots industriales. En esta subsección se presentarán los tipos de robots industriales conocidos y sus características principales, además de su morfología, atendiendo a la estructura mecánica del robot.

1.2.3. Robots industriales:

Los robots industriales a pesar de estar clasificados por su forma, dentro de este grupo se pueden clasificar en función del tipo de articulaciones que tienen sus brazos. Esta clasificación se puede observar en la Tabla 1.1.

Tabla 1.1 - Tipos de brazos manipuladores o industriales.

Manipulador	Articulaciones en el brazo
Cartesiano	3 prismáticas
Cilíndrico	1 rotacional y 2 prismáticas

Esféricos	2 rotacionales y 1 prismática
SCARA	2 rotacionales y 1 prismática
Antropomórfico	3 rotacionales

1.2.3.1. Robot manipulador cartesiano:

Los manipuladores cartesianos se caracterizan por presentar una geometría compuesta por tres articulaciones prismáticas, los ejes de las articulaciones usualmente son ortogonales entre sí. Además, el GDL de cada articulación corresponde con una variable x , y , z ; y su espacio de trabajo viene definido por un paralelepípedo rectangular [14], esto se puede apreciar en la Fig. 1.2.

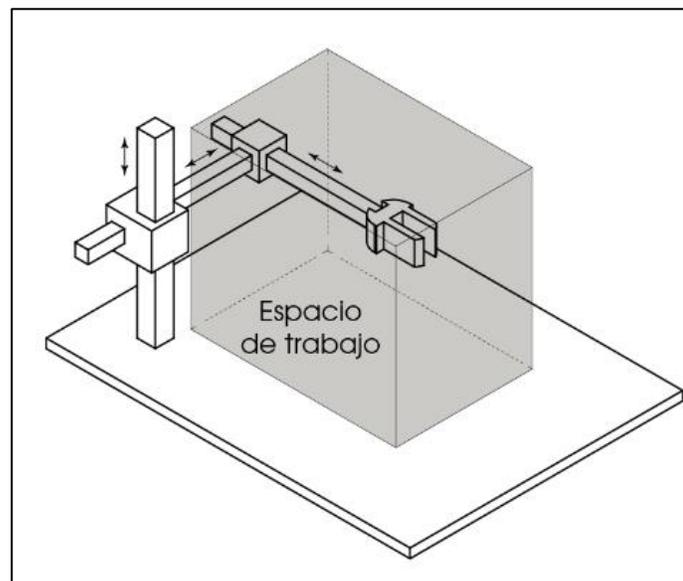


Fig. 1.2 – Manipulador cartesiano [14].

1.2.3.2. Robot manipulador cilíndrico:

Cada GDL de un manipulador cilíndrico corresponde con una variable del espacio en coordenadas cilíndricas. Las coordenadas cilíndricas definen la posición de un punto del espacio mediante un ángulo, una distancia horizontal y una distancia vertical. Además, presenta una geometría compuesta por tres articulaciones, la primera rotacional y las dos siguientes prismáticas, una en horizontal y otra en vertical. Si se observa la Fig. 1.3, el espacio de trabajo es parte de un cilindro, normalmente no es completo debido a las limitaciones mecánicas en el recorrido de la primera articulación [14].

1.2.3.3. Robot manipulador esférico:

Cada GDL de un robot manipulador esférico se corresponde con una variable del espacio en coordenadas esféricas. Las coordenadas esféricas definen la posición de un punto del espacio mediante dos ángulos y una distancia radial. Además, presenta una geometría compuesta por tres articulaciones, las dos primeras rotacionales y la tercera prismática. Al observar la Fig. 1.4 se aprecia que el espacio de trabajo de este manipulador es parte de una esfera [14].

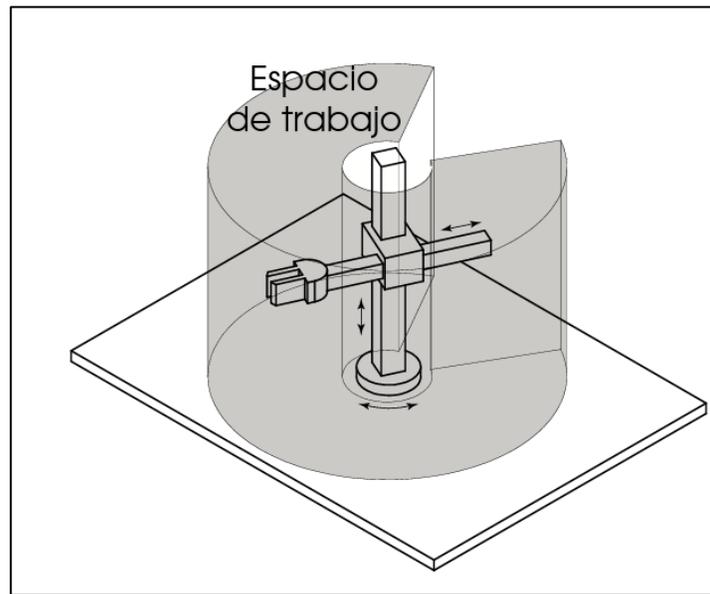


Fig. 1.3 – Manipulador cilíndrico [14].

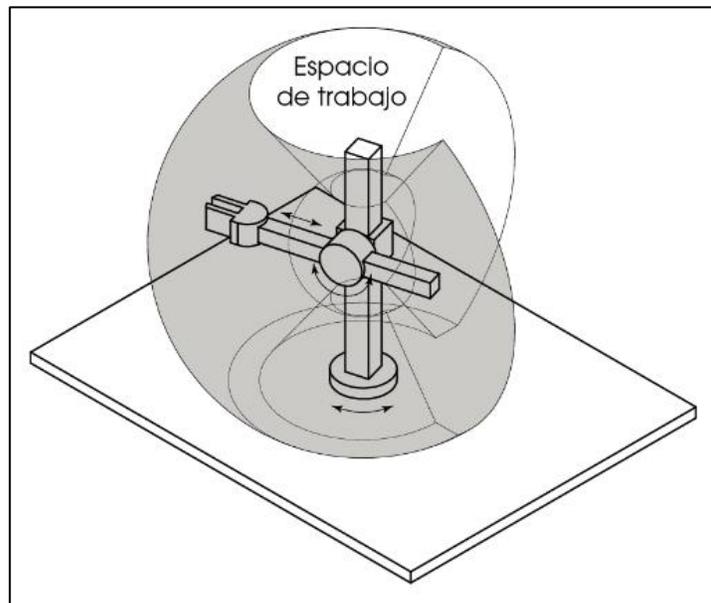


Fig. 1.4 – Manipulador esférico [14].

1.2.3.4. Robot manipulador SCARA:

Los manipuladores SCARA reciben su nombre del acrónimo inglés de Selective Compliance Assembly Robot Arm, que podría traducirse como brazo robótico con adaptabilidad selectiva. Presenta una geometría compuesta por tres articulaciones, las 2 primeras rotacionales y la tercera prismática. La disposición es distinta al esférico, ya que en este caso todos los ejes de movimiento son paralelos. Al observar la Fig. 1.5 se aprecia el espacio de trabajo característico de un robot SCARA.

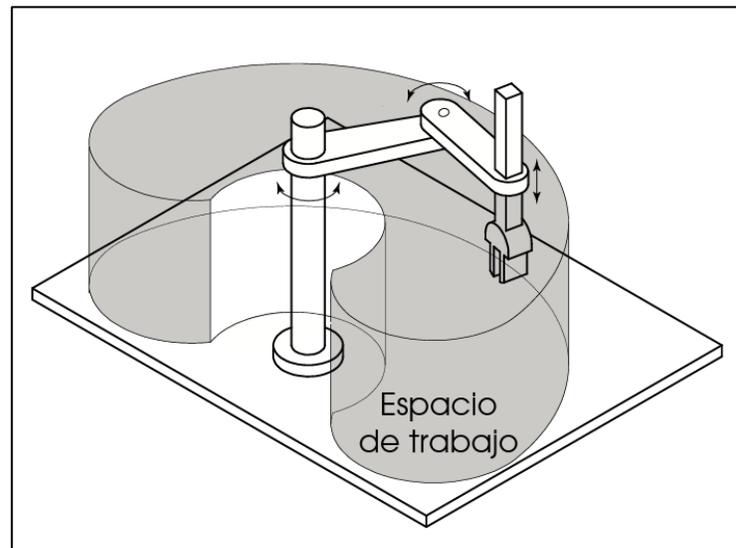


Fig. 1.5 – Manipulador SCARA [14].

1.2.3.5. Robot manipulador antropomórfico:

Los robots antropomórficos presentan una geometría compuesta por tres articulaciones de revolución. Se caracteriza por ser el manipulador más versátil de todos y por su similitud con el brazo humano, de hecho, se suele llamar a sus articulaciones hombro y codo, respectivamente. Este robot no presenta una correspondencia entre los GDL y el espacio de trabajo, esto provoca que sea más difícil de controlar. Si se observa la Fig. 1.6 se puede apreciar el espacio de trabajo de un robot antropomórfico [14].

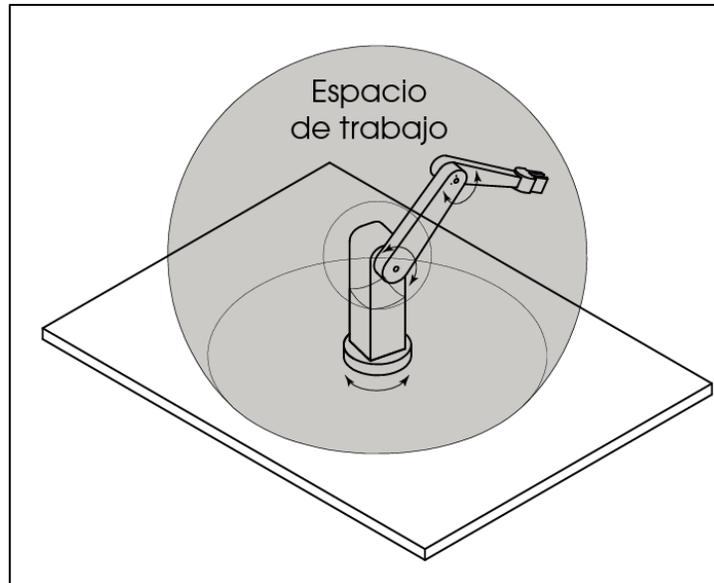


Fig. 1.6 – Manipulador antropomórfico [14].

1.2.4. Aplicaciones de robots industriales

A continuación, se mencionan algunas aplicaciones de los robots industriales. Partiendo por el robot cartesiano, se dice que es un robot simple basado en un sistema de ejes cartesianos X-Y-Z. Dentro de sus aplicaciones principales se encuentra en la manufactura realizada a través de máquinas CNC e impresoras 3D [2].

El robot industrial cilíndrico se utiliza comúnmente en operaciones de ensamblaje, soldaduras por puntos y en moldeado de metales [1]. El robot industrial esférico se utiliza en labores de vigilancia, monitoreo industrial y en desarrollo infantil y entretenimiento [6]. El robot SCARA tiene la principal función de automatizar procesos, dentro de sus principales aplicaciones se encuentran las aplicaciones de ensamblaje, manipulación de materiales de precisión, rutinas pick and place, aplicaciones de atornillado y clasificación de materiales [8].

Por último, el robot antropomórfico tiene una gran cantidad de aplicaciones industriales de las cuales se puede mencionar, paletizado, envasado, soldadura láser, soldadura por arco, soldadura por puntos, carga y descarga de máquinas [5].

1.2.5. Morfología de los robots industriales

Las características más importantes que se deben tener presentes al realizar un diseño mecánico para un manipulador, es su morfología, tipo de articulaciones y largo de los eslabones, esto es

necesario para calcular los torques que necesitarán los actuadores y saber cuántos grados de libertad presenta el diseño.

Los manipuladores industriales son los robots que comúnmente nos encontramos en la industria. Realizan tareas muy variadas, como pintar, soldar, mover objetos o ensamblar [15]. La estructura mecánica de un robot manipulador está compuesta de dos elementos fundamentales:

1. **Eslabones:** elementos, generalmente rígidos, que sirven como estructura al robot. Habitualmente son piezas de acero.
2. **Articulaciones:** elementos que unen los eslabones y les dan movilidad.

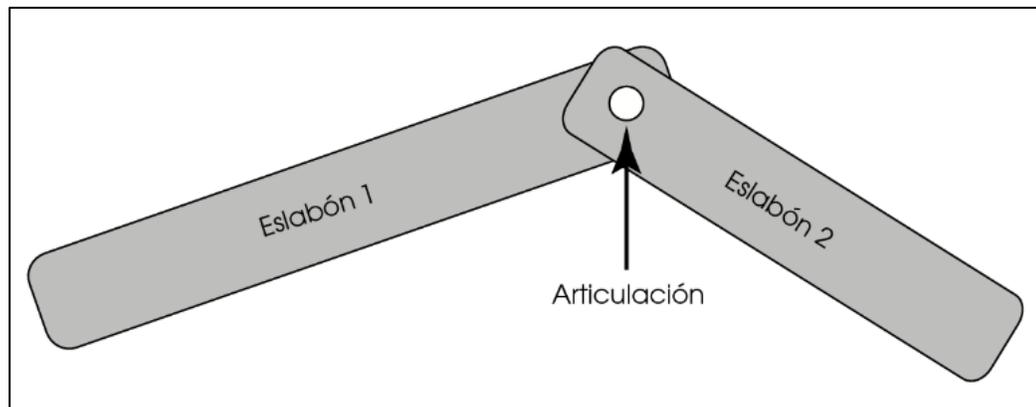


Fig. 1.7 – Ejemplo de dos eslabones con una articulación [14].

A partir de los eslabones y las articulaciones se construyen las diferentes partes del robot, que podemos dividir en:

1. **Brazo:** estructura mecánica que proporciona movilidad. Permite al robot aproximarse al objeto sobre el que se desea trabajar [15].
2. **Muñeca:** estructura mecánica que proporciona destreza. Permite al robot posicionarse y orientarse de forma precisa sobre el objeto una vez que el brazo ha hecho la aproximación [15].
3. **Elemento terminal:** herramienta que realiza una acción. Pueden ser pinzas, soldadores, equipos de pintura, etc. Es el elemento que ejecuta la operación sobre el objeto [15].

La movilidad de un manipulador se obtiene de sus articulaciones, el movimiento de cada articulación puede ser de desplazamiento, de giro o una combinación de ambos [18]. A continuación, se presentan los tipos de articulaciones más utilizados en la robótica industrial:

1. **Prismática o de traslación:** permite a los eslabones moverse en direcciones perpendiculares entre sí.
2. **Rotacional o de revolución:** permite a los eslabones rotar entre ellos.

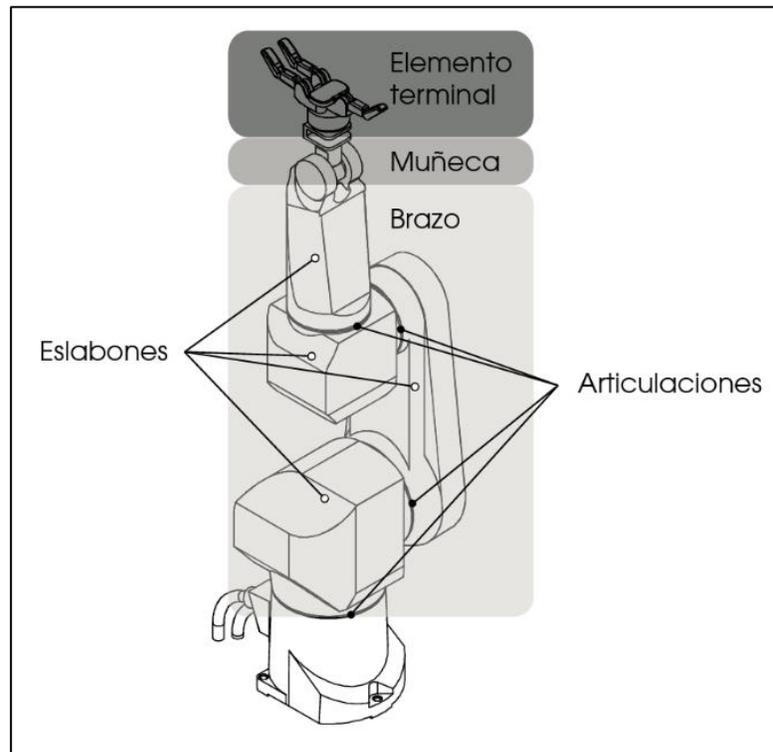


Fig. 1.8 – Elementos de un robot manipulador [14].

1.2.6. Elemento terminal

Los elementos terminales, también llamados efectores finales (end effector) son los encargados de interactuar directamente con el entorno del robot. Si bien un mismo robot industrial es, dentro de unos límites lógicos, versátil y adaptable a una gran variedad de aplicaciones, no ocurre así con los elementos terminales, que son en muchos casos específicamente diseñados para cada tipo de trabajo [14].

Se puede establecer una clasificación de los elementos terminales atendiendo a si se trata de un elemento de sujeción o de una herramienta [14]. Los elementos de sujeción más comunes son

denominados: pinzas o garras, pueden tener un accionamiento neumático o eléctrico. Las herramientas terminales más utilizadas son la soldadura por puntos, por arco o para pintar [18]. Algunos de los elementos más utilizados se presentan en las Fig. 1.9, Fig. 1.10 y Fig. 1.11.



Fig. 1.9 – Pinza de dos dedos con accionamiento neumático [18].



Fig. 1.10 – Ventosa para manipulación por vacío [18].

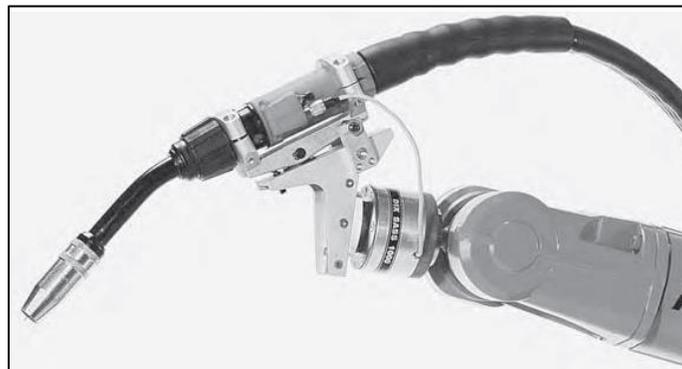


Fig. 1.11 – Antorcha de soldadura al arco [18].

1.2.7. Grados de libertad

Los grados de libertad (GDL) de un robot nos indican el número de movimientos independientes que pueden realizar. Para entenderlo mejor podemos fijarnos en la Fig. 1.12, el robot de la izquierda solo tiene un movimiento, por lo que tiene 1 GDL, mientras que el de la derecha tiene dos movimientos independientes, por lo que tiene 2 GDL [14].

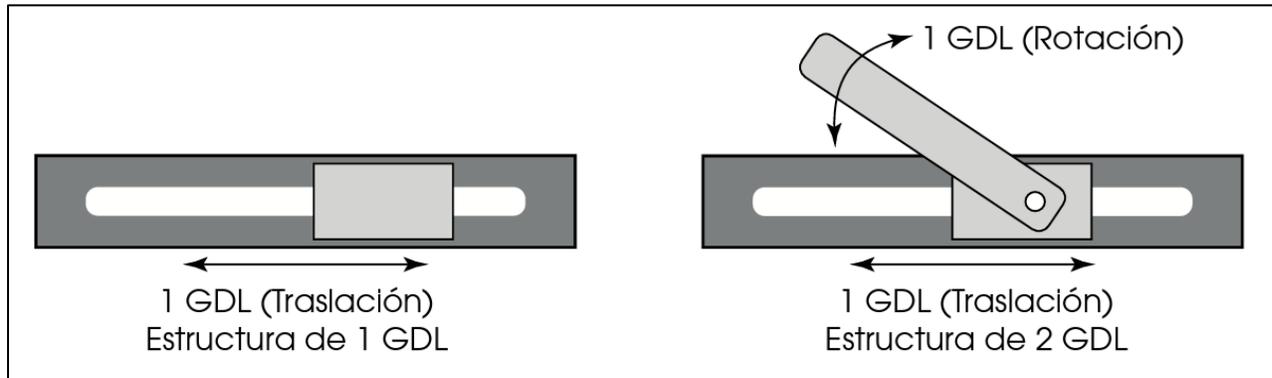


Fig. 1.12 – Ejemplo de sistema con 1 y 2 GDL [14].

1.2.8. Robot antropomórfico

En el desarrollo de la robótica tal y como la entendemos hoy, existen una pluralidad de intereses y tecnología diferentes que hace que sea muy difícil establecer una secuencia lógica y consistente de eventos o descubrimientos en los que apoyarse. No obstante, se pueden establecer algunas líneas maestras para esclarecer el progreso de este campo, teniendo en cuenta que las incorporaciones científico-tecnológicas se suceden a cada momento [1]. Es por esto que en esta subsección se aborda una secuencia lineal de los estudios e investigaciones relacionadas al tema central de esta memoria de título.

En 1954 se crea el primer robot industrial UNIMATE, la primera patente de un robot actual fue registrada en 1954 por George C. Devol. Consistía en un brazo mecánico terminado en una pinza que estaba montado sobre unas guías y cuya secuencia de movimientos estaba codificada en un tambor magnético [14]. Este modelo se denominó UNIMATE y se puede observar en la Fig. 1.13.

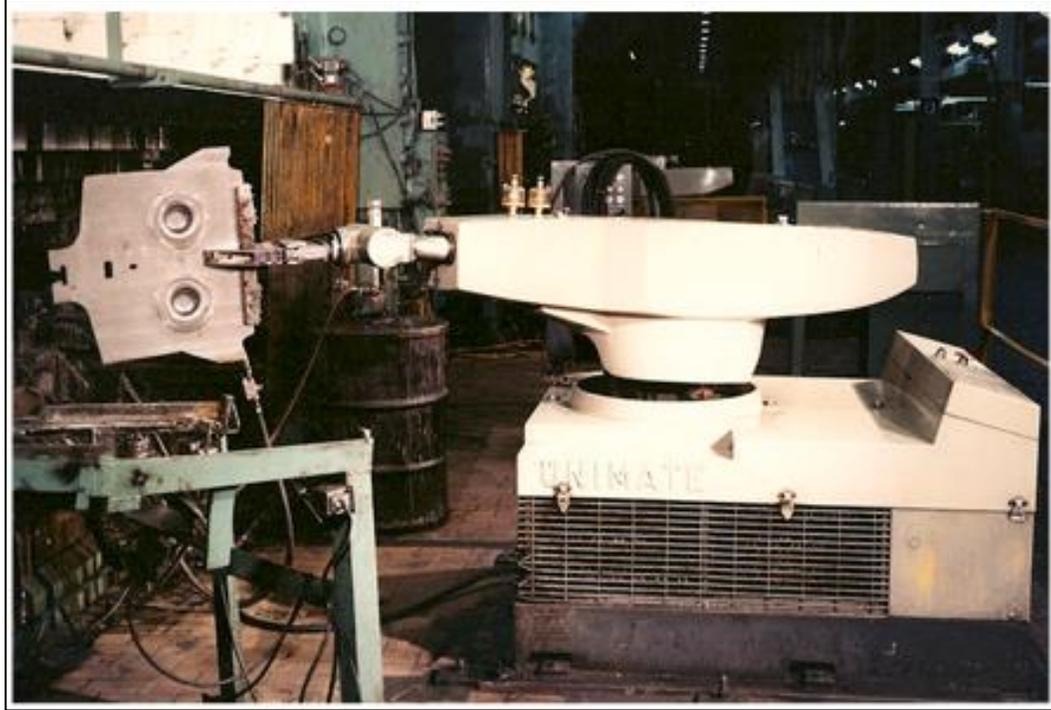


Fig. 1.13 – Primer robot de la historia (UNIMATE) [14].

En 1969 se creó el Stanford Arm, el primer brazo robótico controlado mediante un computador electrónico, ya que el UNIMATE tan solo tenía una memoria magnética en la que almacenaba las instrucciones, pero no un computador completo [14]. Este robot se puede observar en la Fig. 1.14.



Fig. 1.14 – Stanford arm [14].

En 1973, la empresa KUKA escribió la historia como pionera de la robótica con el primer robot industrial del mundo con seis ejes accionados por motor eléctrico, este primer robot fue llamado Famulus [70].



Fig. 1.15 – Primer robot industrial de KUKA, llamado FAMULUS. [70]

En 1981 se crea el primer robot con accionamiento directo, Takeo Kanade presenta el primer brazo robótico de accionamiento directo. Esto quiere decir que los motores que mueven el robot están directamente colocados en el punto donde ejecutan el movimiento, es decir, que no utilizan mecanismos de transmisión, como correas dentadas o cadenas [14]-[15].

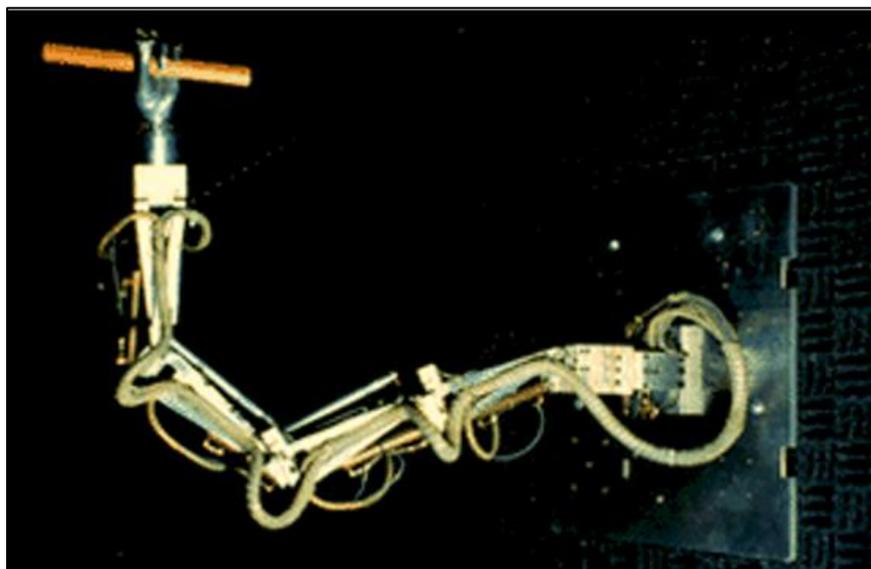


Fig. 1.16 – Brazo robótico de accionamiento directo [14].

En los años finales de la década de los sesenta y principios de la década de los setenta se ponen las bases de la investigación en robótica en las universidades. Se crean departamentos de investigación en inteligencia artificial, y se diseñan los primeros robots móviles con un cierto grado de autonomía, como Shakey por el Stanford Research Institute o el Stanford Cart de la Universidad de Stanford. También se ponen las bases de los diseños actuales de brazos manipuladores, principalmente por Victor Scheinman, con el diseño del brazo de Stanford. Este robot fue el primer manipulador controlado por computador y con accionamiento eléctrico, y que llevaría más tarde al diseño, fabricación y comercialización del robot PUMA [18].

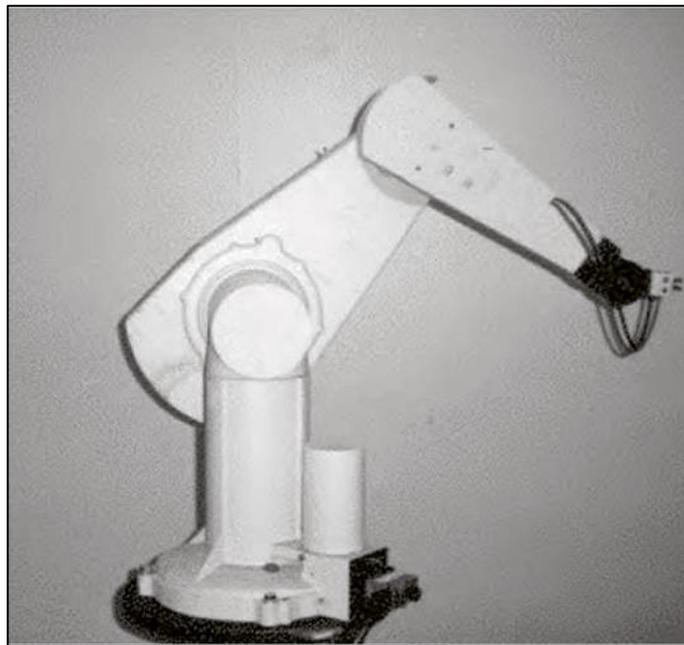


Fig. 1.17 - Robot PUMA [18].

En Europa, la robótica tuvo un despertar más tardío, aunque no menos relevante. En 1973 la firma sueca ASEA construyó el primer robot con accionamiento totalmente eléctrico, el robot IRb6, seguido un año más tarde del IRb60. Posteriormente, ya como ABB, se convertiría en una de las empresas más importantes del mundo en la fabricación de robots industriales, y Suecia uno de los países con más robots per cápita. En este sentido, en 1980 se fundó la Federación Internacional de Robótica con sede en Estocolmo, Suecia [18].

Los manipuladores mencionados anteriormente, establecieron las bases para lo que son, hoy por hoy, los manipuladores robóticos industriales, han mejorado gracias a diferentes

investigaciones, que buscan innovar con nuevos sistemas robóticos. A continuación, se presentan las investigaciones y fuentes encontradas, relacionadas con el tema principal de esta tesis, realizadas en el siglo XXI.

En 2004, en la Universidad Politécnica de Cartagena se plantea crear a SimuRob, que es un simulador bajo Windows del robot antropomórfico IRB-1400. El lenguaje de programación estaba basado en C++ y fue diseñado con el objetivo que los estudiantes pudieran aprender el manejo del robot y la programación de sus movimientos [35]. Posteriormente, en el año 2008 se crea el diseño y simulación de un robot a través de Matlab/Simmechanics, en la cual se muestra el uso de ambos softwares, además explica cómo realizar el diseño de un manipulador y asociarlo al software Matlab/Simmechanics [31]. Al año siguiente, se presenta la simulación de la cinemática directa e inversa de un robot paralelo de 2 GDL [32]-[26] y de cómo controlarlo a través del mismo software utilizado en [31].



Fig. 1.18 - Robot Irb6 de la firma sueca ASEA [18].

En 2010, se presenta una investigación [23] relacionada al estudio de la cinemática de un robot paralelo de 6 GDL, se detalla el desarrollo de la cinemática directa e inversa para controlar el manipulador. El mismo año, se realiza un estudio [25] relacionado al desarrollo y análisis de la

cinemática y dinámica de un robot de 3 GDL. Dos años más tarde, en 2012, se desarrolla un análisis relacionado al mismo tema de [25], la investigación se planteó en la Universidad Zhuhai de Beijing, donde se abordó la creación de un manipulador paralelo de 3 GDL flexible, se estudió la cinemática y la dinámica del manipulador. Además, se presenta un estudio de esfuerzos a través de cargas puntuales en la estructura mecánica del robot [24]. Años posteriores, en 2016, en el Instituto Tecnológico y de Estudios Superiores de Monterrey se propone diseñar diversos prototipos de manipuladores robóticos industriales, entre ellos un robot antropomórfico, con el fin de estudiar la cinemática directa de todos los manipuladores creados, para observar y estudiar los movimientos de los robots industriales [36].

Las investigaciones más recientes encontradas, se exponen en el año 2019, en aquellas se presentan simulaciones en Matlab para un robot antropomórfico de 6 GDL, relacionadas al control de la cinemática directa e inversa del manipulador [19]. Además, en [20] y [21] se presenta el diseño mecánico de 2 manipuladores, uno de 6 GDL y otro de 4 GDL respectivamente, abordando el mismo tema principal, analizar y simular la cinemática directa e inversa de los manipuladores mencionados. El mismo año, Juan Carlos Romero presenta una investigación en la Universidad del Altiplano en Perú relacionada al desarrollo de un sistema de control para prototipo de un robot antropomórfico [34], en la investigación detalla los procesos necesarios actualmente para lograr controlar un robot antropomórfico y realiza diferentes pruebas relacionadas con el uso de una APK para comparar el control del manipulador a través de un joystick o por medio de una computadora.

1.2.9. Actuadores

Los elementos más trascendentales de un manipulador antropomórfico son sus actuadores, ya que son estos lo que definen el correcto movimiento de cada una de sus articulaciones, dependerá de sus actuadores si sus movimientos son suaves o bruscos. Al presentar en esta memoria de título solo como una simulación, se crea una lista de los posibles actuadores que se pueden utilizar en un robot antropomórfico.

Existen diferentes tipos de actuadores, con diferentes accionamientos, entre los más utilizados se destacan los actuadores neumáticos, hidráulicos y eléctricos [18]. Para este caso, el estudio se centra en los actuadores eléctricos y se presentan a continuación:

1.2.8.1. Motor de corriente continua (CC)

Los motores de corriente continua (CC) pueden variar en tamaño y potencia, desde pequeños motores en juguetes y electrodomésticos, hasta grandes mecanismos que impulsan vehículos, trenes, ascensores y accionan trenes de laminación de acero [37]. En electrónica, se usa bastante debido a la facilidad de cambiar su sentido de giro, a través de un puente H, además la velocidad de giro se puede realizar mediante una señal PWM [38]. La técnica de modulación PWM se usa en inversores DC/AC monofásicos y trifásicos. Se basan en la comparación de una señal de referencia modular y una señal portadora de forma triangular o diente de sierra; la comparación genera un tren de pulsos de anchos específico que se utilizan en la conmutación del puente inversor, de la misma manera se utiliza en los motores DC, para controlar la velocidad al variar la energía que se le suministra al motor.



Fig. 1.19 – Motor CC – 12 V.

1.2.8.2 Motor paso a paso

Existen diferentes varios tipos de motores paso a paso, pero el más utilizado en robótica es el de imanes permanentes. Este tipo de motor paso a paso está constituido por un rotor de imanes permanentes y un estator formado por varios electroimanes. Cada uno de estos electroimanes será una fase del motor [14].

El principio de funcionamiento es similar al del motor síncrono, con la diferencia de que en este caso las tensiones que alimentan a los electroimanes del estátor no son funciones senoidales alternas, sino secuencias de pulsos de tensión constante. Esto provoca que el rotor del motor gire

a saltos, no de forma continua. El mayor inconveniente de estos motores es su baja potencia. La Fig. 1.20 muestra un motor paso a paso utilizado comúnmente en pequeños robots o impresoras 3D.



Fig. 1.20 – Motor paso a paso Nema 17.

1.2.8.2 Servomotor

Un servomotor es un actuador rotativo o motor que permite un control preciso en términos de posición angular, aceleración y velocidad, capacidades que un motor normal no tiene. En definitiva, utiliza un motor normal y lo combina con un sensor para la retroalimentación de posición. Pero, los servomotores no son en realidad una clase específica de motor, sino una combinación de piezas específicas, que incluyen un motor de corriente continua o alterna, y son adecuados para su uso en un sistema de control de bucle cerrado [18].

Por lo que una definición más exacta de un servomotor sería la de un servomecanismo de bucle cerrado que utiliza la retroalimentación de posición para controlar su velocidad de rotación y posición. La señal de control es la entrada, ya sea analógica o digital, que representa el comando de posición final para el eje. Por otro lado, el codificador o encoder sirve como sensor, proporcionando retroalimentación de velocidad y posición. En la mayoría de los casos, sólo se informa de la posición. La posición final se informa al controlador y se compara con la entrada de posición inicial, y luego, si hay una discrepancia, se mueve el motor para llegar a la posición correcta [20].



Fig. 1.21 – Servomotor [45].

1.2.10. Tipos de trayectorias

1.2.11.1 Trayectorias punto a punto

En este tipo de trayectorias cada articulación evoluciona desde su posición inicial a la final sin realizar consideración alguna sobre el estado o evolución de las demás articulaciones. Normalmente, cada actuador trata de llevar a su articulación al punto de destino en el menor tiempo posible, pudiéndose distinguir dos casos: movimiento eje a eje y movimiento simultáneo de ejes. A su vez, dentro del segundo tipo, los ejes pueden moverse de manera independiente o coordinada [18].

1.2.11.2 Movimiento eje a eje

En este caso, sólo se mueve un eje cada vez. Comenzará a moverse la primera articulación, y una vez que ésta haya alcanzado su punto final lo hará la segunda, y así sucesivamente. Este tipo de movimiento da como resultado un mayor tiempo de ciclo, teniendo como única ventaja un menor consumo de potencia instantánea por parte de los actuadores y la posible multiplexación de una única etapa de potencia. Su uso en la práctica actual es muy limitado [18].

1.2.11.3 Movimiento simultáneo de ejes

En este caso, todos los actuadores comienzan simultáneamente a mover las articulaciones del robot a una velocidad específica para cada una de ellas. Dado que la distancia a recorrer y las velocidades serán en general diferentes, cada una acabará su movimiento en un instante diferente.

El movimiento del robot no acabará hasta que se alcance definitivamente el punto final, lo que se producirá cuando concluya su movimiento el eje que más tarde. De esta manera, el tiempo total invertido en el movimiento coincidirá con el del eje que más tiempo emplee en realizar su movimiento particular, pudiéndose dar la circunstancia de que el resto de los actuadores hayan forzado su movimiento a una velocidad y aceleración elevada, viéndose obligados finalmente a esperar a la articulación más lenta. Por los motivos expuestos, en la práctica las trayectorias con movimiento simultáneo de los ejes, pero no coordinado, no están por lo general implementadas, siendo sustituidas por las trayectorias coordinadas que se describen a continuación [18].

1.2.11.4 Trayectorias coordinadas o isócronas

Para evitar que algunos actuadores trabajen forzando sus velocidades y aceleraciones, teniendo que esperar después la conclusión del movimiento de la articulación más lenta, puede hacerse un cálculo previo, averiguando cuál es esta articulación y qué tiempo invertirá. Se ralentizará entonces el movimiento del resto de los ejes para que inviertan el mismo tiempo en su movimiento, acabando todos ellos simultáneamente. Se tiene así que todas las articulaciones se coordinan comenzando y acabando su movimiento a la vez, adaptándose todas a la más lenta [18].

El tiempo total invertido en el movimiento es el menor posible y no se demandan aceleraciones y velocidades elevadas a los actuadores de manera inútil. Desde el punto de vista del usuario la trayectoria que describe el extremo del robot no es significativa, siendo ésta impredecible, aunque como es obvio, un conocimiento del modelo y control cinemático del robot permitiría su cálculo. La mayor parte de los robots industriales cuentan con este tipo de trayectorias, denominándose frecuentemente de manera genérica como trayectorias punto a punto [18].

1.2.11.5 Trayectorias continuas

Cuando se pretende que la trayectoria que sigue el extremo del robot sea conocida por el usuario (trayectoria en el espacio cartesiano o de la tarea), es preciso calcular y controlar de manera continua las trayectorias articulares, que, por lo general, presentarán una compleja evolución temporal.

1.2.11. Generación de trayectoria

Normalmente, el usuario del robot indica el movimiento que éste debe realizar especificando las localizaciones espaciales por las que debe pasar el extremo, junto con otros datos, como instantes de paso, velocidades o tipos de trayectorias. Así, por ejemplo, es frecuente especificar que el robot debe ir de un punto inicial hasta otro final, siguiendo en el espacio de la tarea una línea recta a velocidad constante [18].

La trayectoria cartesiana más habitual es la línea recta, utilizándose también en ciertas ocasiones trayectorias circulares. Ambas pueden ser descritas con facilidad de manera analítica en función del tiempo, de modo que es fácil conocer las coordenadas cartesianas por las que se desea que pase el extremo del robot en un instante concreto [31].

Con estos objetivos se propone un algoritmo recursivo que permite seleccionar sólo aquellos puntos de la trayectoria cartesiana estrictamente necesarios, quedando estos más próximos entre sí allí donde se produciría mayor error de la trayectoria, y más espaciados en los tramos en los que las trayectorias deseada y resultante son más cercanas [14].

En la práctica, por lo general es suficiente con seleccionar puntos equidistantes tan cercanos como sea posible, asegurando que las trayectorias articulares puedan ser generadas antes del tiempo en que sean requeridas por el control dinámico del robot [18].

1.2.12. Interpolación de trayectorias

Una vez que se dispone de la secuencia de configuraciones articulares por las que debe pasar el robot, la siguiente función del control cinemático es la de unir esta sucesión de puntos articulares garantizando, junto con las condiciones de configuración-tiempo de paso, que se cumplan las restricciones de velocidad y aceleración máximas asociadas a los actuadores del robot, de manera que se consiga una trayectoria realizable y suficientemente suave.

Para ello deberá seleccionarse algún tipo de función (frecuentemente polinómica) cuyos parámetros o coeficientes se ajustarán al imponer las condiciones de contorno: posiciones, velocidades y aceleraciones. En la selección de esta función debe considerarse que tanto el cálculo de sus parámetros, como su posterior utilización para generar puntos de consigna al control dinámico, debe hacerse en tiempo real, por lo que la simplicidad de la función será un factor para valorar.

A continuación, se presentan las funciones interpoladoras utilizadas con mayor frecuencia. Cada una de ellas ha sido desarrollada para un solo grado de libertad, debiendo quedar claro que el mismo cálculo deberá repetirse para cada uno de los grados de libertad del robot. Cabe indicar que si bien las técnicas de interpolación están planteadas para el espacio articular son igualmente aplicables para el espacio de la tarea [18].

1.2.13.1 Interpoladores lineales

Se pretende que una de las articulaciones q del robot, pase sucesivamente por los valores q^i en los instantes t^i . Una primera solución a este problema consistiría en mantener constante la velocidad de movimiento entre cada 2 valores sucesivos (q^{i-1}, q^i) de la articulación.

La trayectoria entre dos puntos (q^{i-1}, q^i) sería entonces:

$$q(t) = (q^i - q^{i-1}) \frac{t - t^{i-1}}{T} + q^{i-1} \quad t^{i-1} < t < t^i \quad (1.1)$$

donde,

$$T = t^i - t^{i-1} \quad (1.2)$$

Como es obvio (observar Fig. 1.22), esta trayectoria asegura la continuidad de la posición, pero origina saltos bruscos en la velocidad \dot{q} de la articulación, y consecuentemente precisa de aceleraciones \ddot{q} de valor infinito, lo que en la práctica no es posible [18].

1.2.13.2 Interpolador splin cúbico

Para asegurar que la trayectoria que une los puntos por los que tiene que pasar la articulación considerada presente continuidad en velocidad, puede recurrirse a utilizar un polinomio de grado 3 que una cada pareja de puntos adyacentes. De este modo en cada intervalo, al tener cuatro parámetros disponibles se podrán imponer cuatro condiciones de contorno, dos de posición y dos de velocidad. Los valores de las velocidades de paso por cada punto deben, por tanto, ser conocidas a priori. Se consigue así una trayectoria compuesta por una serie de polinomios cúbicos, cada uno válido entre dos puntos consecutivos [18].

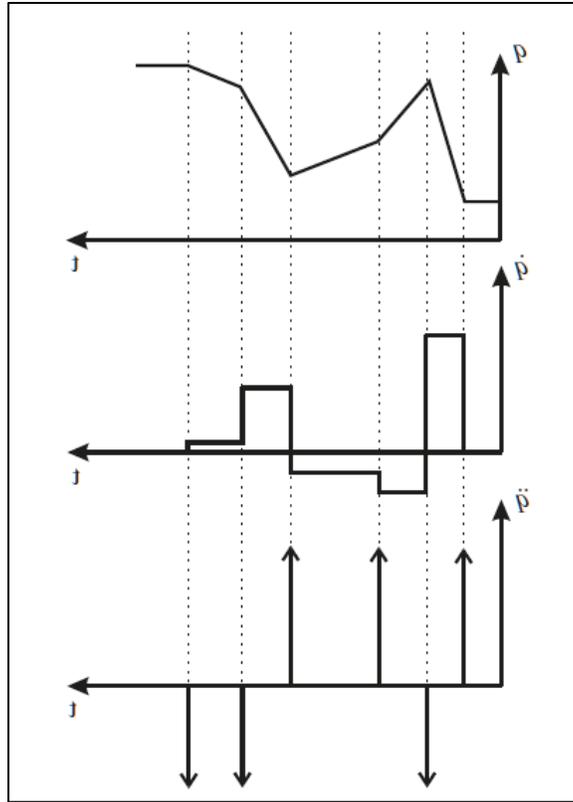


Fig. 1.22 – Posición, velocidad y aceleración para un interpolador lineal [18].

Este conjunto de polinomios concatenados, escogidos de modo que exista continuidad en la posición y velocidad, se denominan splines cúbicos, por ser de tercer grado. La expresión de la trayectoria que une dos puntos adyacentes (q^i, q^{i+1}) será:

$$q(t) = a + b(t - t^i) + c(t - t^i)^2 + d(t - t^i)^3 \quad t^i < t < t^{i+1} \quad (1.3)$$

donde,

$$a = q^i \quad (1.4)$$

$$b = \dot{q}^i \quad (1.5)$$

$$c = \frac{3}{T^2}(q^{i+1} - q^i) - \frac{1}{T}(\dot{q}^{i+1} + 2\dot{q}^i) \quad (1.6)$$

$$d = -\frac{2}{T^3}(q^{i+1} - q^i) + \frac{1}{T^2}(\dot{q}^{i+1} + 2\dot{q}^i) \quad (1.7)$$

$$T = t^{i+1} - t^i \quad (1.8)$$

1.2.13.3 Interpolador splin quíntico

La segunda de las alternativas antes consideradas para fijar las velocidades de paso garantizaba la continuidad de las aceleraciones a costa de dejar libres los valores de las velocidades (no podían ser especificadas como condiciones de contorno). En el caso de que las velocidades de paso vengan impuestas por algún criterio externo (y sean, por tanto, condiciones de contorno) y se precise garantizar la continuidad de las aceleraciones, es necesario aumentar el grado del polinomio que define la trayectoria en cada intervalo hasta 5, disponiendo así de 6 parámetros que permiten ajustar las 6 condiciones de contorno:

$$q(t) = a + b(t - t^{i-1}) + c(t - t^{i-1})^2 + d(t - t^{i-1})^3 + e(t - t^{i-1})^4 + f(t - t^{i-1})^5 \quad (1.9)$$

La imposición de las 6 condiciones de contorno en cada tramo permite obtener los correspondientes coeficientes que definen el polinomio.

1.2.13. Discusión

En base a lo descrito anteriormente, desde un punto de vista industrial, en Chile ha aumentado el uso de este tipo de manipuladores, en mayor rango en el área minera [67]-[68]-[69]. De esta manera, se desea crear un estudio relacionado a un robot industrial, en este caso un robot antropomórfico, utilizando software de diseño 3D y software de programación Matlab.

Respecto a la estructura mecánica, diferentes autores como en [20] y [21] trabajan como una estructura rígida de madera (MDF) con servomotores ubicados en el mismo punto de las articulaciones, esto puede no puede ser lo más apropiado si se necesita mayor velocidad y precisión en un manipulador, ya que se está controlando directamente el manipulador con un control cinemático que excluye las dinámicas del robot y de los actuadores, además en este caso utilizaban trayectorias lineales, donde las velocidades actúan como un escalón y sus aceleraciones como un impulso infinito [27]-[28]-[30].

En esta memoria de título se espera diseñar una estructura mecánica rígida, liviana y de gran resistencia mecánica. Esto se puede lograr utilizando el plástico ABS a través de impresiones 3D, el diseño se adaptará para poder utilizar una impresora 3D estándar del mercado para imprimir cada parte del manipulador. En cuanto a la generación de trayectoria, se plantea realizar una interpolación splin cúbica para generar una trayectoria Pick and Place o trayectoria punto a punto.

1.3. Descripción de la problemática

La era de la robótica ya está presente en muchos nichos industriales, tal y como es la minería. En Chile, la principal empresa cuprífera ya ha dado grandes pasos en este sentido con el desarrollo y la implementación de diversas soluciones automatizadas que han traído respuestas a problemas que van desde la necesidad de agilizar los procesos productivos hasta el reemplazo de mano de obra en tareas que representen un posible riesgo para las personas [68]-[69].

La robótica ha llegado a Chile para quedarse, aunque existe una gran alza en la productividad como la minería y la manufactura, aún existe el problema de conocer el correcto funcionamiento de este tipo de manipuladores [67]. Dicho esto, se plantea crear una memoria de título centrada en el estudio y verificación de un robot antropomórfico de 6 GDL, generando un diseño mecánico basado en un manipulador industrial. Además, se plantea crear una caracterización matemática que permita estudiar las cinemáticas, dinámicas y torques asociados al robot.

Hoy en día, para exportar diseños CAD desde Inventor a Matlab hace falta tener una mayor claridad de cómo se realiza el proceso o tener una forma estandarizada de generar este proceso. De esta manera, se pueden simular cualquier diseño mecánico creado en Inventor en la herramienta de Matlab llamada Simscape Multibody, así se podrían estudiar diferentes estrategias de control sobre diferentes diseños mecánicos.

Finalmente, para resolver esta problemática se verifica el funcionamiento de un robot antropomórfico de 6 GDL al generar una trayectoria con una interpolación cúbica, permitiendo estudiar las posiciones, velocidades y aceleraciones de cada articulación. Además, para corroborar el diseño mecánico creado luego de calcular sus torques de manera, se genera guía de configuraciones y simulaciones en Simscape Multibody con el diseño CAD creado, para comparar los resultados de los torques obtenidos, asumiendo que los actuadores son ideales y que la dinámica del robot no afecta en estos torques.

1.4. Objetivos

1.4.1. Objetivo general

- Diseñar y generar trayectoria para un robot antropomórfico de 6 grados de libertad.

1.4.2. Objetivos específicos

- Calcular las relaciones de transmisión para las primeras 3 articulaciones del manipulador.
- Diseñar la estructura mecánica del manipulador relacionado con los cálculos matemáticos realizados.
- Exportar planos mecánicos desde el software CAD – Autodesk Inventor Professional.
- Simular la cinemática directa e inversa del manipulador.
- Simular la dinámica del robot antropomórfico.
- Explicar la comunicación entre Inventor y Matlab.
- Generar trayectoria para el robot antropomórfico de 6 GDL.
- Simular el diseño mecánico creado en Simscape Multibody.
- Explicar el funcionamiento de Simscape Multibody y sus herramientas para generar una simulación a partir de un modelo CAD.

1.5. Alcances

- Los recursos que sean destinados a este proyecto serán entregados por la Escuela de Ingeniería Civil Mecatrónica de la Facultad de Ingeniería de la Universidad de Talca.
- El robot antropomórfico será de 6 grados de libertad.
- Las simulaciones se harán a través de Matlab y Autodesk Inventor Profesional.
- El manipulador no podrá levantar más de 500 gramos.
- Los torques necesarios para el robot antropomórfico consideran una carga máxima de 0,5 kg.

1.6. Limitaciones

- El robot antropomórfico no contará con un cambiador de pieza en el efector final.

1.7. Metodología

Para lograr diseñar y controlar un robot antropomórfico se deben seguir una serie de pasos para lograr los objetivos propuestos y así cumplir con el objetivo principal. Por consiguiente, los pasos para lograr lo mencionado anteriormente son:

1.7.1. Estudio teórico de los fundamentos de la robótica y mecanismos

El estudio teórico de los fundamentos de la robótica y mecanismos está basado en la búsqueda de información respecto del tema, por lo tanto, se revisan fuentes bibliográficas para obtener los conocimientos básicos sobre la robótica, conceptos fundamentales, historia de la robótica, clasificación y morfología de los robots. También de generar una idea clara del diseño del robot, para que tenga una estructura fuerte y movimientos precisos.

La revisión bibliográfica se realiza a través del portal de biblioteca online de la universidad de Talca, que permite bajar libros relacionados con el tema y mantenerlos por un máximo de 21 días. Además, se recurrirá a los documentos relacionados con la topología a estudiar desde la plataforma IEEE – Xplore, la cual tiene convenio con la universidad de Talca, obteniendo acceso directo a toda la información respecto a investigaciones, conferencias, revistas, publicaciones o libros relacionados con el tema principal de la memoria de título.

1.7.2. Diseño mecánico y simulación de un robot antropomórfico de 6 grados de libertad

Luego de tener claro el diseño a realizar, se procede a crear el diseño mecánico a través de un software. Se utiliza el software Autodesk Inventor Profesional 2020 que tiene convenio con la Universidad de Talca y ofrece una licencia estudiantil con duración de 3 años, permitiendo bajar la versión más apropiada para el PC que se utilizará. Además, es necesario realizar cálculo de torques para conocer los actuadores que se utilizarán y hacer un estudio de esfuerzos según el material de construcción.

1.7.3. Estudio teórico de la cinemática y dinámica del manipulador

En este punto, se necesita volver a las revisiones bibliográficas para obtener información respecto de la teoría planteada para la cinemática directa, cinemática inversa y dinámica de un

robot antropomórfico de 6 grados de libertad. El desarrollo de este punto es muy importante para dar movimiento y crear un control sobre el manipulador, ya que va relacionado directamente a cómo se moverá el robot en el espacio.

1.7.4. Simulación de la cinemática y dinámica del manipulador

Se contrasta la información teórica obtenida en el punto anterior, de esta manera, se crean simulaciones en el software MatLab R2020b para observar el correcto comportamiento de la cinemática y dinámica del manipulador.

1.7.5. Simulación del manipulador en Simmechanics – Multibody

Se plantea generar una comunicación entre Autodesk Inventor Profesional y Matlab, a través de la exportación del ensamblaje final del diseño mecánico del manipulador, desde Inventor hacia Matlab. Así, se puede trabajar con el diseño mecánico directamente en Matlab y poder controlar sus movimientos. En el capítulo 4 de esta memoria de título, se presenta un tutorial para generar la comunicación entre ambos softwares. Además, en este punto, se presentará la simulación gráfica del movimiento del manipulador, a través de su cinemática directa e inversa; y dinámica del robot antropomórfico.

1.7.6. Estudio teórico para la generación de trayectorias

En este punto, se necesita volver a las revisiones bibliográficas para obtener información respecto de la teoría planteada para la generación de trayectoria de un robot antropomórfico de 6 grados de libertad. El desarrollo de este punto es utilizado como implementación de los estudios realizados en cuanto a diseño y simulación, además se quiere contrastar el buen funcionamiento del manipulador en una simulación, respetando el diseño mecánico (torques, masas, esfuerzos, etc) y el control (cinemática y dinámica del manipulador).

1.7.7. Simulación para la generación de trayectorias en un robot antropomórfico de 6 GDL

Se genera una simulación con el software Matlab, utilizando la función de Simmechanics y el software Inventor. Se unen ambos programas para generar una simulación donde el manipulador

diseñado siga una trayectoria y se pueda estudiar el comportamiento de los torques asociados al manipulador, respecto del diseño mecánico realizado.

1.8. Resumen capítulo 1

Se presentan los aspectos generales de la memoria de título, exponiendo los objetivos, alcances, limitaciones y la metodología de trabajo. Se plantean las problemáticas que se quieren solucionar en este proyecto, en el estado del arte se expone una introducción a la robótica, explicando sus orígenes y conceptos claves relacionados al tema. Se clasifican los robots y se decide mostrar los diferentes tipos de robots industriales existentes, debido a que el robot antropomórfico atiende a esta clasificación. De la misma manera, se aborda la morfología de los robots industriales, mostrando sus características principales, seguido de un estudio lineal de los diferentes estudios relacionados a la tesis. Se estudian los diferentes actuadores utilizados en la actualidad y se muestran las bases teóricas para generar una trayectoria.

Capítulo 2. Caracterización matemática de un robot antropomórfico de 6 grados de libertad

2.1. Introducción

La cinemática del robot estudia el movimiento de este con respecto a un sistema de referencia sin considerar las fuerzas que intervienen. Así, la cinemática se interesa por la descripción analítica del movimiento espacial del robot como una función del tiempo y en particular por las relaciones entre la posición y la orientación del extremo final del robot con los valores que toman sus coordenadas articulares [18]. De lo mencionado anteriormente, surgen dos problemas cinemáticos: el problema de la cinemática directa e inversa, esto se puede apreciar conceptualmente en la Fig. 2.1.

Mencionado lo anterior, en este capítulo se espera estudiar analíticamente cinemática directa e inversa, estudiar la dinámica del manipulador a través de una simulación en Matlab, utilizando el método recursivo de Euler-Lagrange. Además, se presenta la dinámica de un actuador de corriente continua, ya que el diseño mecánico se crea a partir de este tipo de actuadores.

2.2. Cinemática directa

El problema cinemático consiste en determinar cuál es la posición y orientación del extremo final del robot, con respecto, a un sistema de coordenadas que se toma como referencia, conocidos los valores de las articulaciones y los parámetros geométricos de los elementos del robot. Denavit y Hartenberg propusieron un método sistemático para describir y representar la geometría espacial de los elementos de una cadena cinemática, y en particular de un robot, con respecto a un sistema de referencia fijo. Este método utiliza una matriz de transformación homogénea para describir la relación espacial entre dos elementos rígidos adyacentes, reduciéndose el problema cinemático directo a encontrar una matriz de transformación homogénea 4×4 que relacione la localización espacial del extremo del robot con respecto al sistema de coordenadas de su base [18].

Existen diferentes métodos para obtener la solución de la cinemática directa para diversos tipos de manipuladores o mecanismos, estos métodos se adaptan dependiendo directamente de los grados de libertad con los que cuenta el mecanismo. A continuación, se mencionan algunos de los

métodos existentes, ya que es importante mostrar que el método utilizado en esta memoria de título no es el único que existe. El primero es el método geométrico, que relaciona las variables en un solo plano XY, este método se utiliza para robots de 2 GDL, ya que desde 3 GDL hacia arriba los cálculos se complican. El siguiente método soluciona el problema cinemático a través de matrices de transformación homogénea y también se encuentra el método mediante el uso de cuaternios. En esta memoria se desarrolla el problema de la cinemática directa a través del algoritmo de Denavit-Hartenberg, este proceso se presenta en la siguiente subsección.

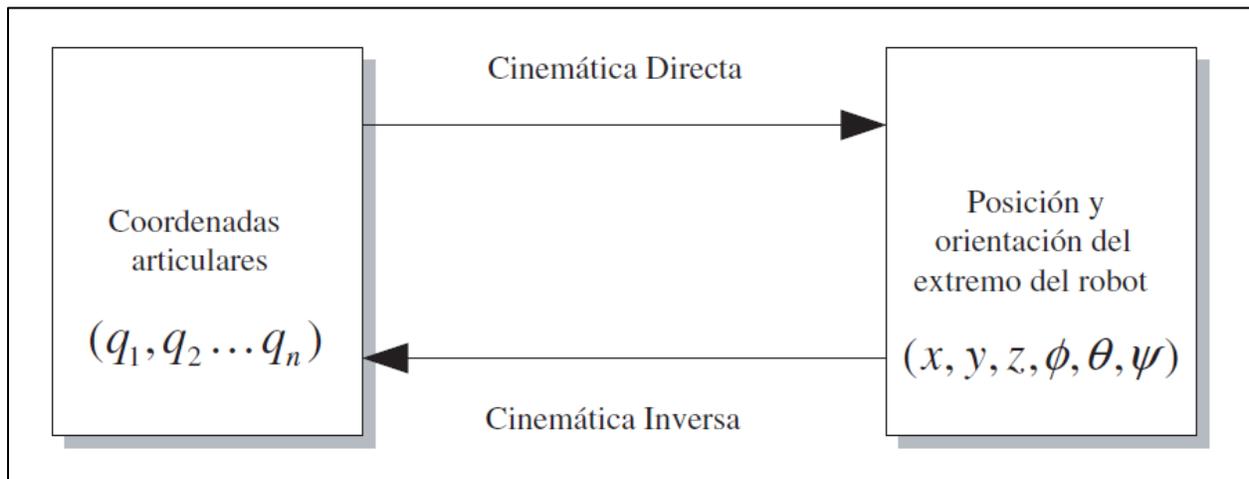


Fig. 2.1– Concepto de cinemática directa e inversa [18].

2.2.1. Algoritmo de Denavit-Hartenberg para la obtención del modelo cinemático directo

El método utilizado en el análisis de la cinemática directa del robot antropomórfico de 6 GDL es el algoritmo de Denavit-Hartenberg. D-H propusieron en 1955 un método matricial que establece la localización que debe tomar cada sistema de coordenadas $\{S_i\}$ ligado a cada eslabón i de una cadena articulada, para poder sistematizar la obtención de las ecuaciones cinemáticas de la cadena completa. Escogiendo los sistemas de coordenadas asociados a cada eslabón según la representación propuesta por D-H, será posible pasar de uno al siguiente mediante 4 transformaciones básicas que dependen exclusivamente de las características geométricas del eslabón [18].

Las 4 transformaciones básicas consisten en una sucesión de rotaciones y traslaciones que permiten relacionar el sistema de referencia del elemento $i - 1$ con el sistema del elemento i . Las

transformaciones en cuestión son las siguientes:

1. Rotación alrededor del eje z_{i-1} definido por un ángulo θ_i .
2. Traslación a lo largo de z_{i-1} definido por una distancia d_i .
3. Traslación a través de x_i definido por una distancia a_i .
4. Rotación alrededor del eje x_i definidor por un ángulo α_i .

Dado que el producto de las matrices no es conmutativo, las transformaciones se realizan en el orden presentado en la Ec. 2.1, de este modo se tiene que:

$${}^{i-1}A_i = \mathbf{Rotz}(\theta_i)\mathbf{T}(0,0,d_i)\mathbf{T}(a_i,0,0)\mathbf{Rotx}(\alpha_i) \quad (2.1)$$

luego, se realiza el producto entre matrices obteniendo:

$${}^{i-1}A_i = \begin{bmatrix} \cos \theta_i & -\sin \theta_i & 0 & 0 \\ \sin \theta_i & \cos \theta_i & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & d_i \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & a_i \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos \alpha_i & -\sin \alpha_i & 0 \\ 0 & \sin \alpha_i & \cos \alpha_i & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (2.2)$$

finalmente,

$${}^{i-1}A_i = \begin{bmatrix} \cos \theta_i & -\cos \alpha_i \sin \theta_i & \sin \alpha_i \sin \theta_i & a_i \cos \theta_i \\ \sin \theta_i & \cos \alpha_i \cos \theta_i & -\sin \alpha_i \cos \theta_i & a_i \sin \theta_i \\ 0 & \sin \alpha_i & \cos \alpha_i & d_i \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (2.3)$$

Donde θ_i , d_i , a_i y α_i son los parámetros de D-H del eslabón i . De este modo, basta con identificar los 4 parámetros mencionados para obtener las matrices ${}^{i-1}A_i$ y relacionar así, todos y cada uno de los eslabones del robot.

2.2.2. Análisis de cinemática directa de un robot antropomórfico de 6 GDL

En la Fig. 2.2 se puede observar la configuración básica de un manipulador antropomórfico de 6 GDL, así se realiza el análisis de la cinemática directa del robot antropomórfico utilizando el algoritmo de Denavit-Hartenberg.

El primer paso del algoritmo es enumerar los eslabones, denotando con un 1 al primer eslabón móvil de la cadena y acabando en n, para este caso n=6. Además, se enumera como eslabón 0 a la

base fija del robot. De la misma manera, se enumeran las articulaciones comenzando en 1 y terminando en n. La primera articulación corresponde al primer grado de libertad del manipulador.

El tercer paso consiste en localizar los ejes de rotación de cada articulación, se debe definir si la articulación es rotativa o prismática, para este caso solo existen articulaciones rotacionales. Lo mencionado anteriormente se puede apreciar en la Fig. 2.3.

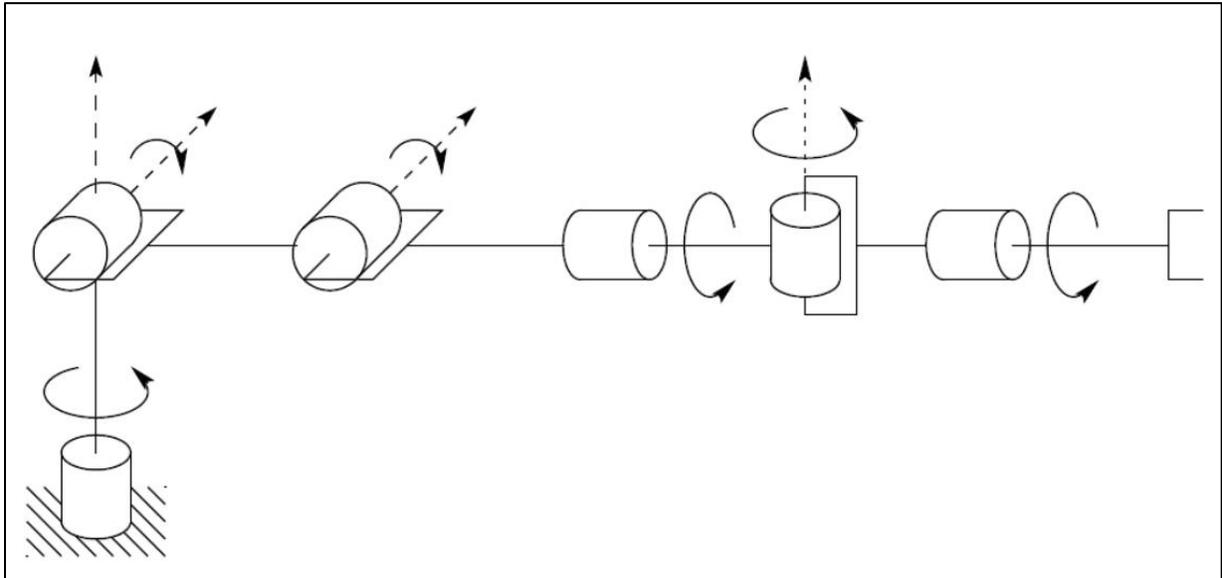


Fig. 2.2 – Configuración de un robot antropomórfico con pinza [41].

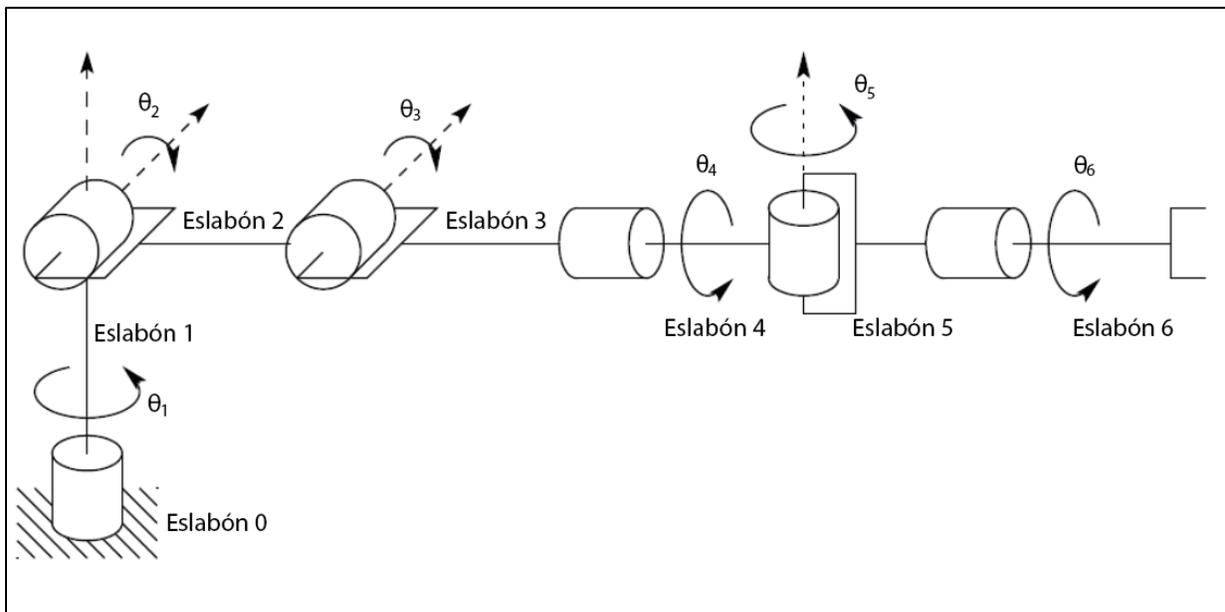


Fig. 2.3 – Configuración de eslabones y articulaciones según D-H.

El siguiente paso es situar el eje z_i sobre el eje de la articulación $i + 1$, desde $i = 0$ hasta $n = 1$. Luego se sitúa el origen del sistema de la base S_0 en cualquier punto del eje z_0 . Los ejes x_0 y y_0 se sitúan de modo que formen un sistema dextrógiro con z_0 . Para ubicar los siguientes ejes z_i para $i = 1$ hasta $n - 1$, se ubican en la intersección del eje z_i con la línea normal común a z_{i-1} y z_i . Finalmente, situar x_i en la línea normal común a z_{i-1} y z_i , de la misma manera, ubicar y_i de modo que forme un sistema dextrógiro con x_i y z_i . Estos pasos se pueden observar gráficamente en la Fig. 2.4.

A continuación, se presentan los 4 parámetros de Denavit-Hartenberg, estos son: $\theta_i, d_i, a_i, \alpha_i$. Estos parámetros dependen directamente de las características geométricas de cada eslabón y de las articulaciones que le unen con el anterior y siguiente. Cada parámetro representa una característica importante dentro del esquema presentado en la Fig. 2.4, en concreto θ_i es el ángulo que forman los ejes x_{i-1} y x_i medido en un plano perpendicular al eje z_{i-1} , utilizando la regla de la mano derecha. d_i es la distancia a lo largo del eje z_{i-1} desde el origen del sistema de coordenadas $(i - 1)$ -ésimo hasta la intersección del eje z_{i-1} con el eje x_i . a_i es la distancia a lo largo del eje x_i que va desde la intersección del eje z_{i-1} con el eje x_i hasta el origen del sistema i -ésimo, en el caso de las articulaciones giratorias. α_i es el ángulo de separación del eje z_{i-1} y el eje z_i , medido en un plano perpendicular al eje x_i , utilizando la regla de la mano derecha.

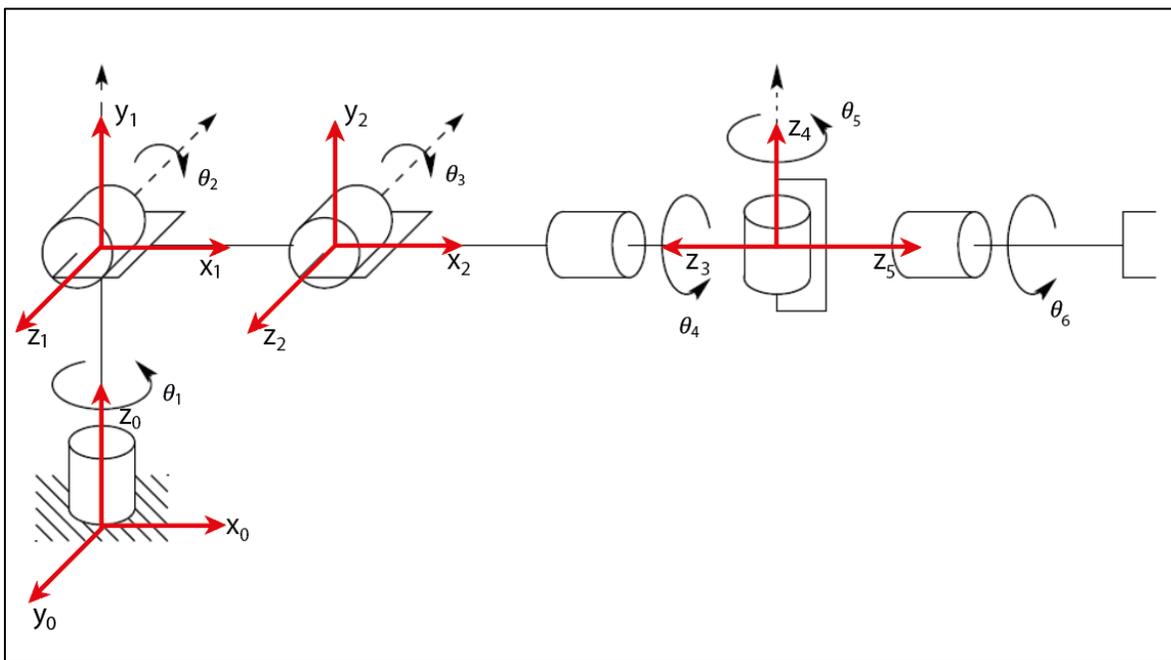


Fig. 2.4 – Ubicación de ejes en cada articulación del manipulador según D-H.

Luego de seguir los pasos anteriormente mencionados, se pueden obtener los parámetros de D-H para un robot antropomórfico de 6 GDL, estos parámetros se presentan en la Tabla 2.1. Además, en la Fig. 2.5 se puede observar más claramente cada componente del manipulador y que forman parte de los parámetros de Denavit-Hartenberg.

Tabla 2.1 – Parámetros de Denavit-Hartenberg para un manipulador de 6 GDL

Articulación	θ_i	d_i	a_i	α_i
q_1	θ_1^*	d_1	0	0
q_2	θ_2^*	0	a_2	0
q_3	θ_3^*	0	a_3	0
q_4	θ_4^*	0	0	-90°
q_5	θ_5^*	0	0	90°
q_6	θ_6^*	d_6	0	0

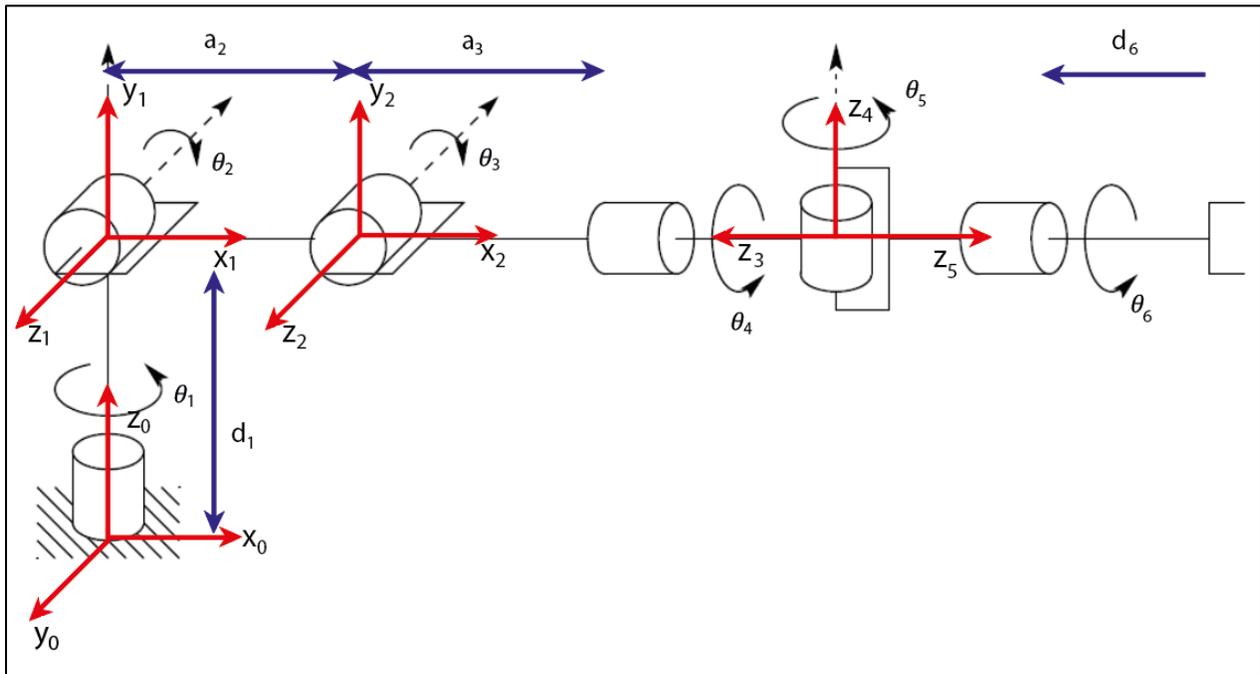


Fig. 2.5 – Características principales de un manipulador de 6 GDL según D-H.

A continuación, se expresan las matrices de transformación definidas en las ecuaciones (2.2) y (2.3),

$${}^0A_1 = \begin{bmatrix} \cos(\theta_1) & 0 & \text{sen}(\theta_1) & 0 \\ \text{sen}(\theta_1) & 0 & -\cos(\theta_1) & 0 \\ 0 & 1 & 0 & d_1 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (2.4)$$

$${}^1A_2 = \begin{bmatrix} \cos(\theta_2) & -\text{sen}(\theta_2) & 0 & a_2 \cos(\theta_2) \\ \text{sen}(\theta_2) & \cos(\theta_2) & 0 & a_2 \text{sen}(\theta_2) \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (2.5)$$

$${}^2A_3 = \begin{bmatrix} \cos(\theta_3) & -\text{sen}(\theta_3) & 0 & a_3 \cos(\theta_3) \\ \text{sen}(\theta_3) & \cos(\theta_3) & 0 & a_3 \text{sen}(\theta_3) \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (2.6)$$

$${}^3A_4 = \begin{bmatrix} \cos(\theta_4) & 0 & \text{sen}(\theta_4) & 0 \\ \text{sen}(\theta_4) & 0 & \cos(\theta_4) & 0 \\ 0 & -1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (2.7)$$

$${}^4A_5 = \begin{bmatrix} \cos(\theta_5) & 0 & \text{sen}(\theta_5) & 0 \\ \text{sen}(\theta_5) & 0 & -\cos(\theta_5) & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (2.8)$$

$${}^5A_6 = \begin{bmatrix} \cos(\theta_6) & -\text{sen}(\theta_6) & 0 & 0 \\ \text{sen}(\theta_6) & \cos(\theta_6) & 0 & 0 \\ 0 & 0 & 1 & d_6 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (2.9)$$

Así pues, se puede calcular la matriz T que indica la localización del sistema asociado al extremo del robot con respecto al sistema de referencia de la base del robot:

$$T = {}^0A_1 {}^1A_2 {}^2A_3 {}^3A_4 {}^4A_5 {}^5A_6 \quad (2.10)$$

$$T = \begin{bmatrix} n_x & s_x & a_x & p_x \\ n_y & s_y & a_y & p_y \\ n_z & s_z & a_z & p_z \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (2.11)$$

Finalmente, se presentan los valores para componente de la matriz presentada en la Ec. (2.11), estos valores se obtienen al multiplicar las matrices homogéneas obtenidas anteriormente, de esta manera se obtiene la cinemática directa de un robot antropomórfico de 6 GDL. Debido a que los resultados son extensos, se emplea una notación para disminuir las expresiones.

$$n_x = -C\theta_6[S\theta_1S\theta_5 - C(\theta_2 + \theta_3 + \theta_4)C\theta_1C\theta_5] - S(\theta_2 + \theta_3 + \theta_4)C\theta_1S\theta_6 \quad (2.12)$$

$$n_y = C\theta_6[C\theta_1S\theta_5 + C(\theta_2 + \theta_3 + \theta_4)C\theta_5S\theta_1] - S(\theta_2 + \theta_3 + \theta_4)S\theta_1S\theta_6 \quad (2.13)$$

$$n_z = C(\theta_2 + \theta_3 + \theta_4)S\theta_6 + S(\theta_2 + \theta_3 + \theta_4)C\theta_5C\theta_6 \quad (2.14)$$

$$s_x = S\theta_6[S\theta_1S\theta_5 - C(\theta_2 + \theta_3 + \theta_4)C\theta_1C\theta_5] - S(\theta_2 + \theta_3 + \theta_4)C\theta_1C\theta_6 \quad (2.15)$$

$$s_y = -S\theta_6[C\theta_1S\theta_5 + C(\theta_2 + \theta_3 + \theta_4)C\theta_5S\theta_1] - S(\theta_2 + \theta_3 + \theta_4)C\theta_6S\theta_1 \quad (2.16)$$

$$s_z = C(\theta_2 + \theta_3 + \theta_4)C\theta_6 - S(\theta_2 + \theta_3 + \theta_4)C\theta_5S\theta_6 \quad (2.17)$$

$$a_x = C\theta_5S\theta_1 + C(\theta_2 + \theta_3 + \theta_4)C\theta_1S\theta_5 \quad (2.18)$$

$$a_y = S\theta_5S\theta_1C(\theta_2 + \theta_3 + \theta_4) - C\theta_1C\theta_5 \quad (2.19)$$

$$a_z = S(\theta_2 + \theta_3 + \theta_4)S\theta_5 \quad (2.20)$$

$$p_x = d_6[C\theta_5S\theta_1 + C(\theta_2 + \theta_3 + \theta_4)C\theta_1S\theta_5] + a_2C\theta_1\theta_2 + a_3C\theta_1\theta_2\theta_3 - a_3C\theta_1S\theta_2\theta_3 \quad (2.21)$$

$$p_y = a_2C\theta_2S\theta_1 - d_6C\theta_2\theta_3S\theta_1 + a_3C\theta_2\theta_3S\theta_1 - a_3S\theta_1\theta_2\theta_3 + d_6C(\theta_2 + \theta_3 + \theta_4)S\theta_1\theta_5 \quad (2.22)$$

$$p_z = d_1 + a_3 S(\theta_2 + \theta_3) + a_2 S\theta_2 + d_6 S\theta_5 [C(\theta_2 + \theta_3) S\theta_4 + S(\theta_2 + \theta_3) C\theta_4] \quad (2.23)$$

2.3. Cinemática inversa

En la sección anterior se mostró como determinar la posición y orientación del extremo final del robot, con respecto, a un sistema de coordenadas que se toma como referencia. En esta sección se ocupa el problema inverso para encontrar los valores que deben adoptar las coordenadas articulares del robot para que su extremo se posicione y oriente según una determinada localización espacial. El problema cinemático directo de una manera sistemática a partir de la utilización de matrices de transformación homogéneas e independientemente de la configuración del robot, no ocurre lo mismo con el problema cinemático inverso, siendo el procedimiento de obtención de las ecuaciones fuertemente dependiente de la configuración del robot [18].

El desarrollo de este tipo de problema cinemático ofrece las siguientes ventajas:

1. En muchas aplicaciones, el problema cinemático inverso ha de resolverse en tiempo real (por ejemplo, en el seguimiento de una determinada trayectoria). Una solución de tipo iterativo no garantiza tener la solución en el momento adecuado.
2. Al contrario de lo que ocurría en el problema cinemático directo, con cierta frecuencia la solución del problema cinemático inverso no es única; existiendo diferentes $n - \text{uplas}$ $[q_1, \dots, q_n]^T$ que posicionan y orientan el extremo del robot del mismo modo. En estos casos una solución cerrada permite incluir determinadas reglas o restricciones que aseguren que la solución obtenida sea la más adecuada de entre las posibles.

Para este caso en específico, al ser un robot de 6 grados de libertad con capacidad de posicionar y orientar su extremo en el espacio, se utiliza el método de desacoplamiento cinemático, que permite resolver los 3 primeros grados de libertad de manera independiente a los 3 últimos grados de libertad. A los 3 primeros GDL se les denomina posicionamiento y a los 3 últimos GDL orientación, de esta manera, un problema de 6 GDL pasa a ser 2 problemas más simples con 3 GDL cada uno. En el siguiente apartado, se muestra el desarrollo de la cinemática inversa de un robot antropomórfico de 6 grados de libertad, utilizando el método geométrico.

2.3.1. Análisis de cinemática inversa de un robot antropomórfico de 6 GDL

Para calcular la cinemática inversa hay 6 parámetros que se deben encontrar, estos son: θ_1, θ_2 y θ_3 que serán los parámetros de posición del robot; θ_4, θ_5 y θ_6 que serán los parámetros de

orientación de la muñeca. Anteriormente se obtuvo la cinemática directa, obteniendo la matriz de cinemática directa deseada que se presenta en la siguiente ecuación:

$${}^6A_0 = \begin{bmatrix} {}^6R_0 & {}^6d_0 \\ 0 & 1 \end{bmatrix} \quad (2.24)$$

Esta matriz de cinemática directa deseada está formada por una submatriz de rotación 6R_0 t un vector de desplazamiento 6d_0 , además la submatriz de rotación se puede dividir en 3 vectores diferentes: n , s y a .

$${}^6A_0 = \begin{bmatrix} {}^6n_0 & {}^6s_0 & {}^6a_0 & {}^6d_0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (2.25)$$

El vector n indica como se encuentra la rotación del x_f con respecto al sistema base, el vector s indica la orientación del y_f con respecto al sistema base y el vector a indica la orientación del z_f con respecto al sistema base, entonces la Ec. (2.24) se puede representar de la siguiente forma:

$${}^6A_0 = \begin{bmatrix} {}^6n_{x_0} & {}^6s_{x_0} & {}^6a_{x_0} & {}^6x_0 \\ {}^6n_{y_0} & {}^6s_{y_0} & {}^6a_{y_0} & {}^6y_0 \\ {}^6n_{z_0} & {}^6s_{z_0} & {}^6a_{z_0} & {}^6z_0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (2.26)$$

Para entender el procedimiento es importante observar la Fig. 2.7, el análisis del problema cinemático inverso para un robot de 6 grados de libertad se puede reducir a resolver 3 vectores. El primer vector desde el sistema base al centro de la muñeca, en este caso, expresado de la forma ${}^m d_0$, el segundo vector desde el sistema base hasta el efector final del robot, representado por ${}^f d_0$ y el tercer vector parte en el centro de la muñeca con dirección al efector final, expresado por $d_6 {}^f a_0$. En forma analítica, según la Fig. 2.6 y lo planteado en la Ec. (2.26) se observa que para calcular el vector $d_6 {}^f a_0$ se necesita el valor de la distancia entre el centro de la muñeca y el punto del efector final multiplicado por un vector de desplazamiento en dirección del eje que une ambas partes del robot, por lo tanto, se utiliza el vector de desplazamiento expresado en la Ec. (2.25): 6a_0 .

Lo mencionado anteriormente se conoce como: desacoplamiento cinemático. Por lo tanto, según lo explicado anteriormente, la Fig. 2.7 se presenta el problema de resolver la cinemática inversa para un robot antropomórfico de 3 GDL, de esta manera se resuelve el primer problema cinemático inverso que corresponde a la posición del manipulador.

Para comenzar el cálculo de θ_1 , θ_2 y θ_3 se genera una proyección en el plano zr, esto se puede apreciar en la Fig. 2.8, donde se resalta el plano en el cual se hará la proyección. El resultado de la proyección en el plano zr, se presenta en la Fig. 2.9.

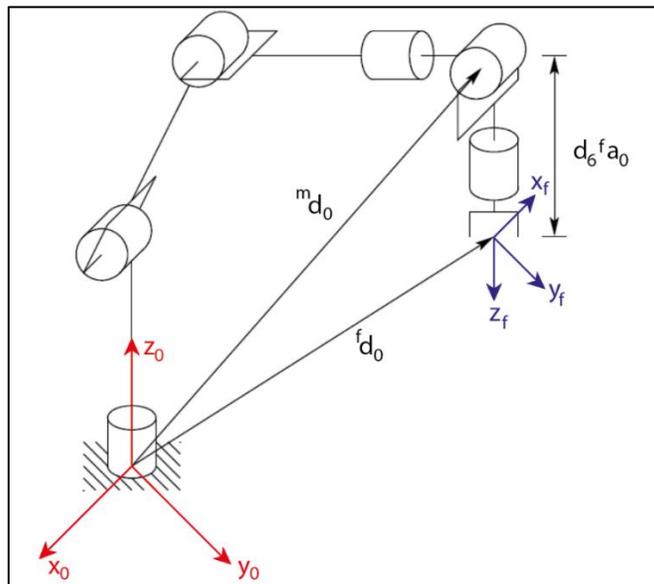


Fig. 2.6 – Problema cinemático inverso para un robot antropomórfico de 6 GDL.

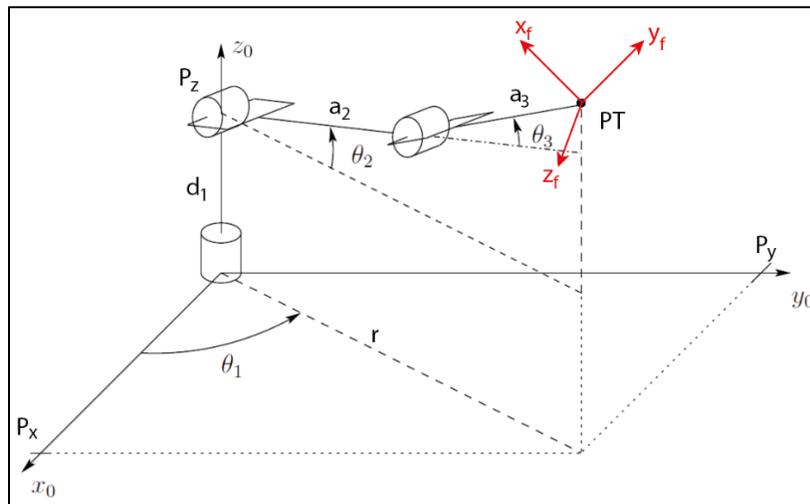


Fig. 2.7 –Desacoplamiento cinemático para un robot antropomórfico de 6 GDL.

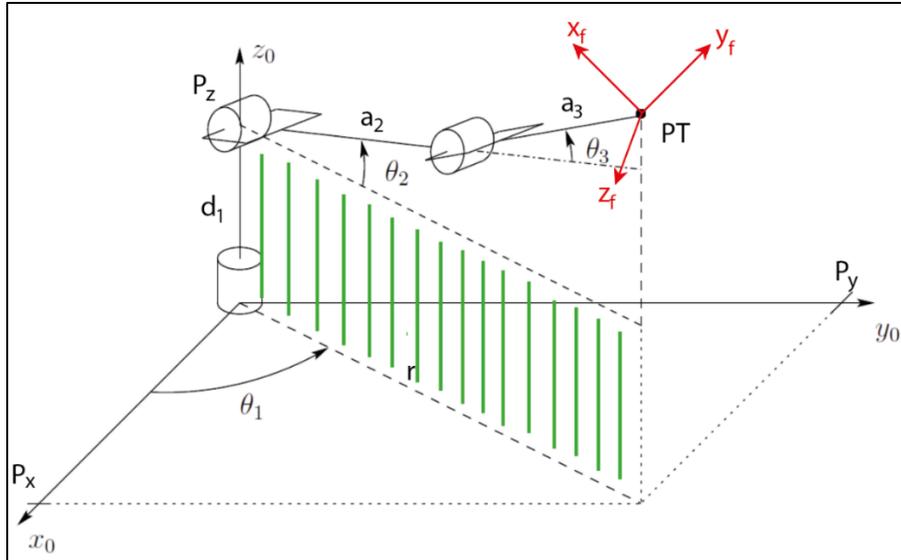


Fig. 2.8 –Proyección del problema cinemático sobre el plano zr.

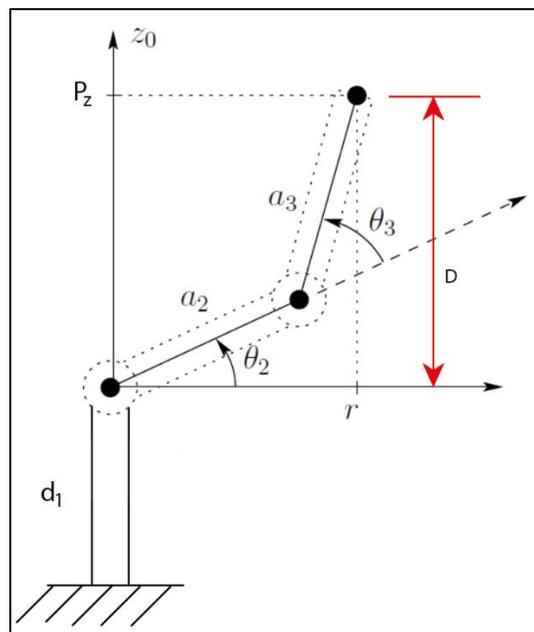


Fig. 2.9 – Resultado de la proyección del problema cinemático sobre el plano zr.

Luego de realizar la proyección, desde la Fig. 2.9 se extrae la información necesaria para obtener los valores de θ_2 y θ_3 . Además, se aprecia que luego de la proyección el problema cinemático inverso de 3 GDL se redujo a resolver la misma situación para un robot de 2 GDL, conocido generalmente como: robot plano.

Para comenzar el análisis se traza una línea recta que une la articulación 1 con la articulación 3, llamada H, así se crea un triángulo rectángulo. La Fig. 2.10 muestra lo anteriormente mencionado, de esta manera se puede plantear lo siguiente.

$$H^2 = D^2 + r^2 \quad (2.27)$$

el valor D, se obtiene de la siguiente manera:

$$D = P_z - d_1 \quad (2.28)$$

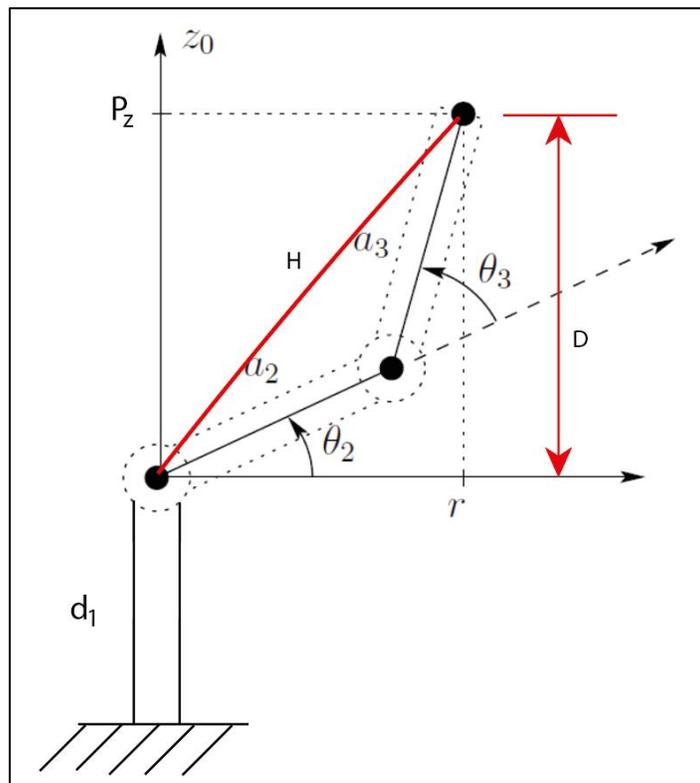


Fig. 2.10 – Trazo de línea recta H.

El valor del cateto r, se obtiene a través de una proyección sobre el plano xy, de la misma manera que en la proyección en zr, se obtiene un triángulo rectángulo. La hipotenusa estará dada por r y sus catetos serán: P_x y P_y . Entonces se plantea lo siguiente:

$$r = \sqrt{P_x^2 + P_y^2} \quad (2.29)$$

al reemplazar los valores de r y D en la Ec. (2.27),

$$H = \sqrt{P_x^2 + P_y^2 + (P_z - d_1)^2} \quad (2.30)$$

Luego, en la Fig. 2.11 se presenta el triángulo que se obtiene al calcular H. Entonces el interés se centra en plantear el valor de μ , obteniendo:

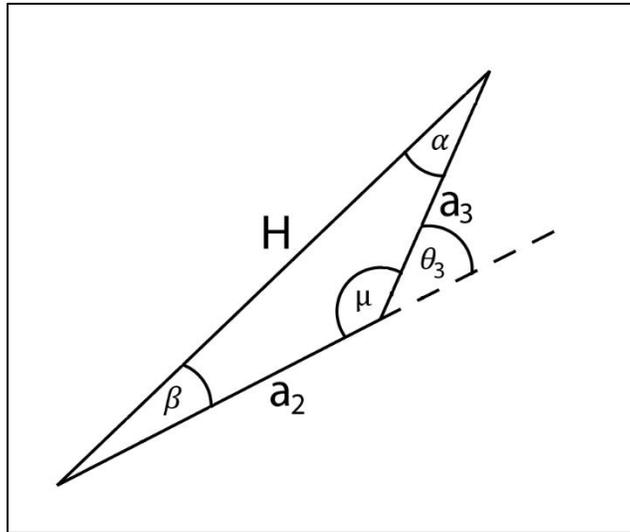


Fig. 2.11 – Triángulo del cual se obtiene el valor de θ_3 .

$$\mu = 180^\circ - \theta_3 \quad (2.31)$$

Ahora si se aplica la ley del coseno en el triángulo de la Fig. 2.11, se obtiene:

$$c^2 = a^2 + b^2 - 2abc\cos(\mu) \quad (2.32)$$

despejando el $\cos(\mu)$,

$$-\cos(\mu) = \frac{c^2 - a^2 - b^2}{2ab} \quad (2.33)$$

al reemplazar los valores obtenidos anteriormente, en la Ec. (2.33),

$$-\cos(\mu) = \frac{P_x^2 + P_y^2 + (P_z - d_1)^2 - a_2^2 - a_3^2}{2a_2a_3} \quad (2.34)$$

sustituyendo el valor de μ ,

$$-\cos(180 - \theta_3) = \frac{P_x^2 + P_y^2 + (P_z - d_1)^2 - a_2^2 - a_3^2}{2a_2a_3} \quad (2.35)$$

utilizando la siguiente identidad trigonométrica, se podrá despejar el valor de θ_3 ,

$$\cos(\pi - \alpha) = -\cos(\alpha) \quad (2.36)$$

reemplazando,

$$\cos(\theta_3) = \frac{P_x^2 + P_y^2 + (P_z - d_1)^2 - a_2^2 - a_3^2}{2a_2a_3} \quad (2.37)$$

sea la función de la tangente de la siguiente forma:

$$\tan(\theta_3) = \frac{\sin(\theta_3)}{\cos(\theta_3)} \quad (2.38)$$

considerando la identidad trigonométrica $\sin^2(\theta_3) + \cos^2(\theta_3) = 1$, se despeja el valor de $\sin(\theta_3)$,

$$\sin(\theta_3) = \sqrt{1 - \cos^2(\theta_3)} \quad (2.39)$$

por lo tanto,

$$\tan(\theta_3) = \frac{\sqrt{1 - \cos^2(\theta_3)}}{\cos(\theta_3)} \quad (2.40)$$

finalmente, si se reemplaza la Ec. (2.37) en la Ec. (2.40), se obtiene

$$\tan(\theta_3) = \frac{\sqrt{1 - \left(\frac{P_x^2 + P_y^2 + (P_z - d_1)^2 - a_2^2 - a_3^2}{2a_2a_3} \right)^2}}{\frac{P_x^2 + P_y^2 + (P_z - d_1)^2 - a_2^2 - a_3^2}{2a_2a_3}} \quad (2.41)$$

aplicando atan, se obtiene el valor de θ_3 , entonces:

$$\theta_3 = \text{atan} \left(\frac{\sqrt{1 - \left(\frac{P_x^2 + P_y^2 + (P_z - d_1)^2 - a_2^2 - a_3^2}{2a_2a_3} \right)^2}}{\frac{P_x^2 + P_y^2 + (P_z - d_1)^2 - a_2^2 - a_3^2}{2a_2a_3}} \right) \quad (2.42)$$

usando una variable auxiliar se puede expresar el resultado de la siguiente manera,

$$C = \frac{P_x^2 + P_y^2 + (P_z - d_1)^2 - a_2^2 - a_3^2}{2a_2a_3} \quad (2.43)$$

$$\theta_3 = \text{atan} \left(\frac{\sqrt{1 - C^2}}{C} \right) \quad (2.44)$$

continuando con θ_2 , se obtiene el triángulo presentado en la Fig. 2.10. Entonces el valor de θ_2 estará dado por:

$$\theta_2 = \alpha_1 - \beta \quad (2.45)$$

el triángulo expresa lo siguiente,

$$\theta_2 = \text{atan} \left(\frac{\text{cateto}_{\text{opuesto}}}{\text{cateto}_{\text{adyacente}}} \right) \quad (2.46)$$

reemplazando,

$$\theta_2 = \text{atan} \left(\frac{(P_z - d_1)}{\sqrt{P_x^2 + P_y^2}} \right) - \text{atan} \left(\frac{a_3 \cos(\theta_3)}{a_2 + a_3 \sin(\theta_3)} \right) \quad (2.47)$$

Se puede observar de la Ec. (2.47) que el valor de θ_2 dependerá directamente del valor de θ_3 , además θ_3 puede tener un valor positivo y negativo, dependiendo si se decide usar una configuración codo arriba o codo abajo.

Para obtener el valor de θ_1 se tiene el triángulo rectángulo de la proyección sobre el eje xy, de esta manera se obtiene la siguiente expresión:

$$\theta_1 = \text{atan}\left(\frac{P_y}{P_x}\right) \quad (2.48)$$

Para resolver la segunda parte del problema cinemático inverso, se plantea lo siguiente:

$${}^6R_3 = ({}^3R_0)^T {}^6R_0 \quad (2.49)$$

Al analizar lo planteado en la Ec. (2.49) se observa que el lado izquierdo de la ecuación, son las matrices de rotación desde 3 a 6, que dependen directamente de los valores de θ_4 , θ_5 y θ_6 , valores que se desea encontrar en el desarrollo de la ecuación. El lado derecho de la ecuación depende de las matrices de rotación del sistema base hasta el centro de la muñeca transpuesto multiplicado por la cinemática directa deseada, este proceso se realizó en el análisis del problema cinemático directo. Dicho esto, y con ayuda del software Matlab para obtener los cálculos, se obtienen los valores de θ_4 , θ_5 y θ_6 . El código que se realizó para calcular estas variables se adjunta en el Anexo B - Códigos.

Para θ_5 , se obtiene la siguiente relación

$$\cos(\theta_5) = R_{13f} \sin(\theta_1) - R_{23f} \cos(\theta_1) \quad (2.50)$$

considerando la identidad trigonométrica $\sin^2(\theta_5) + \cos^2(\theta_5) = 1$, se despeja el valor de $\sin(\theta_5)$,

$$\sin(\theta_5) = \sqrt{1 - \cos^2(\theta_5)} \quad (2.51)$$

sea,

$$K := \cos(\theta_5) = R_{13f} \sin(\theta_1) - R_{23f} \cos(\theta_1) \quad (2.52)$$

reemplazando,

$$\tan(\theta_5) = \frac{\sqrt{1 - \cos^2(\theta_5)}}{\cos(\theta_5)} \quad (2.53)$$

finalmente,

$$\theta_5 = \frac{\sqrt{1 - K^2}}{K} \quad (2.54)$$

para θ_4 se obtiene la siguiente relación,

$$\frac{\cos(\theta_4)\sin(\theta_5)}{\sin(\theta_4)\sin(\theta_5)} = \frac{R_{33f}S(\theta_2 + \theta_3) + R_{13f}C(\theta_2 + \theta_3)C(\theta_1) + R_{23f}S(\theta_2 + \theta_3)S(\theta_1)}{R_{33f}C(\theta_2 + \theta_3) - R_{13f}S(\theta_2 + \theta_3)C(\theta_1) - R_{23f}S(\theta_2 + \theta_3)S(\theta_1)} \quad (2.55)$$

finalmente, al reducir y aplicar $\tan(\theta_4)$ se obtiene,

$$\theta_4 = \text{atan} \left(\frac{R_{33f}S(\theta_2 + \theta_3) + R_{13f}C(\theta_2 + \theta_3)C(\theta_1) + R_{23f}S(\theta_2 + \theta_3)S(\theta_1)}{R_{33f}C(\theta_2 + \theta_3) - R_{13f}S(\theta_2 + \theta_3)C(\theta_1) - R_{23f}S(\theta_2 + \theta_3)S(\theta_1)} \right) \quad (2.56)$$

para θ_6 ,

$$\frac{\sin(\theta_5)\sin(\theta_6)}{\cos(\theta_6)\sin(\theta_5)} = \frac{R_{12f}\sin(\theta_1) - R_{22f}\cos(\theta_1)}{R_{21f}\sin(\theta_1) - R_{11f}\sin(\theta_1)} \quad (2.57)$$

reduciendo,

$$\frac{\sin(\theta_6)}{\cos(\theta_6)} = \frac{R_{12f}\sin(\theta_1) - R_{22f}\cos(\theta_1)}{R_{21f}\sin(\theta_1) - R_{11f}\sin(\theta_1)} \quad (2.58)$$

aplicando $\tan(\theta_6)$,

$$\theta_6 = \text{atan} \left(\frac{R_{12f}\sin(\theta_1) - R_{22f}\cos(\theta_1)}{R_{21f}\sin(\theta_1) - R_{11f}\sin(\theta_1)} \right) \quad (2.59)$$

De esta manera, se resuelve el problema cinemático inverso para un robot antropomórfico de 6 grados de libertad. Además, para finalizar se expresa la posición de la muñeca, que estará dada por:

$$x_m = x_f - d_6 R_{13f} \quad (2.60)$$

$$y_m = y_f - d_6 R_{23f} \quad (2.61)$$

$$z_m = z_f - d_6 R_{33f} \quad (2.62)$$

2.4. Dinámica

En esta subsección se expresa la dinámica para un robot antropomórfico de 6 grados de libertad. Para llevar a cabo este proceso, se utiliza el método de Euler-Lagrange. Los cálculos para realizar este procedimiento son bastante extensos, por lo que se utiliza el software Matlab, de esta manera se puede hacer un cálculo iterativo sin problemas. Así, se podrá encontrar el valor de cada torque asociado a cada coordenada articular y también el modelo dinámico que rige el movimiento de este robot antropomórfico.

En esta memoria de título solo se expresará y mostrará como se puede calcular la dinámica para este tipo de manipulador, ya que para realizar las simulaciones se utilizará la dinámica asociada a los actuadores, en este caso, motores CC. Esto se puede realizar ya que se utilizan transmisiones 1:5, de esta manera se puede obviar la dinámica del robot y usar la dinámica de los actuadores para realizar alguna estrategia de control.

2.4.1. Análisis de la dinámica de un robot antropomórfico de 6 GDL

La dinámica se ocupa de la relación entre las fuerzas que actúan sobre un cuerpo y el movimiento que en él se origina. Por tanto, el modelo dinámico de un robot tiene por objetivo conocer la relación entre el movimiento del robot y las fuerzas implicadas en el mismo [18].

La obtención de este modelo para mecanismos de uno o dos grados de libertad no es excesivamente compleja, pero a medida que el número de grados de libertad aumenta, el planteamiento y obtención del modelo dinámico se complica enormemente. Por este motivo no siempre es posible obtener un modelo dinámico expresado de una forma cerrada, esto es, mediante una serie de ecuaciones, normalmente de tipo diferencial de 2° orden, cuya integración permita conocer qué movimiento surge al aplicar unas fuerzas o qué fuerzas hay que aplicar para obtener un movimiento determinado. El modelo dinámico debe ser resuelto entonces de manera iterativa mediante la utilización de un procedimiento numérico [18].

Siguiendo con lo anterior, el primer paso del método Euler-Lagrange es calcular el Lagrangiano, expresado por:

$$\mathcal{L}(q, \dot{q}) = E_K(q, \dot{q}) - U(q) \quad (2.63)$$

El Lagrangiano es la diferencia entre la energía cinética y la energía potencial. La energía cinética se puede obtener de la siguiente manera:

$$E_K = \frac{1}{2} \dot{q}^T M(q) \dot{q} \quad (2.64)$$

Donde, $M(q)$ es la matriz de inercia o masa, se puede expresar:

$$M(q) = \sum_{i=1}^n m_i J_v^{iT} J_v^i + J_\omega^{iT} R_i \tilde{I}_i R_i^T J_\omega^i \quad (2.65)$$

Donde, \tilde{I}_i es el tensor de inercia del eslabón i en el sistema de referencia del eslabón i , este valor es constante. J_v^i es el Jacobiano de velocidad lineal del centro de masa del eslabón i , J_ω^i es el Jacobiano de velocidad angular del eslabón i , m_i es la masa del eslabón i y R_i es la matriz de rotación del sistema del eslabón i con respecto al sistema base.

La energía potencial se expresa de la siguiente manera,

$$U = - \sum_{i=1}^n m_i g_0^T P_{ci} \quad (2.66)$$

Donde, g_0 es el vector de aceleración de gravedad y P_{ci} es la posición del centro de masa del eslabón i . Luego, al reemplazar los valores de la energía cinética y potencial en el Lagrangiano se obtiene:

$$\mathcal{L}(q, \dot{q}) = \frac{1}{2} \dot{q}^T M(q) \dot{q} + \sum_{i=1}^n m_i g_0^T P_{ci} \quad (2.67)$$

De esta manera, para obtener las ecuaciones de movimiento Euler-Lagrange plantea lo siguiente,

$$\frac{d}{dt} \left(\frac{\partial \mathcal{L}}{\partial \dot{q}} \right)^T - \left(\frac{\partial \mathcal{L}}{\partial q} \right)^T = \tau \quad (2.68)$$

Si se reemplaza el valor del Lagrangiano y resolviendo se obtiene la siguiente expresión,

$$M(q) \ddot{q} + \left(\dot{M}(q) - \frac{1}{2} \dot{q}^T \left(\frac{\partial M(q)}{\partial q} \right)^T \right) \dot{q} + \left(- \sum_{i=1}^n J_v^{iT}(q) m_i g_0 \right) = \tau_{total} \quad (2.69)$$

Ahora para expresar el modelo dinámico se tiene que,

$$C(q, \dot{q}) = \left(\dot{M}(q) - \frac{1}{2} \dot{q}^T \left(\frac{\partial M(q)}{\partial q} \right)^T \right) \dot{q} \quad (2.70)$$

$$g(q) = \left(- \sum_{i=1}^n J_v^{iT}(q) m_i g_0 \right) \quad (2.71)$$

Donde, $C(q, \dot{q})$ es la expresión de Coriolis o fuerzas centrífugas y $g(q)$ es la expresión de la gravedad que actúa en cada eslabón del manipulador.

Finalmente, el modelo dinámico se expresa en la Ec. (2.72):

$$M(q)\ddot{q} + C(q, \dot{q})\dot{q} + g(q) = \tau_{total} \quad (2.72)$$

Donde, q expresa las coordenadas articulares generalizadas, $M(q)$ es la matriz de masa o inercia, $C(q, \dot{q})$ expresa las fuerzas centrífugas y Coriolis, $g(q)$ representa las fuerzas gravitacionales debidas al peso de cada eslabón i y τ_{total} son las fuerzas generalizadas totales no conservativas.

Utilizando Matlab se genera un código iterativo que permite calcular la dinámica para este manipulador de 6 GDL de manera simbólica, para que no sea tan extenso este ítem se presenta el código en el Anexo B – Códigos. Así se puede corroborar los cálculos realizados y se puede utilizar el código para un manipulador con una mayor o menor cantidad de grados de libertad.

Es importante mencionar, que el procedimiento anteriormente descrito para el desarrollo de la dinámica es el proceso analítico matemático, no se muestran resultados ya que se presentarán en el video de las simulaciones [64]. El código trabaja lo anteriormente descrito de manera iterativa para las 6 articulaciones que contiene el robot antropomórfico.

2.4.2. Análisis de la dinámica de los actuadores de un manipulador

Para realizar el análisis se considera un actuador de corriente continua de excitación separada presentado en la Fig. 2.12, así el estudio se divide en dos partes, análisis a la parte eléctrica y a la parte mecánica del motor.

En la parte eléctrica del motor se aplica la ley de los voltajes de Kirchhoff, método más conocido como “mallas”, a través del método se obtiene la siguiente ecuación:

$$V_a = V_{R_a} + V_{L_a} + e_a \quad (2.73)$$

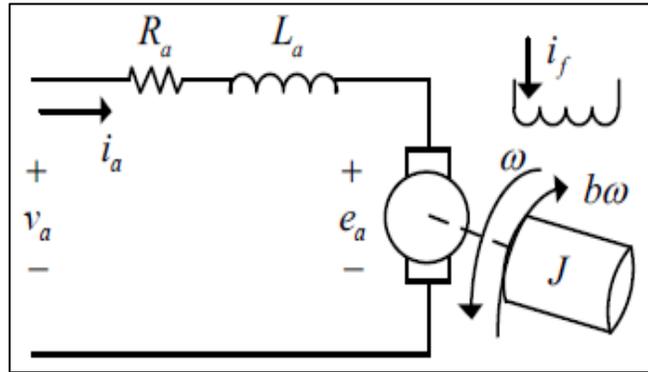


Fig. 2.12 – Motor de corriente continua de excitación separada.

Continuando, se obtienen las siguientes relaciones: $V_{L_a} = L_a \frac{di_a}{dt}$, $V_{R_a} = R_a i_a$ y $e_a = k_\phi \omega$. Si se reemplazan estas relaciones en la Ec. (2.73) se obtiene:

$$V_a = L_a \frac{di_a}{dt} + R_a i_a + k_\phi \omega \quad (2.74)$$

finalmente, al despejar la derivada de la corriente, se obtiene la primera ecuación diferencial que rige el comportamiento dinámico de un motor CC.

$$\frac{di_a}{dt} = \frac{1}{L_a} (V_a - R_a i_a - k_\phi \omega) \quad (2.75)$$

Para la parte mecánica, se aplica la primera ley de Newton que indica:

$$J\alpha = \sum \tau \quad (2.76)$$

donde, J es el momento de inercia del motor, α es la aceleración angular y $\sum \tau$ es la sumatoria de torques que actúan en el motor, obteniendo lo siguiente:

$$J\alpha = \tau_{el\acute{e}ctrico} - \tau_{roce} - T_{carga} \quad (2.77)$$

Al igual que en la parte eléctrica, se utilizan las siguientes relaciones: $\tau_{el\acute{e}ctrico} = T_e = k_\phi i_a$, $\tau_{roce} = T_r = b\omega$ y $\tau_{carga} = T_L$.

Finalmente, reemplazando las relaciones en la Ec. (2.77) y reduciendo, se obtiene la segunda ecuación dinámica que rige el comportamiento de un motor CC.

$$\frac{d\omega}{dt} = \frac{1}{J} (k_{\phi} i_a - b\omega - T_L) \quad (2.78)$$

Las dinámicas de los actuadores no se utilizarán para verificar el movimiento del manipulador a través de la generación de la trayectoria, pero se asume que los actuadores del diseño mecánico son ideales. Por lo tanto, la trayectoria ingresará directamente al diseño mecánico sin un control realimentado, las posiciones de cada articulación cambiarán con respecto al tiempo y se estudiarán a lazo abierto.

2.5. Resumen Capítulo 2

El capítulo contiene la caracterización completa para un robot antropomórfico de 6 GDL, las cinemáticas y dinámicas que actúan sobre este tipo de manipuladores. Estos cálculos se podrán corroborar en el capítulo 4 a través de los resultados obtenidos de las simulaciones realizadas. Es importante mencionar que, si desea realizar algún tipo de control, ya sea cinemático o dinámico, los cálculos realizados anteriormente son necesarios, sin estos datos no se puede realizar un control sobre las articulaciones del robot.

Capítulo 3. Diseño de un robot antropomórfico de 6 grados de libertad

3.1. Introducción

En este capítulo se muestra el diseño mecánico basado en la configuración de un robot antropomórfico. Este tipo de robot cuenta con 6 GDL y su diseño tiene una semejanza con el brazo humano, por lo tanto, el manipulador se puede subdividir en las siguientes partes: base, cadera, brazo, antebrazo, muñeca y pinza.

El diseño se realiza en Inventor Profesional 2020, el software permite crear piezas independientes y ensamblarlas para obtener un modelo final. Además, tiene un complemento que permite exportar el modelo CAD al software Matlab para generar un estudio cinemático o dinámico sobre el robot diseñado.

Finalmente, se genera un cálculo de torques, estos serán necesarios para escoger los actuadores que irán en las articulaciones y además se escoge el tipo de transmisión que se puede utilizar. En este caso, se utilizará una transmisión 1:5, ya que al usar una relación desde 1:5 hacia arriba, se puede realizar un control directamente a los actuadores del robot sin necesidad de utilizar la dinámica del manipulador, asumiendo además que la velocidad del manipulador será lo suficientemente baja para que las fuerzas de Coriolis y centrípeta sean despreciables.

3.2. Cálculo de torques

El concepto de torque está ligado al movimiento de rotación que experimenta un cuerpo rígido bajo fuerzas externas [42]. En la Fig. 3.1 se aprecian los puntos donde actuarán los torques y en la Fig. 3.2 se observan las distancias entre cada punto. El estudio se llevará a cabo de manera estática, de esta manera se pueden obtener valores aproximados de los torques y ayudará en una elección de los actuadores del manipulador. Antes de comenzar los cálculos, se presenta la Tabla 3.1, con parámetros importantes para contrastar los resultados.

Según lo mencionado anteriormente, si se calcula el torque en el punto A de la Fig. 3.1, se obtiene el valor del primer torque:

Tabla 3.1 – Parámetros para los torques

Parámetro	Valor
m_{carga}	0.5 [kg]
m_{pinza}	0.03 [kg]
$m_{muñeca}$	0.288 [kg]
$m_{antebrazo}$	1.130 [kg]
m_{brazo}	1.816 [kg]
d_{AF}	0.12293 [m]
d_{BA}	0.089 [m]
d_{CB}	0.2 [m]
d_{DC}	0.25 [m]
g	$9.8 \frac{m}{s^2}$

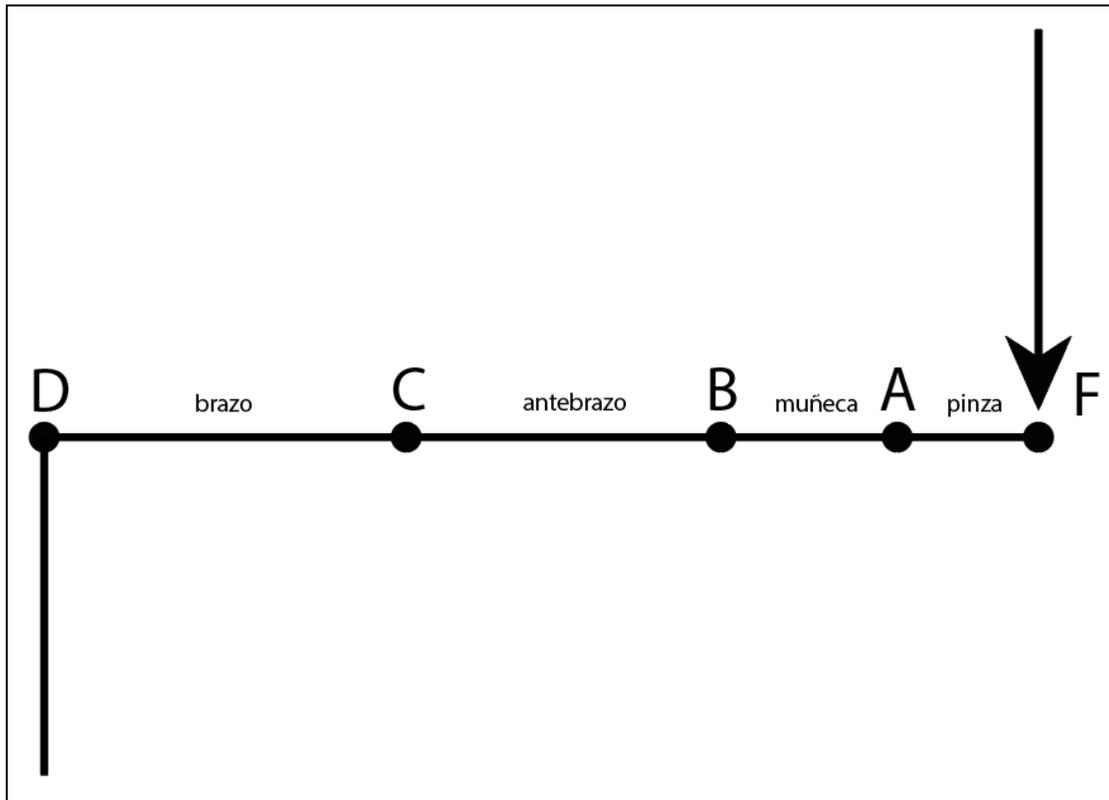


Fig. 3.1 – Planteamiento de torques que actúan en el robot.

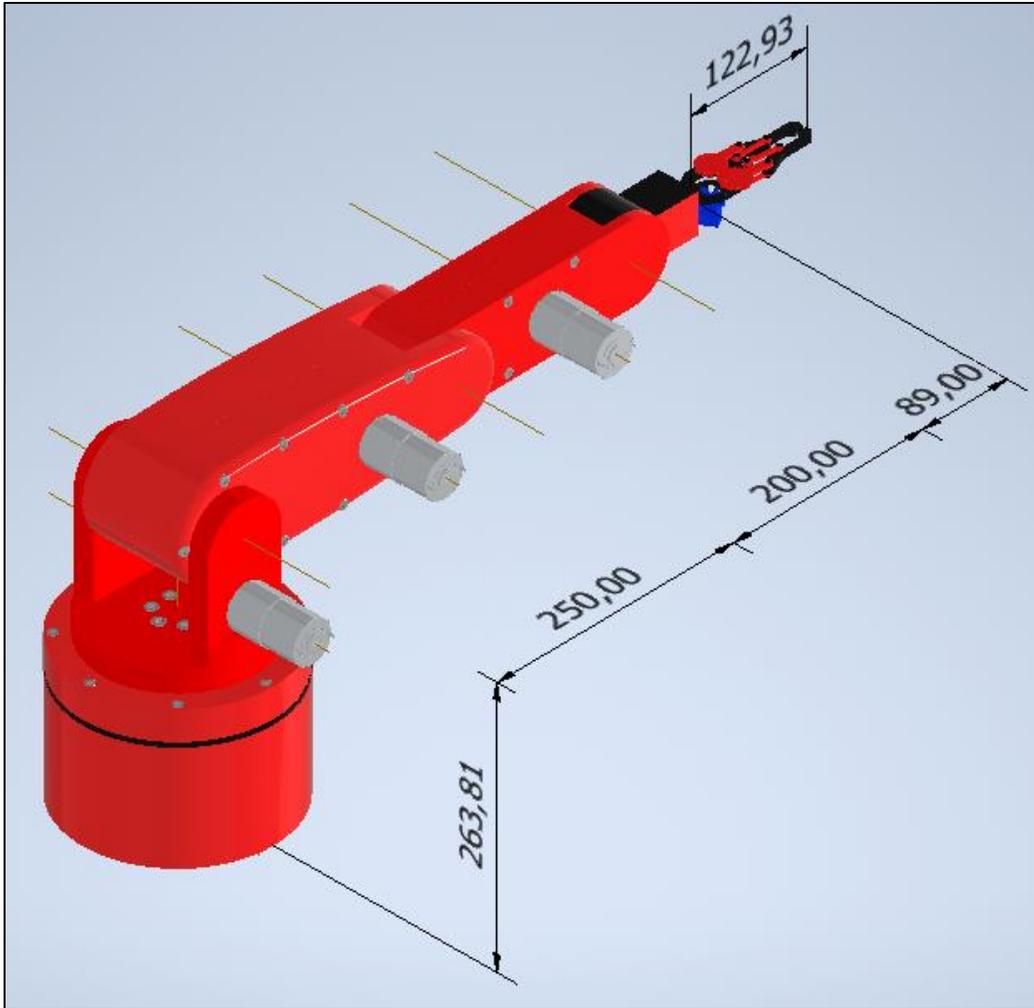


Fig. 3.2 – Distancias entre torques.

$$\sum \tau = Fd \quad (3.1)$$

$$\sum \tau_A = -Fd_{AF} - W_1 \frac{d_{AF}}{2} \quad (3.2)$$

$$\sum \tau_A = -m_{carga}gd_{AF} - W_1 \frac{d_{AF}}{2} \quad (3.3)$$

$$\sum \tau_A = -m_{carga}d_{AF}g - m_{pinza} \frac{d_{AF}}{2} g \quad (3.4)$$

reemplazando parámetros de la Tabla 3.1, se obtiene.

$$\tau_A = -0.62 [Nm] \quad (3.5)$$

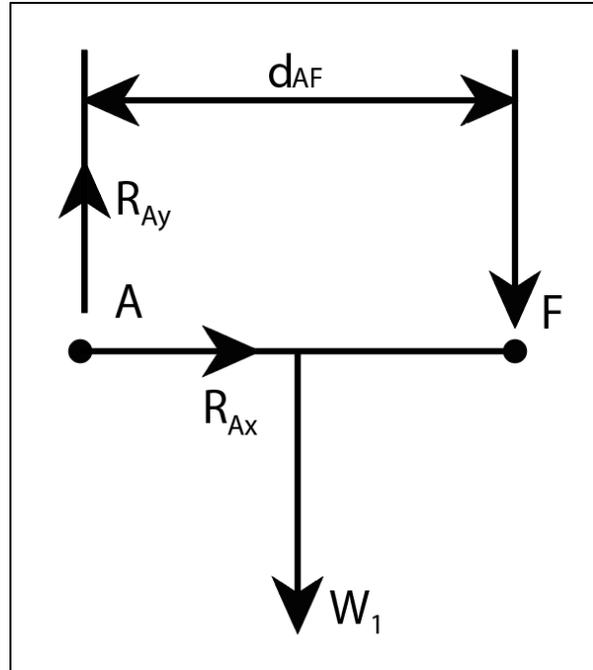


Fig. 3.3 – Torque en el punto A.

Es importante calcular las fuerzas que actúan sobre el punto A, porque para los siguientes torques será necesario conocer el valor de la reacción R_{Ay} .

$$\sum F_x = R_{Ax} = 0 \quad (3.6)$$

$$\sum F_y = R_{Ay} - W_1 - F = 0 \quad (3.7)$$

despejando R_{Ay} en la Ec. (3.7) se obtiene,

$$R_{Ay} = W_1 - F \quad (3.8)$$

$$R_{Ay} = m_{pinza}g - m_{carga}g \quad (3.9)$$

Para continuar se realiza un estudio de fuerza y torques en el punto B, las fuerzas y distancias que actúan en este punto son representadas en la Fig. 3.4.

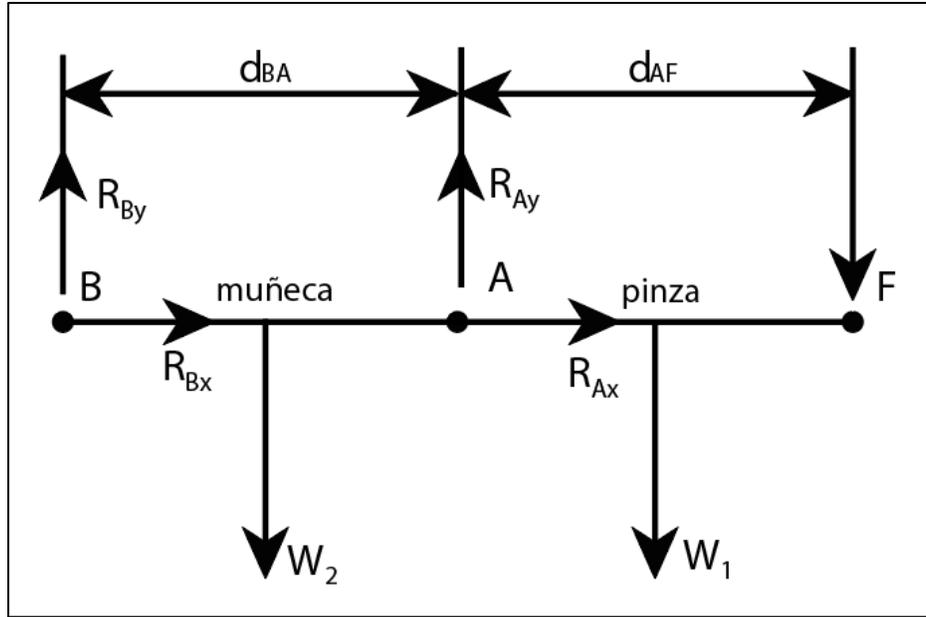


Fig. 3.4 – Torque en el punto B.

Las sumatorias de fuerzas de los ejes x e y, se presentan a continuación:

$$\sum F_x = R_{Ax} + R_{Bx} = 0 \quad (3.10)$$

$$\sum F_y = R_{Ay} + R_{By} - W_1 - W_2 - F = 0 \quad (3.11)$$

$$R_{By} = W_1 + W_2 + F - R_{Ay} \quad (3.12)$$

reemplazando el valor de R_{Ay} en la Ec. (3.12) se obtiene:

$$R_{By} = W_2 + 2F \quad (3.13)$$

ahora al estudiar el torque en el punto B, mostrado en la Fig. 3.4, resulta:

$$\sum \tau_B = R_{Ay}d_{BA} - W_2 \frac{d_{BA}}{2} - F(d_{AF} + d_{BA}) - W_1 \left(\frac{d_{AF}}{2} + d_{BA} \right) \quad (3.14)$$

reemplazando los parámetros se obtiene el torque del punto B.

$$\tau_B = -1.144 \text{ [Nm]} \quad (3.15)$$

Para el punto C se desarrolla el mismo procedimiento anterior guiado por la Fig. 3.5,

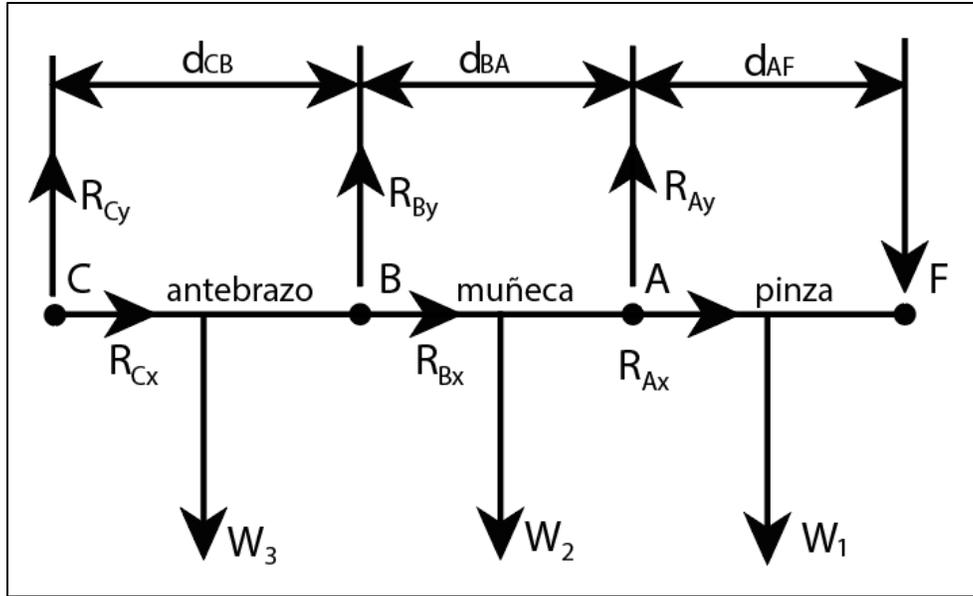


Fig. 3.5 – Torque en el punto C.

$$\sum F_x = R_{Ax} + R_{Bx} + R_{Cx} = 0 \quad (3.16)$$

$$\sum F_y = R_{Ay} + R_{By} + R_{Cy} - W_1 - W_2 - W_3 - F = 0 \quad (3.17)$$

Reemplazando y reduciendo la Ec. (3.17),

$$R_{Cy} = W_3 \quad (3.18)$$

Continuando con el torque en el punto C, se obtiene:

$$\begin{aligned} \sum \tau_C = & R_{Ay}(d_{BA} + d_{CB}) + R_{By}d_{CB} - F(d_{AF} + d_{BA} + d_{CB}) \dots \\ & - W_1\left(\frac{d_{AF}}{2} + d_{BA} + d_{CB}\right) - W_2\left(\frac{d_{BA}}{2} + d_{CB}\right) - W_3\left(\frac{d_{CB}}{2}\right) \end{aligned} \quad (3.19)$$

Reemplazando,

$$\sum \tau_C = -2.84 \text{ [Nm]} \quad (3.20)$$

Finalmente, se procede a calcular el último torque ubicado en el punto D, presentado en la Fig.

3.6,

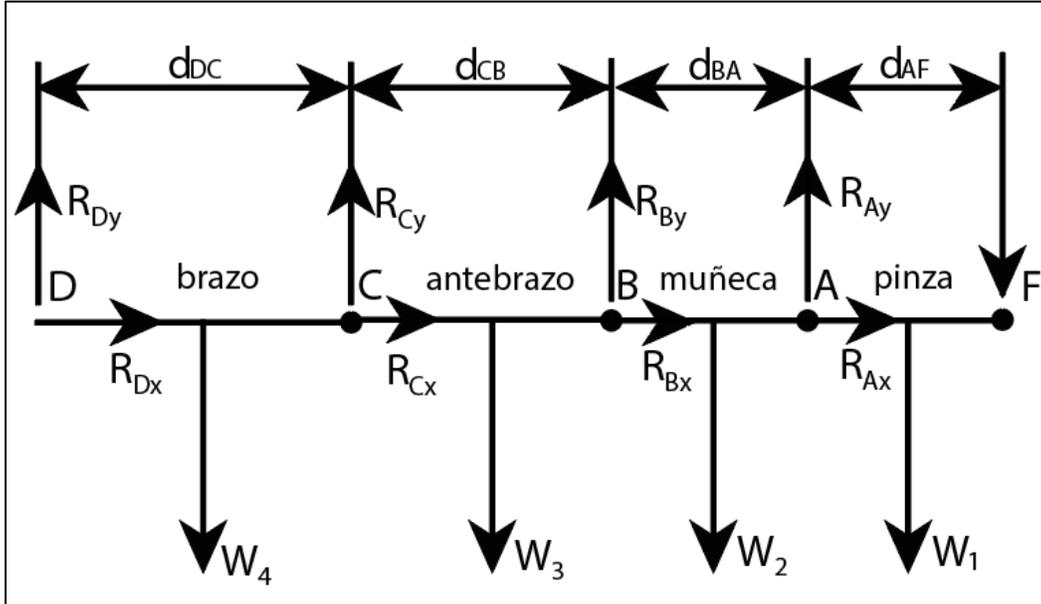


Fig. 3.6 – Torque en el punto D.

$$\sum F_x = R_{Ax} + R_{Bx} + R_{Cx} + R_{Dx} = 0 \quad (3.21)$$

$$\sum F_y = R_{Ay} + R_{By} + R_{Cy} + R_{Dy} - W_1 - W_2 - W_3 - W_4 - F = 0 \quad (3.22)$$

reemplazando y reduciendo,

$$R_{Dy} = W_4 - 2F \quad (3.23)$$

a continuación, se muestra la expresión para el torque en el punto D,

$$\begin{aligned} \sum \tau_D = & R_{Ay}(d_{BA} + d_{CB} + d_{DC}) + R_{By}(d_{CB} + d_{DC}) + R_{Cy}d_{DC} \dots \\ & - F(d_{AF} + d_{BA} + d_{CB} + d_{DC}) - W_1\left(\frac{d_{AF}}{2} + d_{BA} + d_{CB} + d_{DC}\right) \dots \\ & - W_2\left(\frac{d_{BA}}{2} + d_{CB} + d_{DC}\right) - W_3\left(\frac{d_{CB}}{2} + d_{DC}\right) - W_4d_{DC} \end{aligned} \quad (3.24)$$

finalmente, al reemplazar los parámetros se obtiene:

$$\sum \tau_D = -3.63 \text{ [Nm]} \quad (3.25)$$

3.3. Transmisiones del robot

Las transmisiones de este manipulador se elaboran a razón 1:5, con el propósito de que las dinámicas del robot no afecten en el control, ya que desde una relación de transmisión de 1:5 o superior, se puede tomar en consideración solo las dinámicas de los actuadores. Para este caso se utilizarán motores CC para realizar las simulaciones y poder realizar el estudio a las dinámicas. Para realizar el cálculo de las transmisiones se considera la Fig. 3.7 y la Ec. 3.26,

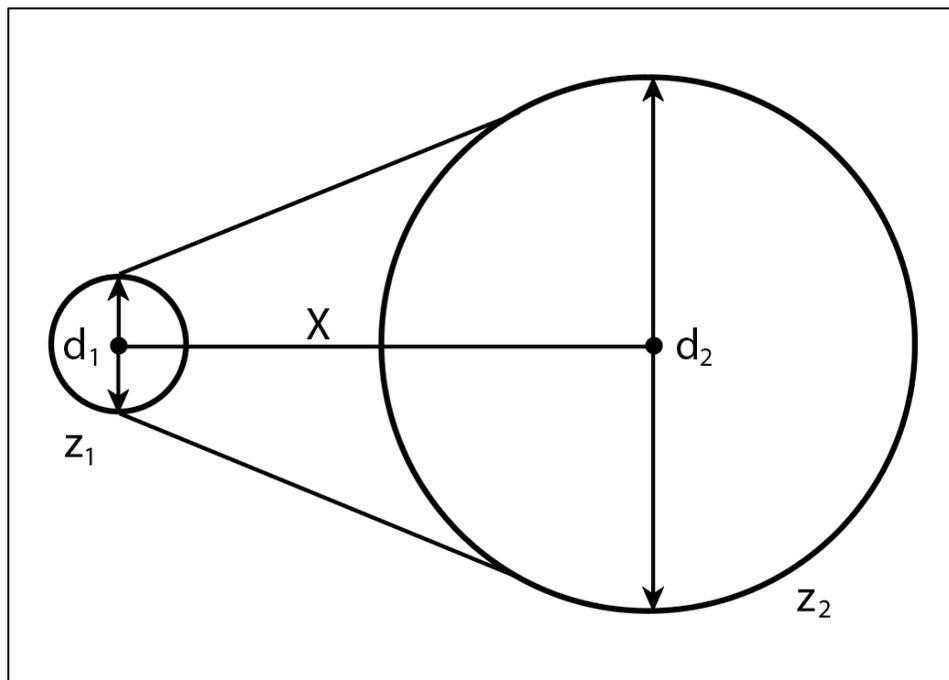


Fig. 3.7 – Planteamiento para las relaciones de transmisión.

$$L_C = 2X + \frac{\pi}{2}(d_1 + d_2) + \frac{(d_2 - d_1)^2}{4X} \quad (3.26)$$

Donde:

L_C	Largo de correa
X	Distancia entre centros
d_1	Polea conducida
d_2	Polea conductora

Los parámetros utilizados en el cálculo de las transmisiones se presentan en la Tabla 3.2 y en la Ec. (3.27) se muestra el valor final del largo de la correa según los valores seleccionados, estos fueron escogidos de un catálogo de transmisiones de correas dentadas [43].

Tabla 3.2 – Parámetros para las transmisiones

Parámetro	Valor
X	55.35 [mm]
d_1	9.68 [mm]
d_2	50.42 [mm]

Por lo tanto, el valor final del largo de la correa se presenta en la Ec. (3.27),

$$L_C = 213.1 [mm] \quad (3.27)$$

Finalmente, según los valores obtenidos se puede revisar el catálogo [43] y obtener la información necesaria para esta relación de transmisión. En la simulación, Inventor utiliza una manera diferente para utilizar las transmisiones, por lo que se utilizan valores aproximados para conservar la relación de transmisión. Además, si se desea implementar este manipulador, se aconseja utilizar una relación de transmisión 16T:80T que cuenta con 16 y 80 dientes respectivamente, conserva la relación de transmisión y su material de construcción es de aluminio. Esta transmisión se utilizará en las articulaciones 1, 2, 3 y 4 ya que según el cálculo anterior son los puntos donde más se requiere torque si se desea utilizar la carga máxima de 500 gramos y en las articulaciones 5 y 6 se utilizarán servomotores, ya que los torques permiten utilizar este tipo de actuadores. Con las transmisiones y torques calculados, se procede a mostrar el diseño de cada pieza del manipulador.

3.4. Diseño de la base

La base está formada por 3 partes principales: base inferior, base superior y el soporte para el motor. Estas partes se unen a través de 8 pernos, antes de completar esta unión se debe fijar el motor al soporte (parte negra de la base). Luego, se fija la polea conducida al motor y la polea conductora al eje de la cadera, que será presentado más adelante. La base inferior cuenta con 4 orificios para fijar el robot a una estructura más rígida como un banco de trabajo. Además, para

obtener un movimiento más suave, se agrega un rodamiento de bolas, que entrará a presión a la base superior y en la parte inferior de la cadera.

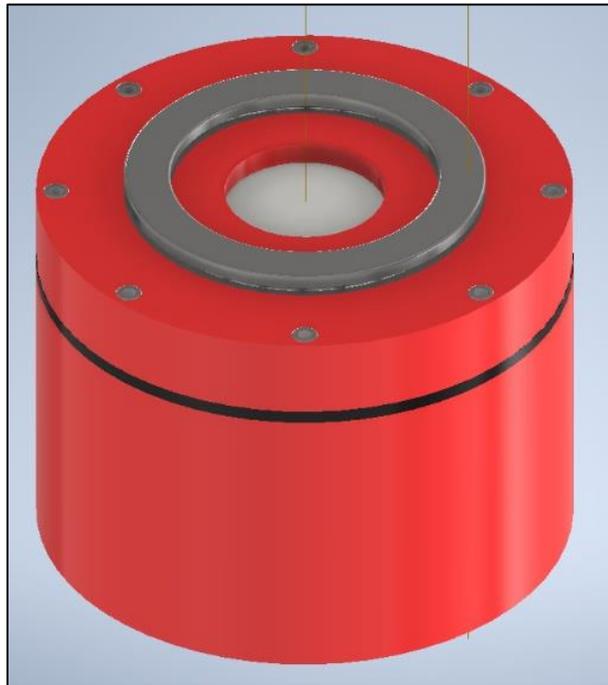


Fig. 3.8 – Ensamblaje de la base.

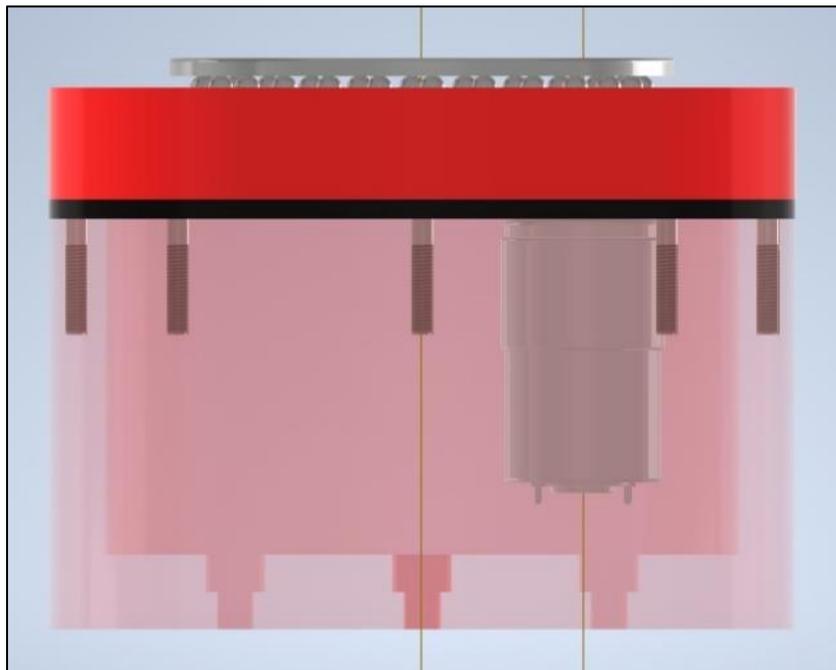


Fig. 3.9 – Ensamblaje actuador a la base.

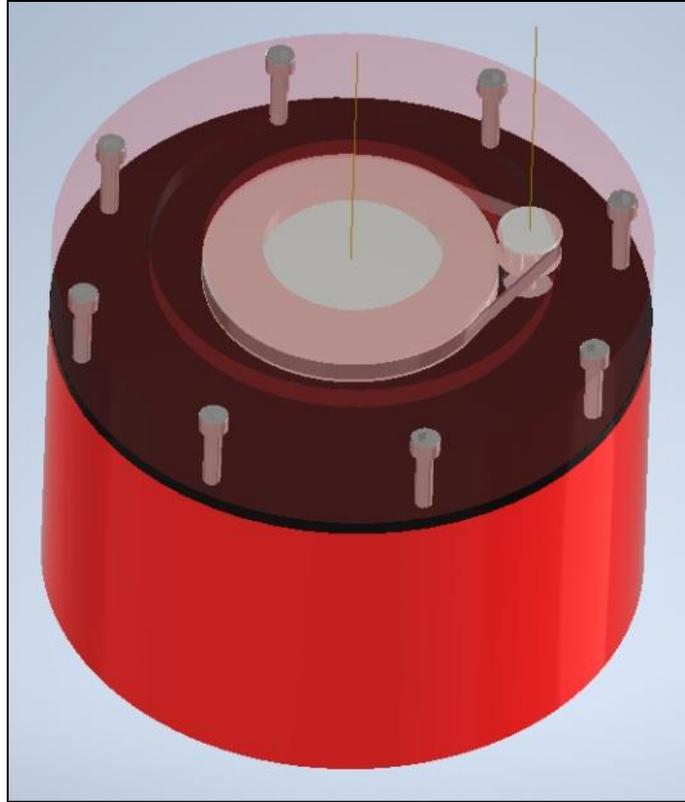


Fig. 3.10 – Transmisión en la base del manipulador.

3.5. Diseño de la cadera

La cadera tiene una estructura con una base y 2 partes elevadas como torres, en donde irá acoplado el brazo. Esta pieza principal asemejará el movimiento rotacional que tiene el ser humano en su cadera y permitirá al manipulador moverse hasta en 360° . Además, contiene un eje unido a la polea conductora de la base, también cuenta con el motor que activará el movimiento del brazo y va acompañado de su respectiva transmisión 1:5 calculada anteriormente. Luego, el eje del brazo irá acoplado a la polea conductora del motor 2, en ambas uniones del eje del brazo con la cadera se ponen rodamientos de bolas para tener un movimiento suavizado.

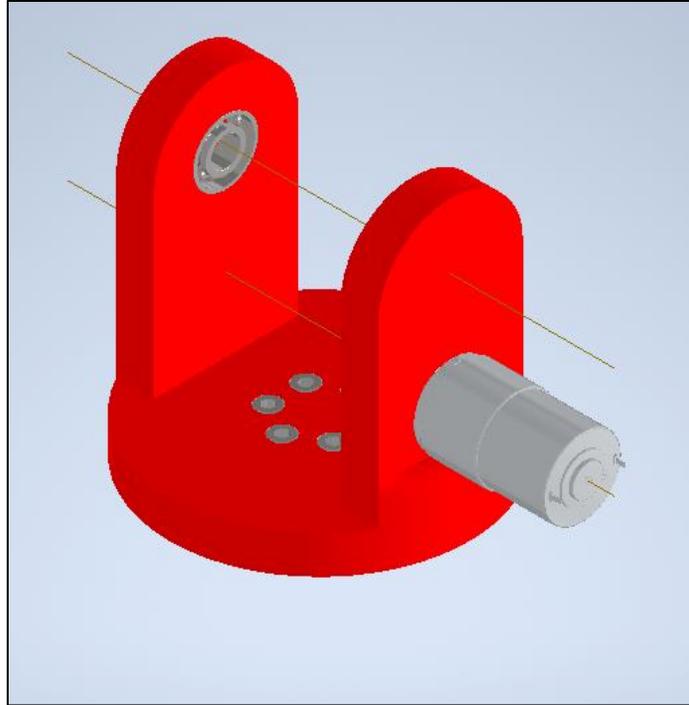


Fig. 3.11 – Ensamblaje de cadera vista principal.

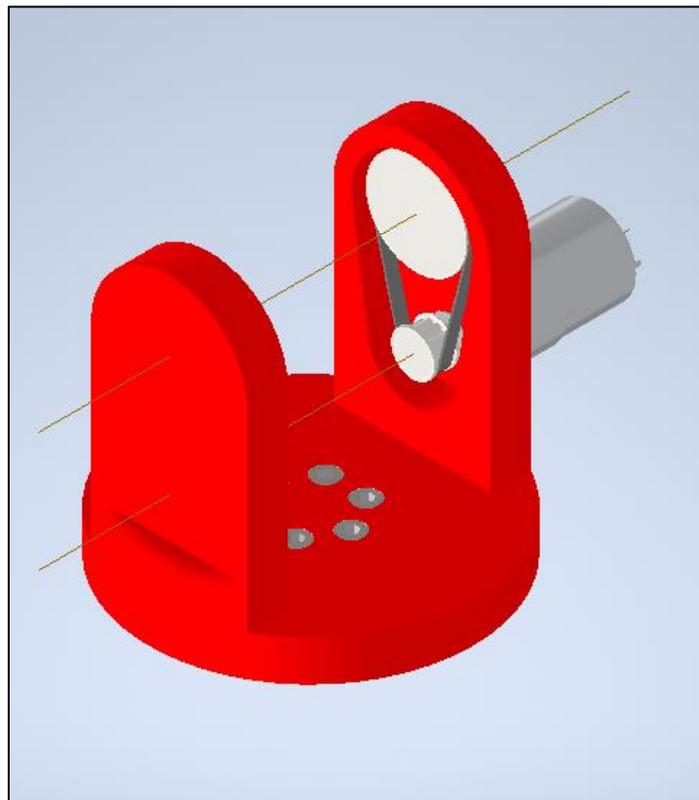


Fig. 3.12 – Transmisión de la cadera.

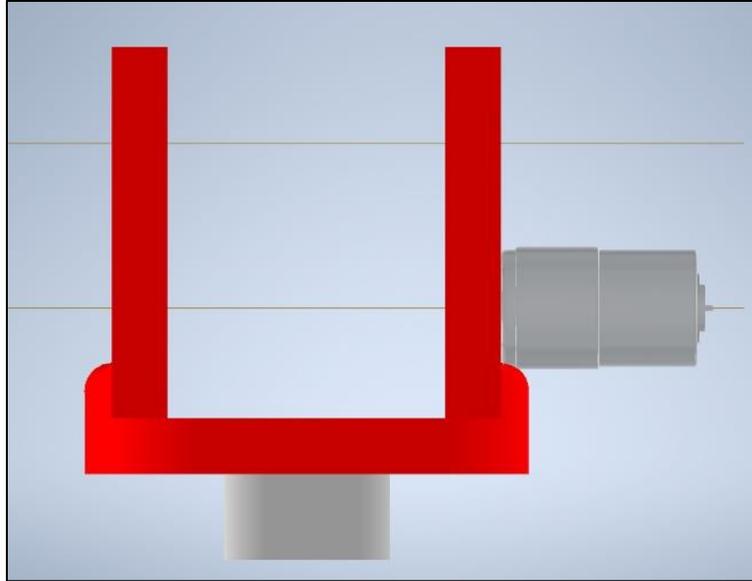


Fig. 3.13 – Ensamblaje de cadera vista frontal.

3.6. Diseño del brazo

El brazo está compuesto por 2 partes principales, las cuales pueden ser unidas a través de 9 pernos. El motor 3, que activará el antebrazo, irá acoplado al brazo y tendrá una transmisión 1:5 oculta en el hueco interior del brazo. El eje que une el brazo a la cadera, por temas de diseño para Matlab, se hizo como una parte que sobresale del brazo, pero en una posible implementación, se espera que ese eje se diseñe aparte y se acople al brazo a través de pernos.

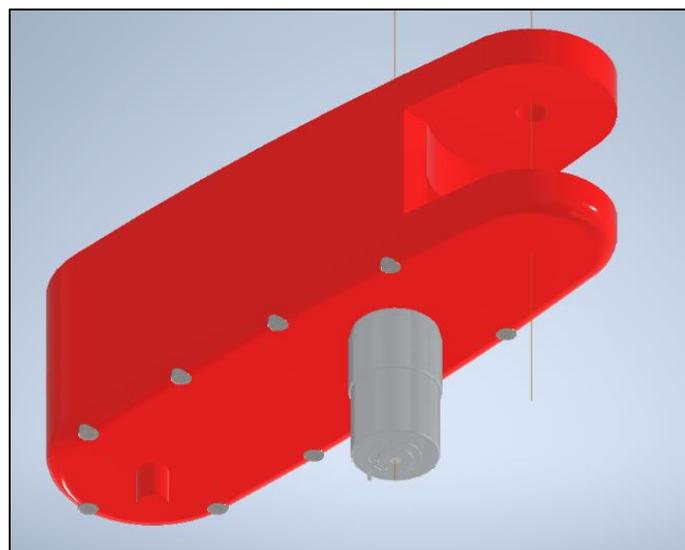


Fig. 3.14 – Ensamblaje del brazo vista principal.

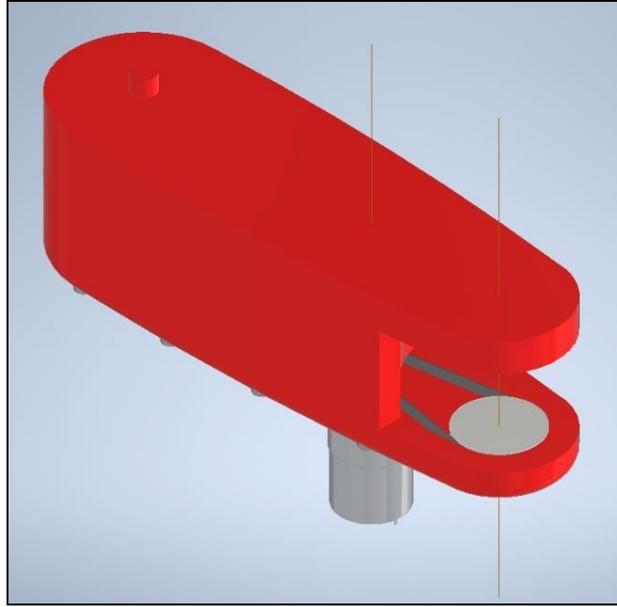


Fig. 3.15 – Ensamblaje del brazo con su transmisión.

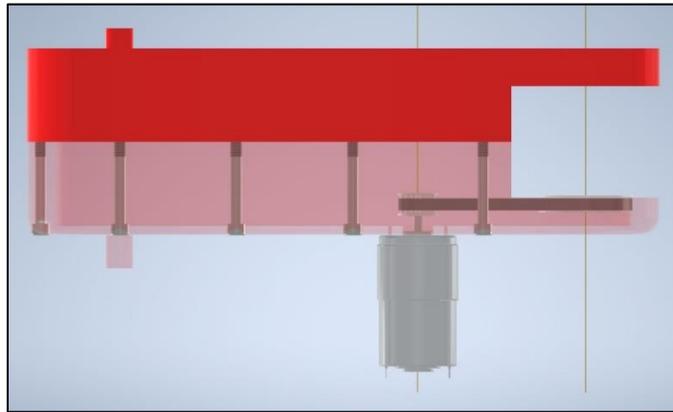


Fig. 3.16 – Ensamblaje de pernos en el brazo, parte A.

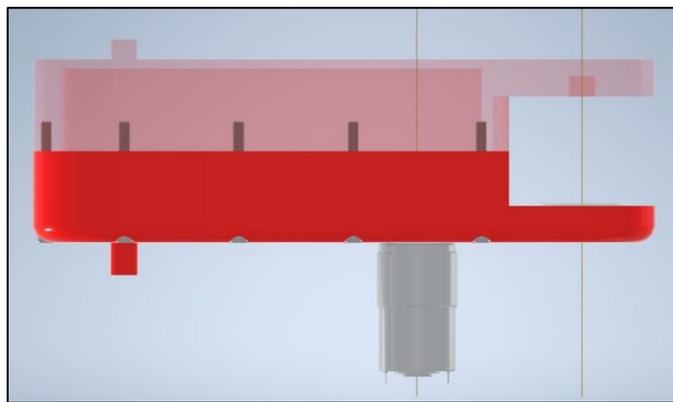


Fig. 3.17 – Ensamblaje de pernos en el brazo, parte B.

3.7. Diseño del antebrazo

El antebrazo tiene la misma forma del brazo, la diferencia está en el largo. La distancia entre los ejes del brazo es de 25 cm y la distancia para el antebrazo es de 20 cm. La transmisión utilizada es la misma que en el brazo.

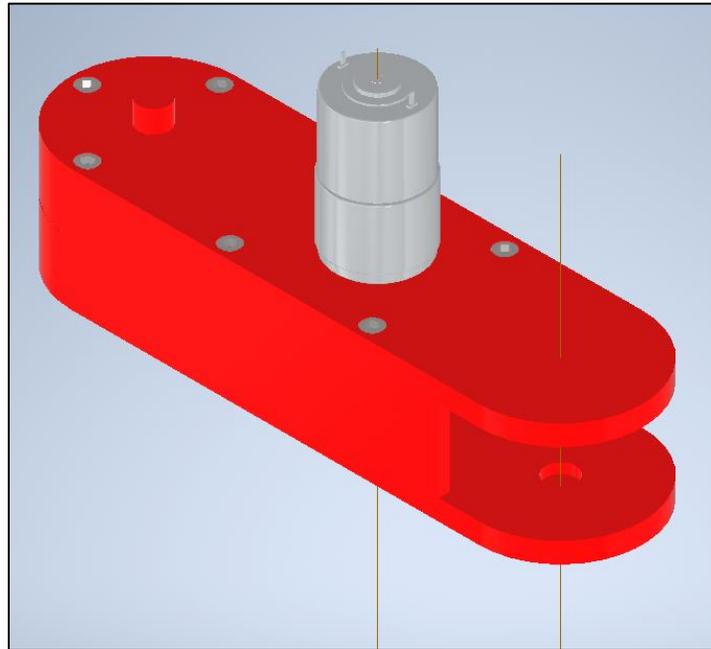


Fig. 3.18 – Ensamblaje del antebrazo.

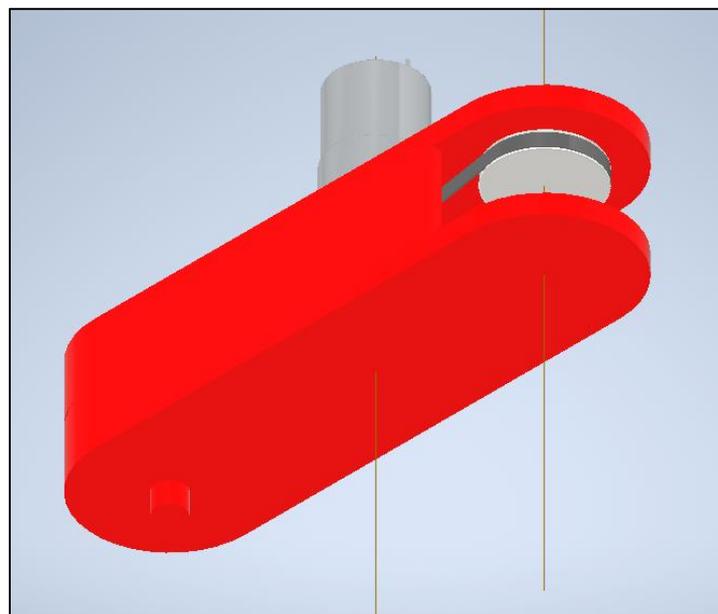


Fig. 3.19 – Ensamblaje del antebrazo con transmisión.

3.8. Diseño muñeca

La muñeca otorga el cuarto y quinto grado de libertad, al igual que en el cuerpo humano. Está acoplada al antebrazo y tiene en el extremo un servomotor que permite dar el giro rotacional al efector final, que, en este caso, será una pinza.

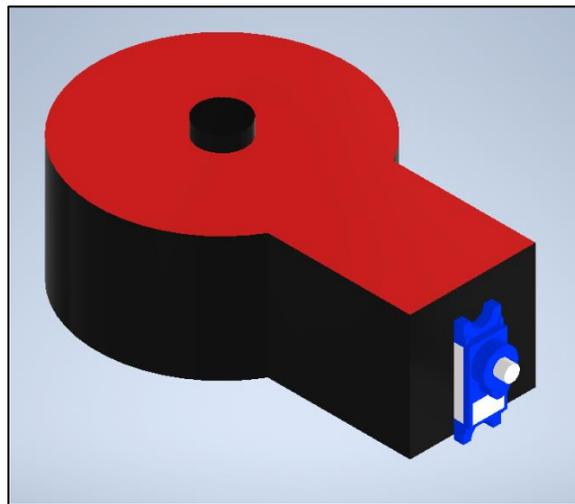


Fig. 3.20 – Ensamblaje muñeca.

3.9. Diseño pinza

La pinza está basada en un diseño universal de engranajes que le permite abrir y cerrar. Este movimiento se realizará a través de un servomotor acoplado a uno de los engranajes.

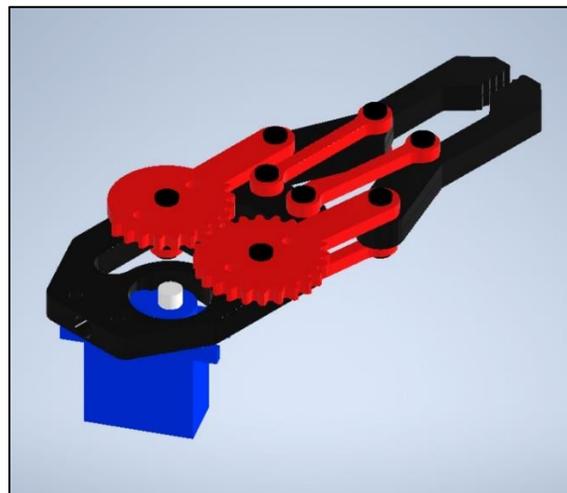


Fig. 3.21 – Ensamblaje pinza.

3.10. Ensamblaje final

En las siguientes figuras se presenta el ensamblaje final del diseño mecánico para un manipulador de 6 GDL.

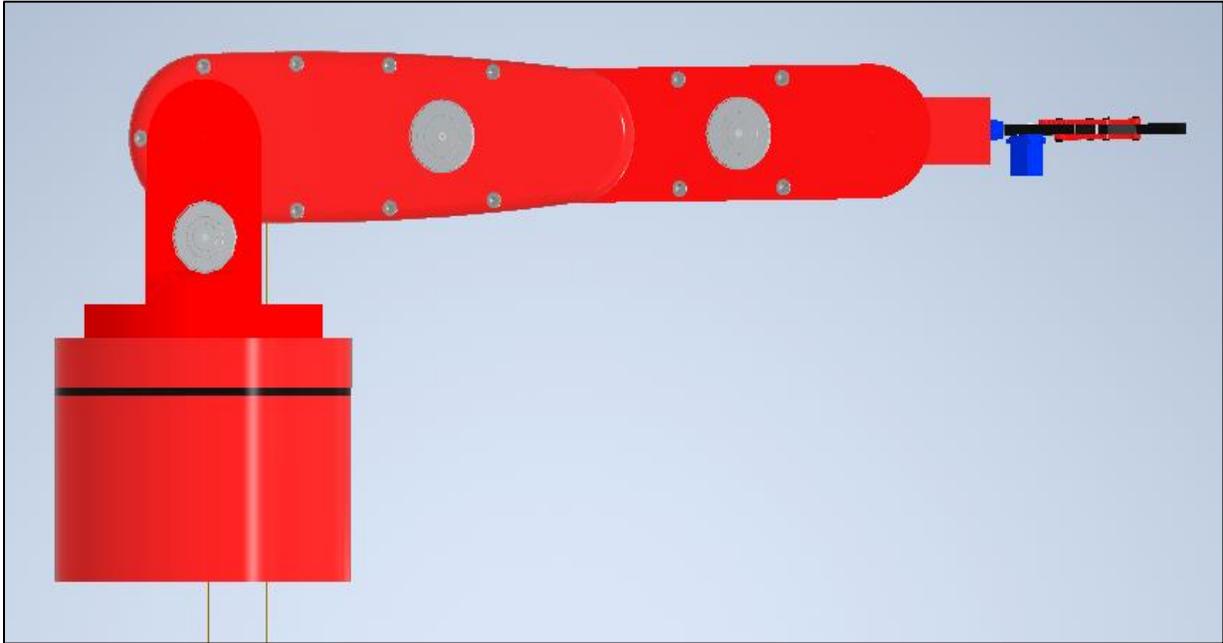


Fig. 3.22 – Ensamblaje robot antropomórfico de 6 GDL, vista frontal.

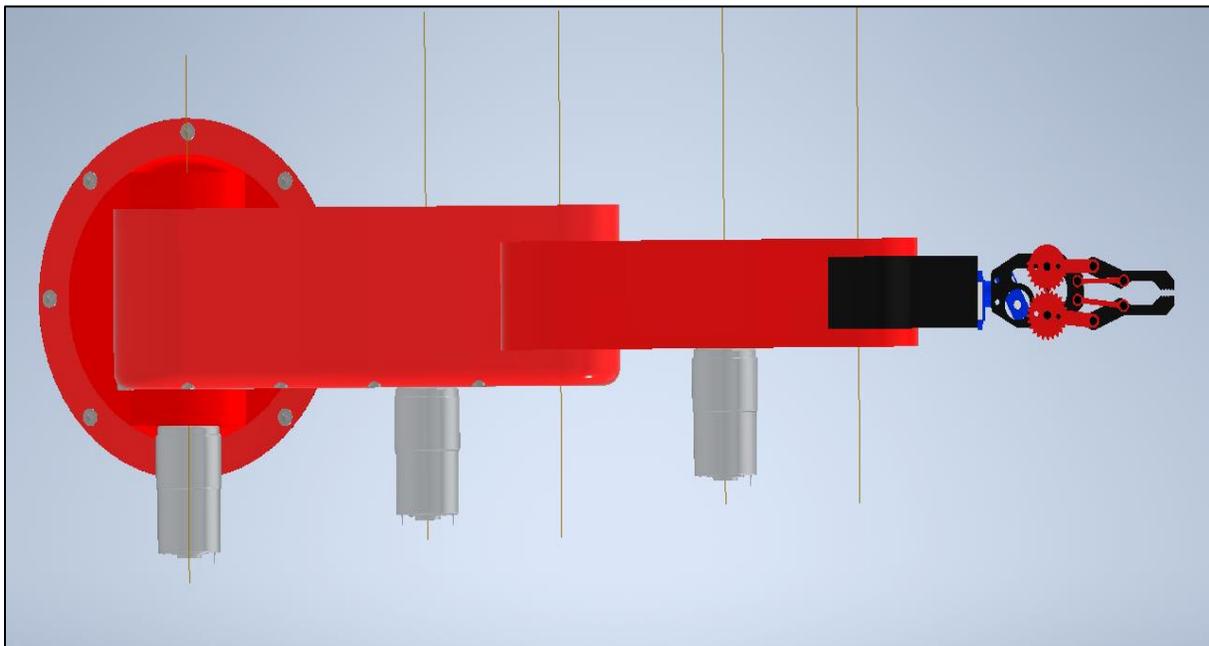


Fig. 3.23 – Ensamblaje robot antropomórfico de 6 GDL vista superior.

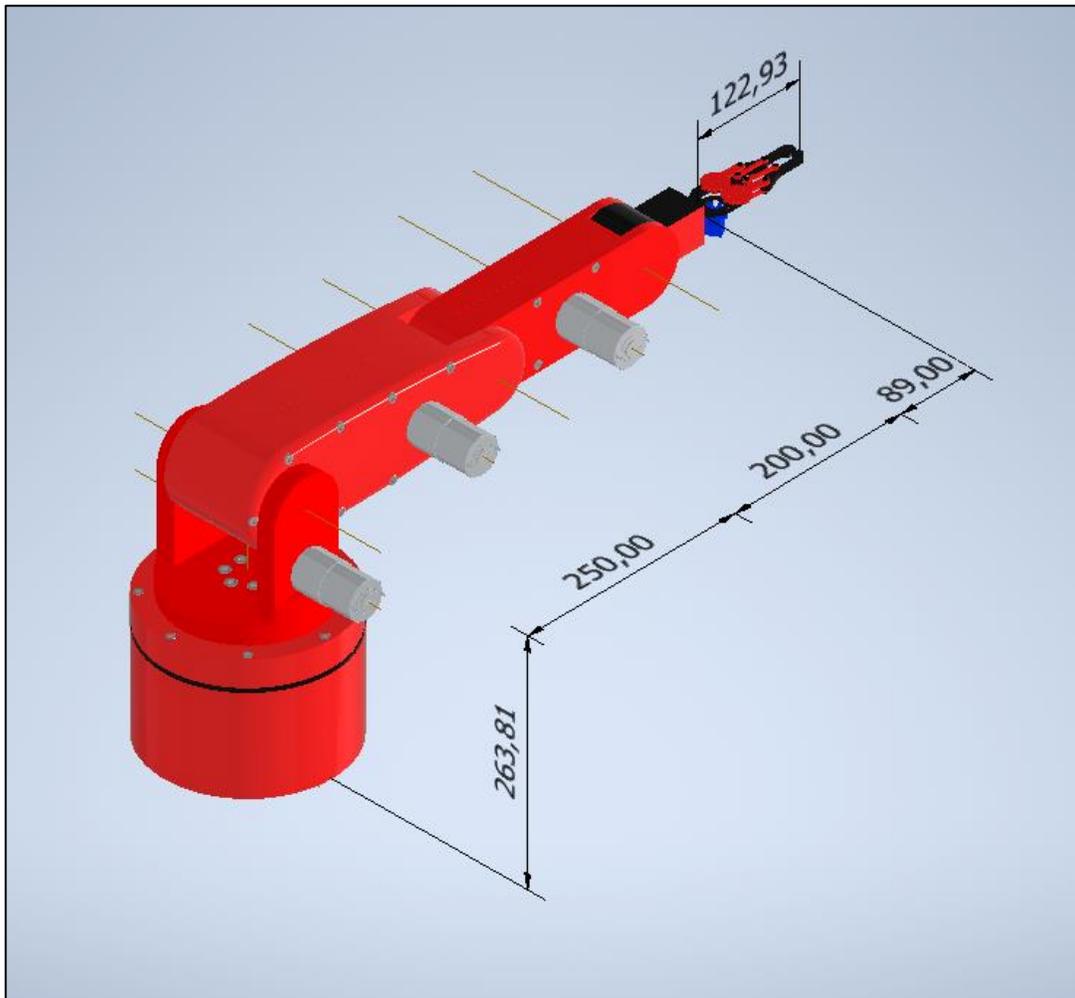


Fig. 3.24 – Ensamblaje final robot antropomórfico de 6 GDL

3.11. Análisis de esfuerzos

En esta subsección se presenta un análisis de esfuerzo para el diseño mecánico realizado, aplicando la carga máxima que se plantea de 0,5 kg. Por lo tanto, se aplica una fuerza de 4.9 [N] aproximadamente. Se crean 2 análisis, el primero en el extremo de la pinza y el segundo en el extremo de la muñeca, esto se realiza debido a que la muñeca será la que soporte el valor de esta carga ejerciendo un mayor torque, como se demostró en los torques calculados anteriormente. Los resultados de estos análisis se presentan a continuación:

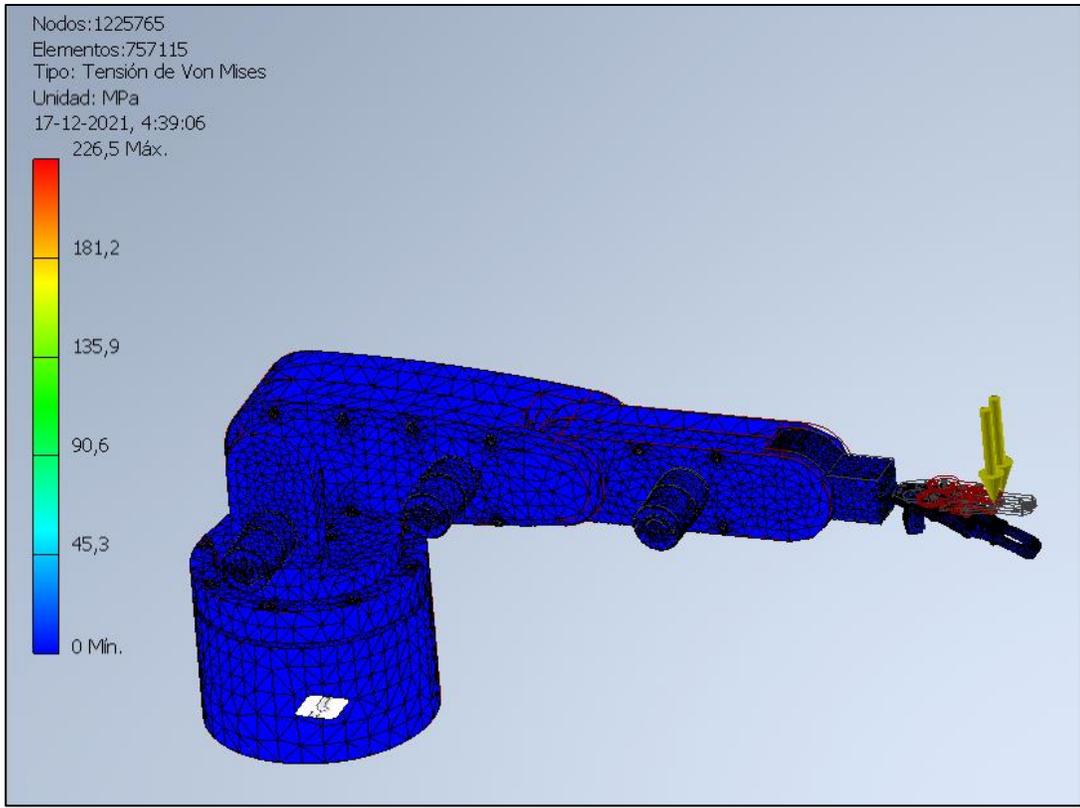


Fig. 3.25 – Tensión de Von Mises al aplicar carga máxima sobre la pinza.

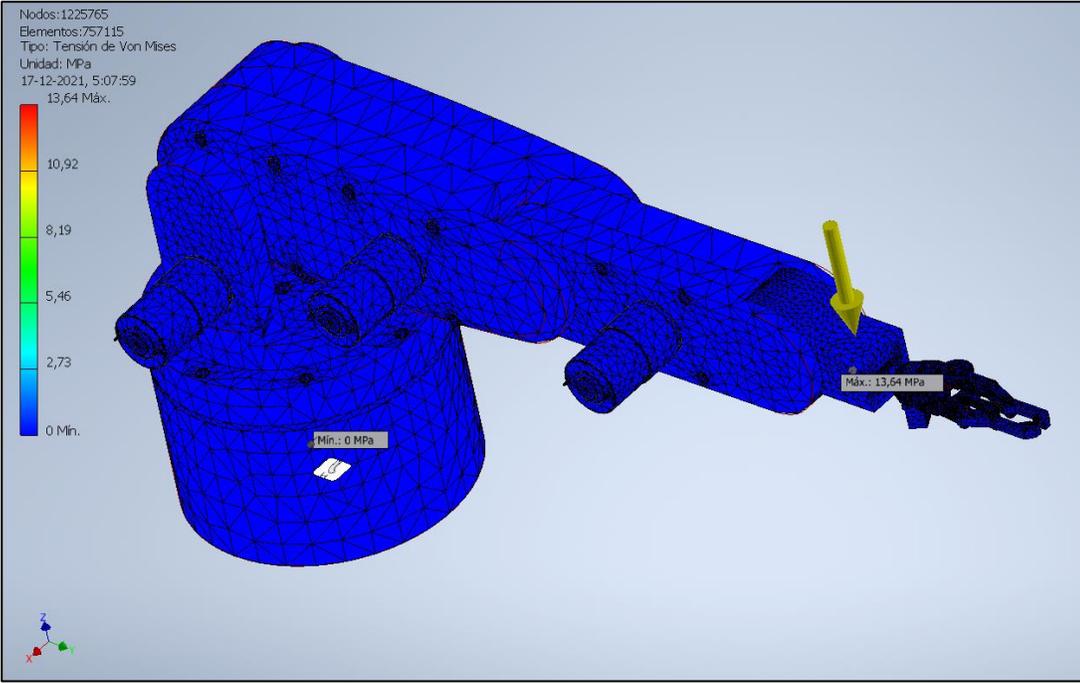


Fig. 3.26 – Tensión de Von Mises al aplicar carga máxima sobre la muñeca.

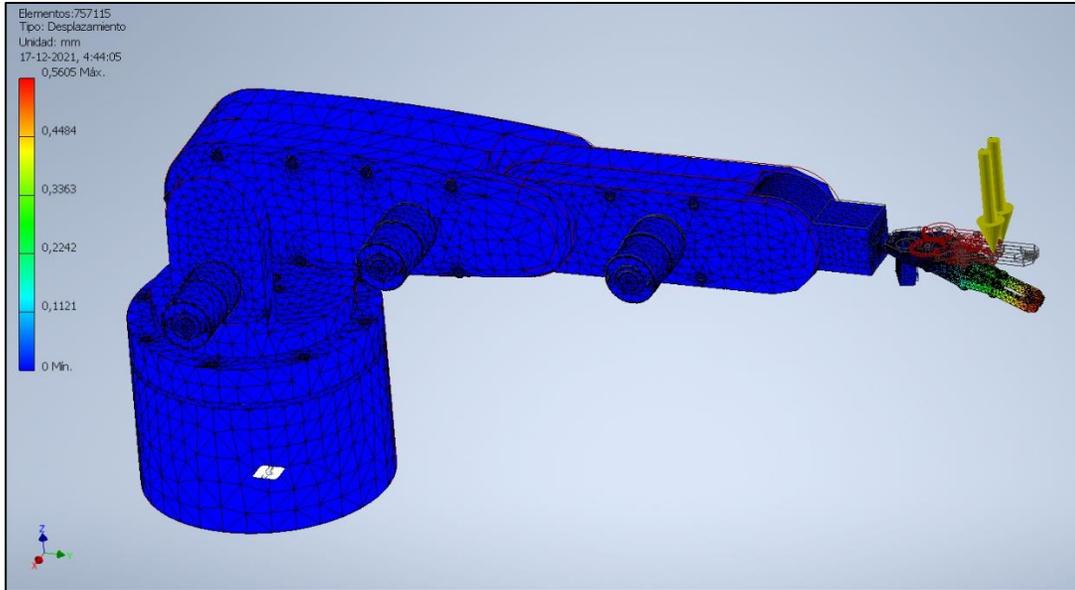


Fig. 3.26 - Desplazamiento del manipulador al aplicar carga maxima sobre la pinza.

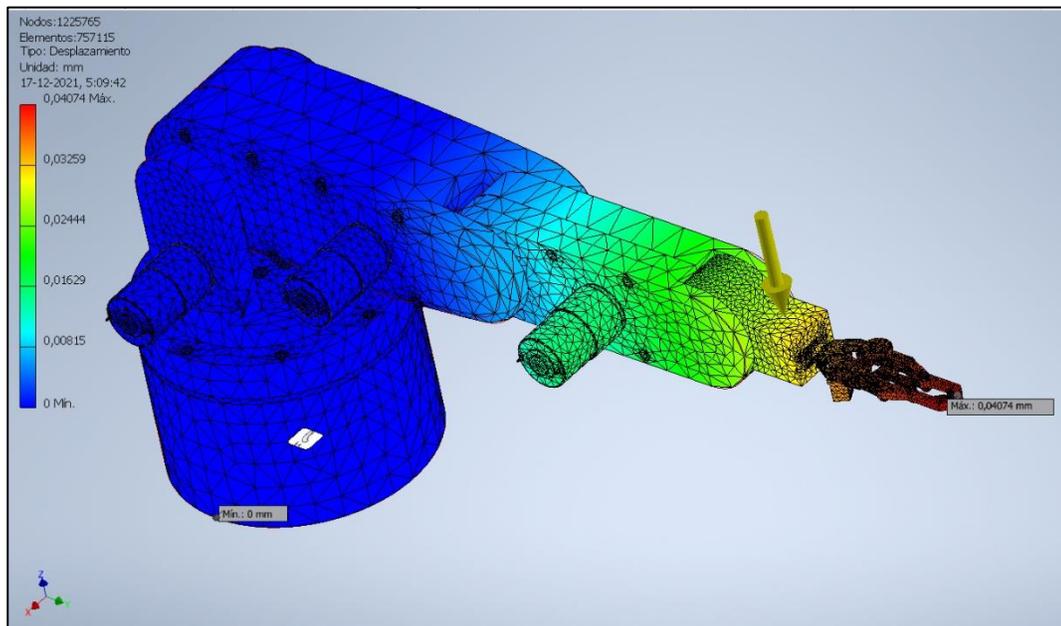


Fig. 3.27 – Desplazamiento del manipulador al aplicar carga maxima sobre la mueca.

Los resultados obtenidos del analisis de esfuerzos son aceptables, ya que el lımite elastico para un material como el ABS es de 32 MPa, al aplicar la fuerza sobre la pinza directamente esta se podra deformar, pero en la practica la mueca sera quien soporte en mayor cantidad esa tension, por lo tanto, los valores obtenidos al aplicar la fuerza sobre la mueca establecen que la estructura

mecánica creada en plástico ABS, soportará la tensión de la carga máxima. Con respecto al desplazamiento, para la carga aplicada en la pinza se obtiene un rango de 0 a 0,4884 [mm] y para la carga aplicada en la muñeca se obtiene un rango de 0 a 0,04074 [mm], por lo tanto, los rangos obtenidos de desplazamiento en ambos casos se encuentran dentro de una tolerancia aceptable para este diseño y tipo de material [71].

3.12. Costos para implementar el manipulador

Los costos presentados en esta subsección son valores aproximados, para estimar el valor final de este manipulador si se desea implementar. Los valores de los actuadores y de las transmisiones recomendadas anteriormente [72]-[73]-[74], se presentan en la Tabla 3.3.

Tabla 3.3 – Valores de actuadores y transmisiones

Descripción	Cantidad	Especificación	Costo
Motor CC	4	3 [Nm] – 12-36 [V]	\$ 154.247
Servomotor	2	SG90	\$ 4216
Transmisión	4	16T:80T 1:5	\$46.040

Para obtener los costos de cada parte del manipulador, se generan archivos STL para ingresarlos al software CURA y obtener cuantos gramos aproximadamente se necesitan para crear cada pieza y cuánto tardará en crearse. Los valores del kilo de ABS rondan los \$16.990 - \$22.990 en Chile más gastos de envío [75]. Además, se genera un cobro de \$100 aproximadamente por cada hora que tarde en construirse la pieza. En la Tabla 3.4 se presentan los gramos y tiempo que tarda cada pieza en construirse y se genera el costo aproximado de generar cada pieza. En las Fig. 3.29 y Fig. 3.30 se aprecia algunas partes del manipulador en el software CURA.

Tabla 3.4 – Costos de las piezas del manipulador

Descripción	Gramos	Tiempo	Costo
Base completa	1644 [g]	132 [h]	\$46.080
Cadera	489 [g]	24 [h]	\$12.180

Brazo	1040 [g]	48 [h]	\$36.000
Antebrazo	460 [g]	24 [h]	\$11.600
Muñeca	196 [g]	24 [h]	\$6.320
Pinza	50 [g]	6 [h]	\$1.600

Los cálculos de costos relacionados a cada pieza del robot se generan al crear una regla de 3 entre el valor del ABS en un valor medio de \$20.000 y los gramos gastados. Se utilizan aproximadamente 3.879 gramos de ABS para implementar el manipulador, aproximadamente 4 rollos de filamento ABS para impresora 3D. Además, dentro de las recomendaciones, el ABS se debe imprimir en lugares cerrados sin corrientes de aire, ojalá con la impresora 3D aislada o cubierta, para evitar flujos de aire que pueden arruinar la impresión de las piezas. Los costos de diseño en este no se consideran, ya que solo se quiere mostrar un valor aproximado del valor del robot si se desea implementar, de la misma manera sucede con los pernos, los rodamientos y la parte electrónica que se quiera aplicar en una posible implementación, estos valores no generarán cambios gigantes en el valor del producto finalizado, que es este caso bordeará los \$320.000. Un cambio brusco en el valor final del producto se puede generar al variar la cantidad de relleno que aplica a la impresión 3D, en este caso se aplica un 50%, pero si aplica un relleno del 20% el valor final de imprimir las piezas del manipulador se puede reducir hasta la mitad.

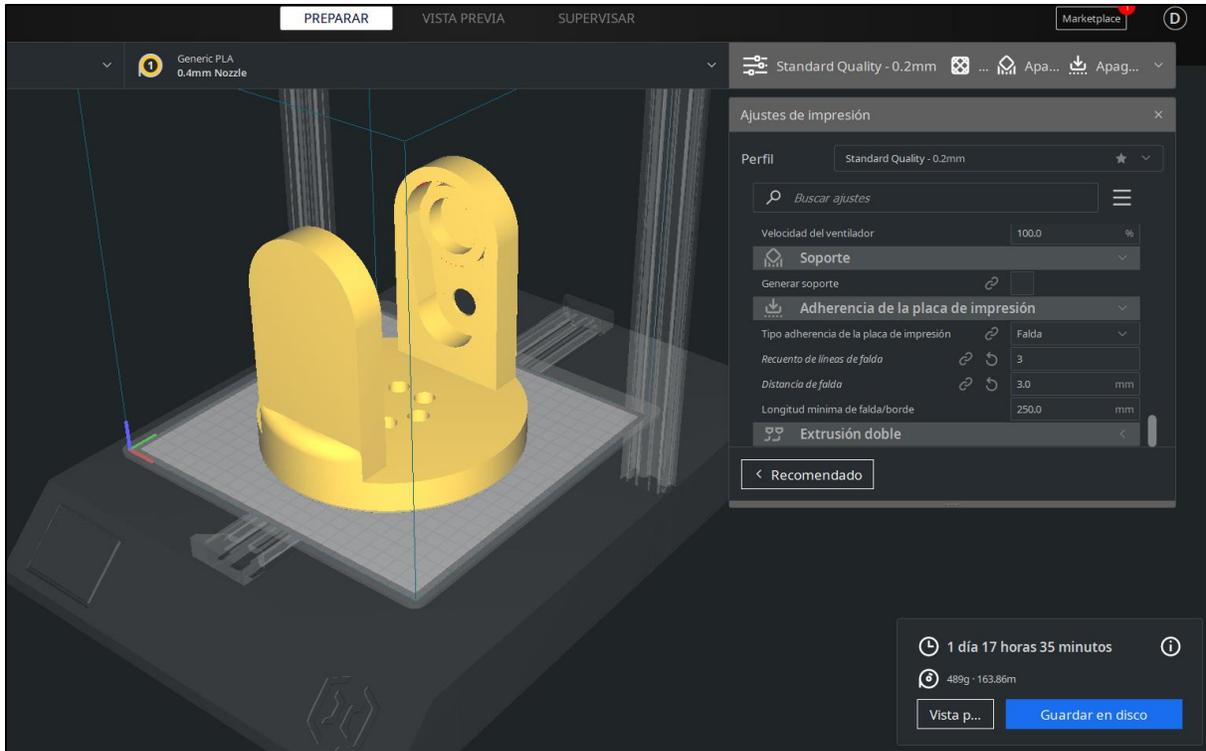


Fig. 3.29 – Estimación de costos de la cadera en el software CURA.

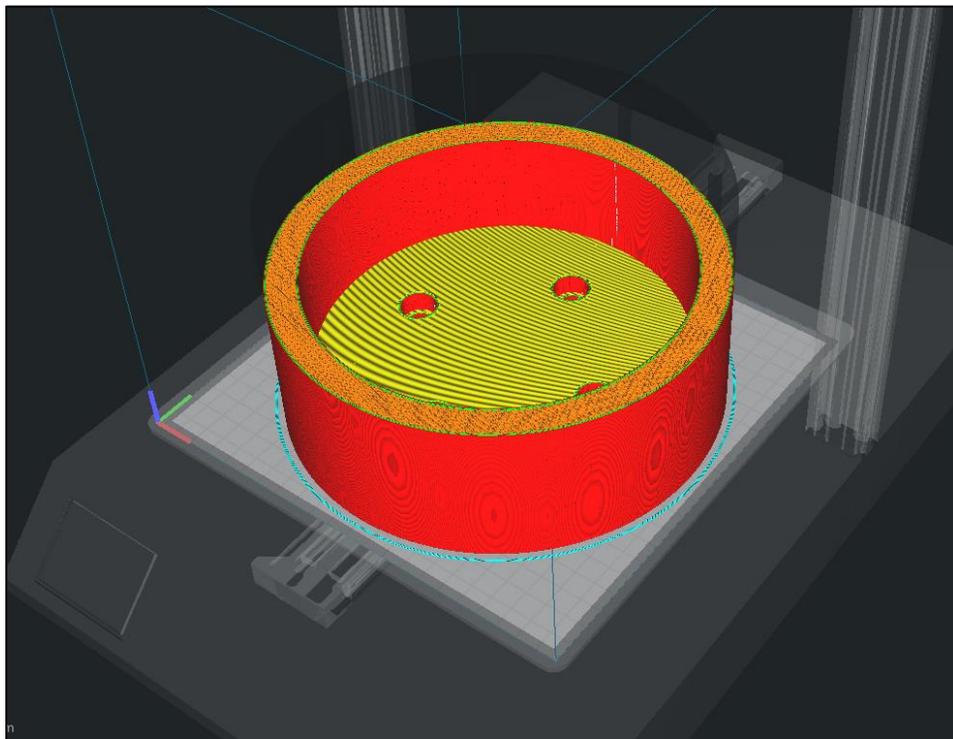


Fig. 3.30 – Estimación de costos una parte de la base en el software CURA (50% de relleno).

3.13. Resumen Capítulo 3

En este capítulo se presentó el diseño mecánico para un robot antropomórfico de 6 GDL, en primer lugar, se calcularon los torques de manera estática en el punto crítico del manipulador. Las masas que se utilizaron son una aproximación obtenida a través del software de diseño 3D, eso puede variar dependiendo del tipo de material que se quiera utilizar para implementarlo. Luego, se calcula las transmisiones con una relación 1:5, se menciona además que sólo se utilizarán en las 4 primeras articulaciones, ya que son las que necesitarán de un mayor torque para moverse. Finalmente, se presentan las piezas y ensamblajes del robot, conservando las partes mencionadas al principio de este capítulo que asemejan a las partes de un brazo humano. Los planos mecánicos se presentan en el Anexo A y los archivos se encuentran disponibles en el enlace [66].

Capítulo 4. Configuraciones necesarias para generar una trayectoria en un robot antropomórfico de 6 grados de libertad

4.1. Introducción

En este capítulo se presentan las herramientas necesarias para crear las simulaciones que permiten estudiar los cálculos matemáticos obtenidos en el capítulo 2 y además de estudiar los torques obtenidos en el capítulo 3. Para realizar estos procesos se necesita generar una comunicación entre Inventor y Matlab, de esta manera se puede exportar el diseño mecánico directamente a Matlab, donde se reconocen las diferentes articulaciones que componen al manipulador.

Para estudiar la cinemática directa se recurre a la librería de Peter Corke, que permite utilizar funciones creadas por el autor relacionadas con robótica, entonces se puede graficar un robot antropomórfico aproximado, conocer su cinemática directa y corroborar si está correcta. Para estudiar la cinemática inversa y la dinámica del manipulador se generan nuevos códigos, los cuales se presentarán como anexos.

Finalmente, se muestra cómo se puede generar una simulación conectando Matlab con Inventor, de esta manera se podrían generar diferentes tipos de estudios relacionados a simulaciones de diseños CAD, generando diferentes tipos de control y estudiar los comportamientos asociados antes de proceder a implementar un manipulador de estas características.

4.2. Comunicación del modelo CAD con Matlab mediante Simscape Multibody Link

Simscape Multibody Link es un complemento que se instala en la aplicación CAD para exportar sus modelos de ensamblaje CAD. El complemento genera los archivos que necesita para importar el modelo en el entorno Simscape Multibody (utilizando la función `smimport`). Se puede instalar el complemento en 4 aplicaciones CAD:

1. Solidworks.
2. Autodesk Inventor.
3. Wildfire.
4. Creo.

Para comenzar a realizar la comunicación entre ambos softwares se necesita tener licenciados ambos productos, para este caso será necesario tener licencia en Matlab y Autodesk Inventor Professional. Además, las instalaciones deben tener la misma arquitectura de sistema, por ejemplo, windows de 64 bits.

Paso 1: realizar instalación de los programas a utilizar.

En este caso, se realiza la instalación de Autodesk Inventor Professional 2019 y Matlab R2020b. Ambos productos se encuentran con licencia estudiantil, gracias a los convenios entregados por la Universidad de Talca.

Paso 2: descarga del plugin/addon.

Al ingresar en [40] se accede directamente a la página de Matlab. En esta página, se observa bajo que parámetros funciona Simscape Multibody Link, en este caso, con Simscape Multibody 3.0 y superior, Matlab R2008b y superior y en sistemas operativos basados en win32 y win64. A continuación, se presenta la Tabla B.1, en la cual se presentan las características necesarias para cada software CAD.

Al observar la tabla B.1, se observa que en la versión de Matlab R2008b se realiza la primera comunicación con el software Solidworks y para el caso de Autodesk Inventor se logra la comunicación en la versión siguiente de Matlab, R2009a. Luego de analizar la tabla B.1, en [40] se presenta un espacio en blanco para introducir el correo, que tiene la licencia de Matlab, de esta manera proceder a descargar el plugin. (Observar Fig. 4.1).

Tabla 4.1 – Características según cada software CAD [40]

Software CAD	Versiones admitidas	Matlab
Solidworks	2001 plus y superior	R2008b y superior
Wildfire, Creo	Creo 1.0 y superior	R2008b y superior

	Wildfire 2.0 y superior	
Autodesk Inventor	2008 y superior	R2009a y superior

Fig. 4.1 – Espacio en blanco para introducir el correo que tienen licencia de Matlab [40].

Fig. 4.2 – Ejemplo de correo institucional que tiene licencia de Matlab [40].

Luego de introducir el correo, como se muestra en la Fig. 4.2, en el paso siguiente se pide ingresar una serie de datos para confirmar la identidad de la persona. Al completar este paso, se muestran las diferentes versiones Simscape Multibody Link, aquí es necesario basarse en la versión de Matlab para escoger la versión correcta. En este caso, la versión de Matlab es R2020b, en la Fig. 4.3 se observan las diferentes versiones y se remarca la versión desde la cual se realiza la descarga del plugin. Antes de descargar el plugin, se debe tener en cuenta el sistema operativo que utiliza el ordenador: Linux, Windows o UNIX. Para este tutorial, se usa un sistema operativo basado en Windows de 64 bits, luego de selecciona esa opción se procede a descargar 2 archivos “smlink.r2020b.win64.zip” y “install.addon.m”.

> Simscape Multibody Link 7.3 – Release 2021a (Simscape Multibody 7.3)

▼ Simscape Multibody Link 7.2 – Release 2020b (Simscape Multibody 7.2)

Simscape Multibody 7.2	
Win64 (PC) Platform	smlink.r2020b.win64.zip install_addon.m
UNIX (64-bit Linux)	smlink.r2020b.glnxa64.zip install_addon.m
Mac OS X (64-bit Intel)	smlink.r2020b.maci64.zip install_addon.m

> Simscape Multibody Link 7.1 – Release 2020a (Simscape Multibody 7.1)

> Simscape Multibody Link 7.0 – Release 2019b (Simscape Multibody 7.0)

> Simscape Multibody Link 6.1 – Release 2019a (Simscape Multibody 6.1)

> Simscape Multibody Link 6.0 – Release 2018b (Simscape Multibody 6.0)

Fig. 4.3 – Archivos seleccionados para este tutorial [40].

Paso 3: Instalación del plugin/addon en Matlab y en el software de CAD.

En este paso, es necesario crear una carpeta que contenga ambos archivos descargados y que sea de fácil acceso. Luego, se procede a ejecutar Matlab como “administrador” y esperar hasta que el programa se inicie. A continuación, se debe ejecutar el comando addpath (‘ubicación de ambos archivos’), para la ubicación de ambos archivos, se puede dar en propiedades uno de los archivos y copiar la dirección directamente (observar Fig. 4.4).

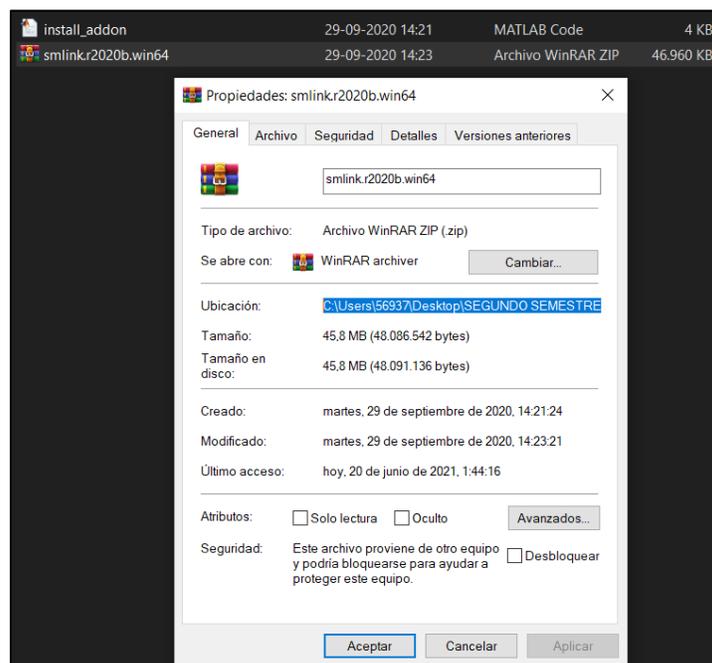


Fig. 4.4 – Carpeta de ubicación de archivos.

Posteriormente, se ejecuta el comando `addpath` y se procede a ejecutar otro comando, este se llama `install_addon` ('nombre del archivo.zip') (observar Fig. 4.5). En este caso el comando a ejecutar sería: `install_addon('smlink.r2020b.win64.zip')`, el código mencionado se puede apreciar en la Fig. 4.5. En la ventana de comandos, aparece el mensaje "instalando y extrayendo archivos". Aquí se debe esperar hasta que aparezca el mensaje "instalación completada", esto se aprecia en la Fig. 4.6. Además, se puede corroborar la instalación revisando la documentación, ejecutando el comando `doc smlink`.

```
>> addpath('C:\Users\56937\Desktop\SEGUNDO SEMESTRE FINAL\Matlab\CAD')
>> install_addon('smlink.r2020b.win64.zip')
Installing smlink...
Extracting archive smlink.r2020b.win64.zip to C:\Program Files\MATLAB\R2020b...
```

Fig. 4.5 – Comandos a ejecutar para la instalación del plugin.

```
Adding directories for smlink to path...
Installation of smlink complete.

To view documentation, type "doc smlink".
```

Fig. 4.6 – Mensaje de instalación completada y revisión de documentación.

Ahora es importante tener Autodesk Inventor Professional cerrado para activar unas preferencias. Luego se debe habilitar el plugin instalado, entonces en el comando de Matlab se escribe el código `smlink_linkinv` (Observar Fig. 4.7) al ejecutar el comando aparece el mensaje de una correcta instalación (Observar Fig. 4.7).

```
>> smlink_linkinv
Registering dll: regsvr32 "C:\Program Files\MATLAB\R2020b\bin\win64\cl_inventor2sm.dll"
>> |
```

Fig. 4.7 – Comando final para conectar con Inventor.

Luego se procede a abrir inventor, aparecerá un mensaje relacionado a que la instalación fue correcta se oprime "Aceptar" y se espera hasta que el programa se ejecute. Aparecerá una alerta de seguridad, se marca Simm Mechanics Link y se da clic en Aceptar. Finalmente, se comprueba

el vínculo entre Matlab e Inventor, para esto en Inventor se accede a Herramientas → Complementos → Sim Mechanics Link, aparece bloqueado por default se desactiva “bloquear” y luego se activa cargado/descargado y cargar automáticamente (Observar Fig. 4.8).

Paso 4: cambiar “comas por puntos”

En inventor se procede a exportar el ensamblaje del diseño mecánico creado (Observar. Fig. 4.9) y se genera una carpeta con el nombre ensamblaje, creando archivos.STEP y un archivo .XML. En el archivo .XML es muy importante cambiar las comas por puntos, ya que Matlab trabaja con puntos, e Inventor trabaja con comas en sus decimales.

Paso 5: Abrir el diseño mecánico en Matlab.

En este paso es importante tener la carpeta abierta en Matlab en el “currentfolder” y luego se usa la función smimport (‘nombre del archivo.XML’), se ejecuta y genera automáticamente el diagrama de bloques de Simulink del diseño mecánico.

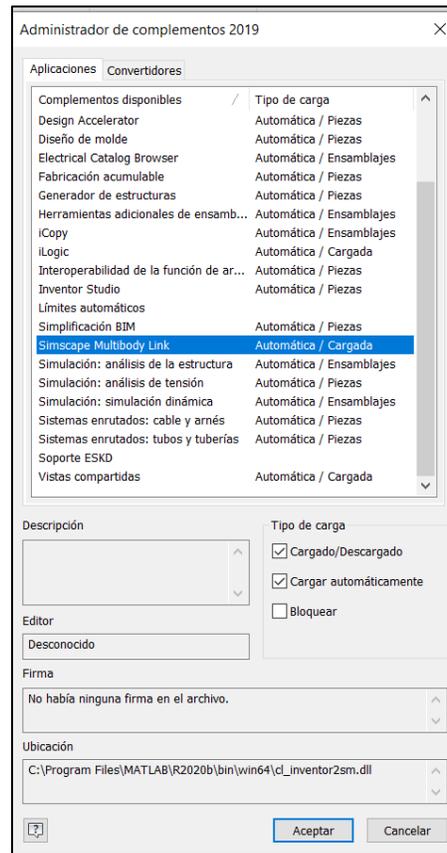


Fig. 4.8 – Comprobación del vínculo de comunicación.

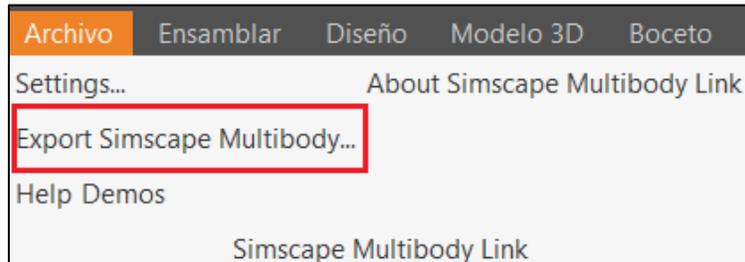


Fig. 4.9 – Complemento para exportar el diseño CAD a Matlab.

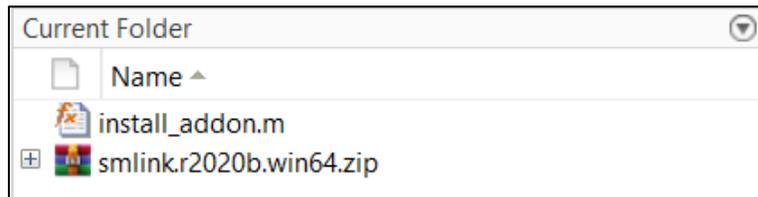


Fig- 4.10 – Current folder donde deben estar los archivos creados por Inventor en Matlab.

4.3. Librería de Peter Corke

La instalación de esta librería se explica visualmente en el video creado para explicar esta memoria de título [64]. La librería de Peter Corke se basa en toolboxes de robótica para Matlab, desarrollados directamente por quien se menciona, estas herramientas permiten generar simulaciones y estudiar diferentes tipos de robots. La instalación de esta librería es sencilla, se debe ir directamente a la página de Peter Corke [65], se ingresa a la sección ‘TOOLBOXES’ y se descarga el archivo ‘RTB10.4.mltbx’ mostrado en la Fig. 4.11. Luego desde dentro del navegador de archivos de MATLAB se ejecuta el archivo descargado, así se instalarán y configurarán las rutas correctamente.



Fig. 4.11 – Archivo para la instalación de librería de Peter Coorke [64].

4.4. Integración del modelo CAD desde Inventor hacia Matlab

En esta subsección, el primer paso es exportar el diseño CAD a Matlab, el diseño realizado se presenta en la Fig. 4.12, este diseño mecánico fue el presentado en el capítulo anterior, en el ensamblaje final del robot antropomórfico de 6 GDL. En la Fig. 4.14, se observa el diagrama de bloques generado automáticamente luego de exportar el diseño mecánico.

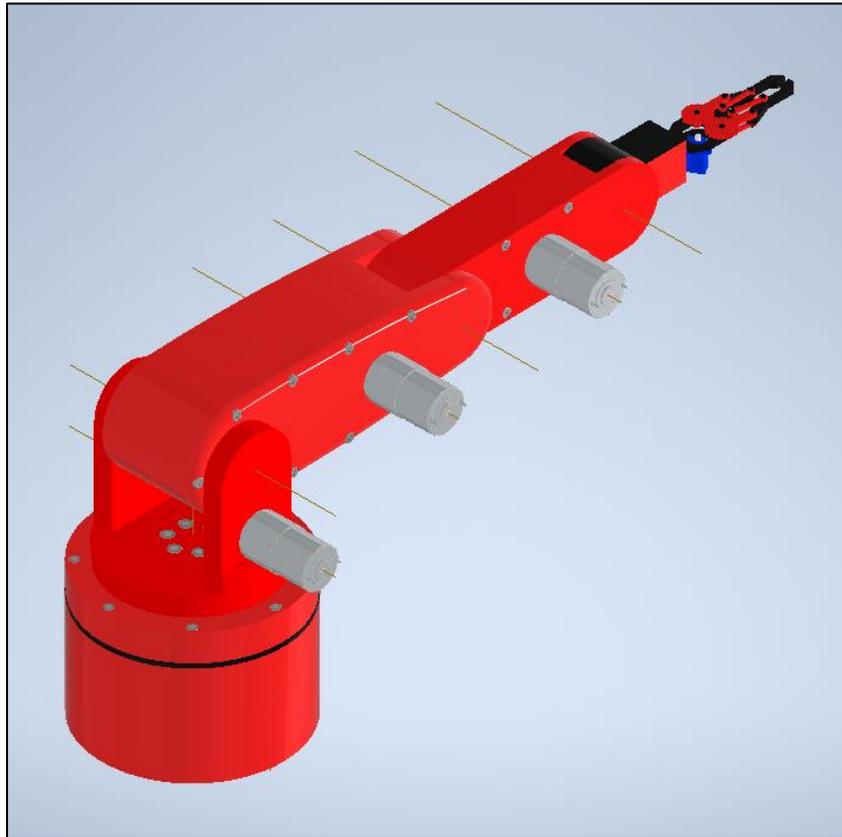


Fig. 4.12 – Diseño mecánico creado en Inventor exportado a Matlab.

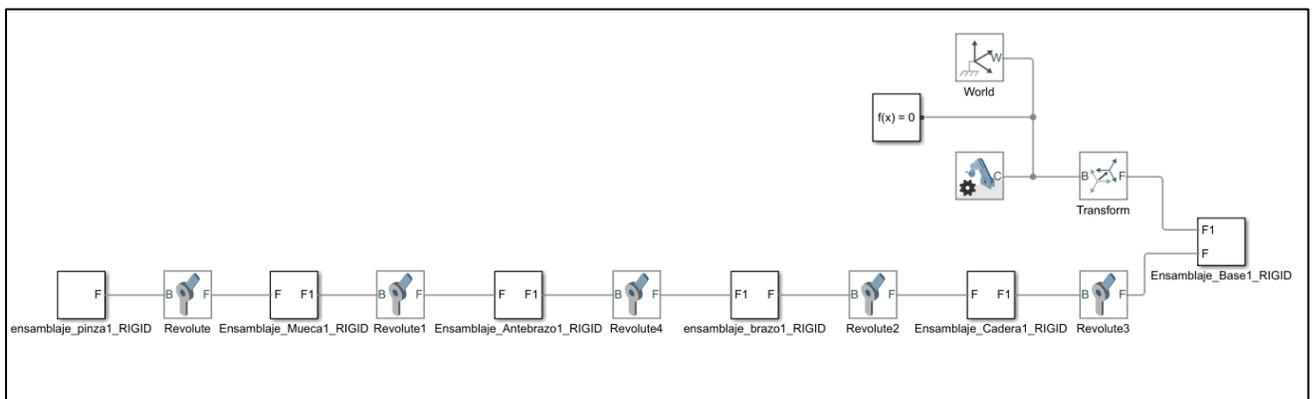


Fig. 4.13 – Bloques generados en Simscape Multibody en Matlab.

En la figura anterior se observa, que existen bloques llamados Revolute, estos son las articulaciones del modelo creado y sobre ellos se puede aplicar una entrada articular para estudiar su movimiento, esto conlleva poder estudiar posiciones, velocidades, aceleraciones y torques de cada articulación. Cada articulación Revolute debe tener la configuración mostrada en la Fig. 4.15 para poder ingresar los valores de la trayectoria. Además, en la Fig. 4.14 se presenta el modelo obtenido al exportar el diseño CAD, observando que ahora se encuentra en el entorno de simulación de Simscape Multibody.

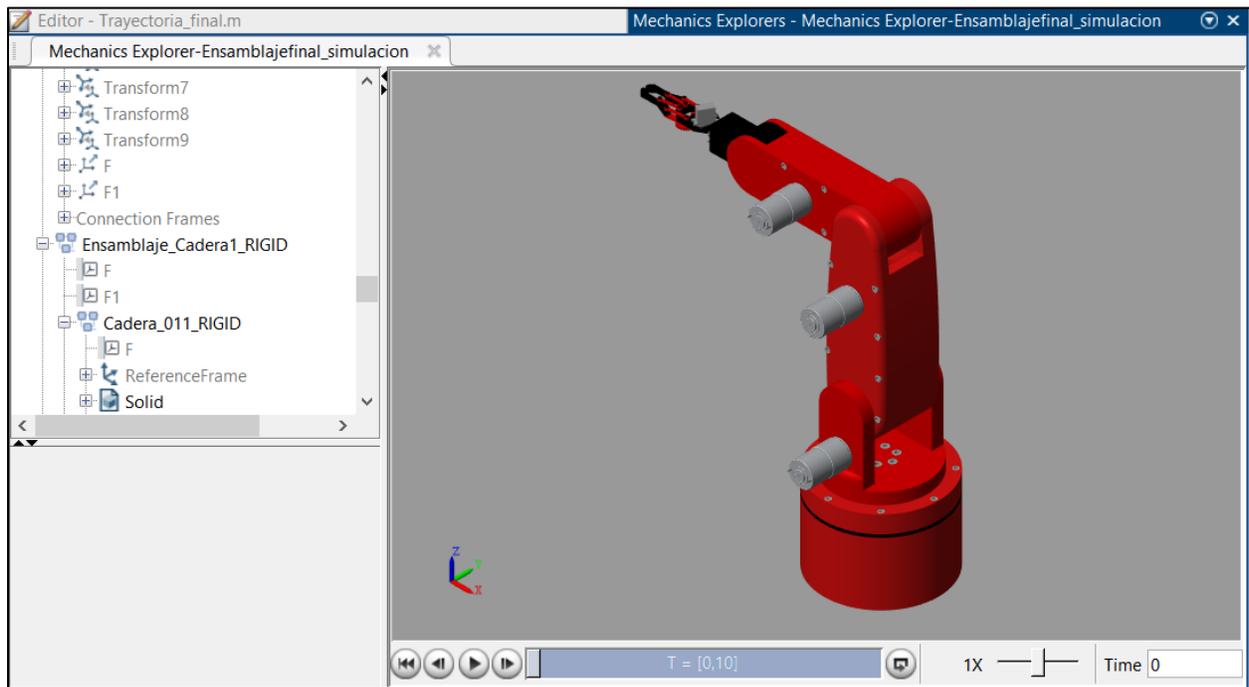


Fig. 4.14 – Robot antropomórfico diseñado en el entorno de simulación.

La configuración de la Fig. 4.15 permite generar una entrada a través de variables articulares y genera un torque automático relacionado al diseño mecánico creado. Entonces el nuevo bloque creado en Revolute tiene entrada q , que permitirá al manipulador moverse al valor q entregado en su entrada. Además, se puede sensar la posición, velocidad, aceleración y torque de cada articulación, para generar el sensado se debe tener la configuración de la Fig. 4.16.

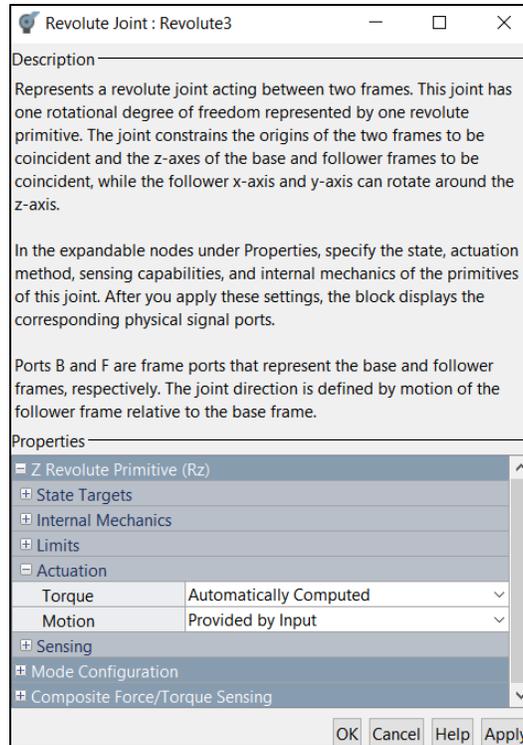


Fig. 4.15 – Configuración de la actuación de cada articulación.

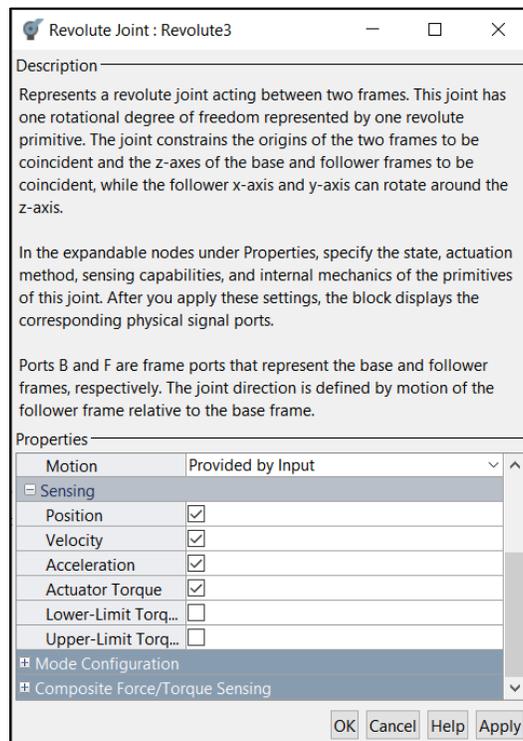


Fig. 4.16 – Configuración para sensor las variables.

En la Fig. 4.17 se presenta otra configuración importante, el bloque Simulink-PS Converter permite la comunicación de Simulink con Simscape Multibody, este debe estar configurado en radianes. Además, la configuración presentada en la Fig. 4.18 permite el correcto comportamiento de la simulación, de lo contrario, la simulación no funcionará.

La simulación tendrá una comunicación directa desde el workspace de Matlab, en donde se creará el código para la trayectoria. La entrada de variables articulares estará fijada en grados, pero como se explicó anteriormente, el bloque Simulink-PS Converter permite el ingreso de datos en radianes, por lo tanto, se debe hacer una transformación de unidades, esto se presenta en la Fig. 4.19. Teniendo en cuenta todas las configuraciones antes mencionadas, se puede ingresar la trayectoria deseada para cada articulación y como última indicación, al exportar el modelo de Inventor a Matlab se genera un código DataFile.m, es importante que este archivo este ejecutado en el comando de Matlab o no cargará el modelo CAD y marcará error la simulación.

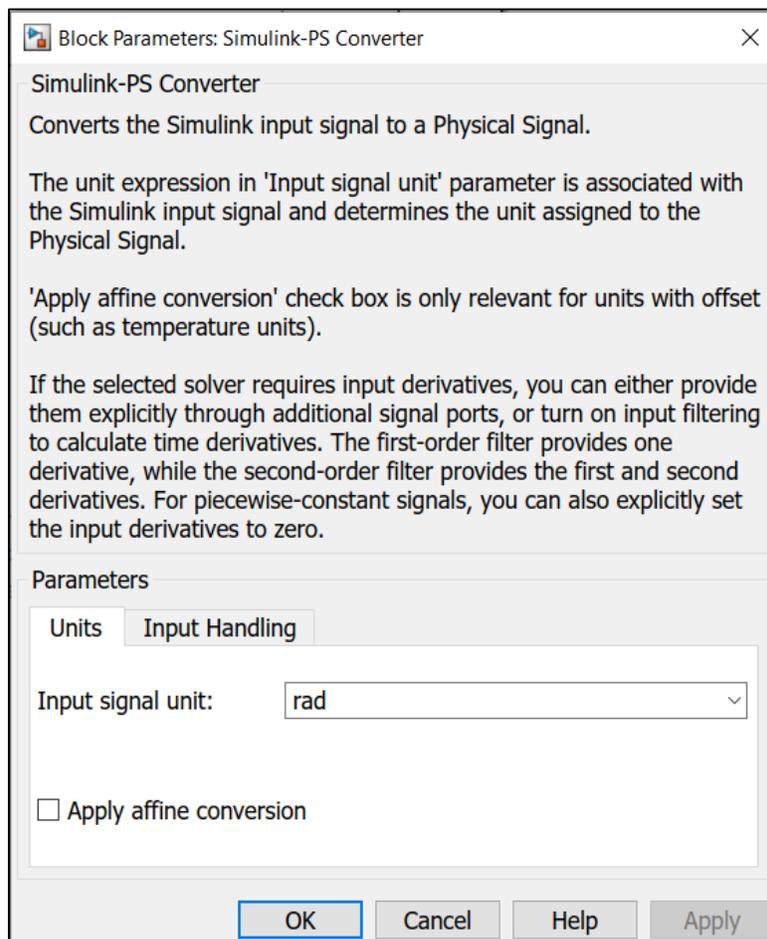


Fig. 4.17 – Configuración del convertidor en radianes.

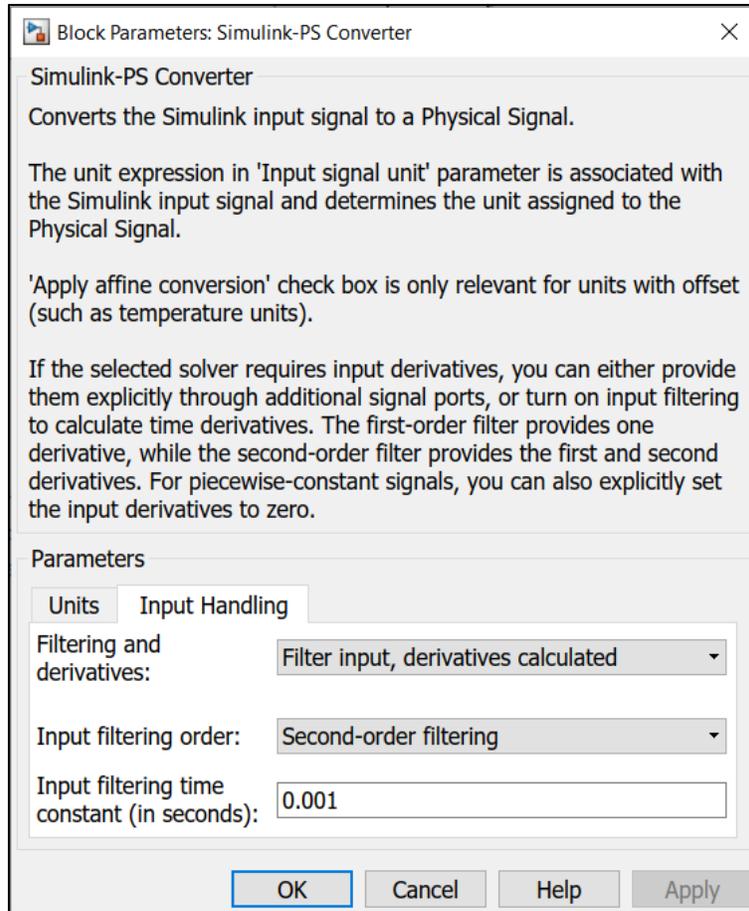


Fig. 4.18 – Configuraciones importantes para que funcione la simulación.

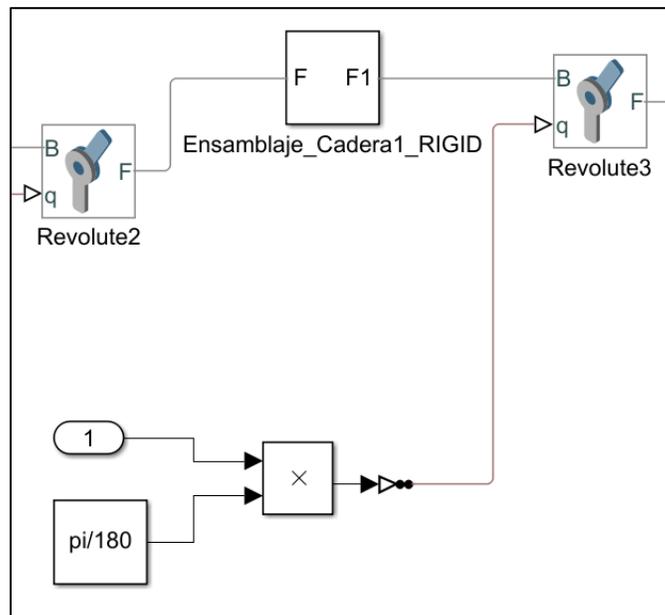


Fig. 4.19 – Transformación de grados a radianes.

A continuación, se presenta la configuración para el ingreso de datos al manipulador creado, cada entrada corresponde a una articulación. Los datos que ingresarán estarán dados por una interpolación cúbica, entre un valor inicial y un valor final para generar una trayectoria.

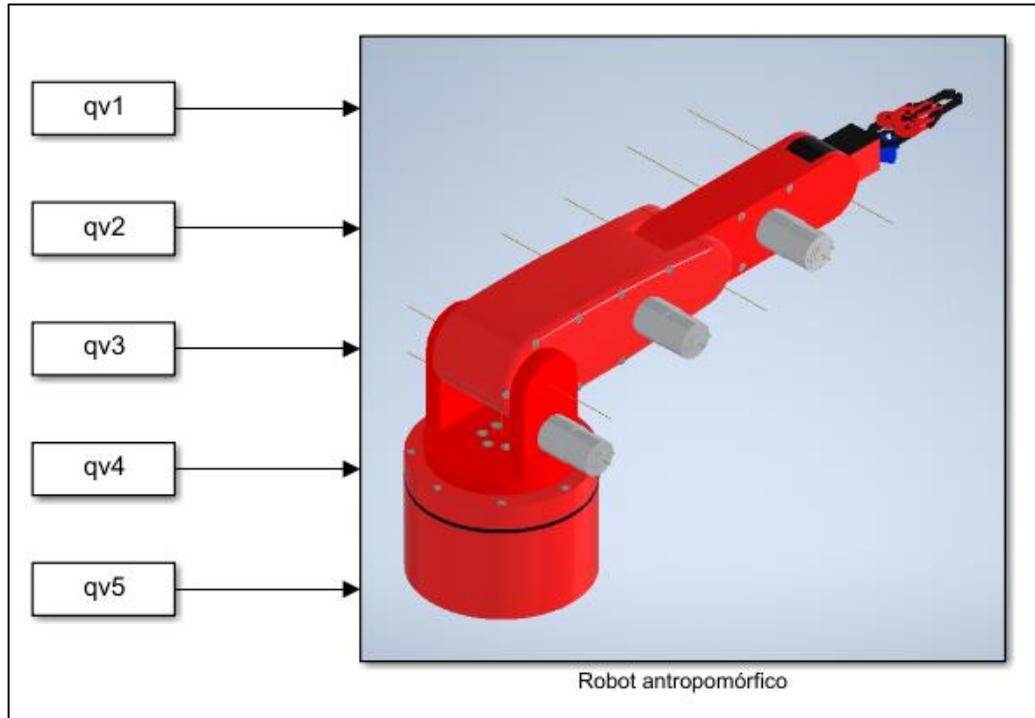


Fig. 4.20 – Entrada de datos al manipulador.

El código para generar la trayectoria se presenta en el Anexo B – Códigos, luego de crear el código se procede a obtener los resultados de la simulación, los cuales se presentan en el siguiente capítulo. Además, se crea un video explicativo relacionado a las simulaciones trabajadas en Matlab [64].

4.4.1. Cinemática directa mediante la librería de Peter Corke

Luego de la instalación de la librería de Peter Corke, se puede hacer un análisis de cinemática directa del manipulador y observar el comportamiento de este, variando el valor en grados de las articulaciones. Para realizar esta simulación se utiliza la función 'Revolute', que permite crear una articulación de revolución, la función recibe los parámetros de D-H calculados en el capítulo 2 de esta memoria. Se definen las 6 articulaciones con la ayuda de la función 'Revolute' y de los parámetros D-H, luego con la función 'SerialLink' se crea el robot, que se aprecia en la Fig. 4.21.

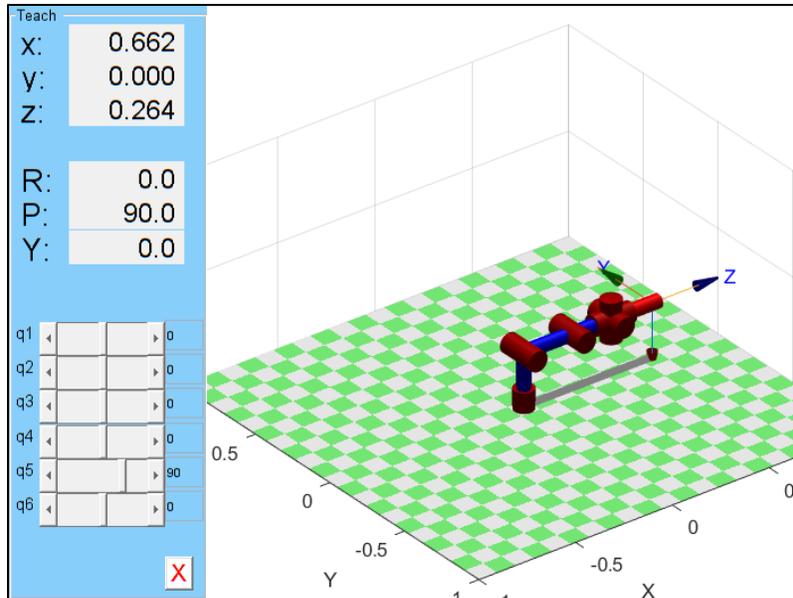


Fig. 4.21 – Simulación cinemática directa con librería de Peter Corke.

La Fig. 4.21 muestra el valor de x,y,z del efector final, que es el punto de interés para la cinemática directa de un robot de este tipo. Para corroborar esta cinemática con respecto al diseño mecánico realizado se calculan las distancias en (x,y,z) del efector final en ambos softwares, obteniendo los mismos resultados en ambos programas, por lo tanto, la cinemática directa funciona de buena manera.

4.4.2. Cinemática inversa mediante la librería de Peter Corke

La simulación de la cinemática inversa se presenta en código a través de Matlab, el punto de interés principal en conocer la cinemática inversa se basa en la posición del efector final. De esta manera, al ingresar los puntos de interés $x=0.662, y=0, z=0.264$ obtenidos de la cinemática directa, los ángulos entregados por la cinemática inversa deberían ser los mostrados en la Fig. 4.21.

4.5. Dinámica

La simulación de la parte dinámica del manipulador se genera de manera simbólica, obteniendo valores para cada torque asociado al manipulador, pero como no se utilizará la dinámica para generar la trayectoria, se deja la referencia del código utilizado en el Anexo B – Códigos.

4.6. Resumen capítulo 4

Este capítulo cumple con entregar las herramientas necesarias para exportar un diseño CAD desde Inventor a Simscape Multibody de Matlab, solucionando la problemática relacionada a la estandarización de las herramientas para exportar modelos CAD de Inventor a Matlab. Par esta memoria de título, se crea una generación de trayectoria a través de un código en el software de programación, pero se espera que en trabajos futuros se implementen controles realimentados con otros tipos de interpoladores.

Capítulo 5. Resultados simulaciones

5.1. Introducción

En este capítulo se presentan las simulaciones obtenidas tras programar una generación de trayectoria al modelo CAD exportado desde Inventor. Se presentan los valores iniciales y finales de la trayectoria, también los resultados del comportamiento de las posiciones, velocidades y aceleraciones de cada articulación del manipulador.

Este capítulo permite verificar lo planteado en la problemática de esta tesis, ya que se puede corroborar el funcionamiento de este manipulador a través de las gráficas obtenidas. Considerando que los actuadores son ideales y que la dinámica no afecta en el comportamiento del manipulador, se obtienen las curvas que describen a una interpolación cúbica.

Finalmente, se verifica el comportamiento de los torques del manipulador, considerando las dinámicas internas que genera el software de Matlab sobre el diseño mecánico creado. Dicho esto, se puede corroborar si los cálculos de torques realizados analíticamente en el capítulo 3 son correctos.

5.2. Generación de trayectoria

La generación de trayectoria se refiere a cuando debe ser calculado el movimiento, con respecto a la posición, velocidad y aceleración de cada articulación [50]. Para esto existen diferentes tipos de movimientos robóticos, entre los que destacan están: el control de trayectoria punto a punto, donde solo es necesario especificar los puntos iniciales y finales, ignorando la trayectoria que describan esos dos puntos, y el control de trayectoria continua donde la ruta que debe seguir el manipulador comprende una línea recta o circular, creando una secuencia de puntos intermedios por donde se debe pasar lo más fielmente posible [51]. Para esta memoria de título se plantea realizar una generación de trayectoria pick and place (trayectoria punto a punto), para confirmar el correcto funcionamiento del diseño mecánico creado en el capítulo anterior, de esta manera se puede dar solución a la problemática planteada, proponiendo estudiar y verificar el comportamiento de un robot industrial antropomórfico de 6 grados de libertad. El comportamiento de lo mencionado anteriormente se puede apreciar de manera esquemática en la Fig. 5.1.

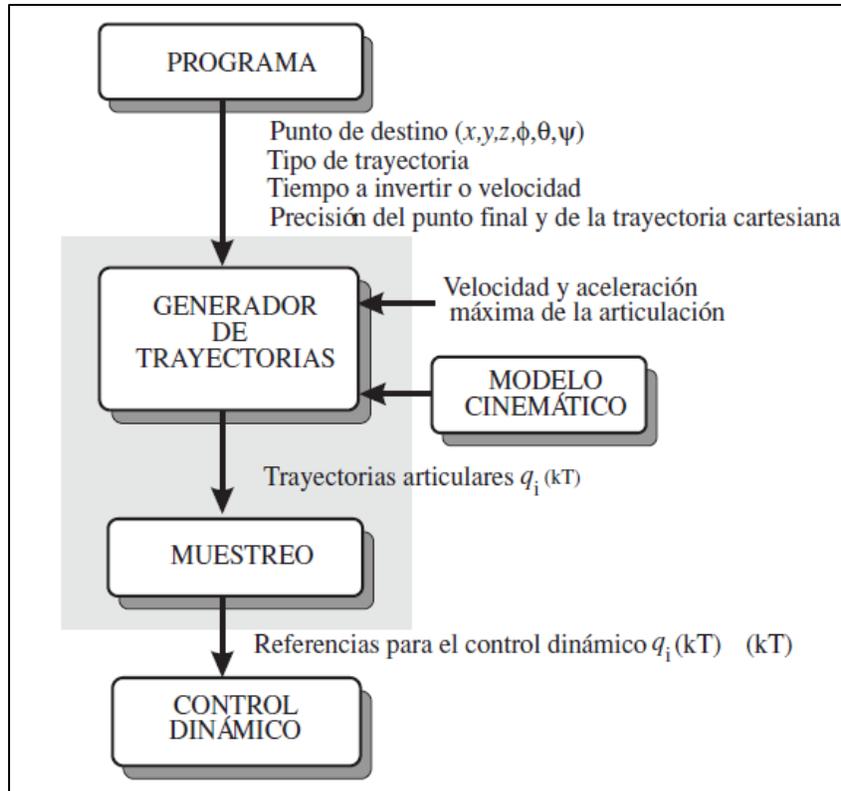


Fig. 5.1 – Diagrama de bloques para un generador de trayectoria [18].

La trayectoria pick and place se desarrolla a través de una interpolación splin cúbica que consiste en un polinomio de tercer grado que proporciona una curva suave entre un punto inicial y final dados. La ecuación principal para comenzar el estudio se presenta en la Ec. (4.1):

La expresión de la trayectoria que une dos puntos adyacentes (q^i, q^{i+1}) será:

$$q(t) = a + b(t - t^i) + c(t - t^i)^2 + d(t - t^i)^3 \quad t^i < t < t^{i+1} \quad (4.1)$$

donde,

$$a = q^i \quad (4.2)$$

$$b = \dot{q}^i \quad (4.3)$$

$$c = \frac{3}{T^2}(q^{i+1} - q^i) - \frac{1}{T}(\dot{q}^{i+1} + 2\dot{q}^i) \quad (4.4)$$

$$d = -\frac{2}{T^3}(q^{i+1} - q^i) + \frac{1}{T^2}(\dot{q}^{i+1} + 2\dot{q}^i) \quad (4.5)$$

$$T = t^{i+1} - t^i \quad (4.6)$$

La Ec. (4.1) muestra la posición de una articulación con respecto al tiempo, los valores de a, b, c y d son los componentes del polinomio de orden 3. Para generar esta trayectoria se genera un código en Matlab, que permite el cálculo de las posiciones de las 5 primeras articulaciones del manipulador. El movimiento de la articulación 6 no se simula en esta tesis, ya que su función es abrir o cerrar la pinza, por lo tanto, no intervendrá en conocer la posición final del efector final.

Además, al obtener la primera y segunda derivada de la Ec. (4.1) se obtienen las velocidades y aceleraciones respectivamente para una articulación del robot antropomórfico. Para la simulación se utilizan valores de tiempo inicial y final, además de posiciones iniciales y finales de cada articulación, estas se presentan en la Tabla 5.1:

Tabla 5.1 – Valores iniciales y finales para la simulación de trayectoria

Variable	Valor inicial	Valor final
q_1	90°	0°
q_2	270°	180°
q_3	90°	180°
q_4	180°	270°
q_5	180°	270°
t	0 [s]	10 [s]

A continuación, se presentan las gráficas de movimiento de las articulaciones respecto a los puntos iniciales y finales dados en la Tabla 5.1 y además los resultados de la trayectoria pick and place. En cada gráfico presentado a continuación, la posición estará dada por la curva en azul, velocidad en curva segmentada roja y aceleración en curva punteada naranja.

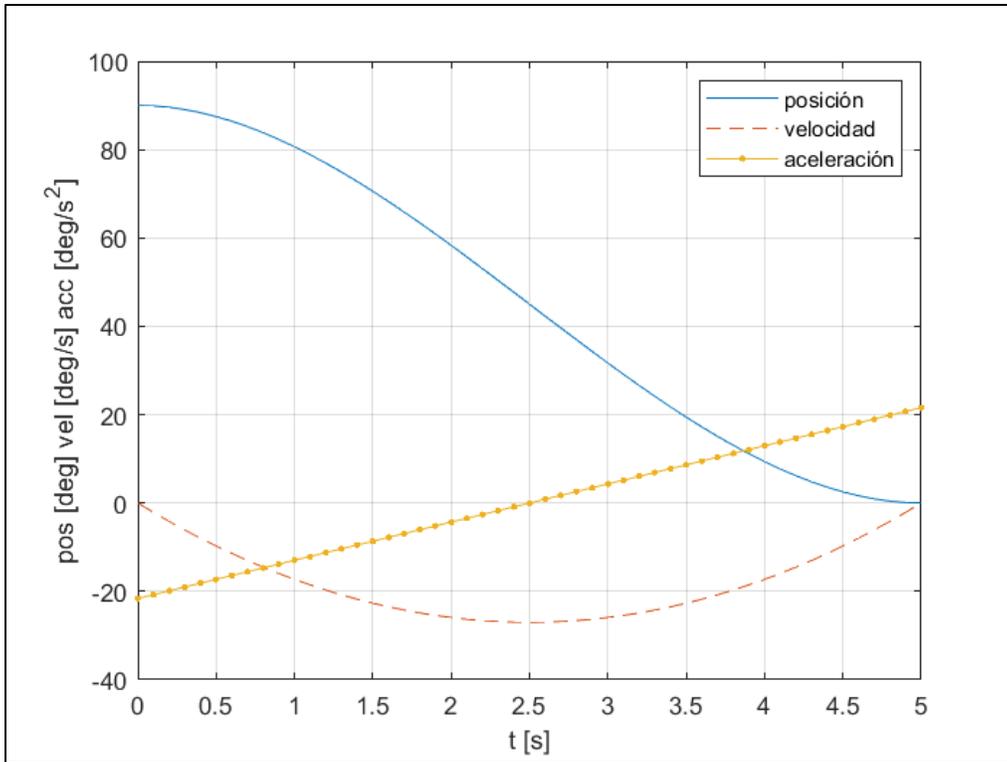


Fig. 5.2 – Movimiento de la articulación 1.

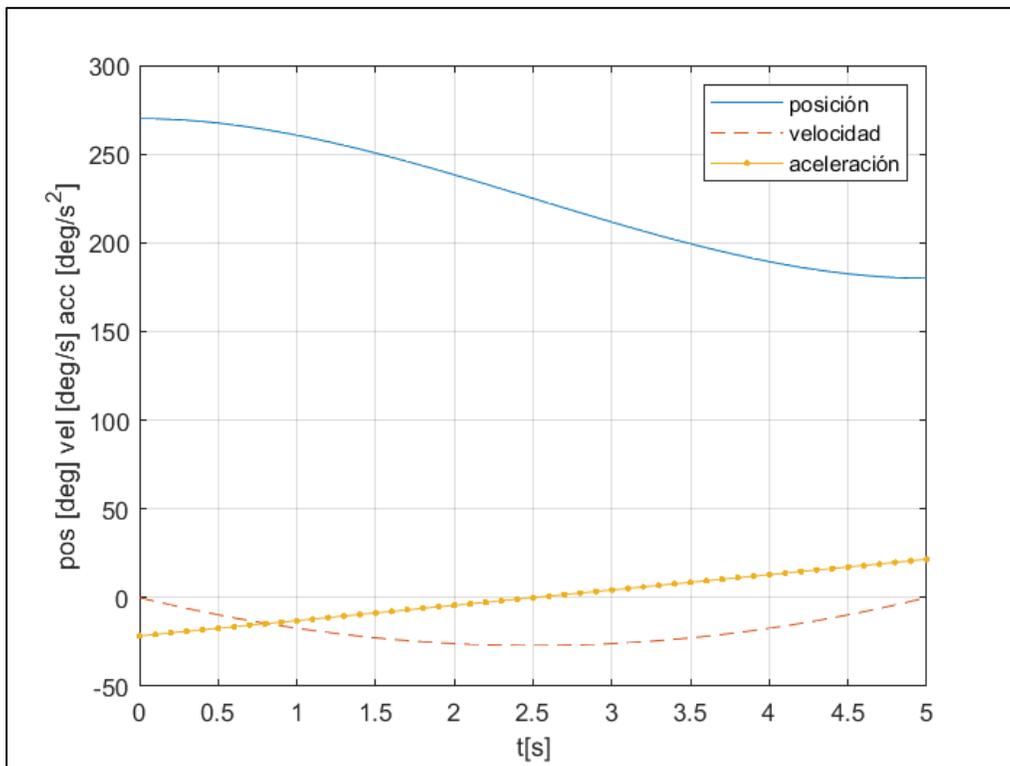


Fig. 5.3 – Movimiento de la articulación 2.

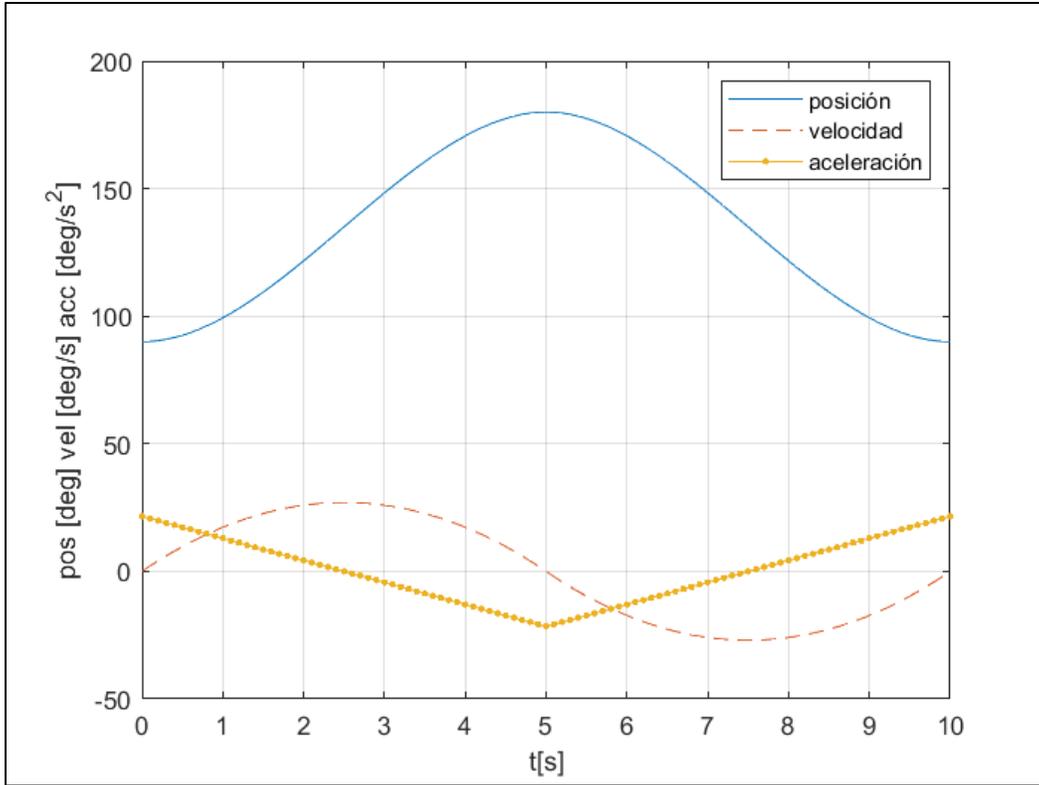


Fig. 5.4 – Movimiento de la articulación 3.

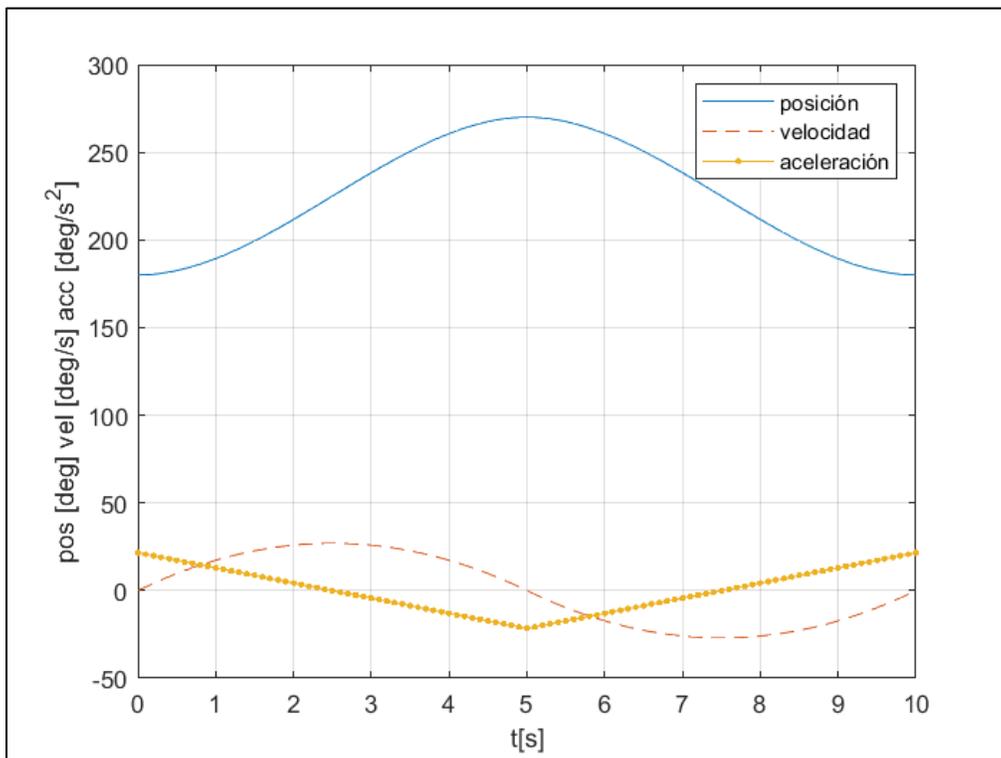


Fig. 5.5 – Movimiento de la articulación 4.

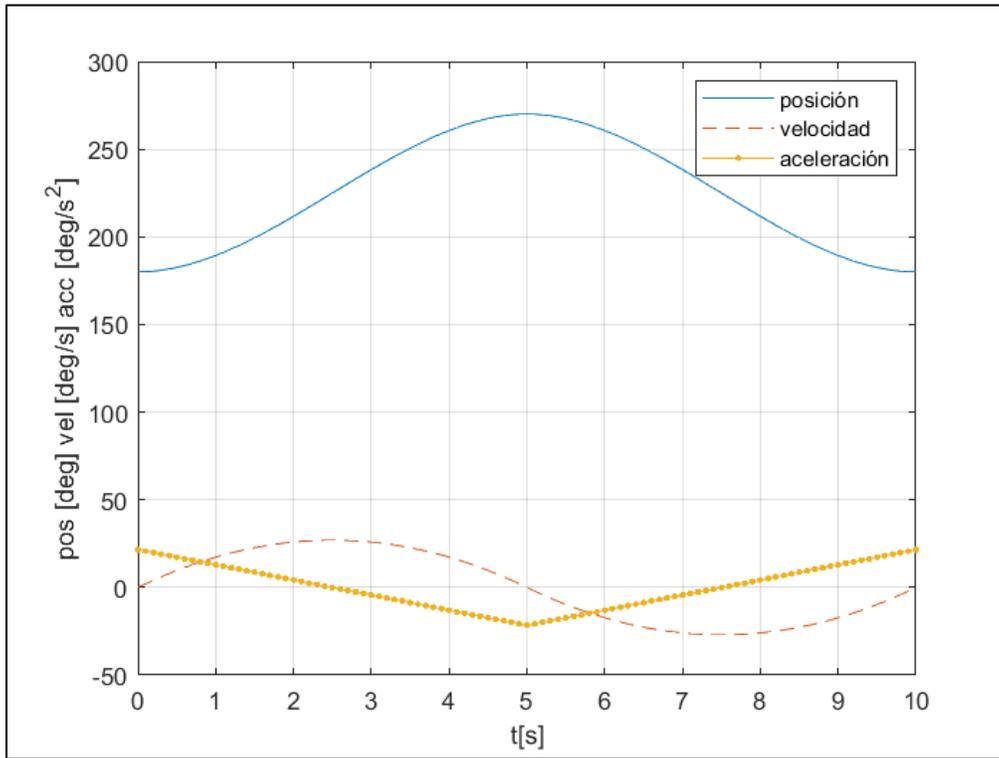


Fig. 5.6 – Movimiento de la articulación 5.

Al observar las gráficas se observa que las curvas azules de posición realizan la trayectoria establecida, siguiendo la referencia dada a través de una curva suave al utilizar un polinomio de tercer grado. Las curvas de velocidad presentadas en líneas segmentadas también siguen la referencia dada al utilizar el polinomio de tercer grado, ya que se desea tener una curva suave y que además el manipulador tenga un velocidad inicial y final igual a cero, para evitar movimientos bruscos, pero teóricamente la velocidad debería ser discreta, esto se demostrará más adelante. Finalmente, las aceleraciones obtenidas en líneas con puntos naranjos también son lo esperado para este tipo de interpolación, ya que en estas gráficas se muestra la aceleración como una recta, pero en realidad las aceleraciones tienden al infinito, esto se podrá demostrar más adelante. A continuación, se muestran las gráficas finales de la trayectoria pick and place con un interpolador cúbico.

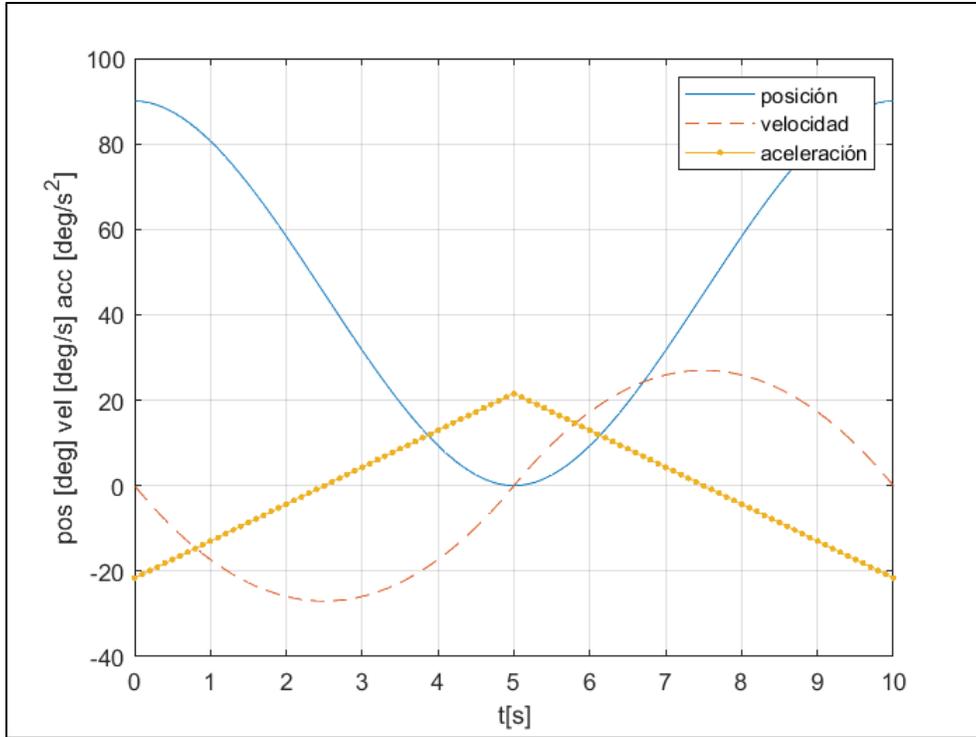


Fig. 5.7 – Trayectoria Pick and Place articulación 1.

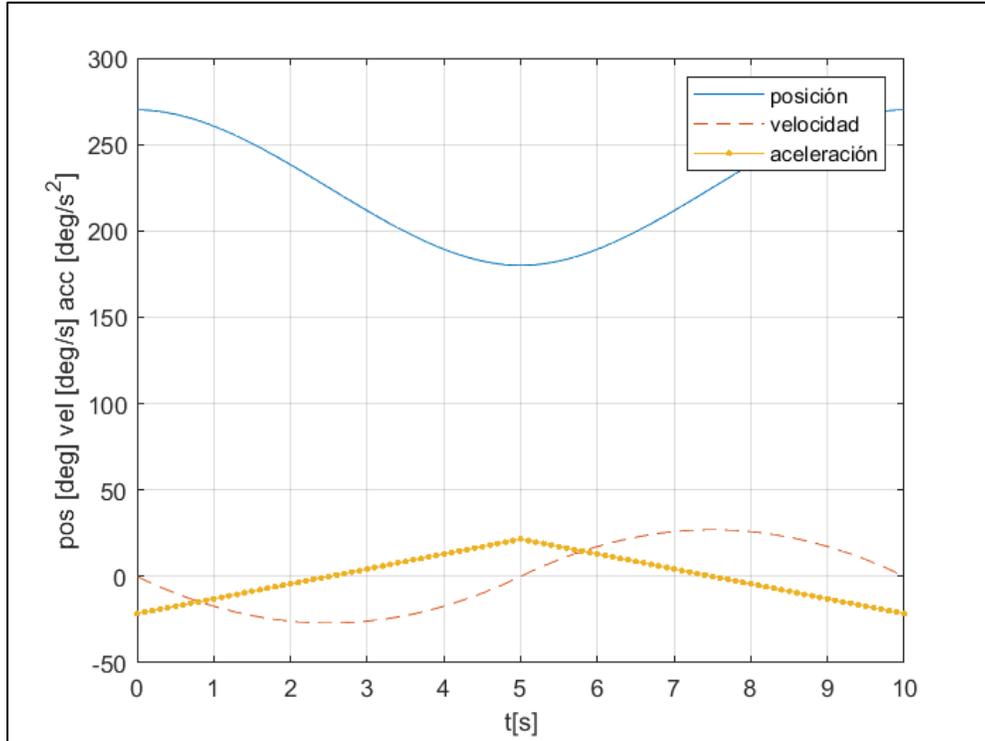


Fig. 5.8 – Trayectoria Pick and Place articulación 2.

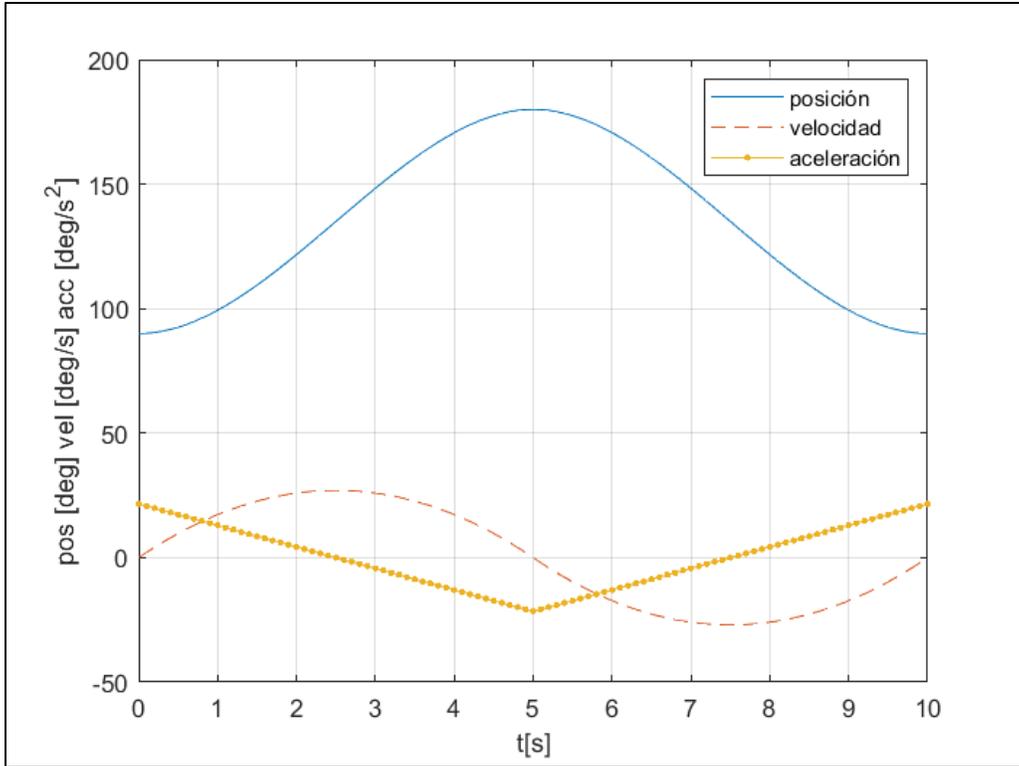


Fig. 5.9 – Trayectoria Pick and Place articulación 3.

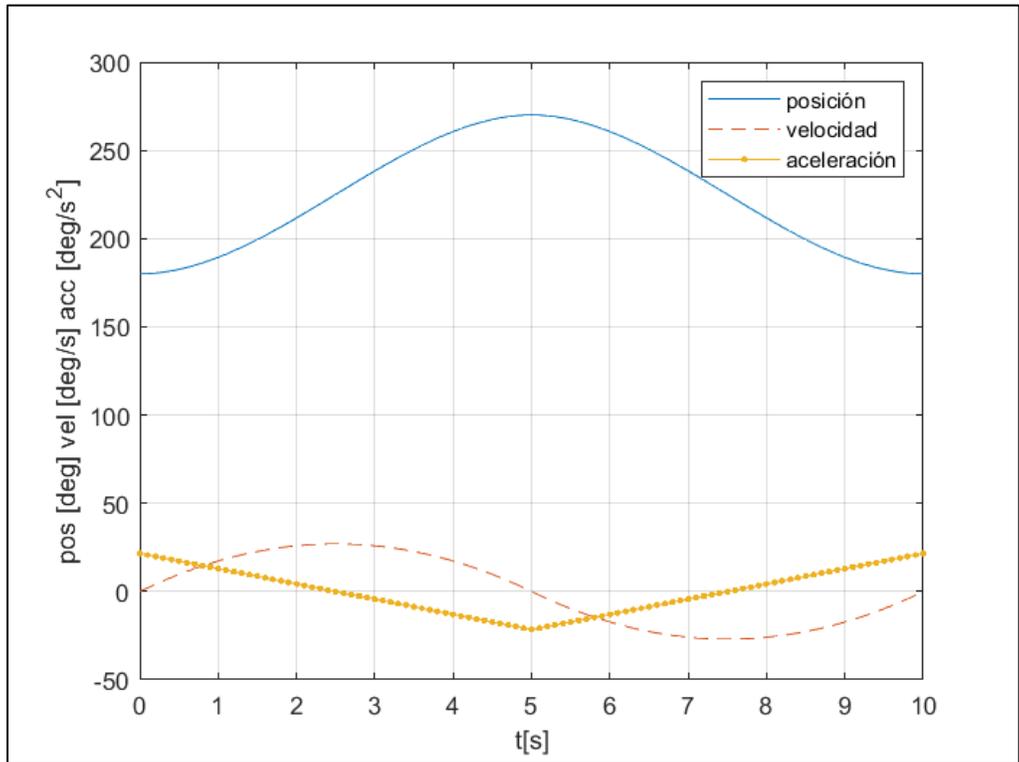


Fig. 5.10 – Trayectoria Pick and Place articulación 4.

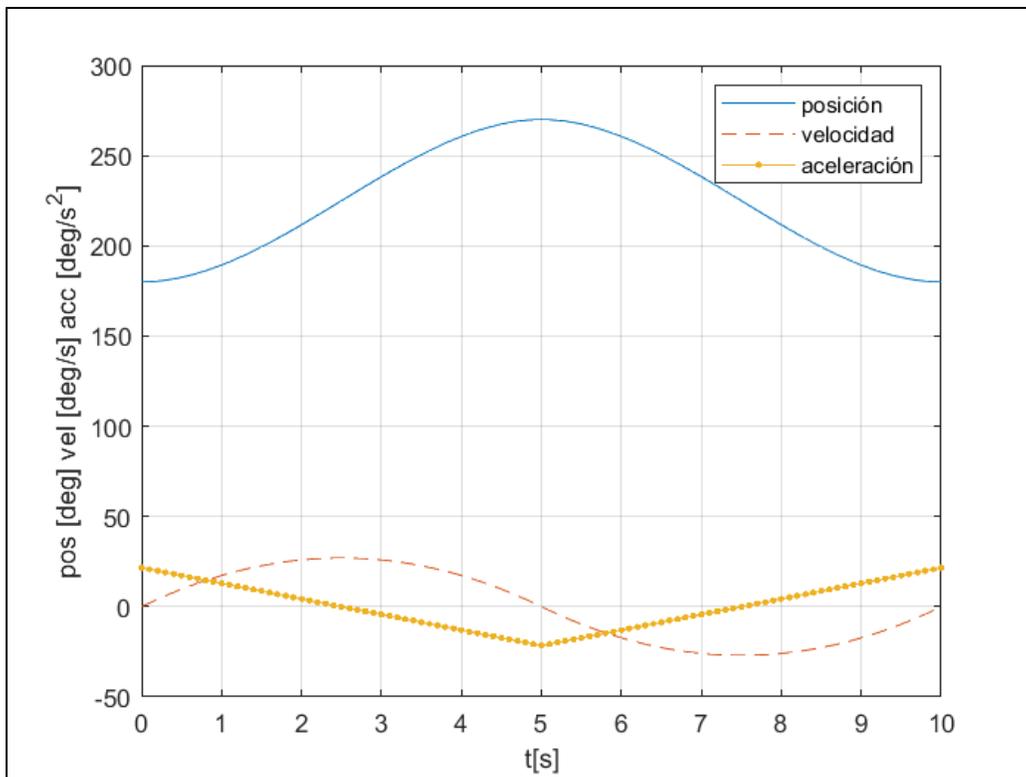


Fig. 5.11 – Trayectoria Pick and Place articulación 5.

En las figuras presentadas con la Trayectoria Pick and Place, se observa como la curva de posición de color azul, pasa por los puntos iniciales y finales establecidos, también se observa la curva de retorno volviendo al punto inicial. En tanto, para las velocidades se obtienen los resultados esperados, ya que se genera una continuidad para la curva pasando por el cero cada vez que se llega a un punto final o inicial, esto significa que la velocidades iniciales y finales de cada actuador serán cero a través de una curva suave. Las aceleraciones al igual que en el punto anterior generan una recta que sube o baja, dependiendo del movimiento de la posición y velocidad, esto se produce ya que no hay una interpolación en las aceleraciones y se generan rectas, también se observa que el manipulador deja de acelerar cuando está en el punto medio de las curvas de velocidad. De esta manera se confirma el buen funcionamiento de la trayectoria con interpolador cúbico.

A continuación, se muestra el resultado de la trayectoria pick and place creada a través de los puntos mostrados en la Tabla 5.1, el movimiento final se puede apreciar en el video explicativo [76], pero para demostrar que el manipulador cumple con la trayectoria se presenta una serie de fotogramas que muestran el correcto funcionamiento del generador de trayectoria.

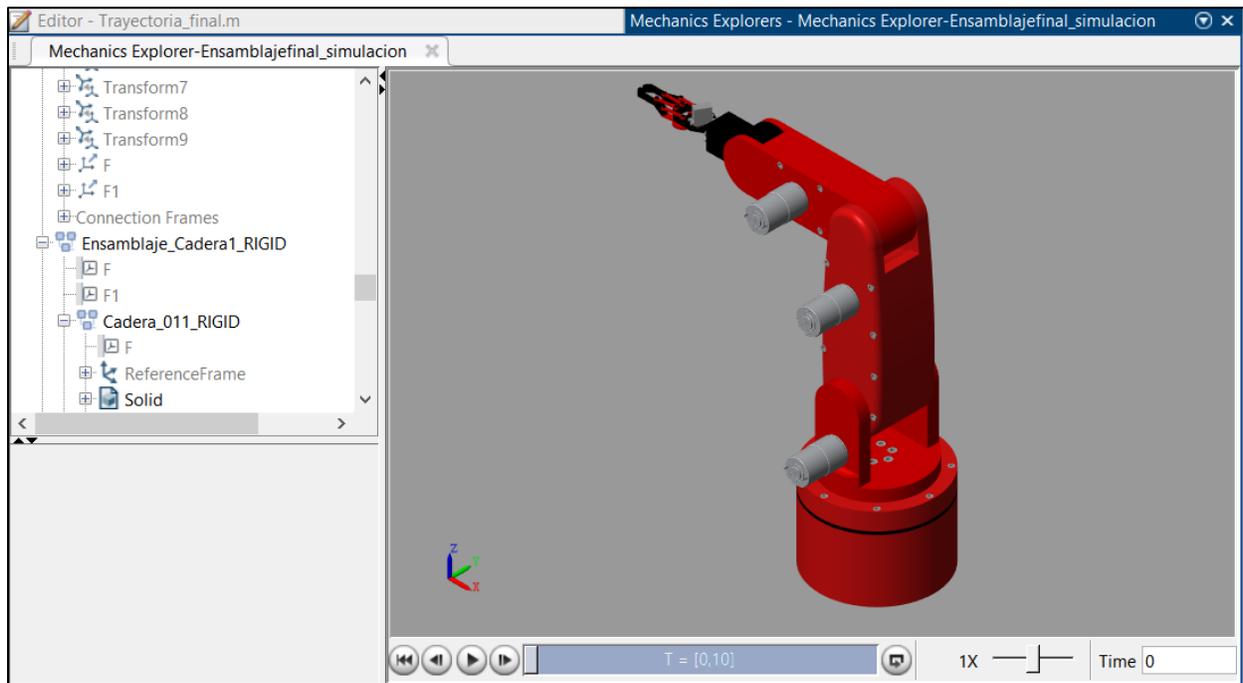


Fig. 5.12 – Posición inicial del manipulador.

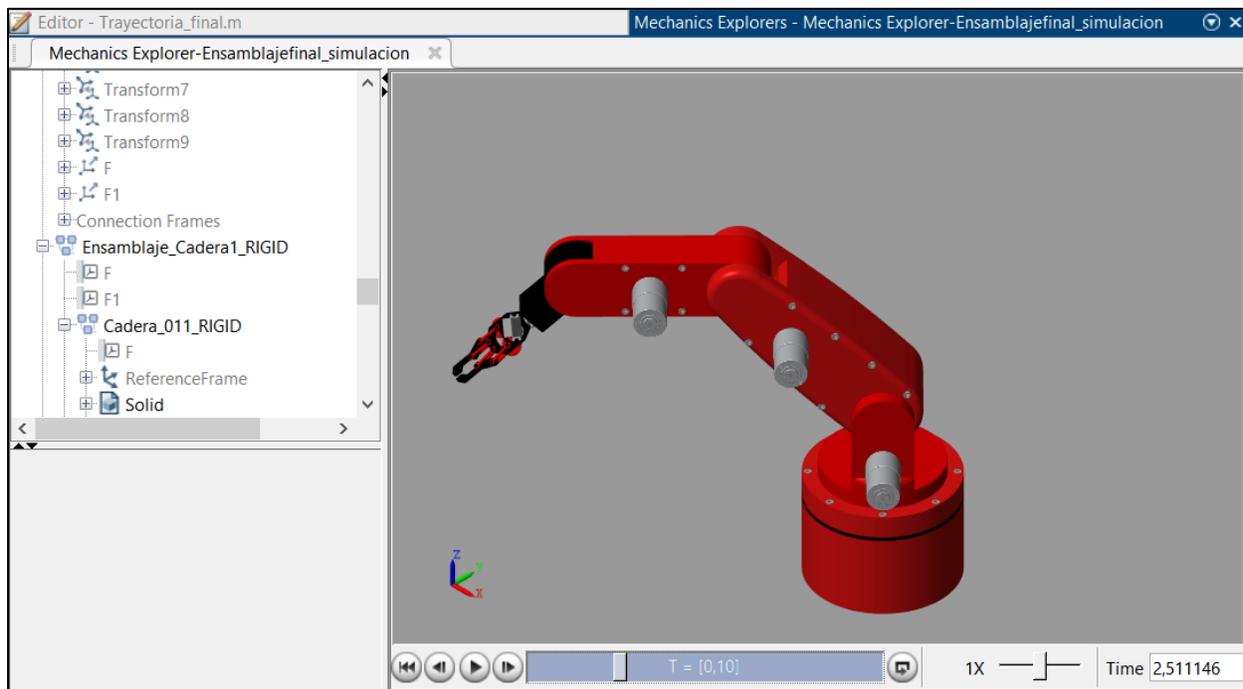


Fig. 5.13 – Posición del manipulador transcurridos 2.5 [s].

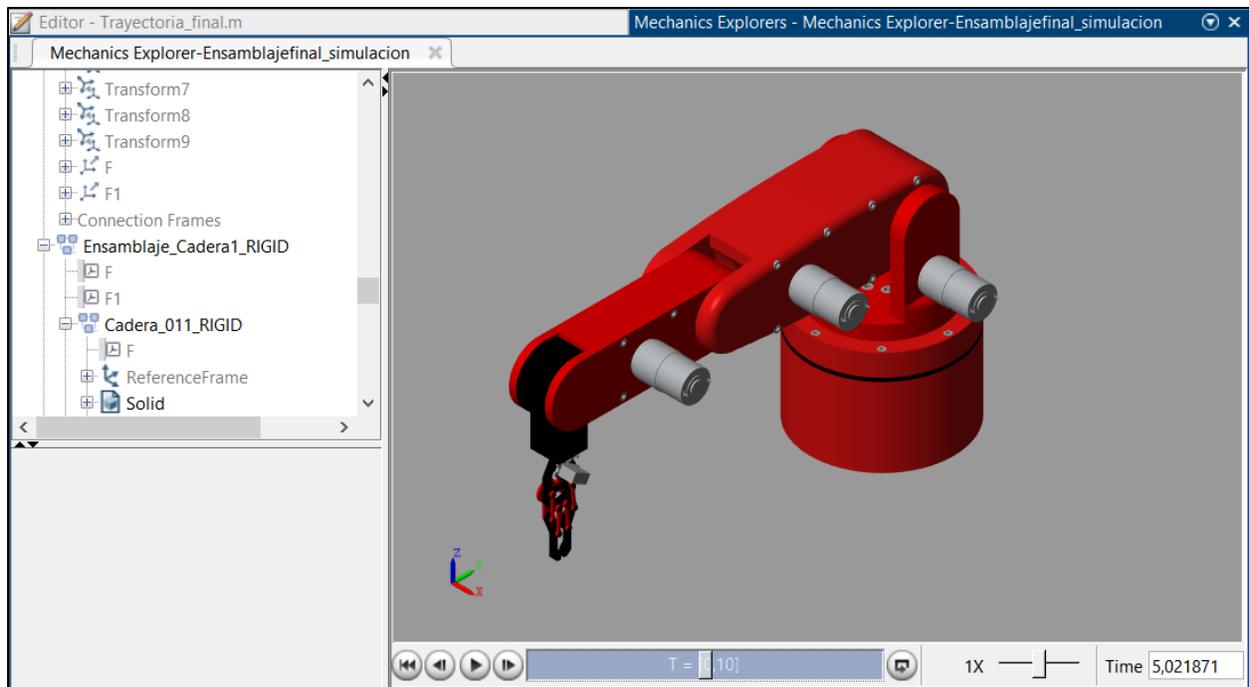


Fig. 5.14 – Posición final del manipulador.

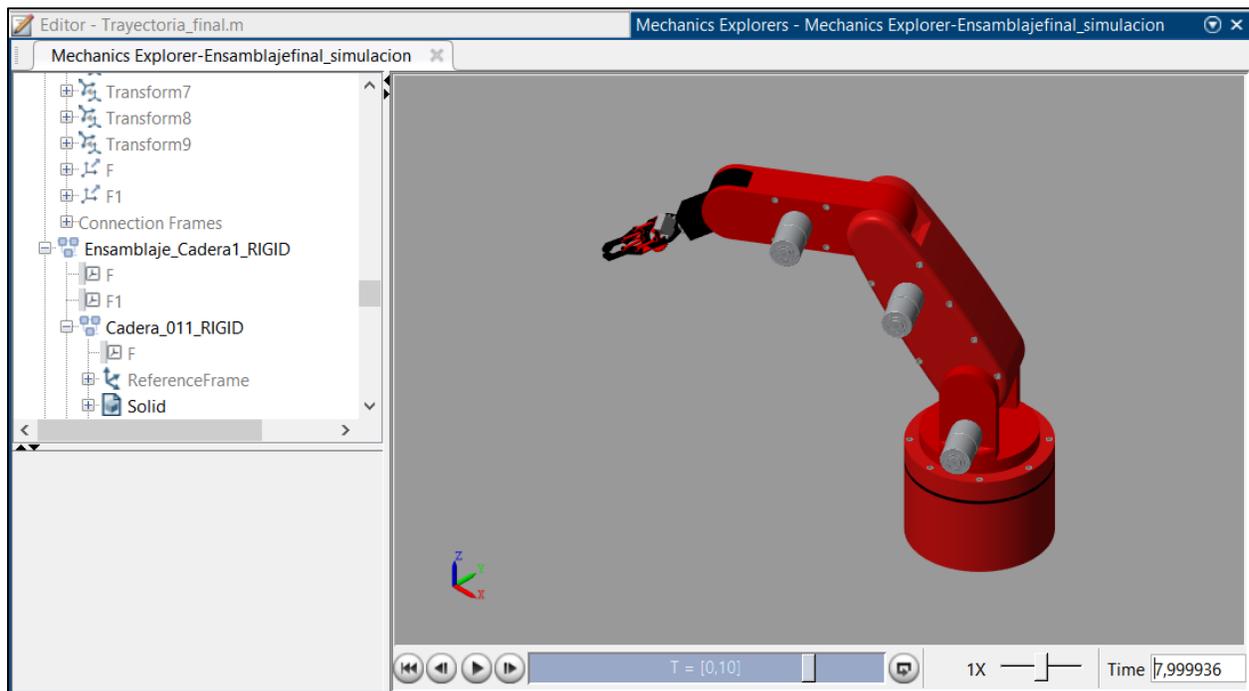


Fig. 5.15 – Posición del manipulador transcurridos 8 [s].

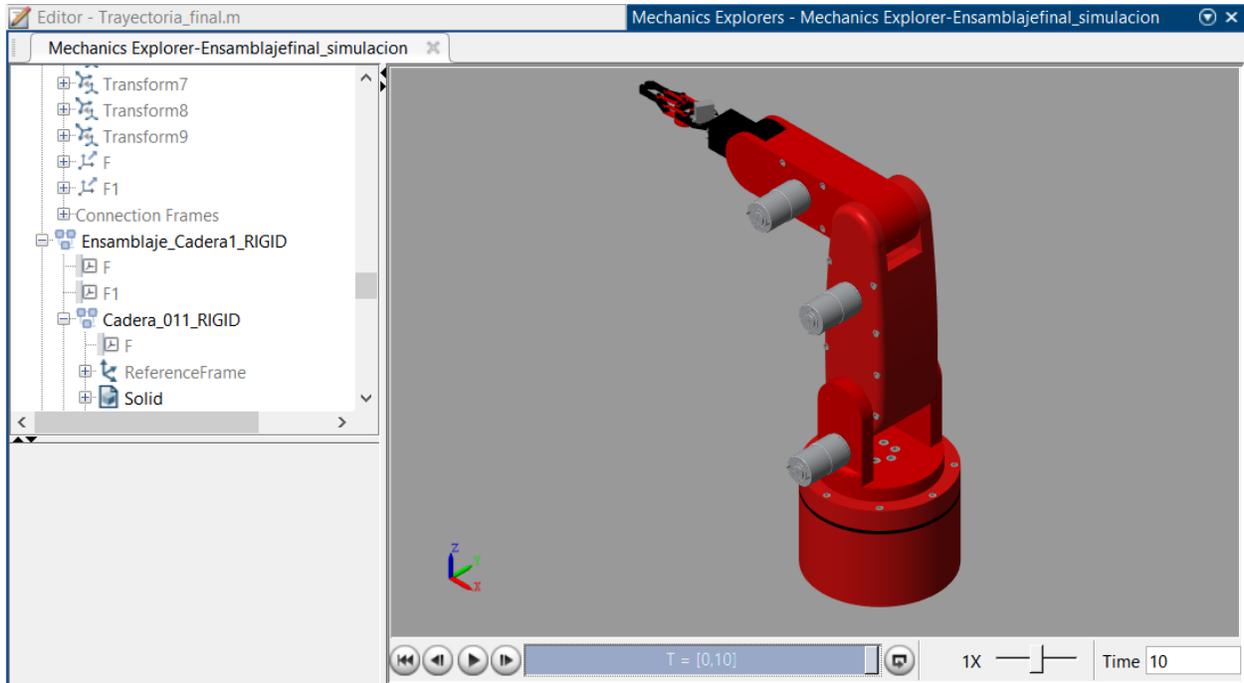


Fig. 5.16 – Posición del manipulador transcurridos 10 [s] (regreso a posición inicial).

Para corroborar el perfecto funcionamiento de la trayectoria, se presenta a continuación las gráficas obtenidas de posición, velocidad y aceleración, directamente desde el Simscape Multibody, de esta manera se pueden comparar los resultados obtenidos. Este procedimiento se realiza solo para la primera articulación, ya que para las demás los resultados serán iguales.

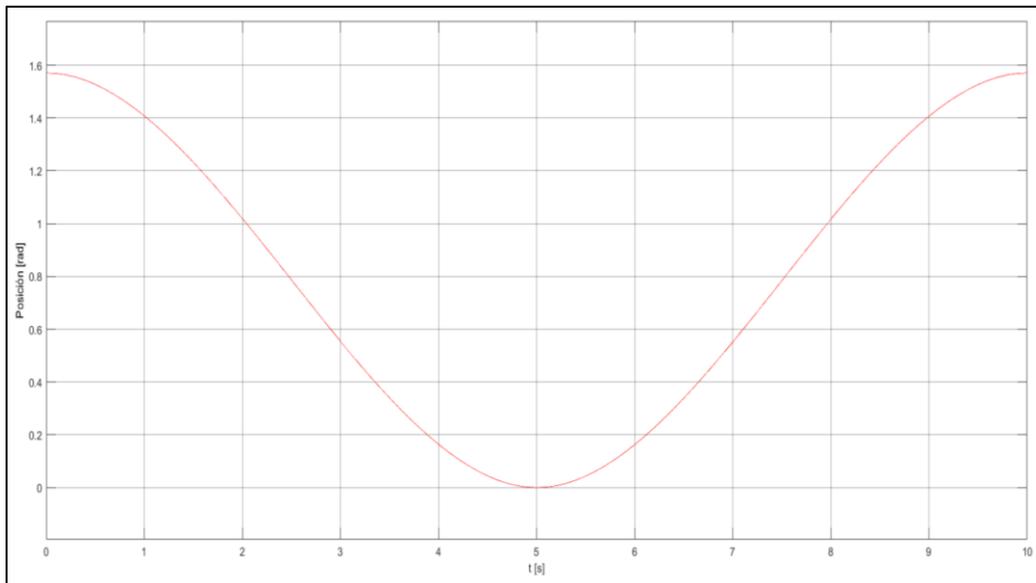


Fig. 5.17 – Posición de trayectoria Pick and Place articulación 1 en Simscape Multibody.

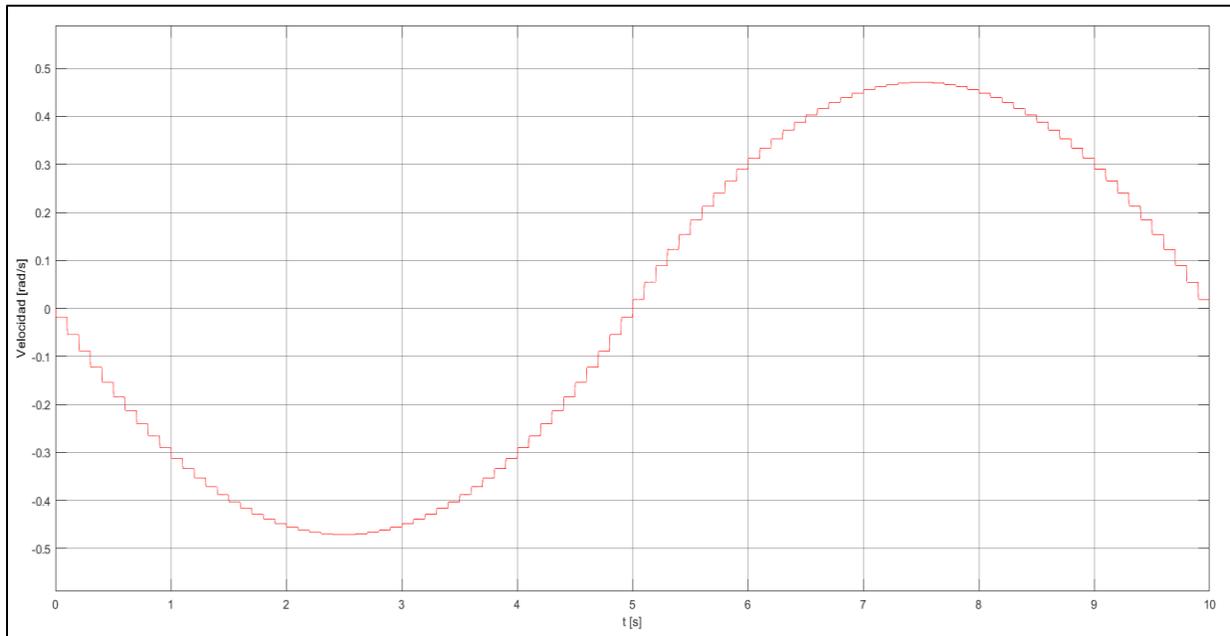


Fig. 5.18 –Velocidad de trayectoria Pick and Place articulación 1 en Simscape Multibody.

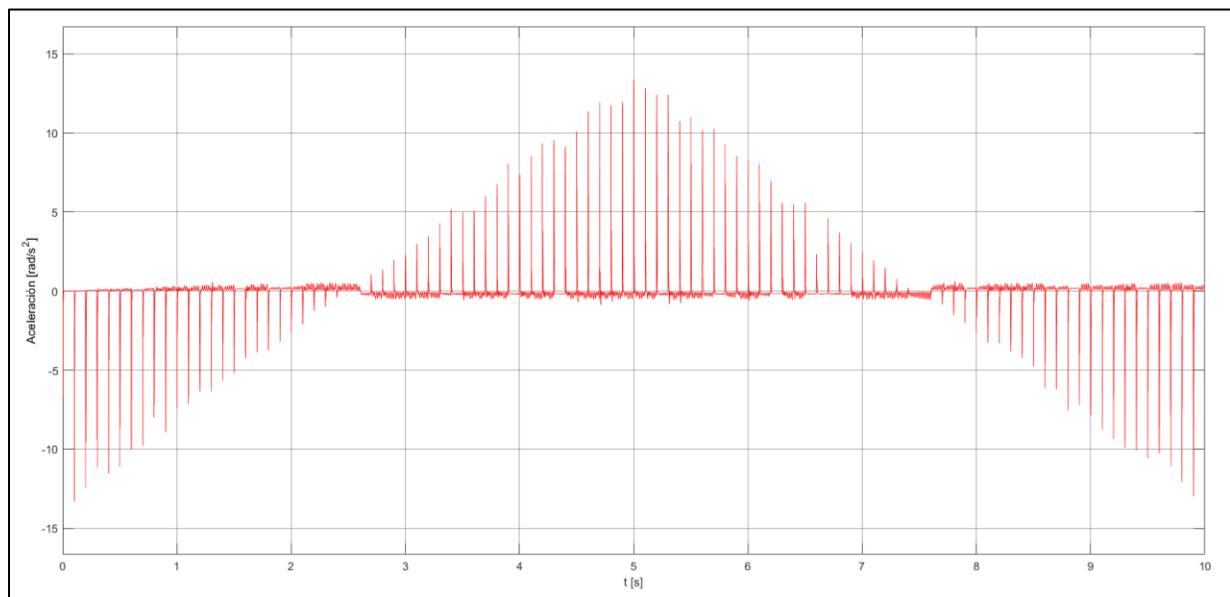


Fig. 5.19 –Aceleración de trayectoria Pick and Place articulación 1 en Simscape Multibody.

Finalmente, se presentan las gráficas obtenidas en Simscape Multibody con los torques del manipulador generados automáticamente, considerando las dinámicas internas del diseño mecánico. Los resultados obtenidos son los esperados, la posición genera una curva suave que

pasa por los puntos de interés, la velocidad es discreta, pero al utilizar un interpolador cúbico genera que la curva se suavice generando pequeño escalones que siguen la referencia dada, por último, la aceleración es una serie de impulsos infinitos que no siguen una referencia deseada, ya que no se genera una interpolación que actúe sobre ella, al no tener una interpolación sobre la aceleración se aleja de lo esperado para controlar un manipulador en la vida real, por consiguiente estas aceleraciones afectan directamente a los torques obtenidos, que se presentan en la siguiente subsección. Además, en el análisis de las velocidades los valores serán discretos o escalones, ya que son las derivadas de los puntos de posición generados en la trayectoria. Estos resultados se obtienen asumiendo que las dinámicas internas de Simscape siguen las referencias dadas, pero esto en la vida real no se verán reflejados de la misma manera. En el caso de las aceleraciones, los impulsos que se generan luego de derivar la velocidad se verán filtrados, de cierta manera, por las dinámicas asociadas al manipulador.

5.3. Torques

En el capítulo 3 se calcularon torques necesarios para mover al manipulador, en las articulaciones 2, 3, 4 y 5 aplicando una carga máxima de 500 gramos. Simscape Multibody permite generar torques automáticos según el diseño mecánico diseñado y exportado a Matlab, entonces se pueden obtener valores aproximados de los torques que necesita el robot para realizar una trayectoria y de esta manera, se puede saber si los torques calculados anteriormente son válidos.

Las gráficas obtenidas para las articulaciones mencionadas anteriormente se presentan a continuación:

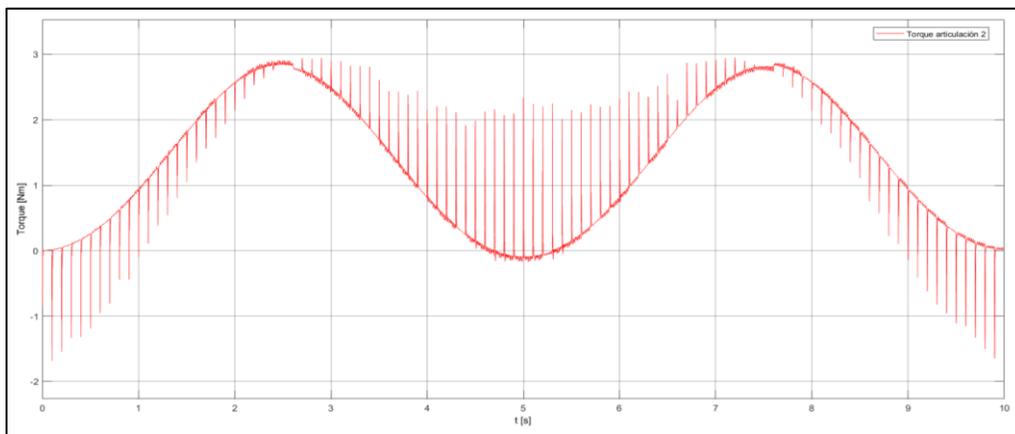


Fig. 5.20 –Torque para la articulación 2.

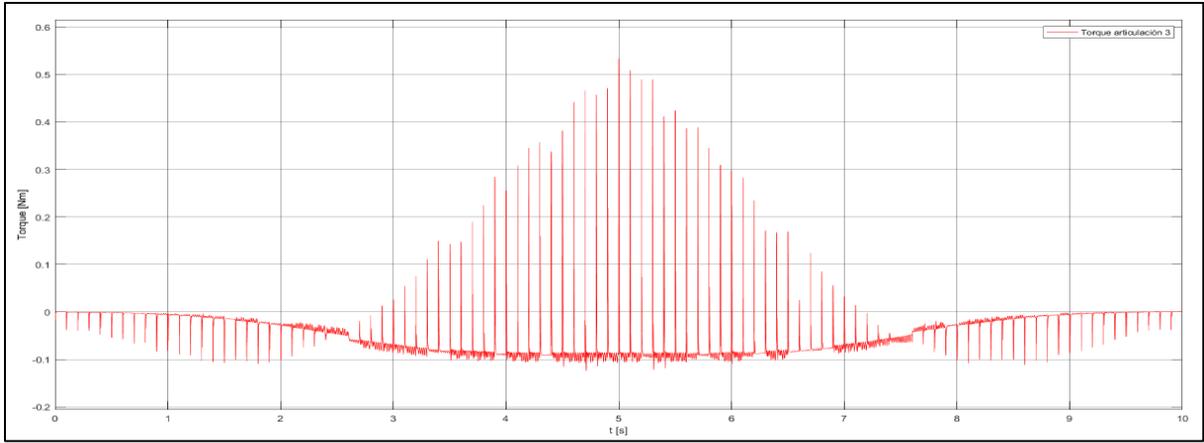


Fig. 5.21 –Torque para la articulación 3.

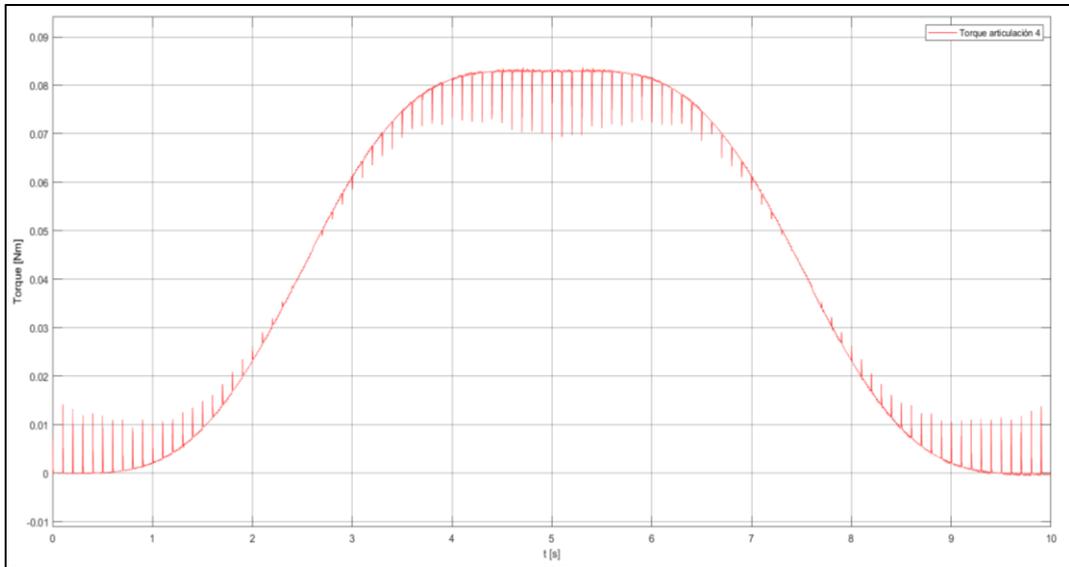


Fig. 5.22 –Torque para la articulación 4.

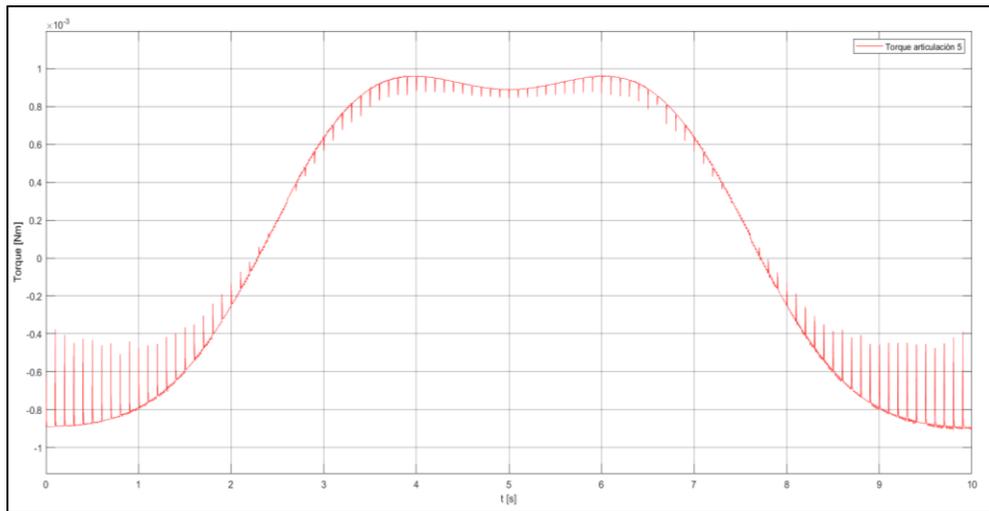


Fig. 5.23 –Torque para la articulación 5.

Los torques obtenidos son menores a los calculados en el capítulo 3 de esta memoria de título, en la articulación 2 se observa que el toque máximo es menor a los 3 [Nm], este valor es el que más se acerca al calculado, pero el cálculo realizado fue de manera estática y aplicando una carga de 500 grs. Esto podría explicar además porque los torques de las articulaciones 3, 4 y 5 son mucho más bajos de los calculados anteriormente, pero se observa que los torques actúan acorde a la trayectoria que se está dando. Las aceleraciones al ser una serie de impulsos infinitos afectan directamente a los torques, esto se puede evidenciar en todas las gráficas de torque presentadas anteriormente.

5.4. Discusión

En este capítulo se simula al manipulador para un rutina o generación de trayectoria, para corroborar que en el tiempo que transcurre la simulación (10 [s]) los torques obtenidos son menores que los calculados analíticamente, de esta manera se puede verificar el diseño mecánico realizado. Con respecto a las dinámicas asociadas, al generar una trayectoria con interpolador cúbico estos valores no fueron utilizados, ya que asume que las velocidades son bajas y no afectarían las fuerzas de Coriolis y centrípetas, además, que se considera la dinámica interna que genera automáticamente el software de Simscape. Para obtener un análisis en mayor profundidad del comportamiento del robot se deberían utilizar las dinámicas asociadas al robot, así se puede estudiar y verificar el funcionamiento de un robot antropomórfico de 6 GDL desde un control de tipo dinámico.

Es importante mencionar que, faltan las dinámicas de los controladores asociadas al control cinemático, por lo tanto, un cambio escalón en la entrada, por ejemplo, de posición, en la práctica no seguirá la referencia como un escalón, sino que, va a tender a seguir una dinámica y esto suaviza las curvas que se obtendrán de aceleraciones.

La trayectoria planteada se desarrolla de buena manera y se obtienen los resultados esperados, pero se aleja de lo que debería ser el movimiento de un manipulador real al no interpolar las aceleraciones, para este caso existe un interpolador quintico, que consiste en un polinomio de grado 5, que incorpora las 2 variables faltantes de aceleración inicial y final, para suavizar la curva y obtener valores reales respecto a lo que debería ser un manipulador. Este punto se plantea como un trabajo futuro.

Los torques calculados se verifican con la dinámica interna que genera Simscape Multibody, por lo tanto, de manera aproximada, se validan los cálculos de torques realizados de manera estática con una carga de 0.5 kg. Este era el punto más importante de verificar para validar el diseño mecánico creado, por lo tanto, se puede discutir que el manipulador está bien diseñado.

Capítulo 6. Conclusiones

6.1. Introducción

En esta memoria de título se presentó el diseño mecánico para un robot antropomórfico de 6 GDL, basado en la clasificación de robots industriales [18]. Se presentó un diseño con un cálculo de torques estáticos para comparar estos valores con las simulaciones de trayectoria y de esa manera verificar que el diseño mecánico se realizó de buena manera. Además, se presentaron los cálculos asociados a las relaciones de transmisión utilizados en las articulaciones 1, 2, 3 y 4, de esta manera, la velocidad de manipulador sería baja, por ende, se podría asumir que las fuerzas de Coriolis y centrípeta no afectarían en las simulaciones.

Se presentó una caracterización matemática completa para un manipulador antropomórfico de 6 GDL, considerando las cinemáticas directa e inversa y, además, su dinámica. Estos cálculos matemáticos se verificaron con simulaciones generadas en Matlab, de esta manera, se verificó el perfecto funcionamiento de la caracterización matemática realizada.

Finalmente, se generó una trayectoria con una interpolación cúbica, para corroborar los movimientos del manipulador y los torques que actúan sobre las articulaciones del robot, asumiendo que los actuadores utilizados tienen un comportamiento ideal.

6.2. Conclusiones

Aludiendo a los objetivos planteados en esta memoria de título, el objetivo principal de diseñar y generar una trayectoria para un robot antropomórfico de 6 GDL se cumple, estudiando sus movimientos a través de las posiciones, velocidades y aceleraciones generadas en la simulación. Además, los objetivos específicos planteados también se completan de manera exitosa, cumpliendo con cada uno de ellos.

Con respecto al diseño mecánico, el movimiento final de la pinza de abrir y cerrar, no se simuló por problemas al generar una relación de revolución con los engranajes creados para dar el movimiento esperado, ya que al realizarlo, el software de Matlab reconocía más de una articulación en la pinza, por lo cual no se podía generar el estudio centrado solo en abrir o cerrar la pinza, de igual manera este movimiento no afecta en los estudios planteados, ya que la pinza igual se muestra en las simulaciones y se observa que llega a los puntos planteados porque el último grado de

libertad no proporciona una orientación en el espacio de la pinza. Además, el manipulador logra levantar una carga máxima de 0,5 [kg] obteniendo una deformación máxima de 13,64 [MPa], lo cual no afectará al diseño creado ya que el material empleado, plástico ABS, tiene un límite elástico de 32 [MPa].

Con respecto a la caracterización matemática, los cálculos obtenidos se pudieron corroborar con las simulaciones creadas en Matlab. La cinemática directa al posicionarse de la misma manera que en el diseño CAD creado, muestra que la posición final del elemento terminal es la misma. De la misma manera, se corrobora la cinemática inversa al ingresar los valores finales de la posición final del elemento terminal, entregando los mismos ángulos de la cinemática directa, por lo tanto, ambas cinemáticas presentaron un buen funcionamiento. La dinámica asociada a este manipulador se calcula de manera general y simbólica, esto quiere decir que quedan expresado en términos generales y no se utilizaron los parámetros obtenidos del diseño y las cinemáticas, esto no se realizó por un tema de complejidad mayor y porque al no realizarlo, no afectaba con cumplir con los objetivos de esta memoria.

Finalmente, la generación de trayectoria se genera de buena manera, permite visualizar el perfecto comportamiento de la posición, alcanzando las referencias establecidas. Lo mismo pasa con la velocidad, al derivar la posición con respecto al tiempo, se genera un escalón de velocidad, pero al utilizar un interpolador cúbico, este escalón se convierte en pequeños escalones que siguen la referencia establecida para la velocidad. Las aceleraciones muestran el comportamiento esperado para este tipo de interpolación, ya que al derivar la velocidad se obtiene un impulso infinito en la aceleración, esto se pudo verificar y se explicó porque sucede este fenómeno. Con respecto a los torques, estos se vieron afectados directamente con los valores de las aceleraciones, pero de igual manera los valores obtenidos fueron más bajos que los calculados. Para el caso de la articulación 2, el torque ejercido en simulación no supera los 3,63 [Nm] nominales calculados analíticamente y de manera estática. Para las articulaciones 3, 4 y 5 los torques simulados son aproximadamente un 80% más pequeños para una trayectoria generada en 10 [s]. Por lo tanto, se pudo llegar a la conclusión de que los torques calculados eran verídicos y se podían utilizar para escoger los actuadores, que, en este caso, serán 4 motores CC para las 4 primeras articulaciones y 2 servomotores para las articulaciones finales, que generan el giro de la muñeca y el abrir y cerrar de la pinza.

6.3. Trabajos futuros

Al finalizar este trabajo de memoria de título, se observan diferentes trabajos a futuro, que se podrían implementar para mejorar lo realizado, estos trabajos se presentan a continuación:

- Implementar el manipulador a través de su diseño mecánico, adaptarlo en sus medidas para poder imprimir las partes del robot en una impresora 3D.
- Estudiar el cambio de actuadores, buscar actuadores que permitan un movimiento más suave y que se puedan controlar con menor complejidad.
- Generar una cotización para obtener los cálculos finales de costos para la fabricación de este manipulador.
- Generar una interpolación quintica que permita interpolar los valores iniciales y finales de las aceleraciones.
- Generar una interpolación lineal y comparar los 3 interpoladores mencionados para decidir cuál es el mejor y se acerca más a la realidad del funcionamiento de un manipulador.
- Crear lazos de control cerrados, retroalimentando la posición para generar un control cinemático o dinámico completo, que incluya las dinámicas que actúan en el manipulador.

Bibliografía

- [1] Caparroso, I. O., Avilés, O., & Bello, J. H. (1999). Una introducción a la robótica industrial. *Ciencia E Ingeniería Neogranadina*, 8, 53-67.
- [2] Reyes, F. (2011). *Robótica-control de robots manipuladores*. Alfaomega grupo editor.
- [3] Román Graván, P., Hervás Gómez, C., & Guisado Lizar, J. L. (2017). Experiencia de innovación educativa con robótica en la Facultad de Ciencias de la Educación de la Universidad de Sevilla (España).
- [4] Salamanca, M. L. P., Lombana, N. B., & Holguín, W. J. P. (2010). Uso de la robótica educativa como herramienta en los procesos de enseñanza. *Ingeniería Investigación y Desarrollo: I2+ D*, 10(1), 15-23.
- [5] Campos, C. L. (2019). *Los Desafíos de la Robótica en Chile: El caso de la movilidad urbana* (Doctoral dissertation, Pontificia Universidad Católica de Valparaíso).
- [6] Basco, A. I., Beliz, G., Coatz, D., & Garnero, P. (2018). *Industria 4.0: fabricando el futuro* (Vol. 647). Inter-American Development Bank.
- [7] Figueroa, M. Á. H., Báez, R. G. V., & Rangel, J. O. C. Propuesta de asistente virtual basado en inteligencia artificial en la integración de un brazo robótico para aplicaciones en la industria 4.0.
- [8] Moreno, I., Muñoz, L., Serracín, J. R., Quintero, J., Patiño, K. P., & Quiel, J. (2012). La robótica educativa, una herramienta para la enseñanza-aprendizaje de las ciencias y las tecnologías. *Teoría de la Educación. Educación y Cultura en la Sociedad de la Información*, 13(2), 74-90.
- [9] Mercader Uguina, J. R. (2017). El impacto de la robótica y el futuro del trabajo. *Revista de la Facultad de Derecho de México*, 67(269), 149-174.
- [10] Bravo Sánchez, F. Á., & Forero Guzmán, A. (2012). La robótica como un recurso para facilitar el aprendizaje y desarrollo de competencias generales.
- [11] Ramírez, P. A. L., & Sosa, H. A. (2013). Aprendizaje de y con robótica, algunas experiencias. *Revista Educación*, 37(1), 43-63.

- [12] Chavarría, M., & Saldaño, A. (2010). La robótica educativa como una innovativa interfaz educativa entre el alumno y una situación-problema. *Didasc@ lia: Didáctica y Educación* ISSN 2224-2643, 1(2), 1-12.
- [13] Lucena Olavarrieta, J. R., Coronel, P., & Pérez, O. (2007). Historia, evolución, estado actual y futuro de la cirugía robótica. *Revista de la Facultad de Medicina*, 30(2), 109-114.
- [14] Fernández, Rodríguez, Raúl, et al. *Robótica educativa*, RA-MA Editorial, 2015.
- [15] Kumar, Saha, Subir. *Introducción a la robótica*, McGraw-Hill España, 2010.
- [16] Arias, J. L. R., & Fonseca, A. R. (2012). Modelamiento matemático de la cinemática directa e inversa de un robot manipulador de tres grados de libertad. *Ingeniería solidaria*, 8(15), 46-52.
- [17] H.S. Lee, S.L. Chang. "Development of a cad/cae/cam system for a robot manipulator". *Journal of Materials Processing Technology*, núm. 140, 2003, pp. 100-104.
- [18] Barrientos, A. (2012). *Fundamentos de robótica (2a. ed.)*. McGraw-Hill España.
- [19] J. Xiao, W. Han y A. Wang, "Investigación de simulación de una cinemática de manipulador de seis grados de libertad basada en la caja de herramientas de MATLAB", Conferencia internacional de 2017 sobre sistemas mecatrónicos avanzados (ICAMechS), Xiamen, 2017, págs. 376-380, doi: 10.1109 / ICAMechS.2017.8316502.
- [20] J. Yu, D. You y J. Liu, "Análisis del método de cinemática inversa para manipulador de seis grados de libertad basado en MATLAB", 2017 3ra Conferencia Internacional IEEE sobre Ciencia de Control e Ingeniería de Sistemas (ICCSSE), Beijing, 2017, págs. 211-215, doi: 10.1109 / CCSSE.2017.8087925.
- [21] Y. Luan, W. Xu, J. Li, D. Zhou, H. Wang y H. Ji, "Modelado y simulación cinemática de un manipulador 4-DOF", Conferencia internacional de 2017 sobre sistemas informáticos, electrónica y control (ICCSEC), Dalian, 2017, págs. 302-305, doi: 10.1109 / ICCSEC.2017.8446769.
- [22] AJ Valencia, M. Mauleoux y C. Castañeda, "Inverse, Direct Kinetics and Differential Kinematic Control of Parallel Robot with Six Degrees of Hexa Freedom - Hunt", 2019 IEEE 10th International Conference on Mechanical and Aerospace Engineering (ICMAE), Bruselas, Bélgica, 2019, págs.563-567, doi: 10.1109 / ICMAE.2019.8881011.
- [23] K. Huang, J. Zhang, F. Wang and J. Xing, "Study on Direct Kinematic Solution for Parallel Robots with 6 Freedoms Based on Interval Analysis," 2010 International Conference on

- Digital Manufacturing & Automation, Changsha, 2010, pp. 455-458, doi: 10.1109/ICDMA.2010.30.
- [24] Q. Zhang, X. Zhang and J. Liang, "Dynamic analysis of planar 3-RRR flexible parallel robot," 2012 IEEE International Conference on Robotics and Biomimetics (ROBIO), Guangzhou, 2012, pp. 154-159, doi: 10.1109/ROBIO.2012.6490959.
- [25] K. Li and R. Wen, "Kinematic and Dynamic Modeling of 3-DOF Micro-motion Manipulators," 2010 International Conference on Measuring Technology and Mechatronics Automation, Changsha City, 2010, pp. 554-557, doi: 10.1109/ICMTMA.2010.823.
- [26] M. Dorosti and J. H. Nobari, "Kinematic and dynamic analysis of 3-DOF Rotary Table manipulator," 2009 Chinese Control and Decision Conference, Guilin, 2009, pp. 5745-5750, doi: 10.1109/CCDC.2009.5195224.
- [27] L. Tang, L. Zhang and F. Ru, "SimMechanics model and LQR control for linear double inverted pendulum based on kalman filter," 2012 International Conference on Systems and Informatics (ICSAI2012), Yantai, 2012, pp. 462-464, doi: 10.1109/ICSAI.2012.6223036.
- [28] B. Sadiq and K. Iqbal, "Mathematical Modelling and Controller Design for Soccer Kick Action Based on SimMechanics in Matlab," 2019 2nd International Conference of Intelligent Robotic and Control Engineering (IRCE), Singapore, Singapore, 2019, pp. 51-55, doi: 10.1109/IRCE.2019.00018
- [29] J. Li and Y. Li, "Workspace solution and PID control of 3-DOF parallel mechanism based on Matlab/SimMechanics," 2017 12th IEEE Conference on Industrial Electronics and Applications (ICIEA), Siem Reap, 2017, pp. 1405-1409, doi: 10.1109/ICIEA.2017.8283059.
- [30] C. Zhang and Z. Zhang, "Modelling and Simulation of SCARA Robot using MATLAB/SimMechanics," 2019 IEEE 3rd Advanced Information Management, Communicates, Electronic and Automation Control Conference (IMCEC), Chongqing, China, 2019, pp. 516-519, doi: 10.1109/IMCEC46724.2019.8984081.
- [31] Yuan Shaoqiang, Liu Zhong and Li Xingshan, "Modeling and simulation of robot based on Matlab/SimMechanics," 2008 27th Chinese Control Conference, Kunming, 2008, pp. 161-165, doi: 10.1109/CHICC.2008.4604913.

- [32] W. Deng, J. Lee and H. Lee, "Kinematics simulation and control of a new 2 DOF parallel mechanism based on Matlab/SimMechanics," 2009 ISECS International Colloquium on Computing, Communication, Control, and Management, Sanya, 2009, pp. 233-236, doi: 10.1109/CCCM.2009.5267894.
- [33] Jianhua Liu, Guojin Chen, Youping Gong and Huipeng Chen, "Modeling and simulation of loader working device based on SimMechanics," Proceedings 2011 International Conference on Transportation, Mechanical, and Electrical Engineering (TMEE), Changchun, 2011, pp. 2054-2057, doi: 10.1109/TMEE.2011.6199620.
- [34] Romero Aguirre, J. C. (2019). Desarrollo de sistema de control para prototipo de robot antropomórfico.
- [35] Cano Izquierdo, J. M., Feliú Batlle, J. J., & Beltrán Blanco, M. (2004). Simurob. Simulador del robot IRB-1400.
- [36] Hernández-Rodríguez, F., & Benavidez-Téllez, E. DISEÑO Y ANÁLISIS DE PROTOTIPOS FUNCIONALES PARA EL ESTUDIO DE CINEMÁTICA DIRECTA EN ROBÓTICA INDUSTRIAL. *Índice*, 4.
- [37] Temirzhanova, Z. (2018). CONTROL DE VELOCIDAD Y DE POSICIÓN DE UN MOTOR DE CC. In *ЭНЕПТИЯ-2018* (pp. 55-56).
- [38] Londoño, L. F. C., & Londoño, G. A. O. Diseño de un Sistema para el Control de Posición de un Motor DC Basado en FPGA. In *CASE Congreso argentino de sistemas embebidos* (Vol. 4, pp. 53-57).
- [39] Posada Contreras, J. (2005). Modulación por ancho de pulso (PWM) y modulación vectorial (SVM). Una introducción a las técnicas de modulación.
- [40] https://la.mathworks.com/campaigns/offers/download_smlink.html
- [41] M. Spong (2005). Robot modeling and control. Wiley.
- [42] Cerón Correa, J. C. Diseño de una secuencia didáctica para la enseñanza aprendizaje del concepto de torque a partir de las máquinas simples. Departamento de Física.
- [43] https://www.skf.com/binaries/pub12/Images/0901d1968040841a-Power-Transmission-catalogue-11015-EN_tcm_12-576542.pdf#cid-576542
- [44] Barrientos, A. (2002). Nuevas aplicaciones de la robótica. Robots de servicio. Avances en robótica y visión por computador. Cuenca, Ediciones Castilla-La Mancha, 288.

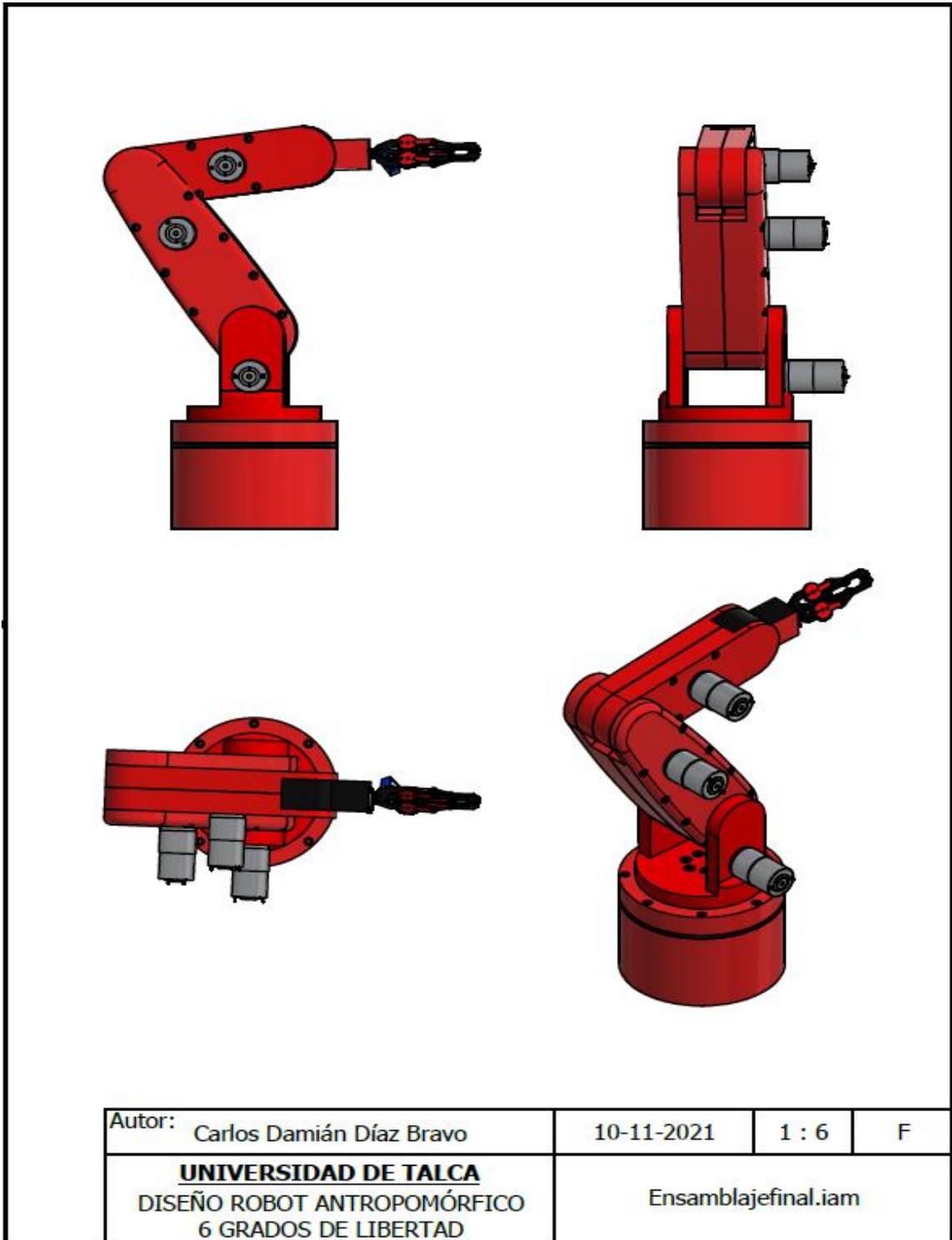
- [45] Turiel, J. P., Marinero, J. C. F., & González, J. R. P. (2002). Aplicaciones de la Robótica: Últimas tendencias y nuevas perspectivas. *Dyna*, 77(3), 61-68.
- [46] Aracil, R., Balaguer, C., & Armada, M. (2008). Robots de servicio. *Revista Iberoamericana de Automática e Informática Industrial RIAI*, 5(2), 6-13.
- [47] P. Corke, Robotics, “Map-Based Planning”, en *Vision and Control, Fundamental Algorithms In MATLAB*, 2011, pp. 102
- [48] P. Corke, Robotics Toolbox for MATLAB, Release 10, 2017 [En línea]. Disponible en <http://petercorke.com/wordpress/toolboxes/robotics-toolbox#Documentation>. Última visita en agosto 2018.
- [49] Pajares, G., Ruz, J. J., Lanillos, P., Guijarro, M., de la Cruz, J. M., & Santos, M. (2008). Generación de trayectorias y toma de decisiones para UAVs. *Revista Iberoamericana de Automática e Informática Industrial RIAI*, 5(1), 83-92.
- [50] Rios, L. H., Bueno, M., & Sánchez, S. (2008). Generación de trayectorias para un robot móvil empleando redes neuronales. *Scientia et technica*, 14(39), 94-99.
- [51] Milanés Hermosilla, D., & Castilla Pérez, A. (2016). Generación de trayectorias para el brazo robótico (ArmX). *Ingeniería Electrónica, Automática y Comunicaciones*, 37(3), 58-71.
- [52] Colnago, G. R., & Messias, M. (2004). Interpolação de dados por spline cúbica utilizando o software MATLAB. In *Congresso Nacional de Matemática Aplicada e Computacional* (Vol. 27, p. 263).
- [53] Trincado, G. U. I. L. L. E. R. M. O., & Vidal, J. (1999). Aplicación de interpolación “spline” cúbica en la estimación de volumen. *Bosque*, 20(2), 3-8.
- [54] Quesada, J. G. (2000). Tutorial de Análisis Numérico Interpolación: Splines cúbicos. Departamento de Informática y Sistemas, Universidad de Las Palmas de Gran Canaria.
- [55] Seuc, A. H., & González, R. (1990). Interpolación mediante splines cúbicos. *Rev. cuba. endocrinol*, 40-8.
- [56] Mohedano Gómez, M. (2010). Implementación de movimientos suaves en robots industriales mediante splines cúbicos: aplicación a un laboratorio remoto de robótica.
- [57] Cajiga, J. S., Soto, J. V., & Ortega, J. P. (2006). Generación de trayectorias para un robot manipulador utilizando procesamiento de imágenes y splines. In *Segundo Congreso Internacional de Ingeniería*, UAQ, Marzo.

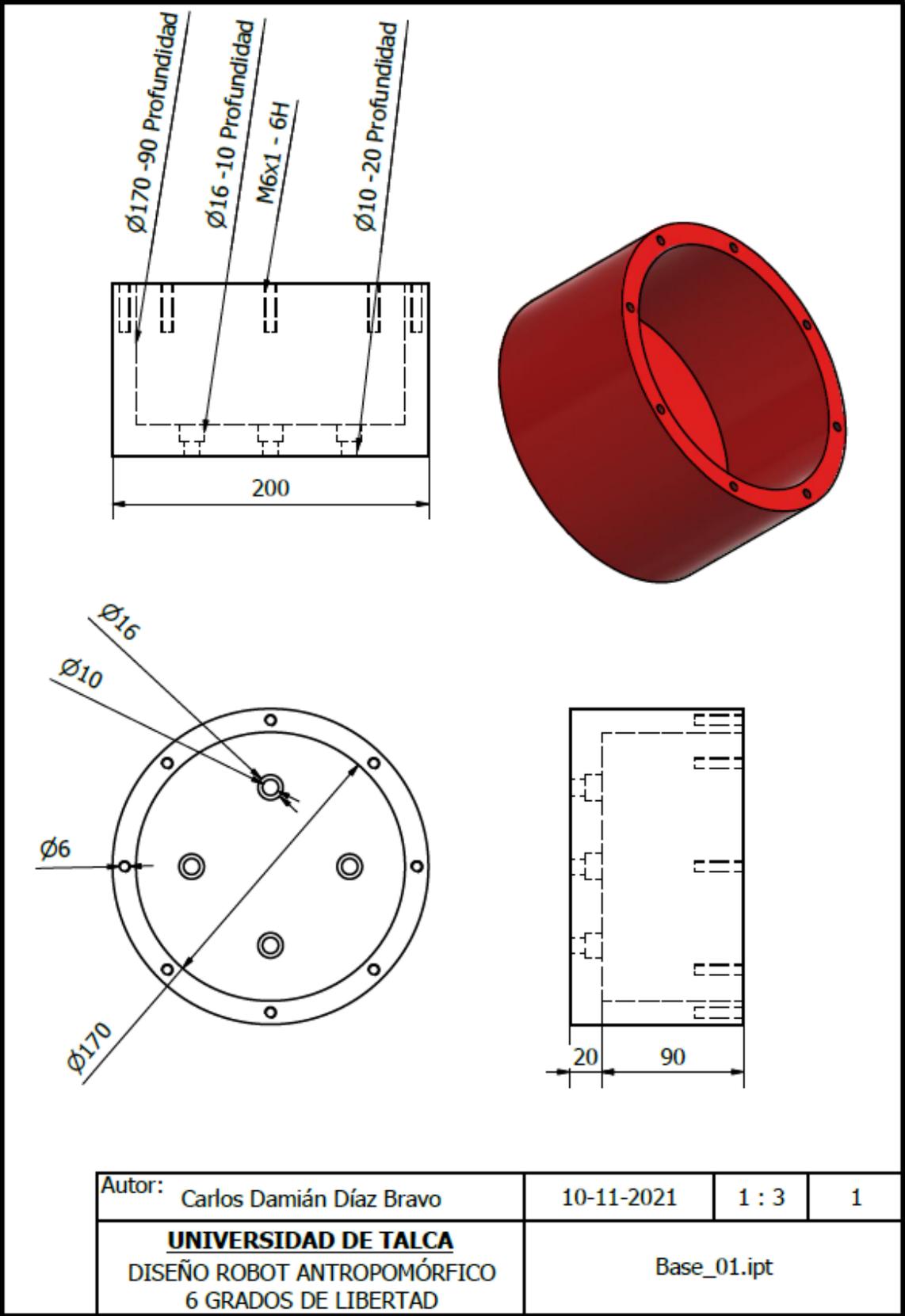
- [58] Suñer, J. L., Valero, F. J., Ródenas, J. J., & Besa, A. (2007). Comparación entre procedimientos de solución de la interpolación por funciones splines para la planificación de trayectorias de robots industriales. In Proceedings of the 8th Congreso Iberoamericano de Ingeniería Mecánica.
- [59] Espinosa, F., Mazo, M., López, E., Urena, J., Rodríguez, F. J., & García, J. J. (1998). Generación de trayectorias y detección de obstáculos mediante splines en el guiado de robots móviles. Centro de Información Tecnológica (CIT), 9(6), 125-134.
- [60] Edgar, A. S. J., Joel, R. Ê. I., Rogelio, R. R. L., & Emilio, V. S. J. Modelo y Simulación de un Brazo Robótico tipo Scara Utilizando Procesamiento de Imágenes y Splines para el Trazado de una Ruta.
- [61] Calderon-Uribe, S., Lara-Rosales, J. R., Ledesma-Carrillo, L. M., & Cabal-Yépez, E. (2019). Interpolaciones polinomiales para aplicaciones en visión robótica. JÓVENES EN LA CIENCIA, 5(1).
- [62] SILVESTRE, I. R., & Roberto, E. (2011). CONTROL PUNTO A PUNTO PARA EL SEGUIMIENTO DE TRAYECTORIAS DE UN ROBOT MOVIL DE RUEDAS TIPO DIFERENCIAL (Doctoral dissertation).
- [63] Dreoni, F., Gil Ortega Linares, M., & Rodríguez Rubio, F. (2003). Identificación y control de un robot manipulador. Jornadas de automática (2003).
- [64] Video explicativo de las simulaciones – link -
<https://www.youtube.com/watch?v=cjv2AS3PB3k>
- [65] <https://petercorke.com/toolboxes/robotics-toolbox/>
- [66] https://alumnosutalca-my.sharepoint.com/:f:/g/personal/carldiaz13_alumnos_utalca_cl/Ej_0aIBN0tpIosPSasvVtp8Bcyw6CwRI7-W6kZMZlqBORg?e=g0STD3
- [67] https://www.codelco.com/cgi-bin/prontus_imprimir.cgi?_URL=http%3A//www.codelco.com/prontus_codelco/site/artic/20110303/pags/20110303132544.html
- [68] <https://www.amchamchile.cl/2017/06/robotica-industrial-expande-sus-areas-de-aplicacion/>
- [69] <https://mirsrobotics.com/blog/2016/09/02/robotica-clave-para-alcanzar-la-mineria-4-0/>
- [70] <https://www.kuka.com/es-es/empresa/acerca-de-kuka/historia>

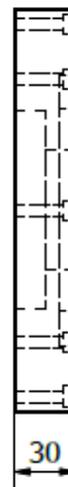
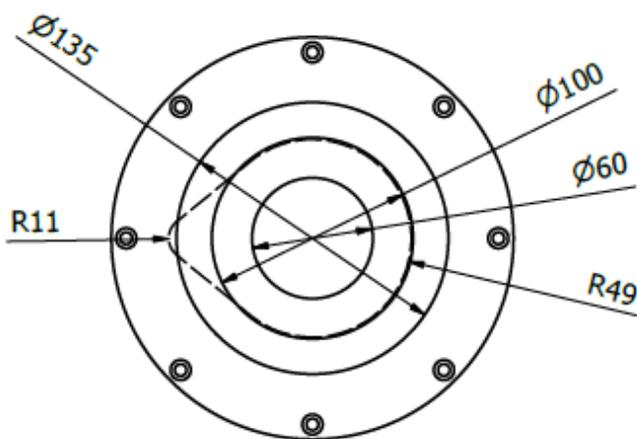
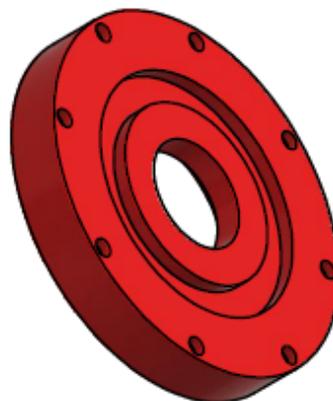
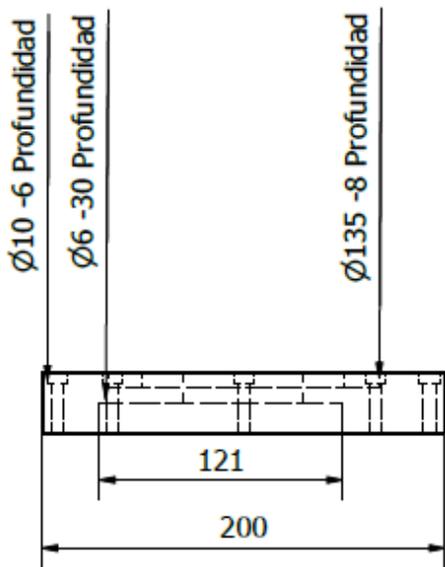
- [71] <https://www.ensingerplastics.com/es-br/semielaborados/plastico/tecaran-abs-grey>
- [72] https://es.aliexpress.com/item/32937751079.html?spm=a2g0o.productlist.0.0.5f247f52WwvZH&algo_pvid=7c21ac09-3c85-4dc9-8918-f7e5b486419b&algo_exp_id=7c21ac09-3c85-4dc9-8918-f7e5b486419b-9&pdp_ext_f=%7B%22sku_id%22%3A%2266157918340%22%7Dhttps://es.aliexpress.com/item/32937751079.html?spm=a2g0o.productlist.0.0.5f247f52WwvZHK&algo_pvid=7c21ac09-3c85-4dc9-8918-f7e5b486419b&algo_exp_id=7c21ac09-3c85-4dc9-8918-f7e5b486419b-9&pdp_ext_f=%7B%22sku_id%22%3A%2266157918340%22%7D
- [73] https://es.aliexpress.com/item/4001149590522.html?spm=a2g0o.productlist.0.0.763d383eJweIGP&algo_pvid=4e06b374-a135-4c6d-aeab-de3f8e73571c&algo_exp_id=4e06b374-a135-4c6d-aeab-de3f8e73571c-1&pdp_ext_f=%7B%22sku_id%22%3A%2210000014899518043%22%7D
- [74] https://es.aliexpress.com/item/32980445293.html?spm=a2g0o.productlist.0.0.338342bfvab489&algo_pvid=4049a68d-93fb-4c1e-af26-ae3d26cfe71b&aem_p4p_detail=202112170017364383075801578040038476989&algo_exp_id=4049a68d-93fb-4c1e-af26-ae3d26cfe71b-4&pdp_ext_f=%7B%22sku_id%22%3A%2266735315440%22%7D
- [75] <https://www.make3d.cl/search?q=ABS>
- [76] <https://youtu.be/i1AuaxLC2MM>

Anexos

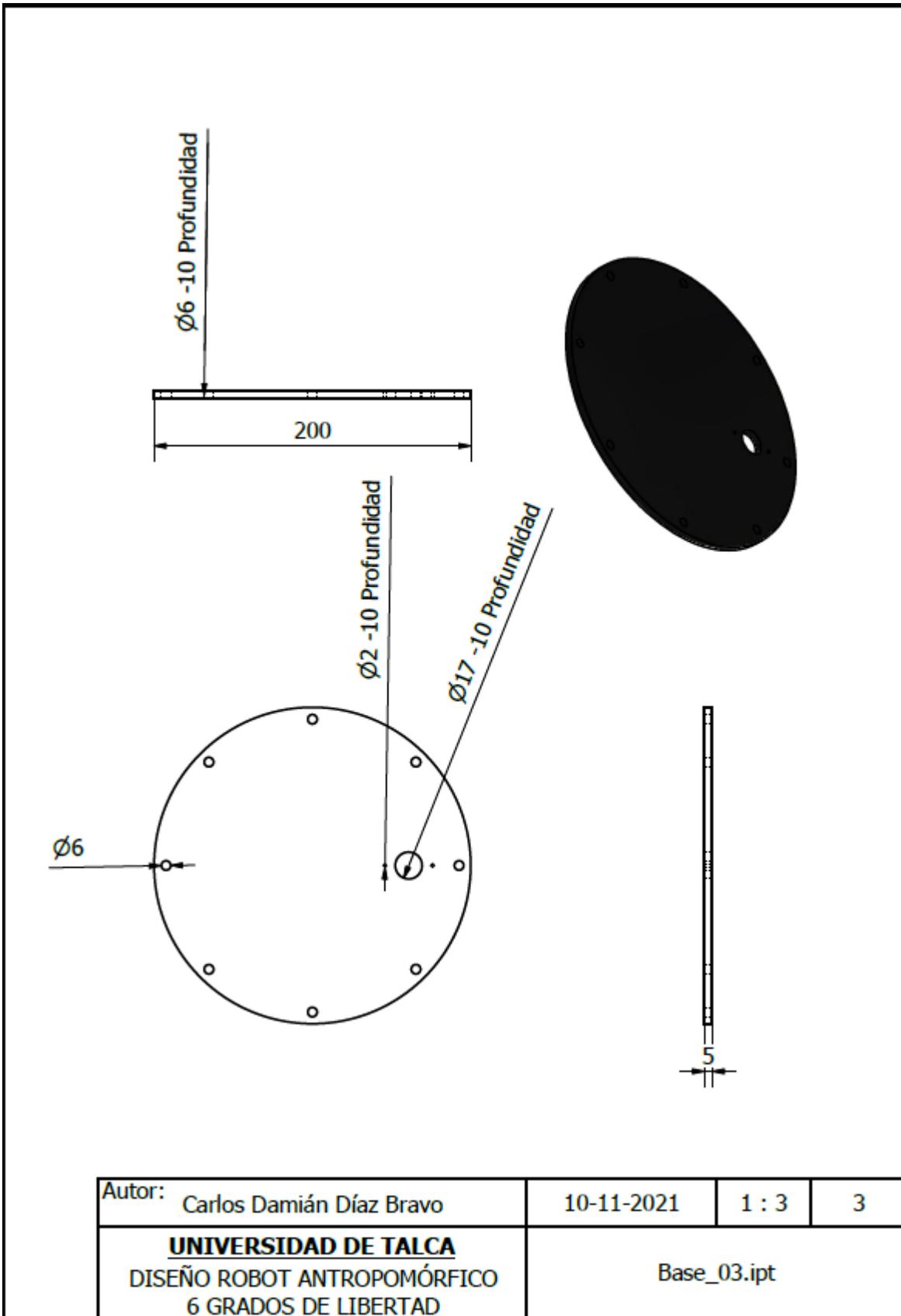
Anexo A. Planos mecánicos

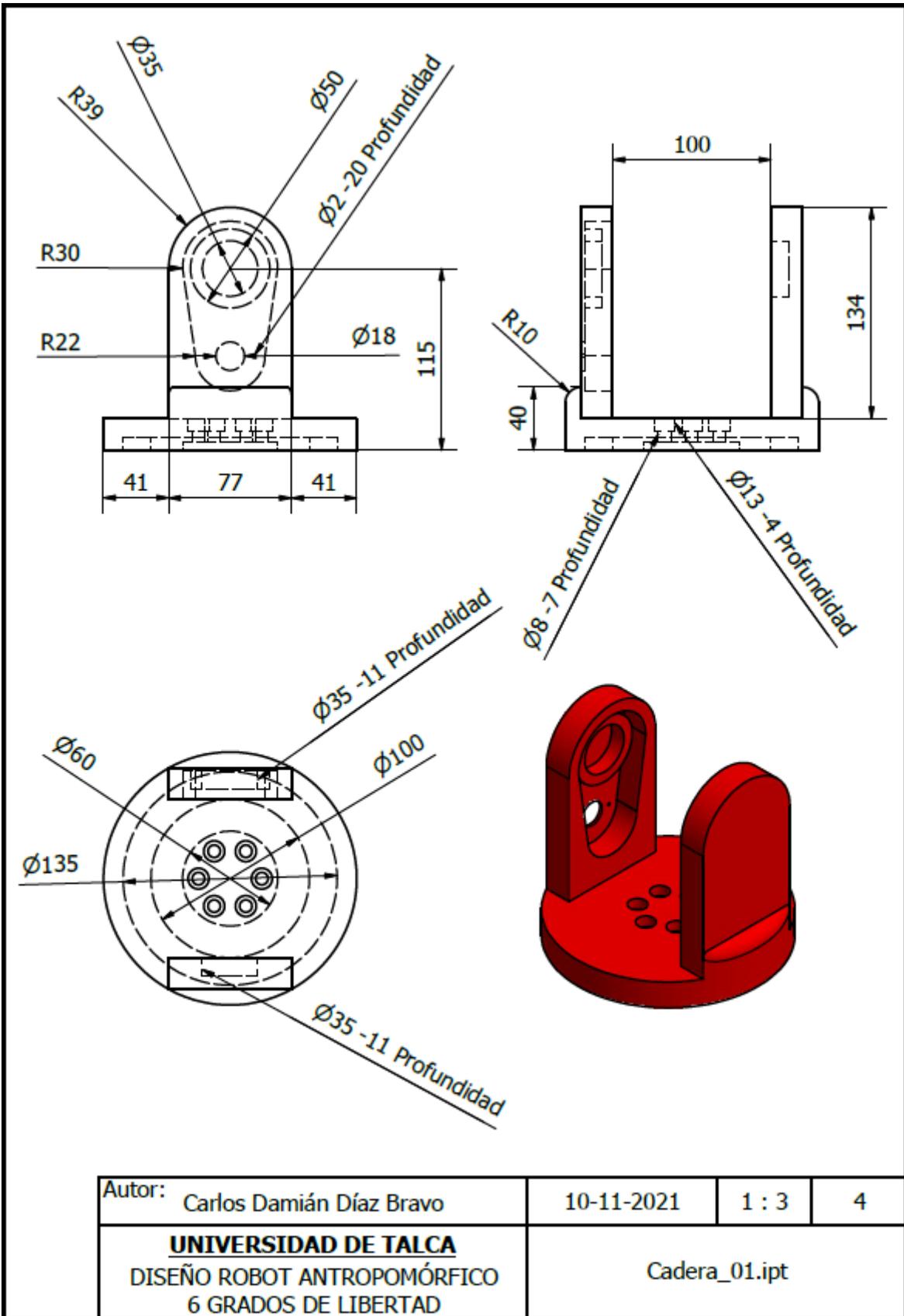




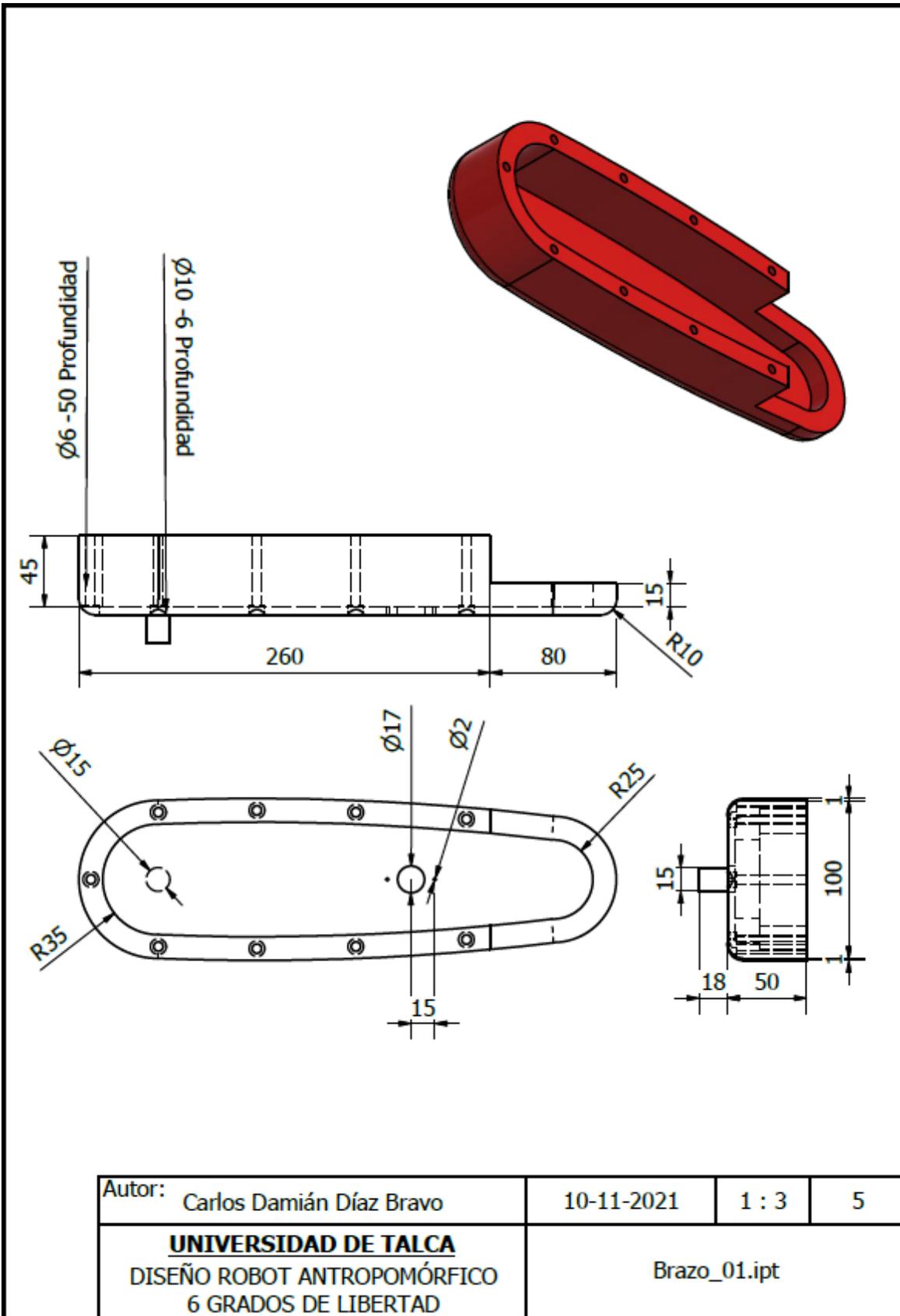


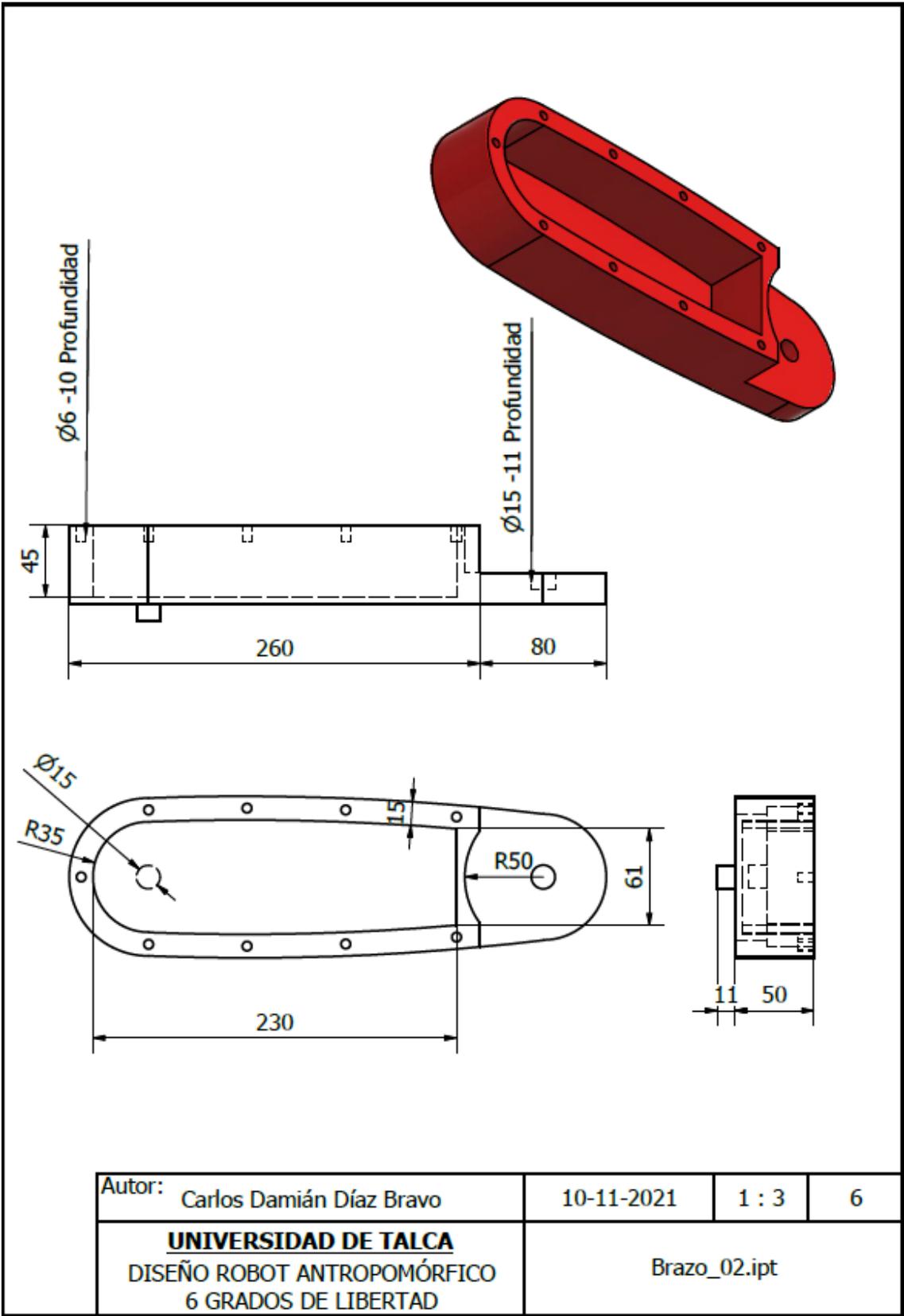
Autor: Carlos Damián Díaz Bravo	10-11-2021	1 : 3	2
UNIVERSIDAD DE TALCA DISEÑO ROBOT ANTROPOMÓRFICO 6 GRADOS DE LIBERTAD		Base_02.ipt	

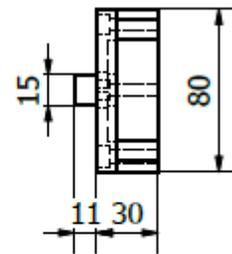
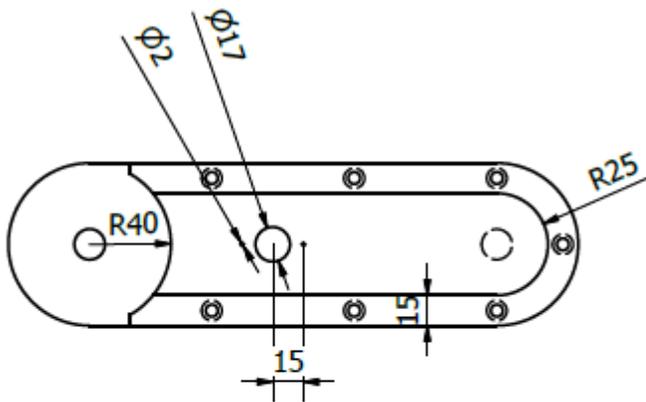
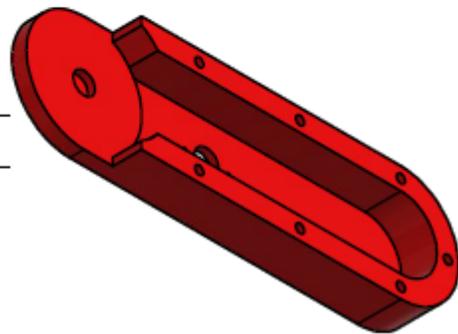
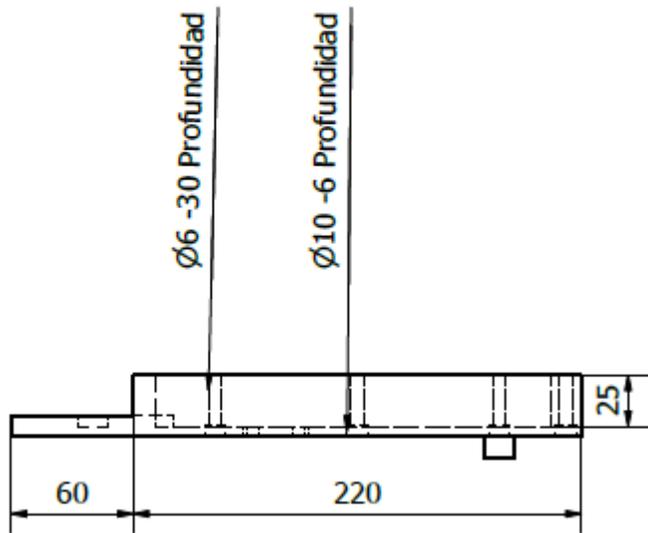




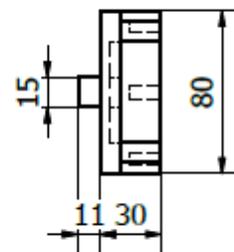
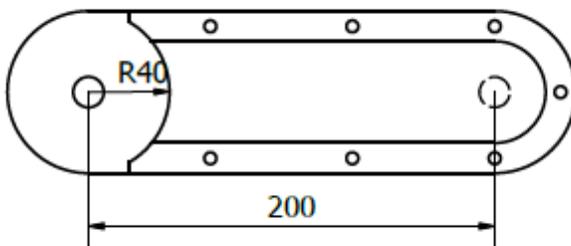
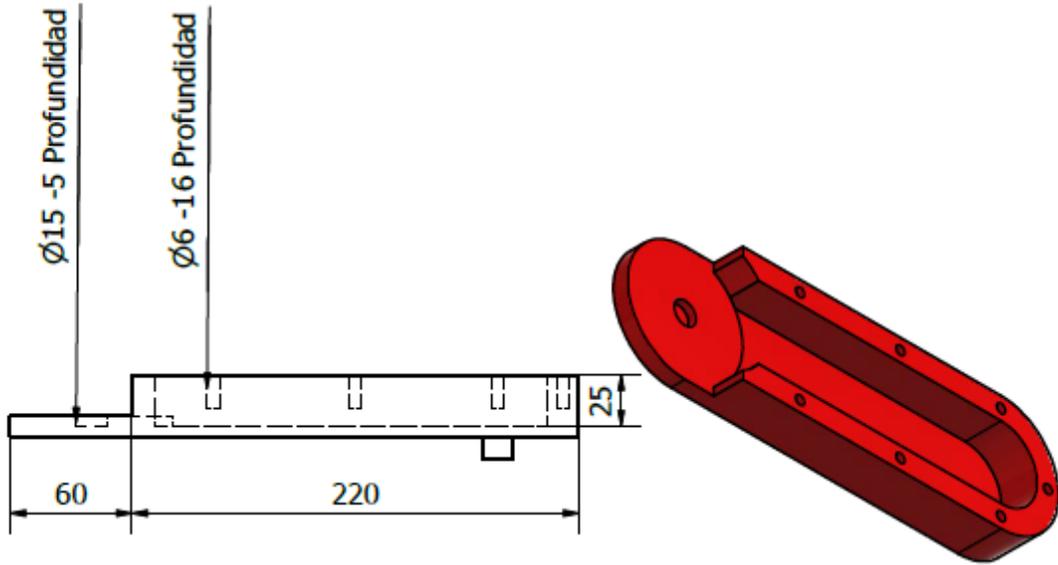
Autor:	Carlos Damián Díaz Bravo	10-11-2021	1 : 3	4
UNIVERSIDAD DE TALCA DISEÑO ROBOT ANTROPOMÓRFICO 6 GRADOS DE LIBERTAD		Cadera_01.ipt		



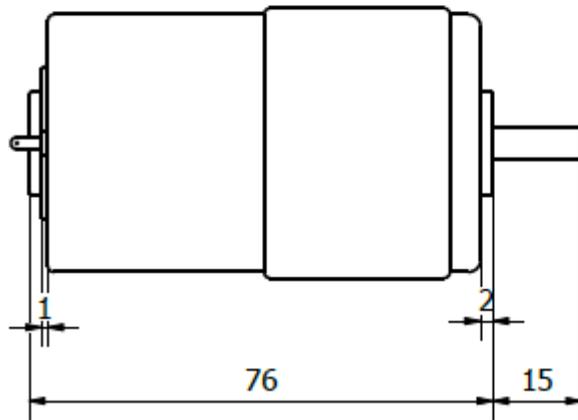
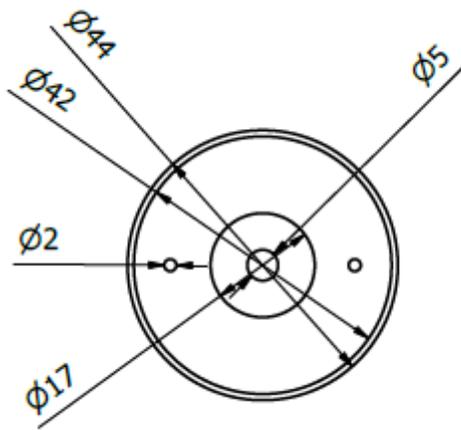
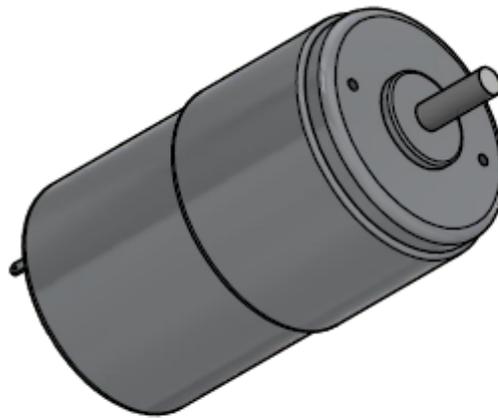
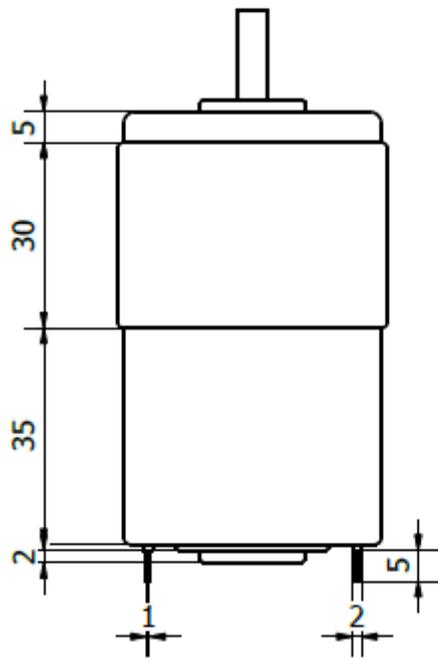




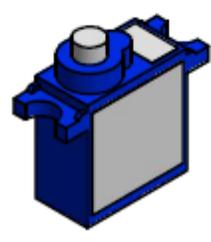
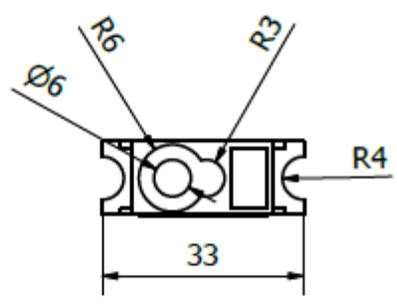
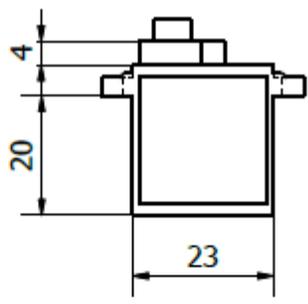
Autor: Carlos Damián Díaz Bravo	10-11-2021	1 : 3	7
UNIVERSIDAD DE TALCA DISEÑO ROBOT ANTROPOMÓRFICO 6 GRADOS DE LIBERTAD	Antebrazo_01.ipt		



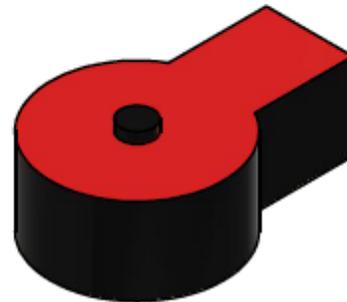
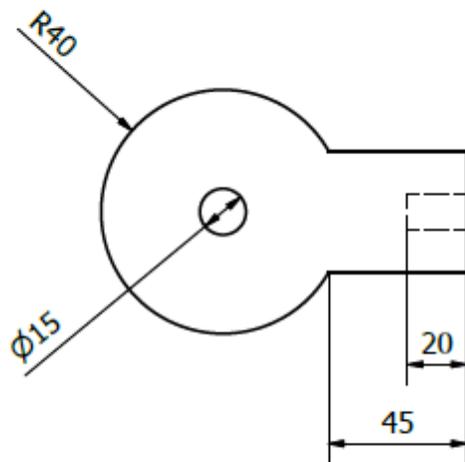
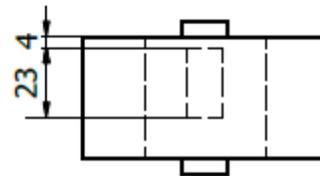
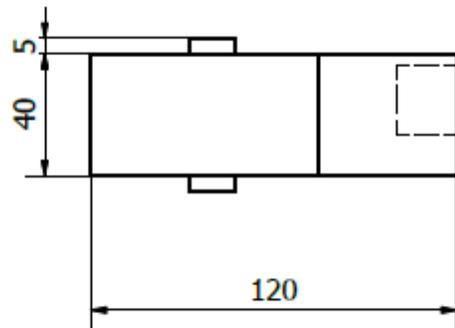
Autor: Carlos Damián Díaz Bravo	10-11-2021	1 : 3	8
UNIVERSIDAD DE TALCA DISEÑO ROBOT ANTROPOMÓRFICO 6 GRADOS DE LIBERTAD		Antebrazo_02.ipt	



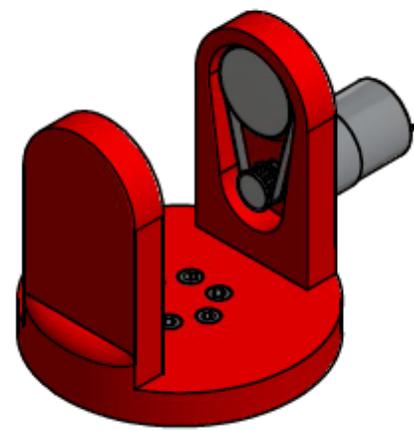
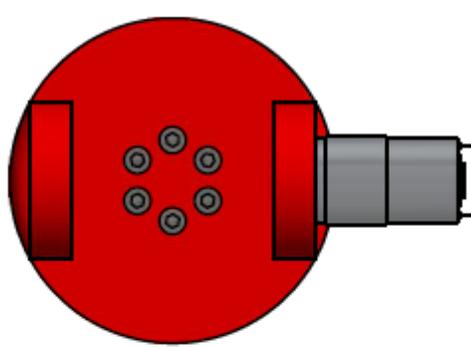
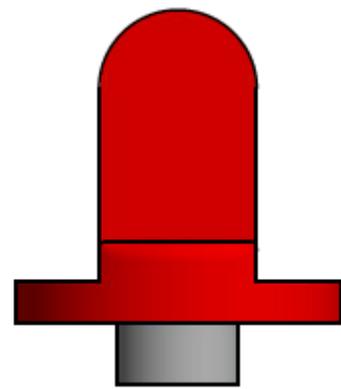
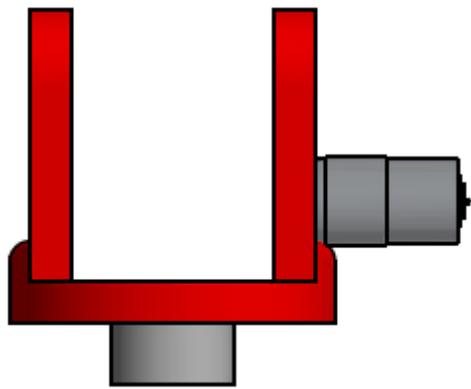
Autor: Carlos Damián Díaz Bravo	10-11-2021	1 : 1	9
UNIVERSIDAD DE TALCA DISEÑO ROBOT ANTROPOMÓRFICO 6 GRADOS DE LIBERTAD		Motor corriente continua.iam	



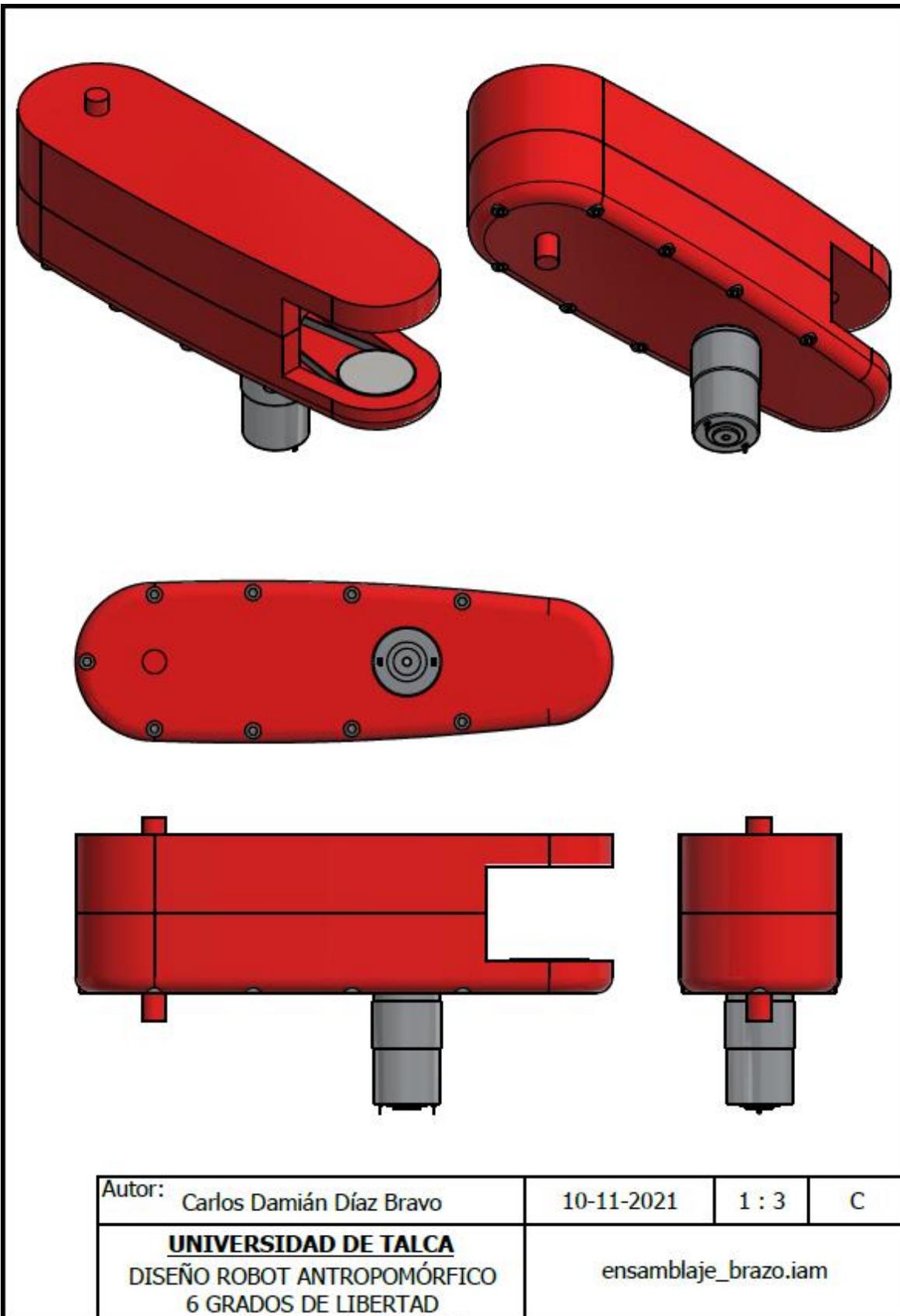
Autor: Carlos Damián Díaz Bravo	10-11-2021	1 : 1	10
<p align="center"> UNIVERSIDAD DE TALCA DISEÑO ROBOT ANTROPOMÓRFICO 6 GRADOS DE LIBERTAD </p>	<p align="center">Servomotor.ipt</p>		

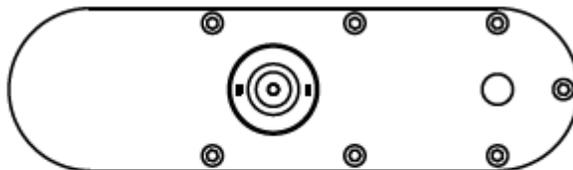
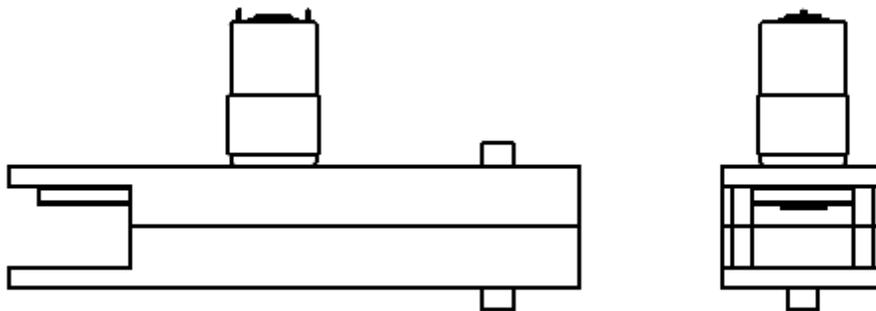
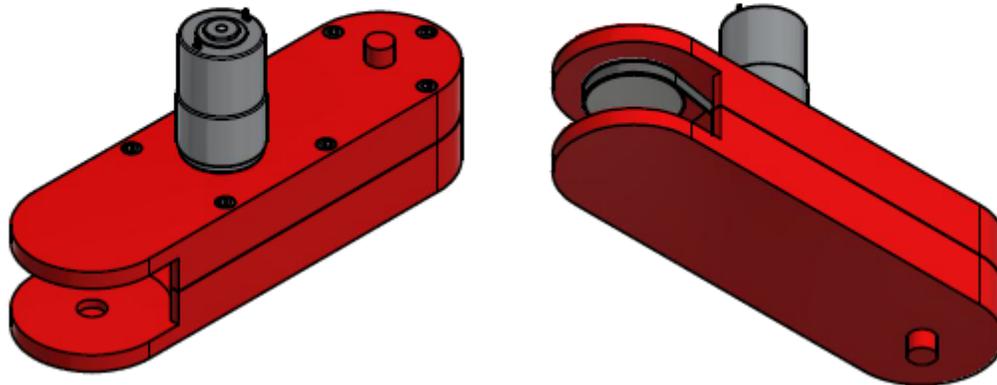


Autor: Carlos Damián Díaz Bravo	10-11-2021	1 : 2	11
UNIVERSIDAD DE TALCA DISEÑO ROBOT ANTROPOMÓRFICO 6 GRADOS DE LIBERTAD	Muñeca_01.ipt		

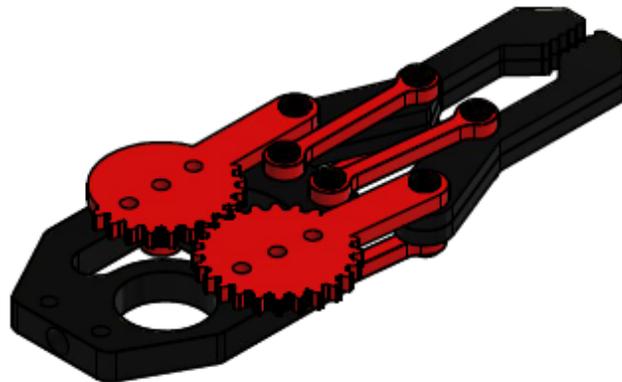
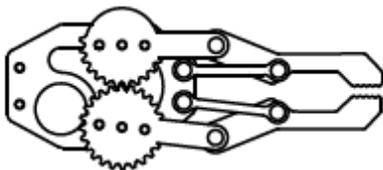


Autor: Carlos Damián Díaz Bravo	10-11-2021	1 : 3	B
UNIVERSIDAD DE TALCA DISEÑO ROBOT ANTROPOMÓRFICO 6 GRADOS DE LIBERTAD	Ensamblaje_Cadera.iam		





Autor: Carlos Damián Díaz Bravo	10-11-2021	1 : 3	D
UNIVERSIDAD DE TALCA DISEÑO ROBOT ANTROPOMÓRFICO 6 GRADOS DE LIBERTAD	Ensamblaje_Antebrazo.iam		



Autor: Carlos Damián Díaz Bravo	10-11-2021	1 : 2	E
UNIVERSIDAD DE TALCA DISEÑO ROBOT ANTROPOMÓRFICO 6 GRADOS DE LIBERTAD	ensamblaje_pinza.iam		

Anexo B. Códigos

%% Cinemática directa para un manipulador de 6 grados de libertad

% Carlos Damián Díaz Bravo

% Universidad de Talca - Facultad de Ingeniería

% Ingeniería Civil Mecatrónica

clear all;

close all;

clc;

%% Se definen las constantes a utilizar

d1=0.26381;

a2=0.25;

a3=0.20;

d4=0.089;

d6=0.21193;

%% Se define el tipo de articulación y los parámetros de D-H

L1=Revolute ('a', 0, 'alpha', pi/2, 'd', d1); %Eslabón 1

L2=Revolute ('a', a2, 'alpha', 0, 'd', 0); %Eslabón 2

L3=Revolute ('a', a3, 'alpha', -pi/2, 'd', 0); %Eslabón 3

L4=Revolute ('a', 0, 'alpha', -pi/2, 'd', 0); %Eslabón 4

L5=Revolute ('a', 0, 'alpha', pi/2, 'd', 0); %Eslabón 5

L6=Revolute ('a', 0, 'alpha', 0, 'd', d6); %Eslabón 6

%% Se crea el robot

bot=SerialLink([L1,L2,L3,L4,L5,L6]);

```
%% Se genera la animación en una posición Home
```

```
ws=([-1 1 -1 1 0 1]);
```

```
%bot.plot([pi/4, pi/4, 2], 'workspace', ws);
```

```
bot.teach([0,0,0,0,0,0], 'workspace', ws, 'noname');
```

```
%Cinemática inversa de un robot antropomórfico de 3 grados de libertad
```

```
%Carlos Damián Díaz Bravo
```

```
%Universidad de Talca - Facultad de Ingeniería
```

```
%Ingeniería Civil Mecatrónica
```

```
function [q] = CINEMATICA_INVERSA_ROBOT_ANTROPOMORFICO(xd,yd,zd)
```

```
%Se declaran los parámetros constantes del manipulador
```

```
d1=0.3;
```

```
a2=0.25;
```

```
a3=0.2;
```

```
%Se define un cociente
```

```
C= (xd.^2 + yd.^2 + (zd - d1).^2 - a2^2 - a3^2)/(2*a2*a3);
```

```
C2= sqrt(1-C.^2);
```

```
q3 = atan2(real(C2),C);
```

```
S2 = sqrt(xd.^2+yd.^2);
```

```
q2 = atan2(zd-d1,real(S2))-atan2(a3*sin(q3),a2+a3*cos(q3));
```

```
q1 = atan2(yd,xd);
```

```
q = [q1*180/pi; q2*180/pi; q3*180/pi];
```

```
end
```

```

%% Dinámica para un manipulador de 6 GDL
%Carlos Damián Díaz Bravo
%Universidad de Talca - Facultad de Ingeniería
%Ingeniería Civil Mecatrónica

clc;
clear all;
close all;

%% Definición de simbólicos
%% Definición de posiciones, velocidades y aceleraciones

syms th1 th2 th3 th4 th5 th6
syms th1p th2p th3p th4p th5p th6p
syms th1pp th2pp th3pp th4pp th5pp th6pp

%% Definición de masas e inercias

global m1 m2 m3 m4 m5 m6
global Ixx1 Iyy1 Izz1 Ixx2 Iyy2 Izz2 Ixx3 Iyy3 Izz3 Ixx4 Iyy4 Izz4 Ixx5 Iyy5 Izz5 Ixx6 Iyy6 Izz6

%% Definición de Torques

syms tau1 tau2 tau3 tau4 tau5 tau6

%% Definición largo de eslabones y distancias al centro de masa

global d_1 a_2 a_3 d_6
global lc1 lc2 lc3 lc4 lc5 lc6

%% Definición de valor de pi

```

```
PI= sym(pi);
```

```
%% Definición de gravedad
```

```
global g
```

```
%% Declaración de constantes
```

```
g=9.81;
```

```
d_1=0.2;
```

```
a_2=0.15;
```

```
a_3=0.1;
```

```
d_6=0.05;
```

```
lc1=d_1/2;
```

```
lc2=a_2/2;
```

```
lc3=a_3/2;
```

```
lc4=0;
```

```
lc5=0;
```

```
lc6=d_6/2;
```

```
m1=0.2;
```

```
m2=0.1;
```

```
m3=0.1;
```

```
m4=0.05;
```

```
m5=0.05;
```

```
m6=0.05;
```

```
Ixx1= 0.2;
```

```
Ixx2= 0.1;
```

```
Ixx3= 0.1;  
Ixx4= 0.05;  
Ixx5= 0.05;  
Ixx6= 0.05;
```

```
Iyy1= 0.2;  
Iyy2= 0.1;  
Iyy3= 0.1;  
Iyy4= 0.05;  
Iyy5= 0.05;  
Iyy6= 0.05;
```

```
Izz1= 0.2;  
Izz2= 0.1;  
Izz3= 0.1;  
Izz4= 0.05;  
Izz5= 0.05;  
Izz6= 0.05;
```

```
%% Coordenadas generalizadas
```

```
%% Vector de posiciones
```

```
q=[th1; th2; th3; th4; th5; th6];
```

```
%% Vector de velocidades
```

```
qp=[th1p; th2p; th3p; th4p; th5p; th6p];
```

```
%% Vector de aceleraciones
```

```
qpp=[th1pp; th2pp; th3pp; th4pp; th5pp; th6pp];
```

```
%% Parámetros Denavit-Hartenberg
```

```
H01 = simplify(DH_robot(th1,d_1,0,PI/2));
```

```
H12 = simplify(DH_robot(th2,0,a_2,0));
```

```
H23 = simplify(DH_robot(th3,0,a_3,0));
```

```
H34 = simplify(DH_robot(th4,0,0,-PI/2));
```

```
H45 = simplify(DH_robot(th5,0,0,PI/2));
```

```
H56 = simplify(DH_robot(th6,d_6,0,0));
```

```
%% Cinemática Directa
```

```
H00=eye(4);
```

```
H02=H01*H12;
```

```
H03=H01*H12*H23;
```

```
H04=H01*H12*H23*H34;
```

```
H05=H01*H12*H23*H34*H45;
```

```
H06=H01*H12*H23*H34*H45*H56;
```

```
T= simplify(H01*H12*H23*H34*H45*H56);
```

```
%% Vectores para la simulación
```

```
xx=[H00(1,4) H01(1,4) H02(1,4) H03(1,4) H04(1,4) H05(1,4) H06(1,4)]; %Componentes en x de  
los orígenes de cada uno de los sistemas
```

```
yy=[H00(2,4) H01(2,4) H02(2,4) H03(2,4) H04(2,4) H05(2,4) H06(2,4)]; %Componentes en y de  
los orígenes de cada uno de los sistemas
```

```
zz=[H00(3,4) H01(3,4) H02(3,4) H03(3,4) H04(3,4) H05(3,4) H06(3,4)]; %Componentes en z de  
los orígenes de cada uno de los sistemas
```

```
%% Propagación de las velocidades
```

%% Se definen las submatrices de rotación

R01 = H01(1:3,1:3);

R12 = H12(1:3,1:3);

R23 = H23(1:3,1:3);

R34 = H34(1:3,1:3);

R45 = H45(1:3,1:3);

R56 = H56(1:3,1:3);

%% Vectores de posición

P01 = H01(1:3,4);

P12 = H12(1:3,4);

P23 = H23(1:3,4);

P34 = H34(1:3,4);

P45 = H45(1:3,4);

P56 = H56(1:3,4);

%% Definir la velocidad cero lineal y rotacional

v00=[0;0;0]; %El manipulador está fijo en la base 0,0,0

w00=[0;0;0];

v11=R01.*(v00+cross(w00,P01));

w11=R01.*w00+th1p*[0;0;1];

v22=R12.*(v11+cross(w11,P12));

w22=R12.*w11+th2p*[0;0;1];

v33=R23.*(v22+cross(w22,P23));

w33=R23.*w22+th3p*[0;0;1];

```
v44=R34.*(v33+cross(w33,P34));  
w44=R34.*w33+th4p*[0;0;1];
```

```
v55=R45.*(v44+cross(w44,P45));  
w55=R45.*w44+th5p*[0;0;1];
```

```
v66=R56.*(v55+cross(w55,P56));  
w66=R56.*w55+th6p*[0;0;1];
```

```
%% Cálculo del Jacobiano del robot  
%% Jacobiano completo del robot
```

```
J66=[diff(v44(1),th1p) diff(v44(1),th2p) diff(v44(1),th3p) diff(v44(1),th4p) diff(v44(1),th5p)  
diff(v44(1),th6p);...  
diff(v44(2),th1p) diff(v44(2),th2p) diff(v44(2),th3p) diff(v44(2),th4p) diff(v44(2),th5p)  
diff(v44(2),th6p);...  
diff(v44(3),th1p) diff(v44(3),th2p) diff(v44(3),th3p) diff(v44(3),th4p) diff(v44(3),th5p)  
diff(v44(3),th6p);...  
diff(w44(1),th1p) diff(w44(1),th2p) diff(w44(1),th3p) diff(w44(1),th4p) diff(w44(1),th5p)  
diff(w44(1),th6p);...  
diff(w44(2),th1p) diff(w44(2),th2p) diff(w44(2),th3p) diff(w44(2),th4p) diff(w44(2),th5p)  
diff(w44(2),th6p);...  
diff(w44(3),th1p) diff(w44(3),th2p) diff(w44(3),th3p) diff(w44(3),th4p) diff(w44(3),th5p)  
diff(w44(3),th6p)];
```

```
%% Energía Cinética
```

```
%% Centros de masa
```

```
P01c = subs(H01(1:3,4),d_1,lc1);  
P12c = subs(H12(1:3,4),a_2,lc2);  
P23c = subs(H23(1:3,4),a_3,lc3);
```

```
P34c = subs(H34(1:3,4),0,0);
P45c = subs(H45(1:3,4),0,0);
P56c = subs(H56(1:3,4),d_6,lc6);
```

```
%% Matrices de Inercia
```

```
I1=[Ixx1 0 0; 0 Iyy1 0; 0 0 Izz1];
I2=[Ixx2 0 0; 0 Iyy2 0; 0 0 Izz2];
I3=[Ixx3 0 0; 0 Iyy3 0; 0 0 Izz3];
I4=[Ixx4 0 0; 0 Iyy4 0; 0 0 Izz4];
I5=[Ixx5 0 0; 0 Iyy5 0; 0 0 Izz5];
I6=[Ixx6 0 0; 0 Iyy6 0; 0 0 Izz6];
```

```
%% Cálculo de la energía cinética
```

```
k1=(1/2)*m1*(v11+cross(w11,P01c)).*((1/2)*m1*(v11+cross(w11,P01c)))+((1/2)*w11.*(I1*w
11));
k2=(1/2)*m2*(v22+cross(w22,P12c)).*((1/2)*m2*(v22+cross(w22,P12c)))+((1/2)*w22.*(I2*w
22));
k3=(1/2)*m3*(v33+cross(w33,P23c)).*((1/2)*m3*(v33+cross(w33,P23c)))+((1/2)*w33.*(I3*w
33));
k4=(1/2)*m4*(v44+cross(w44,P34c)).*((1/2)*m4*(v44+cross(w44,P34c)))+((1/2)*w44.*(I4*w
44));
k5=(1/2)*m5*(v55+cross(w55,P45c)).*((1/2)*m5*(v55+cross(w55,P45c)))+((1/2)*w55.*(I5*w
55));
k6=(1/2)*m6*(v66+cross(w66,P56c)).*((1/2)*m6*(v66+cross(w66,P56c)))+((1/2)*w66.*(I6*w
66));
```

```
kt=k1+k2+k3+k4+k5+k6;
```

```
%% Energía potencial
```

```
%% Extracción de alturas de los elementos de zz
```

```

h1=subs(H01(3,4),d_1,lc1);
h2=subs(H02(3,4),a_2,lc2);
h3=subs(H03(3,4),a_3,lc3);
h4=subs(H04(3,4),0,0);
h5=subs(H05(3,4),0,0);
h6=subs(H06(3,4),d_6,lc6);

```

```

%% Cálculo de la energía potencial

```

```

u1=m1*g*h1;
u2=m2*g*h2;
u3=m3*g*h3;
u4=m4*g*h4;
u5=m5*g*h5;
u6=m6*g*h6;

```

```

ut=u1+u2+u3+u4+u5+u6;

```

```

%% Cálculo del Lagrangiano

```

```

L=kt-ut;

```

```

%% Ecuaciones de movimiento

```

```

Qp=[th1p;th2p;th3p;th4p;th5p;th6p;th1pp;th2pp;th3pp;th4pp;th5pp;th6pp];

```

```

dQ=[diff((diff(L,th1p)),th1),diff((diff(L,th1p)),th2),diff((diff(L,th1p)),th3),diff((diff(L,th1p)),th4
),diff((diff(L,th1p)),th5),diff((diff(L,th1p)),th6),...

```


dQ=[diff((diff(L,th5p)),th1),diff((diff(L,th5p)),th2),diff((diff(L,th5p)),th3),diff((diff(L,th5p)),th4),diff((diff(L,th5p)),th5),diff((diff(L,th6p)),th6),...

diff((diff(L,th5p)),th1p),diff((diff(L,th5p)),th2p),diff((diff(L,th5p)),th3p),diff((diff(L,th5p)),th4p),diff((diff(L,th5p)),th5p),diff((diff(L,th6p)),th6p)];

tau5=dQ*Qp-diff(L,th5);

dQ=[diff((diff(L,th6p)),th1),diff((diff(L,th6p)),th2),diff((diff(L,th6p)),th3),diff((diff(L,th6p)),th4),diff((diff(L,th6p)),th5),diff((diff(L,th6p)),th6),...

diff((diff(L,th6p)),th1p),diff((diff(L,th6p)),th2p),diff((diff(L,th6p)),th3p),diff((diff(L,th6p)),th4p),diff((diff(L,th6p)),th5p),diff((diff(L,th6p)),th6p)];

tau6=dQ*Qp-diff(L,th6);

%% Modelo dinámico en forma matricial

M=[diff(tau1,th1pp),diff(tau1,th2pp),diff(tau1,th3pp),diff(tau1,th4pp),diff(tau1,th5pp),diff(tau1,th6pp);...

diff(tau2,th1pp),diff(tau2,th2pp),diff(tau2,th3pp),diff(tau2,th4pp),diff(tau2,th5pp),diff(tau2,th6pp);...

diff(tau3,th1pp),diff(tau3,th2pp),diff(tau3,th3pp),diff(tau3,th4pp),diff(tau3,th5pp),diff(tau3,th6pp);...

diff(tau4,th1pp),diff(tau4,th2pp),diff(tau4,th3pp),diff(tau4,th4pp),diff(tau4,th5pp),diff(tau4,th6pp);...

$\text{diff}(\tau_5, \text{th1pp}), \text{diff}(\tau_5, \text{th2pp}), \text{diff}(\tau_5, \text{th3pp}), \text{diff}(\tau_5, \text{th4pp}), \text{diff}(\tau_5, \text{th5pp}), \text{diff}(\tau_5, \text{th6p})$;...

$\text{diff}(\tau_6, \text{th1pp}), \text{diff}(\tau_6, \text{th2pp}), \text{diff}(\tau_6, \text{th3pp}), \text{diff}(\tau_6, \text{th4pp}), \text{diff}(\tau_6, \text{th5pp}), \text{diff}(\tau_6, \text{th6p})$];

$M11 = [\text{diff}(M(1,1), \text{th1}), \text{diff}(M(1,1), \text{th2}), \text{diff}(M(1,1), \text{th3}), \text{diff}(M(1,1), \text{th4}), \text{diff}(M(1,1), \text{th5}), \text{diff}(M(1,1), \text{th6})] * \text{qp};$

$M12 = [\text{diff}(M(1,2), \text{th1}), \text{diff}(M(1,2), \text{th2}), \text{diff}(M(1,2), \text{th3}), \text{diff}(M(1,2), \text{th4}), \text{diff}(M(1,2), \text{th5}), \text{diff}(M(1,2), \text{th6})] * \text{qp};$

$M13 = [\text{diff}(M(1,3), \text{th1}), \text{diff}(M(1,3), \text{th2}), \text{diff}(M(1,3), \text{th3}), \text{diff}(M(1,3), \text{th4}), \text{diff}(M(1,3), \text{th5}), \text{diff}(M(1,3), \text{th6})] * \text{qp};$

$M14 = [\text{diff}(M(1,4), \text{th1}), \text{diff}(M(1,4), \text{th2}), \text{diff}(M(1,4), \text{th3}), \text{diff}(M(1,4), \text{th4}), \text{diff}(M(1,4), \text{th5}), \text{diff}(M(1,4), \text{th6})] * \text{qp};$

$M15 = [\text{diff}(M(1,5), \text{th1}), \text{diff}(M(1,5), \text{th2}), \text{diff}(M(1,5), \text{th3}), \text{diff}(M(1,5), \text{th4}), \text{diff}(M(1,5), \text{th5}), \text{diff}(M(1,5), \text{th6})] * \text{qp};$

$M16 = [\text{diff}(M(1,6), \text{th1}), \text{diff}(M(1,6), \text{th2}), \text{diff}(M(1,6), \text{th3}), \text{diff}(M(1,6), \text{th4}), \text{diff}(M(1,6), \text{th5}), \text{diff}(M(1,6), \text{th6})] * \text{qp};$

$M21 = [\text{diff}(M(2,1), \text{th1}), \text{diff}(M(2,1), \text{th2}), \text{diff}(M(2,1), \text{th3}), \text{diff}(M(2,1), \text{th4}), \text{diff}(M(2,1), \text{th5}), \text{diff}(M(2,1), \text{th6})] * \text{qp};$

$M22 = [\text{diff}(M(2,2), \text{th1}), \text{diff}(M(2,2), \text{th2}), \text{diff}(M(2,2), \text{th3}), \text{diff}(M(2,2), \text{th4}), \text{diff}(M(2,2), \text{th5}), \text{diff}(M(2,2), \text{th6})] * \text{qp};$

$M23 = [\text{diff}(M(2,3), \text{th1}), \text{diff}(M(2,3), \text{th2}), \text{diff}(M(2,3), \text{th3}), \text{diff}(M(2,3), \text{th4}), \text{diff}(M(2,3), \text{th5}), \text{diff}(M(2,3), \text{th6})] * \text{qp};$

$M24 = [\text{diff}(M(2,4), \text{th1}), \text{diff}(M(2,4), \text{th2}), \text{diff}(M(2,4), \text{th3}), \text{diff}(M(2,4), \text{th4}), \text{diff}(M(2,4), \text{th5}), \text{diff}(M(2,4), \text{th6})] * \text{qp};$

$M25 = [\text{diff}(M(2,5), \text{th1}), \text{diff}(M(2,5), \text{th2}), \text{diff}(M(2,5), \text{th3}), \text{diff}(M(2,5), \text{th4}), \text{diff}(M(2,5), \text{th5}), \text{diff}(M(2,5), \text{th6})] * \text{qp};$

$M52=[\text{diff}(M(5,2),\text{th}1),\text{diff}(M(5,2),\text{th}2),\text{diff}(M(5,2),\text{th}3),\text{diff}(M(5,2),\text{th}4),\text{diff}(M(5,2),\text{th}5),\text{diff}(M(5,2),\text{th}6)]*qp;$

$M53=[\text{diff}(M(5,3),\text{th}1),\text{diff}(M(5,3),\text{th}2),\text{diff}(M(5,3),\text{th}3),\text{diff}(M(5,3),\text{th}4),\text{diff}(M(5,3),\text{th}5),\text{diff}(M(5,3),\text{th}6)]*qp;$

$M54=[\text{diff}(M(5,4),\text{th}1),\text{diff}(M(5,4),\text{th}2),\text{diff}(M(5,4),\text{th}3),\text{diff}(M(5,4),\text{th}4),\text{diff}(M(5,4),\text{th}5),\text{diff}(M(5,4),\text{th}6)]*qp;$

$M55=[\text{diff}(M(5,5),\text{th}1),\text{diff}(M(5,5),\text{th}2),\text{diff}(M(5,5),\text{th}3),\text{diff}(M(5,5),\text{th}4),\text{diff}(M(5,5),\text{th}5),\text{diff}(M(5,5),\text{th}6)]*qp;$

$M56=[\text{diff}(M(5,6),\text{th}1),\text{diff}(M(5,6),\text{th}2),\text{diff}(M(5,6),\text{th}3),\text{diff}(M(5,6),\text{th}4),\text{diff}(M(5,6),\text{th}5),\text{diff}(M(5,6),\text{th}6)]*qp;$

$M61=[\text{diff}(M(6,1),\text{th}1),\text{diff}(M(6,1),\text{th}2),\text{diff}(M(6,1),\text{th}3),\text{diff}(M(6,1),\text{th}4),\text{diff}(M(6,1),\text{th}5),\text{diff}(M(6,1),\text{th}6)]*qp;$

$M62=[\text{diff}(M(6,2),\text{th}1),\text{diff}(M(6,2),\text{th}2),\text{diff}(M(6,2),\text{th}3),\text{diff}(M(6,2),\text{th}4),\text{diff}(M(6,2),\text{th}5),\text{diff}(M(6,2),\text{th}6)]*qp;$

$M63=[\text{diff}(M(6,3),\text{th}1),\text{diff}(M(6,3),\text{th}2),\text{diff}(M(6,3),\text{th}3),\text{diff}(M(6,3),\text{th}4),\text{diff}(M(6,3),\text{th}5),\text{diff}(M(6,3),\text{th}6)]*qp;$

$M64=[\text{diff}(M(6,4),\text{th}1),\text{diff}(M(6,4),\text{th}2),\text{diff}(M(6,4),\text{th}3),\text{diff}(M(6,4),\text{th}4),\text{diff}(M(6,4),\text{th}5),\text{diff}(M(6,4),\text{th}6)]*qp;$

$M65=[\text{diff}(M(6,5),\text{th}1),\text{diff}(M(6,5),\text{th}2),\text{diff}(M(6,5),\text{th}3),\text{diff}(M(6,5),\text{th}4),\text{diff}(M(6,5),\text{th}5),\text{diff}(M(6,5),\text{th}6)]*qp;$

$M66=[\text{diff}(M(6,6),\text{th}1),\text{diff}(M(6,6),\text{th}2),\text{diff}(M(6,6),\text{th}3),\text{diff}(M(6,6),\text{th}4),\text{diff}(M(6,6),\text{th}5),\text{diff}(M(6,6),\text{th}6)]*qp;$

$Mp=[M11,M12,M13,M14,M15,M16;...$

$M21,M22,M23,M24,M25,M26;...$

$M31,M32,M33,M34,M35,M36;...$

$M41,M42,M43,M44,M45,M46;...$

$M51,M52,M53,M54,M55,M56;...$

$M51,M62,M63,M64,M65,M66];$

$k=1/2*\text{transpose}(qp)*M*qp;$

$dk=[\text{diff}(k,\text{th1});\text{diff}(k,\text{th2});\text{diff}(k,\text{th3});\text{diff}(k,\text{th4});\text{diff}(k,\text{th5});\text{diff}(k,\text{th6})];$

$C=Mp*qp-dk;$

$a1=\text{subs}(\text{tau1},\text{th1p},0);$

$a2=\text{subs}(a1,\text{th2p},0);$

$a3=\text{subs}(a2,\text{th3p},0);$

$a4=\text{subs}(a3,\text{th4p},0);$

$a5=\text{subs}(a4,\text{th5p},0);$

$a6=\text{subs}(a5,\text{th6p},0);$

$a7=\text{subs}(a6,\text{th1pp},0);$

$a8=\text{subs}(a7,\text{th2pp},0);$

$a9=\text{subs}(a8,\text{th3pp},0);$

$a10=\text{subs}(a9,\text{th4pp},0);$

$a11=\text{subs}(a10,\text{th5pp},0);$

$a12=\text{subs}(a11,\text{th6pp},0);$

$b1=\text{subs}(\text{tau2},\text{th1p},0);$

$b2=\text{subs}(b1,\text{th2p},0);$

$b3=\text{subs}(b2,\text{th3p},0);$

$b4=\text{subs}(b3,\text{th4p},0);$

$b5=\text{subs}(b4,\text{th5p},0);$

$b6=\text{subs}(b5,\text{th6p},0);$

$b7=\text{subs}(b6,\text{th1pp},0);$

$b8=\text{subs}(b7,\text{th2pp},0);$

$b9=\text{subs}(b8,\text{th3pp},0);$

$b10=\text{subs}(b9,\text{th4pp},0);$

$b11=\text{subs}(b10,\text{th5pp},0);$

$b12=\text{subs}(b11,\text{th6pp},0);$

c1=subs(tau3,th1p,0);
c2=subs(c1,th2p,0);
c3=subs(c2,th3p,0);
c4=subs(c3,th4p,0);
c5=subs(c4,th5p,0);
c6=subs(c5,th6p,0);
c7=subs(c6,th1pp,0);
c8=subs(c7,th2pp,0);
c9=subs(c8,th3pp,0);
c10=subs(c9,th4pp,0);
c11=subs(c10,th5pp,0);
c12=subs(c11,th6pp,0);

d1=subs(tau4,th1p,0);
d2=subs(d1,th2p,0);
d3=subs(d2,th3p,0);
d4=subs(d3,th4p,0);
d5=subs(d4,th5p,0);
d6=subs(d5,th6p,0);
d7=subs(d6,th1pp,0);
d8=subs(d7,th2pp,0);
d9=subs(d8,th3pp,0);
d10=subs(d9,th4pp,0);
d11=subs(d10,th5pp,0);
d12=subs(d11,th6pp,0);

e1=subs(tau5,th1p,0);
e2=subs(e1,th2p,0);
e3=subs(e2,th3p,0);
e4=subs(e3,th4p,0);
e5=subs(e4,th5p,0);

e6=subs(e5,th6p,0);
e7=subs(e6,th1pp,0);
e8=subs(e7,th2pp,0);
e9=subs(e8,th3pp,0);
e10=subs(e9,th4pp,0);
e11=subs(e10,th5pp,0);
e12=subs(e11,th6pp,0);

f1=subs(tau6,th1p,0);
f2=subs(f1,th2p,0);
f3=subs(f2,th3p,0);
f4=subs(f3,th4p,0);
f5=subs(f4,th5p,0);
f6=subs(f5,th6p,0);
f7=subs(f6,th1pp,0);
f8=subs(f7,th2pp,0);
f9=subs(f8,th3pp,0);
f10=subs(f9,th4pp,0);
f11=subs(f10,th5pp,0);
f12=subs(f11,th6pp,0);

G=[a12;b12;c12;d12;e12;f12];

```
% Generación de trayectoria articular para un robot
% antropomórfico de 6 grados de libertad
% aplicando una función cúbica
% Carlos Damián Díaz Bravo
% Universidad de Talca - Facultad de Ingeniería
% Ingeniería Civil Mecatrónica
```

```
clear all;
close all;
clc;
```

```
%% Parámetros
```

```
% Posición inicial de las articulaciones
```

```
q1o=90;
q1or=0;
```

```
q11=270;
q11r=180;
```

```
q12=90;
q12r=180;
```

```
q13=180;
q13r=270;
```

```
q14=180;
q14r=270;
```

```
% Posición final de las articulaciones
```

```
q1f=0;
```

```
q1fr=90;
```

```
q2f=180;
```

```
q2fr=270;
```

```
q3f=180;
```

```
q3fr=90;
```

```
q4f=270;
```

```
q4fr=180;
```

```
q5f=270;
```

```
q5fr=180;
```

```
% Definición de tiempo
```

```
tfr=5;
```

```
tf=5;
```

```
div=0.1;
```

```
%% Cálculo del movimiento de las articulaciones
```

```
i=0;
```

```
for t=0:div:5
```

```
    i=i+1;
```

```
    tiempo(i)=t;
```

```
    %Movimiento de la articulación 1
```

```

a10=q1o;
a11=0;
a12=3/(tf*tf)*(q1f-q1o);
a13=-2/(tf*tf*tf)*(q1f-q1o);
qpos_1(i)=a10+a11*t+a12*t*t+a13*t*t*t;
qvel_1(i)=a11+2*a12*t+3*a13*t*t;
qacc_1(i)=2*a12+6*a13*t;
%Movimiento de la articulación 2
a20=q11;
a21=0;
a22=3/(tf*tf)*(q2f-q11);
a23=-2/(tf*tf*tf)*(q2f-q11);
qpos_2(i)=a20+a21*t+a22*t*t+a23*t*t*t;
qvel_2(i)=a21+2*a22*t+3*a23*t*t;
qacc_2(i)=2*a22+6*a23*t;
%Movimiento de la articulación 3
a30=q12;
a31=0;
a32=3/(tf*tf)*(q3f-q12);
a33=-2/(tf*tf*tf)*(q3f-q12);
qpos_3(i)=a30+a31*t+a32*t*t+a33*t*t*t;
qvel_3(i)=a31+2*a32*t+3*a33*t*t;
qacc_3(i)=2*a32+6*a33*t;
%Movimiento de la articulación 4
a40=q13;
a41=0;
a42=3/(tf*tf)*(q4f-q13);
a43=-2/(tf*tf*tf)*(q4f-q13);
qpos_4(i)=a40+a41*t+a42*t*t+a43*t*t*t;
qvel_4(i)=a41+2*a42*t+3*a43*t*t;
qacc_4(i)=2*a42+6*a43*t;

```

```

%Movimiento de la articulación 5
a50=q14;
a51=0;
a52=3/(tf*tf)*(q5f-q14);
a53=-2/(tf*tf*tf)*(q5f-q14);
qpos_5(i)=a50+a51*t+a52*t*t+a53*t*t*t;
qvel_5(i)=a51+2*a52*t+3*a53*t*t;
qacc_5(i)=2*a52+6*a53*t;
end

j=0;
k=0;
for t1=0:div:10

if (t1<=5)
j=j+1;
tiempol(j)=t1;
%Movimiento de la articulación 1 de retorno
a10r=q1or;
a11r=0;
a12r=3/(tfr*tfr)*(q1fr-q1or);
a13r=-2/(tfr*tfr*tfr)*(q1fr-q1or);
qpos_1r(j)=a10r+a11r*t1+a12r*t1*t1+a13r*t1*t1*t1;
qvel_1r(j)=a11r+2*a12r*t1+3*a13r*t1*t1;
qacc_1r(j)=2*a12r+6*a13r*t1;
%Movimiento de la articulación 2 de retorno
a20r=q11r;
a21r=0;
a22r=3/(tfr*tfr)*(q2fr-q11r);
a23r=-2/(tfr*tfr*tfr)*(q2fr-q11r);

```

```

qpos_2r(j)=a20r+a21r*t1+a22r*t1*t1+a23r*t1*t1*t1;
qvel_2r(j)=a21r+2*a22r*t1+3*a23r*t1*t1;
qacc_2r(j)=2*a22r+6*a23r*t1;
%Movimiento de la articulación 3 de retorno
a30r=q12r;
a31r=0;
a32r=3/(tfr*tfr)*(q3fr-q12r);
a33r=-2/(tfr*tfr*tfr)*(q3fr-q12r);
qpos_3r(j)=a30r+a31r*t1+a32r*t1*t1+a33r*t1*t1*t1;
qvel_3r(j)=a31r+2*a32r*t1+3*a33r*t1*t1;
qacc_3r(j)=2*a32r+6*a33r*t1;
%Movimiento de la articulación 4 de retorno
a40r=q13r;
a41r=0;
a42r=3/(tfr*tfr)*(q4fr-q13r);
a43r=-2/(tfr*tfr*tfr)*(q4fr-q13r);
qpos_4r(j)=a40r+a41r*t1+a42r*t1*t1+a43r*t1*t1*t1;
qvel_4r(j)=a41r+2*a42r*t1+3*a43r*t1*t1;
qacc_4r(j)=2*a42r+6*a43r*t1;
%Movimiento de la articulación 5 de retorno
a50r=q14r;
a51r=0;
a52r=3/(tfr*tfr)*(q5fr-q14r);
a53r=-2/(tfr*tfr*tfr)*(q5fr-q14r);
qpos_5r(j)=a50r+a51r*t1+a52r*t1*t1+a53r*t1*t1*t1;
qvel_5r(j)=a51r+2*a52r*t1+3*a53r*t1*t1;
qacc_5r(j)=2*a52r+6*a53r*t1;
end

if (t1>=5)
    k=k+1;

```

```

        tiempo2(k)=t1;
    end

end

% Trayectoria en Simscape

tfinal=[tiempo tiempo2];
posf1= [qpos_1 qpos_1r];
posf2= [qpos_2 qpos_2r];
posf3= [qpos_3 qpos_3r];
posf4= [qpos_4 qpos_4r];
posf5= [qpos_5 qpos_5r];

velf1= [qvel_1 qvel_1r];
velf2= [qvel_2 qvel_2r];
velf3= [qvel_3 qvel_3r];
velf4= [qvel_4 qvel_4r];
velf5= [qvel_5 qvel_5r];

accf1= [qacc_1 qacc_1r];
accf2= [qacc_2 qacc_2r];
accf3= [qacc_3 qacc_3r];
accf4= [qacc_4 qacc_4r];
accf5= [qacc_5 qacc_5r];

qv1=[tfinal;posf1]';
qv2=[tfinal;posf2]';
qv3=[tfinal;posf3]';
qv4=[tfinal;posf4]';

```

```
qv5=[tfinal;posf5]';
```

```
qv1v=[tfinal;velf1]';
```

```
qv2v=[tfinal;velf2]';
```

```
qv3v=[tfinal;velf3]';
```

```
qv4v=[tfinal;velf4]';
```

```
qv5v=[tfinal;velf5]';
```

```
qv1a=[tfinal;accf1]';
```

```
qv2a=[tfinal;accf2]';
```

```
qv3a=[tfinal;accf3]';
```

```
qv4a=[tfinal;accf4]';
```

```
qv5a=[tfinal;accf5]';
```

```
%% Resultados
```

```
format short
```

```
figure
```

```
plot(tiempo,qpos_1,'-',tiempo,qvel_1,'--',tiempo,qacc_1,'.-');
```

```
xlabel('t [s]')
```

```
ylabel('pos [deg] vel [deg/s] acc [deg/s^2]')
```

```
grid on
```

```
figure
```

```
plot(tiempo,qpos_2,'-',tiempo,qvel_2,'--',tiempo,qacc_2,'.-');
```

```
xlabel('t[s]')
```

```
ylabel('pos [deg] vel [deg/s] acc [deg/s^2]')
```

```
grid on
```

```
figure
```

```
plot(tiempo,qpos_3,'-',tiempo,qvel_3,'--',tiempo,qacc_3,'.-');
```

```
xlabel('t[s]')
ylabel('pos [deg] vel [deg/s] acc [deg/s^2]')
grid on
```

```
figure
plot(tiempo,qpos_4,'-',tiempo,qvel_4,'--',tiempo,qacc_4,'.-');
xlabel('t[s]')
ylabel('pos [deg] vel [deg/s] acc [deg/s^2]')
grid on
```

```
figure
plot(tiempo,qpos_5,'-',tiempo,qvel_5,'--',tiempo,qacc_5,'.-');
xlabel('t[s]')
ylabel('pos [deg] vel [deg/s] acc [deg/s^2]')
grid on
```

```
figure
plot(tiempo2,qpos_1r,'-',tiempo2,qvel_1r,'--',tiempo2,qacc_1r,'.-');
xlabel('t[s]')
ylabel('pos [deg] vel [deg/s] acc [deg/s^2]')
grid on
```

```
figure
plot(tiempo2,qpos_2r,'-',tiempo2,qvel_2r,'--',tiempo2,qacc_2r,'.-');
xlabel('t[s]')
ylabel('pos [deg] vel [deg/s] acc [deg/s^2]')
grid on
```

```
figure
plot(tiempo2,qpos_3r,'-',tiempo2,qvel_3r,'--',tiempo2,qacc_3r,'.-');
xlabel('t[s]')
```

```
ylabel('pos [deg] vel [deg/s] acc [deg/s^2]')  
grid on
```

```
figure  
plot(tiempo2,qpos_4r,'-',tiempo2,qvel_4r,'--',tiempo2,qacc_4r,'.-');  
xlabel('t[s]')  
ylabel('pos [deg] vel [deg/s] acc [deg/s^2]')  
grid on
```

```
figure  
plot(tiempo2,qpos_5r,'-',tiempo2,qvel_5r,'--',tiempo2,qacc_5r,'.-');  
xlabel('t[s]')  
ylabel('pos [deg] vel [deg/s] acc [deg/s^2]')  
grid on
```

```
figure  
plot(tfinal,posf1,'-',tfinal,velf1,'--',tfinal,accf1,'.-');  
xlabel('t[s]')  
ylabel('pos [deg] vel [deg/s] acc [deg/s^2]')  
grid on
```

```
figure  
plot(tfinal,posf2,'-',tfinal,velf2,'--',tfinal,accf2,'.-');  
xlabel('t[s]')  
ylabel('pos [deg] vel [deg/s] acc [deg/s^2]')  
grid on
```

```
figure  
plot(tfinal,posf3,'-',tfinal,velf3,'--',tfinal,accf3,'.-');  
xlabel('t[s]')  
ylabel('pos [deg] vel [deg/s] acc [deg/s^2]')
```

```
grid on
```

```
figure
```

```
plot(tfinal,posf4,'-',tfinal,velf4,'--',tfinal,accf4,'.-');
```

```
xlabel('t[s]')
```

```
ylabel('pos [deg] vel [deg/s] acc [deg/s^2]')
```

```
grid on
```

```
figure
```

```
plot(tfinal,posf5,'-',tfinal,velf5,'--',tfinal,accf5,'.-');
```

```
xlabel('t[s]')
```

```
ylabel('pos [deg] vel [deg/s] acc [deg/s^2]')
```

```
grid on
```