

ESCUELA TÉCNICA SUPERIOR DE INGENIEROS
INDUSTRIALES Y DE TELECOMUNICACIÓN

UNIVERSIDAD DE CANTABRIA



Trabajo Fin de Grado

**ROBOT MANIPULADOR DIBUJANTE SCARA:
DISEÑO Y SIMULACIÓN POR ORDENADOR**

**Drawer SCARA Robot Manipulator: Design and
Simulation by Computer**

Para acceder al Título de

**GRADUADO EN INGENIERÍA EN
TECNOLOGÍAS INDUSTRIALES**

Autor: Julián Alexei Carrera de la Red

Septiembre – 2021

RESUMEN

El tema de este trabajo se inscribe dentro del actual y novedoso campo de la robótica, concretamente, de los robots manipuladores. Se estudia la morfología, la cinemática y la dinámica de los manipuladores industriales, por medio de diversos métodos de análisis, así como los diferentes sistemas de control, para que puedan realizar trayectorias complejas en un plano.

Como aplicación original se incluye el diseño y simulación con un modelo de robot manipulador dibujante SCARA, que permite la escritura de caracteres gráficos y alfanuméricos.

Se utiliza el software Autodesk Inventor para el diseño gráfico de la geometría de la estructura del mecanismo y sus transmisiones; también contamos con las múltiples herramientas de la plataforma Matlab/Simulink para las técnicas de control y software de simulación de la cinemática y dinámica del manipulador.

PALABRAS CLAVE: Robot Dibujante SCARA; Diseño; Simulación

ABSTRACT

The subject of this work falls within the current and novel field of robotics, specifically, of manipulative robots. The morphology, kinematics and dynamics of industrial manipulators are studied by means of various analytical methods, as well as the different control systems, so that they can perform complex trajectories on a plane. The original application includes the design and simulation with a SCARA drawer robot manipulator model, which allows the writing of graphic and alphanumeric characters. The Inventor program is used for the graphic design of the geometry of the structure of the mechanism and its transmissions; we also have the multiple tools of the Matlab/Simulink platform for control techniques and software for simulation of the kinematics and dynamics of the mechanism.

KEY WORDS: Drawer Robot SCARA; Design; Simulation

ÍNDICE

1	INTRODUCCIÓN.....	9
1.1	Justificación del tema.....	9
1.2	Objetivos	13
1.3	Estructura del TFG.....	13
2	FUNDAMENTO TEÓRICO DE UN MANIPULADOR	14
2.1	Definición de robot manipulador.....	14
2.2	Morfología de un manipulador.....	14
2.2.1	Estructura mecánica.....	14
2.2.2	Transmisiones	17
2.2.3	Actuadores	18
2.2.4	Sensores	18
2.2.5	Elementos terminales	20
2.2.6	Tipología.....	21
2.2.7	Concreción sobre el robot SCARA	23
2.3	Cinemática	25
2.3.1	Cinemática directa	25
2.3.2	Parámetros DH.....	26
2.3.3	Cinemática inversa.....	28
2.3.4	Cinemática diferencial	29
2.4	Dinámica	30
2.4.1	Formulación Lagrange-Euler	32
2.4.2	Formulación Newton-Euler	35
2.4.3	Modelo de un motor de corriente continua	36
2.5	Planeación de trayectorias.....	39
2.5.1	Cómo realizar las trayectorias.....	40
2.5.2	Trayectorias en espacio de tarea	42
2.5.3	Trayectorias en el espacio articular	43
2.6	Sistema de control	45
2.6.1	Esquema general.....	45
2.6.2	Estrategia de control: desacoplo articular	46
2.6.3	Respuesta del sistema.....	47
2.6.4	Reguladores	48
2.7	Software específico de robótica	50
2.7.1	Software de modelado.....	50

2.7.2	Software de análisis y simulación	52
3	DISEÑO Y SIMULACIÓN DEL MODELO SCARA.....	59
3.1	DISEÑO	59
3.1.1	Espacio de trabajo.....	60
3.1.2	Dimensiones y estructura del manipulador	61
3.1.3	Características de los motores elegidos:	64
3.1.4	Transmisiones	66
3.1.5	Cinemática del manipulador	69
3.1.6	Características dinámicas de los eslabones	72
3.1.7	Eector final.....	75
3.2	Simulación.....	76
3.2.1	Generación de trayectorias/puntos de control	76
3.2.2	Simulación cinemática	82
3.3	Simulación dinámica completa	87
4	RESULTADOS	95
4.1	Simulación cinemática	95
4.1.1	Trabajando en el espacio cartesiano	97
4.1.2	Trabajando en el espacio articular.....	99
4.2	Comparación de resultados de la simulación cinemática.....	104
4.3	Simulación dinámica	107
5	CONCLUSIONES.....	112
6	REFERENCIAS Y BIBLIOGRAFÍA	115
7	ANEXOS	118
7.1	Planos de los elementos diseñados	118
7.1.1	Base	118
7.1.2	Elemento 2	119
7.1.3	Elemento 3	120
7.1.4	Elemento 4	121
7.1.5	Pinza	122
7.2	Hoja de datos de los motores	123
7.3	Más ejemplos de trayectorias realizadas.....	125

Índice de Figuras

Figura 1.1. Instalación anual de robots industriales por sector.	9
Figura 1.2. Robot Da Vinci.....	10
Figura 1.3. Robot Atlas desarrollado por BostonDynamics	11
Figura 1.4. Robot escritor Quincy	11
Figura 2.1. Robot manipulador y su equivalente humano	15
Figura 2.2. Robot soldador ABB	21
Figura 2.3. Robot pintor ABB	21
Figura 2.4. Detalle pinza Festo.....	21
Figura 2.5. Especificaciones robot SCARA (Robot Omron Cobra).....	24
Figura 2.6. Explicación gráfica de cinemática directa e inversa	25
Figura 2.7. Parámetros DH para un eslabón giratorio	26
Figura 2.8. Nomenclatura de la cadena cinemática	35
Figura 2.9. Esquema motor de corriente continua.....	36
Figura 2.10. Subsistema mecánico de un motor de corriente continua	38
Figura 2.11. Flujo de trabajo en el espacio cartesiano	41
Figura 2.12. Flujo de trabajo en el espacio articular	41
Figura 2.13. Esquema general del sistema de control.....	46
Figura 2.14. Estructura de control desacoplado de un robot de 4 grados de libertad	47
Figura 2.15. Posibles respuestas dinámicas de un sistema	48
Figura 2.16. Diagrama de bloques de un regulador PID	50
Figura 2.17. Ejemplo de robot en Robotics System Toolbox.....	54
Figura 2.18. Bloque World Frame de la librería Simscape Multibody	55
Figura 2.19. Bloque Mechanism Configuration y parámetros de la librería Simscape Multibody.....	55
Figura 2.20. Bloque Solver Configuration de la librería Simscape Multibody	55

Figura 2.21. Bloque Reference Frame de la librería Simscape Multibody	55
Figura 2.22. Bloque Rigid transform y parámetros de la librería Simscape Multibody.....	56
Figura 2.23. Bloque Joint y parámetros de la librería Simscape Multibody	56
Figura 2.24. Ejemplo de importación de modelo 4GDL a Simscape Multibody	56
Figura 2.25. Bloque representante de una pieza en Simscape Multibody.....	57
Figura 2.26. Características de la pieza importada en Simscape Multibody	57
Figura 2.27. PID tuner App de Simulink Control Designer	58
Figura 3.1. Vista en perspectiva del modelo SCARA diseñado	60
Figura 3.2 Manipulador, plano de tarea y espacio de trabajo.....	60
Figura 3.3. Espacio de trabajo óptimo para una hoja DIN A4.....	61
Figura 3.4. Cotas generales del manipulador	62
Figura 3.5. Plástico PLA de bq. Detalle de su ficha de datos	63
Figura 3.6. Ubicación de motores y transmisiones.....	64
Figura 3.7. Motor Nema 17.....	65
Figura 3.8. Motor MG996R	66
Figura 3.9. Detalle de actuadores y transmisiones de las dos primeras articulaciones	67
Figura 3.10. Rueda dentada transmisión 3	67
Figura 3.11. Engranajes transmisión 4	68
Figura 3.12. Detalle de actuadores y transmisiones de las dos últimas articulaciones	68
Figura 3.13. Esquema robot SCARA	69
Figura 3.14. Visualización robot SCARA en RoboAnalyzer.....	69
Figura 3.15. Identificación de variables robot SCARA	71
Figura 3.16. Base del robot en Autodesk Inventor	72
Figura 3.17. Pieza 1 del robot en Autodesk Inventor	73
Figura 3.18. Pieza 2 del robot en Autodesk Inventor	73

Figura 3.19. Pieza 3 del robot en Autodesk Inventor	73
Figura 3.20. Eslabón final del robot en Autodesk Inventor	73
Figura 3.21. Motor Nema 17 en Autodesk Inventor	73
Figura 3.22. Motor MG996R en Autodesk Inventor	74
Figura 3.23. Transmisión 1 del robot en Autodesk Inventor	74
Figura 3.24. Transmisión 2 del robot en Autodesk Inventor	74
Figura 3.25. Transmisión 3 del robot en Autodesk Inventor	74
Figura 3.26. Transmisión 4 del robot en Autodesk Inventor	75
Figura 3.27. Detalle Efecto final en Autodesk Inventor	76
Figura 3.28. Referencia efecto final.....	76
Figura 3.29. Puntos de control de caracteres alfanuméricos	77
Figura 3.30. Paso 1 Imagen a puntos de control	80
Figura 3.31. Paso 2 Imagen a puntos de control	80
Figura 3.32. Paso 3 Imagen a puntos de control	80
Figura 3.33. Paso 3 Imagen a puntos de control	80
Figura 3.34. Paso 5 Imagen a puntos de control	81
Figura 3.35. Paso 6 Imagen a puntos de control	81
Figura 3.36. Paso 7 Imagen a puntos de control	81
Figura 3.37. Paso 8 Imagen a puntos de control	81
Figura 3.38. Paso 9 Imagen a puntos de control	81
Figura 3.39. Animación SCARA dibujante en Robotics System Toolbox.....	86
Figura 3.40. Esquema del modelo dinámico completo en Simulink.....	88
Figura 3.41. Modelo en Simulink de motor de corriente continua	89
Figura 3.42 Modelo en Simulink de motor corriente continua acoplado a articulación prismática.....	90
Figura 3.43. Regular PID articulación 1	91

Figura 3.44 Regulador PID articulación 2.....	92
Figura 3.45 Regulador PID articulación 3.....	92
Figura 3.46 Regulador PID articulación 4.....	93
Figura 3.47 Modelo mecánico SCARA importado con Simscape Multibody	93
Figura 3.48 Detalle de tramas, elementos y articulaciones en Simulink.....	94
Figura 4.1 Imagen original Gato.jpeg.....	95
Figura 4.2 Imagen original Estrella.jpeg.....	95
Figura 4.3 Comparativa densidad de puntos de control	96
Figura 4.4 Animación del modelo dinámico con Mechanics Explorer.....	107
Figura 4.5 Trayectoria resultante en modelo dinámico sin etapa de control ajustada	108
Figura 4.6 Trayectoria resultante en modelo dinámico con etapa de control ajustada	108
Figura 4.7 Resultados simulación dinámica articulación 1	109
Figura 4.8 Resultados simulación dinámica articulación 2	109
Figura 4.9 Resultados simulación dinámica articulación 3	110
Figura 4.10 Resultados simulación dinámica articulación 4	110

1 INTRODUCCIÓN

1.1 JUSTIFICACIÓN DEL TEMA

Desde que en 1961 se instalara el primer robot industrial UNIMATION en General Motors, el sector de la robótica aplicada al mundo industrial ha tenido un gran crecimiento y relevancia hasta convertirse en imprescindible en la gran mayoría de industrias en la actualidad. El informe de *World Robotics 2020 Industrial Robots* muestra un récord de 2,7 millones de robots industriales que operan en fábricas de todo el mundo, un aumento del 12%. [1]

Los robots manipuladores industriales han revolucionado el sector de la fabricación. Cada vez se diseñan manipuladores más rápidos, versátiles e inteligentes. Hoy en día se les pide que hagan mucho más que limitarse a realizar tareas tradicionales repetitivas, pesadas o incluso peligrosas, como la soldadura y el manejo de materiales. Están asumiendo más capacidades y rasgos "humanos" como la destreza, la memoria y el reconocimiento de objetos. Como resultado, se instalan en la línea de producción asumiendo trabajos como: ensamblar electrónica minuciosa, recoger y empaquetar, probar o inspeccionar productos, etc... En definitiva, en multitud de sectores de la industria están optando por su instalación para seguir compitiendo en el mercado, dada la ventaja de tiempo y calidad que supone tener un robot. Este hecho se muestra en la Figura 1.1., extraída de *Summary Word Robotics 2020 industrial Robots*. [1]

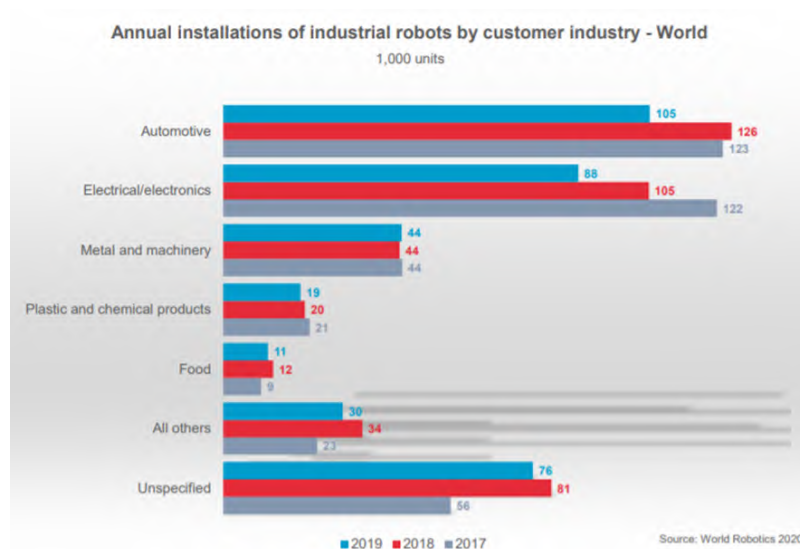


Figura 1.1. Instalación anual de robots industriales por sector.

Fuente: [1]

Las aplicaciones de los robots no se limitan al ámbito industrial y de producción. Existen robots destinados a la medicina, como el famoso robot quirúrgico Da Vinci. El Hospital Universitario Marqués de Valdecilla, referente en Cantabria, tiene dos de estos robots y ha realizado con ellos cerca de 3.000 intervenciones. Así se refiere en *eldiario.es* en marzo de 2021, cuando se incorporó su segundo Da Vinci.

El robot Da Vinci permite dar cobertura a las necesidades quirúrgicas de las distintas especialidades en cirugías complejas de forma mínimamente invasiva, ofreciendo una asistencia con la máxima calidad. Con esta incorporación, Valdecilla continuará siendo un referente en cirugía robótica, tanto en aspectos asistenciales como en la formación e investigación. [2]



Figura 1.2. Robot Da Vinci.

Fuente: <https://www.gruporecoletas.com/noticias/da-vinci-xi-la-tecnologia-mas-puntera-al-servicio-de-los-pacientes/>

Otro ejemplo muy llamativo es el robot humanoide Atlas, desarrollado por BostonDynamics. Hace unos meses se hizo famoso porque sus creadores hicieron un [vídeo](#) donde este robot bailaba y saltaba de manera tan fluida que resultaba increíble. Lleva los límites de la robótica a niveles que se pensaban de ciencia ficción. Las especificaciones de este robot son impresionantes. Es un robot móvil que tiene un total de 28 articulaciones accionadas por una unidad de potencia hidráulica compacta que lleva consigo. El sistema de control que lo gobierna incorpora algoritmos complejos que toman decisiones a partir de estímulos dinámicos que involucran a todo el cuerpo y el entorno para planificar sus movimientos. Todas estas características las consigue incorporar en una estructura de 1.5 m de altura y pesando solo 89 kilogramos. Es un peso muy reducido teniendo en cuenta la gran cantidad

de articulaciones que posee accionadas hidráulicamente; esto es posible gracias a que sus piezas están impresas en 3D, para darle la relación fuerza-peso necesaria que le permita realizar los saltos. [3]



Figura 1.3. Robot Atlas desarrollado por BostonDynamics

Fuente: <https://www.bostondynamics.com/sites/default/files/2021-08/atlas-dynamic.jpg>

Un último ejemplo de modelo de robot es el que ha motivado la elección de este tema para nuestro Trabajo Fin de Grado, se trata del robot Quincy, que puede escribir letras, números y representar un total de 64 imágenes, que tiene precargadas en la memoria. Así los niños aprenden la grafía de las letras de una manera más divertida y entretenida.



Figura 1.4. Robot escritor Quincy

Fuente: <https://www.whatsnew.com/2020/05/16/quincy-el-robot-con-el-que-los-ninos-aprenderan-a-escribir-dibujar-y-los-numeros/>

Algo parecido pretendemos con el diseño y simulación de nuestro robot, cuya aplicación sea dibujar figuras en una hoja de papel.

En definitiva, nuestra investigación en este ámbito de la robótica se refiere a un tema de plena actualidad y lo seguirá siendo en un futuro fuertemente automatizado como el que se

presenta. Se prevé que se desarrollarán algoritmos que controlen los robots de una manera más precisa y rápida, con mejoras en visión artificial para el reconocimiento de objetos, toma de decisiones e incorporando Inteligencia Artificial para entrenarlos y hacerlos más eficientes en cualquier tarea que se pueda encomendar.

Además, la robótica es una ciencia que reúne e integra en sí misma conocimientos multidisciplinarios de: Mecánica, Electrotecnia, Automática, Diseño y Computación, disciplinas variadas de las que un ingeniero industrial debe tener noción. De modo que realizar el Trabajo Fin de Grado que da acceso al título de Graduado en Ingeniería en Tecnologías Industriales relacionado con la robótica se considera la mejor manera para ejemplificar y mostrar los conocimientos multidisciplinarios adquiridos en la formación académica.

Por otra parte, la principal motivación para nuestra investigación era realizar un manipulador que contribuyese positivamente a la mejora de la sociedad. Previamente se han expuesto dos ejemplos de robots famosos que realizan un desempeño extraordinario y son clasificados de los mejores del mundo. Estos proyectos tan ambiciosos implican un presupuesto y una cantidad de trabajadores muy grande. Sin embargo, la robótica no debe ser vista como algo lejano e inaccesible que sólo se lo pueden permitir las empresas con un fin meramente productivo. Se ha mostrado un ejemplo de un robot educativo, que puede dibujar para evidenciar que los robots pueden servir de ayuda en el aprendizaje de las niñas y niños y que así estos desarrollen gusto por esta rama de la ciencia. Siguiendo esta idea hemos desarrollado el diseño y simulación del robot manipulador SCARA, que permite representar imágenes y caracteres alfanuméricos.

Para llevar a cabo el diseño y la simulación se han utilizado herramientas de software muy útiles y potentes usadas por las empresas de todo el mundo. El modelo se ha diseñado con Autodesk Inventor, software de modelado CAD 3D parametrizado, y analizado cinemática y dinámicamente con Matlab y Simulink, haciendo uso de sus múltiples toolboxes empleadas en diseño de robots y modelado de sistemas multifísicos como son Robotic System Toolbox, Simscape, Image Processing Toolbox o Simulink Control Design entre otras. La Universidad de Cantabria proporciona licencias a sus estudiantes para poder usar estos programas, de otra manera sería imposible afrontar el precio que cuestan estas herramientas.

1.2 OBJETIVOS

Nuestro TFG pretende alcanzar, entre otros, los siguientes objetivos:

- Diseñar la estructura física, transmisiones y actuadores que comprenden el robot manipulador dibujante SCARA.
- Crear un método de importación de imágenes que determine los puntos de control de la trayectoria a seguir por el manipulador.
- Determinar el mejor método para planificar las trayectorias, buscando un equilibrio entre calidad gráfica en términos de suavidad y ajuste con la referencia tomada y rendimiento.
- Simular cinemática y dinámicamente el robot SCARA diseñado, comprendiendo los resultados que se obtienen.

1.3 ESTRUCTURA DEL TFG

Tras esta breve presentación, nuestro trabajo se organiza en cuatro grandes apartados:

- Una fundamentación teórica general que contiene la Morfología y Tipología; Cinemática, Dinámica, Planeación de trayectorias, Control y Herramientas de software.
- La aportación central lo constituyen los apartados de Diseño y Simulación de nuestro modelo de robot manipulador dibujante.
- Tras la simulación, presentamos el análisis de nuestros resultados.
- Finalmente, se incluyen unas reflexiones a modo de Conclusiones y Perspectivas a futuro para seguir investigando.

Cierran nuestro trabajo las Referencias bibliográficas y algunos anexos ilustrativos.

2 FUNDAMENTO TEÓRICO DE UN MANIPULADOR

El concepto de robot es demasiado amplio; existen tantos sistemas mecánicos y tantas configuraciones que casi a cualquier aparato que realice movimientos precisados por orden computacional se le puede llamar robot. Dentro de este ámbito tan amplio, nuestra investigación se centra en un tipo de robot que se utiliza en la industria llamado robot manipulador o manipulador industrial.

2.1 DEFINICIÓN DE ROBOT MANIPULADOR

El punto de partida para conocer qué es un robot pasa por revisar y citar las definiciones que aportan diferentes organizaciones sobre qué es un robot industrial de tipo manipulador:

Un robot industrial, según la *International Organization for Standardization* (ISO), es un mecanismo accionado programable en dos o más ejes con un grado de libertad, moviéndose dentro de su entorno, para realizar tareas previstas.

La *International Federation of Robotics* se basa en la definición que da ISO del término “robot industrial”: un manipulador multipropósito reprogramable, controlado automáticamente, programable en tres o más ejes, que puede ser fijo en su lugar o móvil para su uso en aplicaciones de automatización industrial. [4]

2.2 MORFOLOGÍA DE UN MANIPULADOR

Un sistema robótico moderno se compone de gran cantidad de subsistemas que deben estar completamente integrados y relacionados para poder llevar a cabo las funciones esperadas. Físicamente en un robot manipulador podemos encontrar subsistemas principales como: la estructura mecánica, transmisiones, sistema de accionamiento, sistema sensorial, elementos terminales, sistema de control y alimentación. Estos sistemas no son exclusivos de la robótica, sino que se usan en otras máquinas, pero en un robot se aprovechan sus funcionalidades y las integran creando un sistema más complejo. [5]

2.2.1 Estructura mecánica

La estructura mecánica de un robot está formada por una serie de cuerpos unidos por medio de articulaciones que permiten el giro y el desplazamiento relativo entre los eslabones que une. Por ejemplo, los manipuladores industriales se asemejan a la morfología de un brazo

humano, por lo que algunos de sus términos se denominan cuerpo, brazo, codo o muñeca. Se muestra en la Figura 2.1.

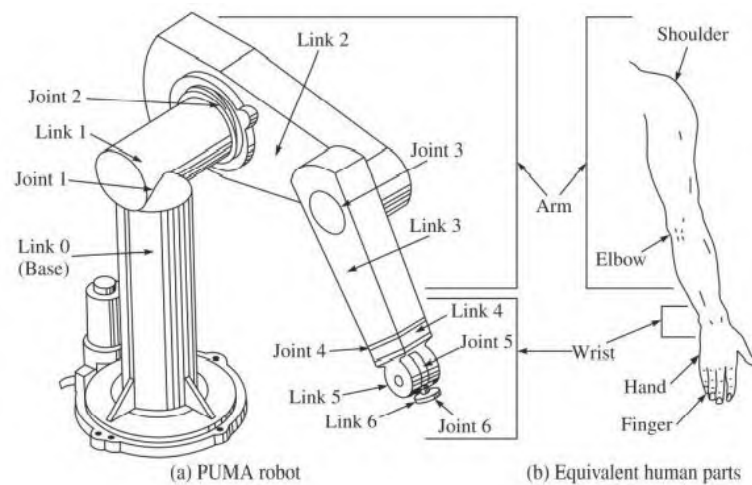


Figura 2.1. Robot manipulador y su equivalente humano

Fuente: [6]


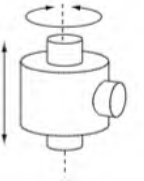
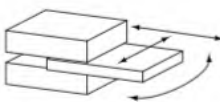

Los movimientos que las articulaciones le permiten realizar al mecanismo pueden ser: de giro, desplazamiento o una combinación de ambos. El movimiento que puede realizar una articulación respecto de la anterior se denomina grado de libertad (GDL). Dependiendo del tipo de articulación, se permitirán más o menos movimientos. La clase del par cinemático vendrá dado por el número de grados de libertad que permite. Es decir, si permite sólo 1 gdl será un par de clase I, si permite movimiento en 2 gdl será de clase II, etc.

Franz Reuleaux, en 1876, indicó los posibles movimientos relativos entre dos elementos de contacto, así podrán existir cinco tipos de articulaciones [7]. Se muestran en la Tabla 2.1:

Tabla 2.1 Tipos de articulaciones

Fuente: [6]

Dibujo	Articulación	Grados de libertad (GDL)
	Prismática	1 GDL

	Rotación	1 GDL
	Cilíndrica	2 GDL
	Planar	3 GDL
	Esférica o Rótula	3 GDL

En la práctica, en robótica sólo se usan las articulaciones de 1 GDL, es decir, las articulaciones de rotación y prismática. Si un manipulador requiere de más grados de libertad en una articulación, en su fase de diseño se tomará como que son varias articulaciones de 1 GDL, pero unidas por eslabones con longitud nula.

Para posicionar y orientar un cuerpo en cualquier posición en el espacio se necesitan definir seis parámetros; tres para la posición y tres para la orientación. De este modo, si se desea que un robot pueda posicionar y orientar su extremo perfectamente en el espacio se necesitará que este al menos tenga 6 GDL.

Cuantos más grados de libertad tenga un robot, mayor versatilidad tendrá para realizar tareas que requieran colocar su extremo en cualquier posición. Aunque la mayoría de los robots modernos ya cuentan de 6 GDL, hay algunas tareas más específicas como las labores de montaje que se necesitan más grados de libertad para permitir que el extremo alcance la posición objetivo incluso, si hay elementos en el espacio de trabajo que obstaculizan su desplazamiento. [8]

Por otra parte, si la tarea está restringida a un plano y no necesita de orientación tridimensional se puede optar por un robot de 4 / 5 GDL. Por ejemplo, este tipo se utiliza en

las labores de pintura de superficies planas. Si el número de grados de libertad es mayor que los que necesitamos para realizar la tarea, decimos que el robot es redundante.

2.2.2 Transmisiones

Como su propio nombre indica, las transmisiones son los elementos que transmiten el movimiento desde los actuadores hasta las articulaciones. Los reductores también se incluirán dentro de este conjunto, puesto que modifican el par y la velocidad de la salida del actuador, consiguiendo los valores adecuados para el movimiento de los elementos del robot.

Una característica principal de un robot es la velocidad con la que puede modificar su posición el extremo. Una alta velocidad de trabajo implica grandes aceleraciones positivas y negativas en un corto periodo de tiempo, con lo que las transmisiones sufren si los elementos tienen momentos de inercia elevados. Un sistema automático estará formado por transmisiones de todo tipo dependiendo de cómo esté diseñado y las funciones que se espera que realice. Por lo general, se pueden clasificar por el movimiento de entrada y el movimiento de salida:

Circular-Circular. Es el tipo que usan los engranajes o las cadenas. El par transmitido es elevado y se puede transmitir a gran distancia; pero generan mucho ruido, son propensos a la holgura y a la deformación.

Circular-Lineal. Este tipo lo usan los tornillos sin fin. Como ventaja es que la holgura es reducida, pero se genera mucho rozamiento y necesitan estar correctamente lubricados.

Lineal-Circular. En esta transmisión se basa la cremallera, la holgura que tiene está en la media, pero el rozamiento y la dificultad en el control son sus mayores desventajas.

De una transmisión se espera que tenga poca holgura para que esta no afecte al movimiento que transmite, aunque no se puede evitar completamente porque al tratarse de elementos mecánicos están sometidos a rozamiento y desgaste. También se espera que pueda transmitir de manera continua un par elevado a una determinada distancia.

El mecanismo de transmisión se complementa con los reductores. Estos elementos se necesitan para reducir la velocidad del motor eléctrico en un único paso. Se escogen aquellos con un diseño ligero y compacto que produzcan el mínimo rozamiento posible.

2.2.3 Actuadores

Según varios autores en el libro *Fundamentos de robótica* (2007), los actuadores son elementos que aportan el par necesario para que los cuerpos del manipulador se muevan de manera precisa. Se pueden clasificar dependiendo de la energía que utilizan para realizar el movimiento en actuadores neumáticos si utilizan aire, hidráulicos si utilizan un fluido líquido, o eléctricos si el par lo aporta un motor eléctrico.

Cada tipo de actuador tiene sus ventajas e inconvenientes:

Los actuadores neumáticos usan aire comprimido a una presión entre 5 y 10 bares. Son rápidos, baratos, sencillos y robustos. En contra, es difícil realizar un control continuo por lo que su precisión es limitada, necesitan una instalación especial y provocan mucho ruido.

Los actuadores hidráulicos funcionan a base de fluidos a una presión entre 50 y 100 bares. Tienen una alta relación potencia-peso, autolubrificantes, alta capacidad de carga, rápidos y buena estabilidad frente a cargas estáticas. En contra, su mantenimiento es difícil, son caros, requieren de una instalación especial, son susceptibles de fugas.

Los actuadores eléctricos ejercen el par requerido con motores eléctricos de corriente continua o servomotores, con motores paso a paso o con motores de corriente alterna. En general, la utilización de estos motores trae ventajas por su fácil control, sencillez y precisión. La gran desventaja es que no pueden ejercer mucha potencia. [9] [5]

2.2.4 Sensores

Los manipuladores necesitan información sobre el entorno y de su propio estado para realizar las tareas con la precisión y velocidad adecuada indicada por el sistema de control. [10]

Un sensor es un elemento físico que emplea algún tipo de fenómeno natural para detectar una variable y así poder medirla. Dependiendo del tipo que sea puede captar señales de posición, velocidad, aceleración, fuerza, temperatura...

La información relacionada con el estado del manipulador, es decir, posición, velocidad, aceleración de sus articulaciones y cuerpos la obtiene con sensores internos. La información relacionada con su entorno, es decir, posición de obstáculos, objetos, temperatura... la captan con sensores externos.

Sensores de posición

Los sensores de posición miden la posición de cada articulación, es decir, el ángulo de articulación de un robot. De estos ángulos de articulación, se puede encontrar la posición del efector final, es decir, su posición y orientación, por medio de la cinemática directa que se verá más adelante. Para el control de posición angular se emplean fundamentalmente sensores denominados encoders y resolvers. [9]

Los encoders se basan en los pulsos que emite un fotorreceptor cuando le llega un haz de luz que atraviesa un disco marcado. Hay dos tipos: encoders incrementales y absolutos.

Los encoders incrementales se basan en un disco transparente al que se le han agregado unas marcas a una distancia fija y de un fotorreceptor. Se proyecta un haz de luz hacia el disco y el fotorreceptor capta luz con el giro del disco que deja pasar luz entre cada marca. Llevando una cuenta del número de pulsos que capta el fotorreceptor se puede saber el ángulo girado. La resolución de este sensor depende de cuántas marcas se pueden realizar en el disco.

Los encoders absolutos funcionan de manera parecida que los incrementales, se diferencian en que las marcas siguen un patrón en código binario consiguiendo que cada posición quede determinada de manera única. La resolución de este tipo de sensor viene definida por el número de bits del encoder, desde 28 a 219.

Los resolvers se basa en utilizar bobinas, al girar, el acoplamiento entre estas varía haciendo que la señal dependa del seno del ángulo de giro. La resolución con este método sólo está limitada al convertir la señal analógica en digital y la electrónica asociada. Aun así, este método es más recomendable que el de encoder óptico.

Sensores de velocidad

La información sobre la velocidad de las articulaciones y los elementos del robot es necesaria para que este funcione correctamente desde el punto de vista dinámico. Esta información puede venir dada directamente por el propio accionamiento del motor, aunque otras veces el sistema de control es la que se encarga de estos parámetros.

El sensor que se usa es un taco generatriz. Esta emite una tensión proporcional a la velocidad de giro de su eje, por ejemplo, 10 mV por rpm. Otra alternativa es usar los datos proporcionados por los sensores de posición y derivarlos para obtener sus velocidades. [9]

Sensores externos

Estos sensores dan información al sistema sobre el entorno, como objetos u obstáculos que el robot debe tener en cuenta al realizar sus movimientos. De esta manera, el comportamiento es más flexible, permite realizar tareas más complejas y toma de decisiones en tiempo real.

Los sensores permiten detectar variables como el alcance, la proximidad y el contacto. El alcance y la proximidad son parámetros que el robot necesita para el guiado de las articulaciones en el entorno, mientras que los sensores de contacto se usan en los elementos terminales detectando la fuerza y/o presión con que se agarran las herramientas u objetos.

2.2.5 Elementos terminales

Para que un robot tenga una funcionalidad completa, usualmente en su extremo lleva acoplado un efector final que le permita interactuar con el entorno y realizar la tarea para la que se ha diseñado. Pueden ser tanto elementos de agarre como herramientas.

Un robot industrial debe ser versátil, por esta razón actualmente la mayoría de empresas fabricantes de manipuladores diseñan el efector final como un módulo intercambiable que se adapta a cada tarea que se precisa.

Este elemento terminal puede ser una pinza para agarrar, sostener y mover objetos. Se distingue entre los posibles agarres: mecánico, ventosas, magnético, adhesivos, ganchos...; se optará por un agarre u otro dependiendo de las especificaciones de peso, forma, tamaño del objeto que debe agarrar y de la fuerza requerida.

Algunas de las herramientas más frecuentes son: pinza para soldadura por puntos donde dos electrodos se cierran sobre la pieza a soldar; pistola de pintura; atornillador; taladro de agujeros. En todas estas operaciones la herramienta debe estar acoplada a la muñeca del manipulador y será una extensión de esta.



Figura 2.2. Robot soldador ABB Figura 2.3. Robot pintor ABB Figura 2.4. Detalle pinza Festo

Fuente Figura 2.2: <https://new.abb.com/products/robotics/es/robots-industriales/irb-1520id>

Fuente Figura 2.3: <https://new.abb.com/products/robotics/es/aplicaciones-por-industrias/plasticos/tratamientos-con-soplete-y-pintura>

Fuente Figura 2.4: https://www.festo.com/cms/es-ve_ve/50802.htm

2.2.6 Tipología

Según se observa en IFR, de acuerdo con los sistemas de uniones que tengan los robots, se pueden clasificar por su estructura mecánica en los siguientes tipos: [4]

Robots articulados: aquellos que su brazo tiene al menos tres juntas rotacionales para posicionarlo. El espacio de trabajo que ocupan es esférico o cuasi-esférico. La estructura de este tipo de robots es la más similar a la morfología del cuerpo humano, pues constan de articulaciones rotacionales y angulares.

Robot cartesiano: aquellos que tienen tres juntas prismáticas y cuyos ejes se corresponden con el sistema de coordenadas cartesianas. El espacio de trabajo es un prisma y se utilizan cuando el área de trabajo que deben cubrir es extensa y el extremo no necesita de un posicionamiento angular variable.

Robot cilíndrico: aquellos que la disposición de sus ejes se corresponde con el sistema de coordenadas cilíndricas. El espacio de trabajo es cilíndrico.

Robot SCARA: es un acrónimo de Selective Compliance Articulated Robot Arm. Su estructura es similar al robot de tipo cilíndrico, pero la rotación se realiza por medio de dos eslabones. Este robot puede realizar movimientos horizontales de gran alcance. Un SCARA clásico permite moverse libremente en el plano X-Y, mientras que en el eje Z es rígido. Pero, acoplando un efector final que permita el desplazamiento vertical, se completa el diseño del robot aportándole una dimensión extra que permite realizar una variedad de operaciones como la manipulación de materiales.

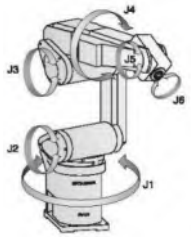
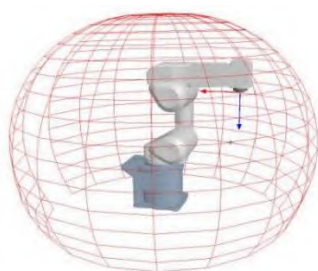

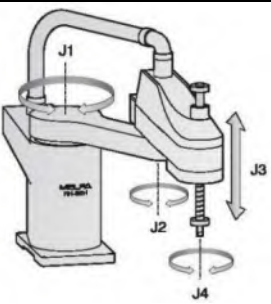
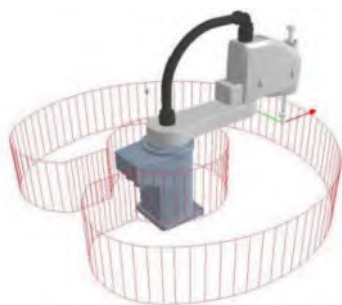

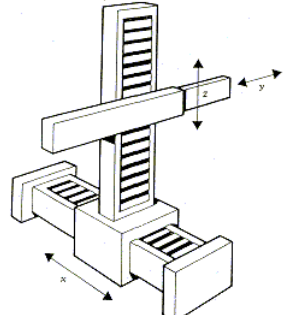


Robot delta o paralelo: su característica principal es que está formado por cadenas cinemáticas cerradas cuyo extremo es una plataforma móvil que se conecta a la base mediante varias cadenas cinemáticas independientes consiguiendo así la traslación y la rotación de su extremo. Destaca en aplicaciones de alta velocidad de empaque, fabricación, ensamblaje y manejo de materiales.

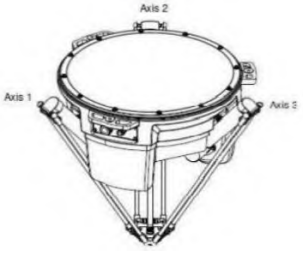


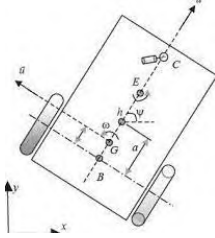
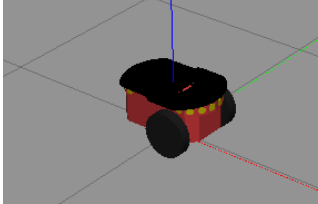

Otros: robots que no entrarían en la definición de los anteriores.

En la siguiente tabla se recoge la estructura de cada tipo de robot y el espacio de trabajo que abarcan, además de una aplicación real en las que se utilizan.

Tabla 2.2 Tipos de robots

Fuente: [11]

Esquema	Estructura cinemática y espacio de trabajo	Robot en operación
 <p>Robot articulado</p>		 <p>Trabajando en una línea de producción</p>
 <p>Robot SCARA</p>		 <p>Trabajando en un laboratorio</p>
 <p>Robot cartesiano</p>		 <p>Organizador de stocks</p>

 <p>Robot paralelo</p>		 <p>Jugando al ping-pong</p>
 <p>Robot móvil</p>		 <p>Robot aspirador</p>

Fuentes imágenes de columna robot en operación:

Robot articulado: <https://new.abb.com/products/robotics/es/robots-industriales/irb-1100>

Robot SCARA: fragmento de vídeo: Epson T-Series SCARA Robots | Pick & Place Demo
<https://youtu.be/UBcRFoShSFI>

Robot cartesiano: <https://italproject.net/soluciones/paletizadores/paletizadores-movimiento-cartesiano/>

Robot paralelo: fragmento de vídeo: Robot Ping-Pong: <https://youtu.be/u3L8vGMDYD8>

Robot Movil: <https://es.mathworks.com/help/robotics/examples/control-a-differential-drive-robot-in-simulink-and-gazebo.html>

2.2.7 Concreción sobre el robot SCARA

El nombre del robot SCARA es un acrónimo de “Selective Compliance Assembly Robot Arm”, o en castellano “Brazo de Robot de Montaje Selectivamente Adaptable”. Se especializan en realizar trabajos repetitivos a gran velocidad y con gran precisión, por eso fundamentalmente se utilizan en aplicaciones de ensamblaje, Pick & Place, manipulación de materiales de precisión, embalaje, aplicaciones de atornillado, aplicación de adhesivos, clasificación de materiales, inspección de piezas y control de calidad.

Según consta en el portal de internet *Revista de Robots* (27, abril, 2021), [12]el robot manipulador Scara se utiliza mayoritariamente en los sectores industriales de la automoción, en la industria alimentaria, en el sector farmacéutico, aeroespacial y en la manipulación de componentes electrónicos.

Como se ha mencionado en el apartado anterior, la estructura de un manipulador SCARA es similar al robot de tipo cilíndrico pero la rotación se realiza por medio de dos eslabones.

Los modelos más usuales de robot SCARA combinan articulaciones rotacionales con una prismática, consiguiendo 3 o 4 grados de libertad, siendo su configuración RRP o RRPR respectivamente. Nuestra investigación se centra en un manipulador de 4 grados de libertad. El robot posee cuatro articulaciones con ejes paralelos entre sí, tres rotacionales y una prismática. La articulación prismática desplaza verticalmente el efector final (z), las dos primeras articulaciones rotacionales ubican el efector en las coordenadas (x, y) y la cuarta articulación rotacional proporciona un grado de orientación. [6]

En la siguiente figura se muestran las características del Robot Omron Cobra. Se puede ver que posee una gran precisión y velocidad. La capacidad de carga (5kg) en relación con el peso del robot, teniendo en cuenta el alcance (600mm), es notable en comparación con otro tipo de modelos como el de tipo brazo robótico articulado.



Repetitividad:	XY ± 0.02 mm, Z ± 0.01 mm, Theta $\pm 0.005^\circ$
Tiempo de Ciclo	Ráfaga 0.43 s, Sostenido 0.60 s
Montaje	Mesa/piso
Peso	31 kg
Configuración Opcional	Controller: eMotionBlox-40, ACE Sight: Si, ePLC Connect: Si, ePLC I/O: Si
Capacidad de Carga (Kg)	5 kg
Ejes	4
Rango	Articulación 1 $\pm 125^\circ$, Articulación 2 $\pm 145^\circ$, Articulación 3 200 mm, Articulación 4 $\pm 360^\circ$
Alcance (mm)	600 mm

Figura 2.5. Especificaciones robot SCARA (Robot Omron Cobra)

Fuente: <https://www.vinssa.com/robot-omron-cobra-scara/>

2.3 CINEMÁTICA

El análisis cinemático es necesario para conocer la posición, velocidad y aceleración de los distintos elementos del sistema robótico. En este análisis no se tiene en cuenta la masa, las fuerzas inerciales de los cuerpos ni las fuerzas que generan el movimiento, solo la geometría de los eslabones y el tipo de articulación que los une. [13]

En cinemática existen dos tipos de problemas: el problema cinemático directo aborda el cálculo de la posición del efector final en función de la de sus eslabones anteriores. El problema cinemático inverso lo resuelve de manera contraria, es decir, a partir del movimiento del efector final busca la posición que deben tener el resto de articulaciones para que se de esa configuración.

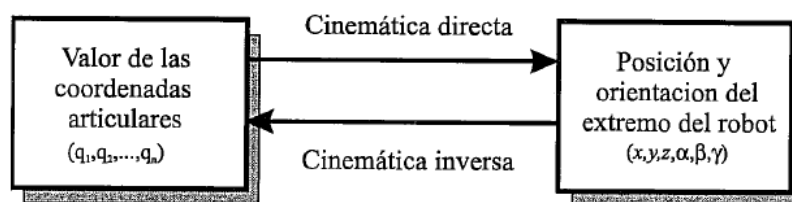


Figura 2.6. Explicación gráfica de cinemática directa e inversa

Fuente: [5]

2.3.1 Cinemática directa

El problema cinemático directo consiste en determinar de manera completa en el espacio la posición y orientación del extremo final (normalmente una herramienta acoplada) respecto de un sistema de referencia fijo. Se determina, como hemos dicho arriba, a partir de las coordenadas de las articulaciones y la geometría de los eslabones.

Para que un elemento libre esté perfectamente definido en un espacio tridimensional se necesitan seis componentes, tres para indicar su posición y tres para indicar su orientación. La forma más sencilla y utilizada es por medio de las coordenadas cartesianas, aunque también es posible emplear coordenadas cilíndricas y esféricas. [5]

Los movimientos, ya sean lineales o rotacionales, se registran en el vector q , denominado vector de estado o vector de coordenadas generalizadas. Este vector contiene el valor instantáneo de cada uno de los grados de libertad.

$$q = \begin{bmatrix} \theta_1 \\ d_2 \\ \vdots \\ \theta_6 \end{bmatrix} \quad 1$$

Es fundamental definir una metodología que ubique de manera única e inequívoca los sistemas coordenados que serán empleados para describir el estado de cada articulación del sistema; uno de los procedimientos más empleados es el conocido como Parámetros de Denavit-Hartenberg (DH), que seguidamente comentamos.

2.3.2 Parámetros DH

La convención DH se aplica a robots de cadena cinemática abierta y consiste en una serie de reglas y pasos que se han adoptado en la comunidad robótica para estandarizar la obtención de modelos cinemáticos. Sitúa los sistemas coordenados asociados a cada eslabón a partir de los valores de las articulaciones de una manera única. Con ello se obtiene una representación del modelo con parametrización mínima. [14]

Empezando por la base del robot, se calcula cada uno de los sistemas de coordenadas de los eslabones a partir de cuatro parámetros, conocidos como parámetros de Denavit-Hartenberg (DH) y que dependen exclusivamente de las características constructivas del robot. [5]

Los cuatro parámetros que definen la transformación son:

- θ . Rotación alrededor del eje z_{i-1}
- d . Traslación a lo largo del eje z_{i-1} una distancia d
- a . Traslación a lo largo del eje x_{i-1} una distancia a
- α . Rotación alrededor del eje x_{i-1}

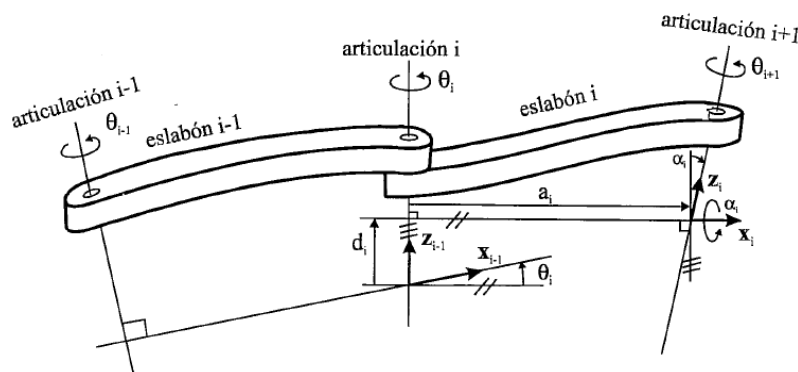


Figura 2.7. Parámetros DH para un eslabón giratorio

Fuente: [5]

El robot se puede encontrar en una configuración cualesquiera, pero es recomendable que sea una posición sencilla de analizar o la posición de inicio. Para obtener los parámetros DH se seguirán los siguientes pasos:

1. Identificar los eslabones, articulaciones y ejes articulares. Numerar los eslabones desde 0 (base) hasta $n-1$ (efector final). Las articulaciones desde 1 hasta n , identificando si son de tipo rotacional o prismático. Siendo n el número de grados de libertad. En cada eslabón habrá una articulación de inicio ($n - 1$) y otra de final (n).
2. El eje z del sistema coordenado debe estar orientado con su dirección coincidente al eje de rotación o la dirección en la que permite el movimiento traslacional a la articulación.
3. El origen del sistema de referencia de cada eslabón se coloca en la intersección del eje z_i con la línea normal común a z_{i-1} y z_i . Si ambos ejes se cortan, se coloca en el punto de corte, si fuesen paralelos se situaría en la articulación $i+1$.
4. En cada articulación ajustar el eje x en la línea normal común a z_{i-1} y a z_i
5. El eje y se situará de modo que constituya un sistema dextrógiro con x_i y z_i
6. Una vez colocados los ejes de coordenadas de cada trama, se procederá a identificar los parámetros θ , d , α , a de cada eslabón.

θ_i como el ángulo que se debe girar tomando como eje x_i para que x_{i-1} y x_i queden paralelos.

d_i como la distancia que hay que desplazarse por el eje z_{i-1} para que x_i y x_{i-1} queden alineados

a_i como la distancia que habría que desplazar el nuevo S_{i-1} para que coincidiese con S_i a lo largo del eje x_i

α_i como el ángulo que se debe girar tomando como eje x_i para que S_{i-1} coincidiese perfectamente con S_i

Convención Denavit-Hartenberg (DH) formal

La convención DH se puede expresar con una matriz de transformación homogénea 4×4 que relaciona los sistemas coordenados de cada articulación $n-1$ y n . Para pasar de uno a otro se

multiplican las matrices de rotación y traslación que dependen de los parámetros geométricos de la cadena cinemática. [13]

$${}^{n-1}T_n = T_{rotación_z}(\theta) \cdot T_{traslación_x}(a) \cdot T_{traslación_z}(d) \cdot T_{rotación}(\alpha) \quad 2$$

Teniendo en cuenta que las matrices de traslación indican movimientos en ejes separados, se pueden unificar estas traslaciones en una sola matriz, quedando:

$${}^{n-1}T_n = T_{rotación_z}(\theta) \cdot T_{traslación_{xz}}(a, d) \cdot T_{rotación}(\alpha) \quad 3$$

La forma matricial quedará de la siguiente manera:

$${}^{n-1}T_n = \begin{bmatrix} \cos\theta & -\sin\theta & 0 & 0 \\ \sin\theta & \cos\theta & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} 1 & 0 & 0 & a \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & d \\ 0 & 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos\alpha & -\sin\alpha & 0 \\ 0 & \sin\alpha & \cos\alpha & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad 4$$

Al multiplicar las tres matrices se llega a la expresión general para la transformada entre dos articulaciones, expresada por el convenio DH:

$${}^{n-1}T_n(\theta_n) = \begin{bmatrix} \cos\theta_n & -\sin\theta_n \cos\alpha_n & \sin\theta_n \sin\alpha_n & a_n \cos\theta_n \\ \sin\theta_n & \cos\theta_n \cos\alpha_n & -\cos\theta_n \sin\alpha_n & a_n \sin\theta_n \\ 0 & \sin\alpha_n & \cos\alpha_n & d_n \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad 5$$

Con DH se utiliza una variable y tres parámetros: el ángulo θ es variable cuando se trata de articulaciones rotacionales, mientras que a , d y α son parámetros que definen geométricamente el eslabón y la posición de los sistemas de referencia.

Si la articulación es prismática, la variable es d y los parámetros fijos son a , α y θ .

De esta manera, tres parámetros son constantes y dependiendo del tipo de articulación será θ o d variable.

2.3.3 Cinemática inversa

En la mayoría de las aplicaciones la trayectoria que debe seguir el efector final está definida por una serie de puntos de control por los que debe pasar con una orientación determinada. La cinemática inversa resuelve este problema, determina la posición de cada articulación en cada punto por el que debe pasar. [15]

Resolver la cinemática inversa del sistema es una tarea más complicada que la cinemática directa cuando se analizan sistemas de varios grados de libertad por las siguientes razones:

- Las ecuaciones que se resuelven generalmente son no lineales, por lo que no siempre se llega a una solución única.

- Pueden existir múltiples soluciones, o incluso infinitas, en el caso de una cinemática redundante.
- Puede que no haya soluciones admisibles. Esto depende de la estructura cinemática del manipulador.

La existencia de soluciones está garantizada solo si la posición y orientación del efector final pertenecen al espacio de trabajo del manipulador. Un manipulador de menos de 6 grados de libertad no puede orientarse en una posición cualquiera del espacio 3D.

Para resolver la cinemática inversa no existe un método genérico optimizado, ya que depende fuertemente de la geometría del manipulador y la configuración de sus articulaciones. Se puede plantear su resolución usando álgebra, analizando la geometría o mediante métodos numéricos iterativos.

La solución algebraica sería la mejor opción cuando se controla un manipulador en tiempo real, pero no siempre garantiza alcanzar una solución, ya que calcula las posiciones articulares a través de las transformaciones algebraicas de la cinemática directa.

El método geométrico busca relaciones geométricas entre las coordenadas del extremo del robot, sus coordenadas articulares y las dimensiones de sus elementos. Es único para cada tipo de manipulador, característica que lo hace inviable para ser un método genérico. A medida que aumenta el número de grados de libertad del sistema, se vuelve una tarea muy complicada de realizar.

La formulación usada en los métodos numéricos para la resolución general del problema cinemático inverso es normalmente de convergencia más lenta que la admisible en el control en tiempo real de los manipuladores robótico. Por eso se prefieren soluciones analíticas en las que se obtienen directamente los valores de las variables articulares. [16]

2.3.4 Cinemática diferencial

El estudio de la cinemática diferencial define las relaciones entre las velocidades de las articulaciones \dot{q} y la velocidad del extremo del manipulador \dot{X} . El vector \dot{q} recoge las velocidades instantáneas de las articulaciones de un sistema de n grados de libertad, mientras que el vector \dot{X} recoge las velocidades cartesianas del extremo. [15]

$$\dot{q} = [\dot{q}_1 \quad \dot{q}_2 \quad \cdots \quad \dot{q}_n]^T$$

$$\dot{X} = [\dot{X}_n \quad \dot{Y}_n \quad \dot{Z}_n \quad \omega_{Xn} \quad \omega_{Yn} \quad \omega_{Zn}]^T \quad 6$$

La relación entre estos vectores es directamente proporcional y está descrita por una matriz denominada Jacobiano. Esta es una herramienta muy importante en el análisis cinemático del sistema; en concreto, es útil para hallar configuraciones singulares, analizar la redundancia de las articulaciones, para el análisis de la cinemática diferencial inversa y derivar algoritmos de dinámica. La cinemática diferencial directa se expresa de la siguiente manera:

$$\dot{X} = J \dot{q} \quad 7$$

donde $J(q)$ es la matriz jacobiana del robot formada por componentes relacionadas con velocidades lineales y angulares. Se representa de la siguiente manera:

$$J(q) = \begin{bmatrix} J_v(q) \\ J_\omega(q) \end{bmatrix} \quad 8$$

Para realizar el paso contrario, es decir, obtener las expresiones que relacionan la velocidad del efector final con las articulaciones, se precisa hacer el análisis cinemático diferencial inverso. Se deberá calcular la inversa de la matriz jacobiana. Existirá en el caso que sea una matriz cuadrada (no debe tener articulaciones redundantes) y su determinante es distinto de cero (no hay singularidades en ese punto).

$$\dot{q} = J^{-1}(q) \dot{X} \quad 9$$

Es frecuente que existan valores de posición donde el jacobiano es singular. En estas singularidades se produce una pérdida de grados de libertad en el espacio cartesiano, es decir, en ese punto existen direcciones en las que no se puede mover el efector. Esta situación se da en los límites del espacio de trabajo, o en su interior, cuando dos o más ejes de articulación están alineados. El análisis cinemático debe tener en cuenta esto y evitarlo siempre que pueda.

2.4 DINÁMICA

La dinámica de un manipulador describe matemáticamente por medio de ecuaciones la relación entre el movimiento y las fuerzas y/o momentos que se deben aplicar para que se origine. En el modelo dinámico se relacionan:

- La posición del robot definida por sus variables articulares y sus derivadas: velocidad y aceleración.
- Las fuerzas y momentos aplicados en las articulaciones.
- Los parámetros geométricos y dinámicos del robot tales como dimensión, centro de masa, masa e inercia de sus elementos.

El cálculo del modelo dinámico se vuelve más complejo cuantos más grados de libertad tenga el manipulador; se puede conocer el movimiento que se origina a partir de las fuerzas aplicadas resolviendo un conjunto de ecuaciones, normalmente, de tipo diferencial de 2º orden. Esto es un proceso complicado en el que muchas veces se hace uso de métodos iterativos muy costosos en términos computacionales y de tiempo.

Aunque es un trabajo complicado, es imprescindible si se quiere realizar una simulación del movimiento del robot, un diseño de la estructura mecánica del robot, dimensionamiento de los actuadores o un diseño del control dinámico del robot. [6]

Un análisis dinámico completo de un manipulador no sólo comprende la dinámica de la estructura mecánica (barras y eslabones), sino también los sistemas de transmisión, actuadores y elementos electrónicos. Para tener todo esto en cuenta en la simulación, se necesita un entorno que permita incorporar estos componentes y trabaje con diferentes sistemas físicos. La herramienta elegida es Simscape, la cual permite crear modelos de sistemas físicos dentro del entorno de Simulink.

En el manipulador se pueden plantear los modelos dinámicos directo e inverso:

El modelo dinámico directo expresa la evolución temporal de las coordenadas articulares del manipulador en función de las fuerzas y pares aplicados por los actuadores.

El modelo dinámico inverso obtiene las fuerzas y pares que se necesita aplicar para que el mecanismo se mueva con unas condiciones cinemáticas de referencia que se marcan como objetivo.

Existen varias formulaciones para resolver la dinámica del sistema, las principales son las de Lagrange-Euler y Newton-Euler. Basándose en estas formulaciones, diversos autores han propuesto algoritmos con objeto de hacerlos más eficientes computacionalmente. Cada una de las formulaciones son equivalentes en cuanto a que intentan describir la dinámica del

manipulador lo más fiel posible, sin embargo, cada método tiene beneficios y desventajas en determinadas aplicaciones por las propias características del método.

Por ejemplo, con la formulación Lagrange-Euler se llega a una serie de ecuaciones diferenciales no lineales de 2º orden bien estructuradas donde aparecen de manera clara los diversos pares y fuerzas que intervienen en el movimiento (inercia, Coriolis, gravedad) muy útiles para el estudio de estrategias de control en el espacio articular, pero, al aplicarlas en tareas en tiempo real, no funcionan tan bien por el elevado tiempo de computación que requieren dichas operaciones con matrices de transformación homogénea. [5]

2.4.1 Formulación Lagrange-Euler

El modelo dinámico de un manipulador se puede obtener de una manera sistemática aplicando la formulación de Lagrange-Euler. Esta formulación se apoya en el concepto de coordenadas generalizadas, las cuales determinan completamente la configuración de cuerpos de un sistema mecánico, y en el lagrangiano L . El lagrangiano es una función escalar que se puede definir como la diferencia entre la energía cinética K y potencial U del sistema mecánico que se está estudiando: [15] [6]

$$L(q, \dot{q}) = K(q, \dot{q}) - U(q) \quad 10$$

Las ecuaciones del movimiento se pueden expresar a partir del lagrangiano como la fuerza generalizada ξ_i asociada al vector de coordenadas generalizadas q_i de la siguiente manera:

$$\frac{d}{dt} \frac{\partial \mathcal{L}}{\partial \dot{q}_i} - \frac{\partial \mathcal{L}}{\partial q_i} = \xi_i \quad 11$$

La energía potencial depende directamente de la posición que ocupa cada elemento en el espacio. En el caso de sólidos rígidos, la energía potencial de cada elemento se define como la cantidad de trabajo necesario para desplazar el centro de masa desde una referencia horizontal hasta la posición final bajo la influencia de la gravedad. La energía potencial total será la suma de las energías potenciales parciales.

$$U = \sum_{i=1}^n U_i = - \sum_{i=1}^n m_i c_i^T g \quad 12$$

donde:

m_i es la masa del eslabón i

c_i^T es el vector que contiene la posición del centro de masas del eslabón i

g es el vector representativo de la gravedad

La energía cinética es la energía que posee un cuerpo por estar en movimiento con una velocidad lineal o angular determinada. En el caso del sólido rígido se puede expresar de esta manera:

$$k = \frac{1}{2} m v^T v + \frac{1}{2} \omega^T I \omega \quad 13$$

Para estudiar un manipulador formado por n articulaciones se debe hacer uso de una formulación que involucre las coordenadas generalizadas del sistema. Las velocidades lineal y angular se pueden expresar mediante la matriz Jacobiana y las derivadas del vector de coordenadas generalizadas así:

$$\begin{aligned} v_i &= J_{v_i}(q) \dot{q} \\ \omega_i &= J_{\omega_i}(q) \dot{q} \end{aligned} \quad 14$$

La energía cinética total será la suma de las energías cinéticas de cada eslabón. Se considerará la masa del elemento como m_i , la velocidad del centro de masa como v_i , la velocidad angular como ω_i y el tensor de inercia I_i .

$$K = \sum_{i=1}^n k_i = \frac{1}{2} \sum_{i=1}^n (m_i v_i^T v_i + \omega_i^T I_i \omega_i) \quad 15$$

Ambos términos del sumatorio se pueden unificar bajo una matriz simétrica y definida positiva I llamada Generalized Inertia matriz (GIM), consiguiendo así que la expresión de energía potencial dependa directamente del vector derivado de coordenadas generalizadas:

$$K = \frac{1}{2} \sum_{i=1}^n (m_i v_i^T v_i + \omega_i^T I_i \omega_i) = \frac{1}{2} \sum_{i=1}^n \dot{q}^T \bar{I}_i \dot{q} \quad 16$$

donde la matriz $n \times n$ \bar{I}_i vendrá definida por:

$$\begin{aligned} \bar{I}_i &= m_i J_{v,i}^T J_{v,i} + J_{\omega,i}^T I_i J_{\omega,i} \\ I &= \sum_{i=1}^n \bar{I}_i \\ K &= \frac{1}{2} \dot{q}^T I \dot{q} \end{aligned} \quad 17$$

La matriz GIM se calculará como el sumatorio de las matrices \bar{I}_i , consiguiendo de esta manera una expresión de la energía cinética del sistema dependiente del vector de coordenadas generalizado.

A partir de las expresiones de la energía potencial y cinemática se desarrolla el lagrangiano, al sustituir en la ecuación 10:

$$L = K - U = \sum_{i=1}^n \frac{1}{2} \dot{q}^T \bar{I}_i \dot{q} + m_i c_i^T g \quad 18$$

Recordando la expresión 12, si se desarrollan las derivadas de las ecuaciones del movimiento sustituyendo el lagrangiano por la expresión 18 se obtienen:

$$\frac{\partial \mathcal{L}(q, \dot{q})}{\partial \dot{q}} = \frac{\partial}{\partial \dot{q}} \left[\frac{1}{2} \dot{q}^T I \dot{q} \right] - \frac{\partial U(q)}{\partial \dot{q}} = I \dot{q} \quad 19$$

$$\frac{d}{dt} \left[\frac{\partial \mathcal{L}(q, \dot{q})}{\partial \dot{q}} \right] = I(q) \ddot{q} + \dot{I}(q) \dot{q} \quad 20$$

$$\frac{\partial \mathcal{L}(q, \dot{q})}{\partial q} = \frac{\partial}{\partial q} \left[\frac{1}{2} \dot{q}^T I \dot{q} \right] - \frac{\partial U(q)}{\partial q} \quad 21$$

Sustituyendo las expresiones 20 y 21 en la ecuación del movimiento se obtiene:

$$\xi = I(q) \ddot{q} + \dot{I}(q) \dot{q} - \frac{\partial}{\partial q} \left[\frac{1}{2} \dot{q}^T I \dot{q} \right] + v(\dot{q}, f_e) + \frac{\partial U(q)}{\partial q} \quad 22$$

Finalmente, las ecuaciones del movimiento de un manipulador de n grados de libertad agrupando los términos, identificándolos y renombrándolos se pueden expresar con esta estructura: [6] [5]

$$\underline{Q} = I(\underline{q}) \underline{\ddot{q}} + C(\underline{q}, \underline{\dot{q}}) \underline{\dot{q}} + F(\underline{\dot{q}}) + G(\underline{q}) \quad 23$$

donde

\underline{q} es el vector que recoge las posiciones de las articulaciones y por tanto la posición completa del manipulador

$\underline{\dot{q}}$ es el vector de velocidades de las articulaciones

$\underline{\ddot{q}}$ es el vector de aceleraciones de las articulaciones

I describe el tensor de inercia del manipulador

C describe las fuerzas centrípetas y de Coriolis

F describe la viscosidad y la fricción de Coulomb (normalmente este término no es considerado en los análisis dinámicos)

G describe el efecto de la gravedad

\underline{Q} es el vector de fuerzas externas asociado al vector de coordenadas generalizadas \underline{q}

2.4.2 Formulación Newton-Euler

La formulación Newton-Euler se usa principalmente para resolver el problema dinámico inverso, es decir, se obtienen los pares y fuerzas que se deben aplicar a las articulaciones para producir el movimiento requerido definido por posiciones, velocidades y aceleraciones de los elementos. [17]

Esta formulación parte del equilibrio de fuerzas aplicando la segunda ley de Newton y/o su equivalente en rotaciones, la ley de Euler en cada eslabón.

$$\sum F = ma \quad \sum T = I\alpha + \omega \times (I\omega) \quad 24$$

Desarrollando correctamente estas ecuaciones para cada elemento de la cadena cinemática que forma el manipulador se llega a una formulación recursiva en la que se obtienen la posición, velocidad y aceleración del eslabón i en referencia a la base del robot a partir de sus eslabones anteriores. Así, partiendo del eslabón 1 se llega hasta el eslabón n . De manera parecida, partiendo de los datos cinemáticos, se obtienen las fuerzas y momentos que actúan en cada eslabón i referidos a la base del robot a partir de las fuerzas externas del eslabón $i+1$. Es decir, se recorre el manipulador desde el eslabón extremo n hasta el elemento base 1.

El algoritmo realiza los cálculos a partir de vectores (operaciones vectoriales como producto vectorial y escalar) en vez de operaciones matriciales como hace el método Lagrange-Euler, siendo este algoritmo más eficiente que el anterior. De hecho, la complejidad computacional de la formulación recursiva de Newton-Euler es del orden $O(n)$, haciendo posible su utilización, aunque el manipulador tenga un número de grados de libertad considerable. En la siguiente Figura 2.8 se muestra la nomenclatura que se utiliza en este método.

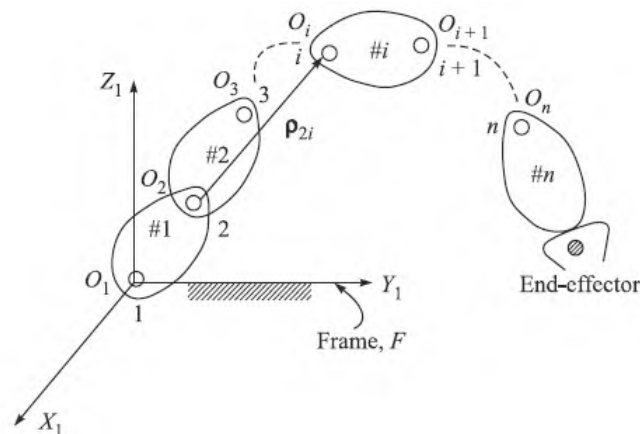


Figura 2.8. Nomenclatura de la cadena cinemática

Fuente: [6]

2.4.3 Modelo de un motor de corriente continua

Un motor de corriente continua (CC) permite transformar energía eléctrica de entrada en energía mecánica rotativa de salida. Está constituido por dos partes fundamentales: una parte fija llamada estátor y una parte móvil llamada rotor. El estátor pertenece al circuito magnético inductor, le da rigidez al conjunto e induce el campo magnético. El rotor pertenece al campo magnético inducido y gira por efecto del campo producido por el estátor. [18]

El principio de funcionamiento de este motor es que un conductor excitado por corriente eléctrica en el seno de un campo magnético experimentará una fuerza. De esta manera, al inducir el estátor un campo magnético por medio de imanes permanentes o con bobinas arrolladas en su cuerpo (depende del tamaño del motor de cc) y el rotor estar recorrido por una corriente eléctrica, se generará un par de giro sobre el rotor.

La ventaja fundamental de los motores de c.c, frente a los motores de corriente alterna c.a., históricamente ha sido su mayor grado de flexibilidad para el control de la velocidad y del par. Actualmente gracias al desarrollo de la electrónica los motores de corriente alterna se han adaptado para mejorar sus prestaciones en cuanto a variabilidad de la velocidad y par.

El modelado del motor de corriente continua controlado por el inducido se obtiene desarrollando las ecuaciones diferenciales que rigen su funcionamiento, tienen varios sistemas diferenciados: el sistema eléctrico, el sistema magnético y el sistema mecánico. [5].

En la Figura 2.9 se puede ver un esquema:

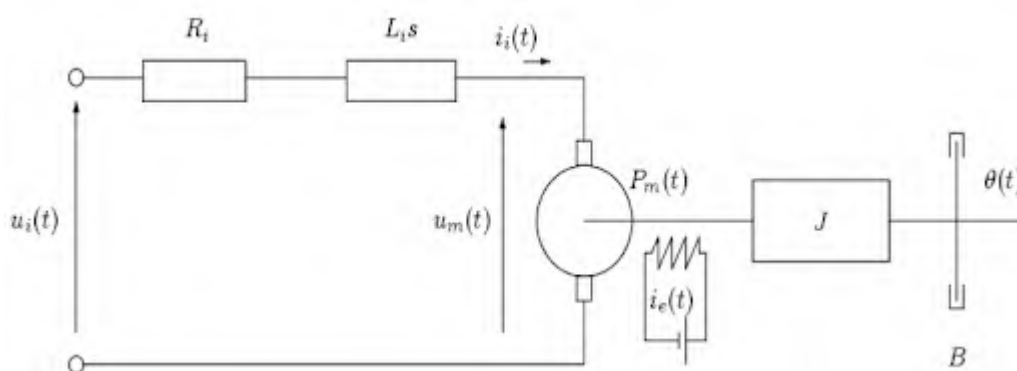


Figura 2.9. Esquema motor de corriente continua

Fuente: <http://isa.uniovi.es/~idiaz/ADSTel/Practicas/ModeladoMotorCC.html>

El subsistema electromagnético está formado por el inductor que proporciona el campo magnético y la corriente de excitación i_e que provoca una corriente en el inducido i_i . El esquema eléctrico del motor consta de una fuente de alimentación u_i , la resistencia e

inductancia del inducido R_i y L_i s respectivamente y una fuerza contraelectromotriz u_m que ejerce el par motor. Aplicando la segunda ley de Kirchhoff a la malla:

$$u_i(t) = R_i \cdot i_i(t) + L_i \cdot \frac{di_i(t)}{dt} + u_m(t) \quad 25$$

$$V = R_i I_i + V_{exc} + E$$

Para determinar el proceso de transformación de energía mecánica en eléctrica del motor de c.c se ha de obtener una ecuación del balance de potencias. Para esto, si se multiplican ambos miembros por I_i resulta:

$$VI_i = R_i I_i^2 + V_{esc} I_i + EI_i \quad 26$$

Estos términos tienen significado físico:

$$P_i = VI_i \quad \text{Potencia eléctrica absorbida por el inducido del motor.}$$

$$P_{cut} = R_i I_i^2 \quad \text{Pérdidas en el cobre del inducido}$$

$$P_{esc} = V_{esc} I_i \quad \text{Pérdidas en los contactos de las escobillas}$$

$$P_m = EI_i \quad \text{Potencia electromagnética desarrollada por el motor}$$

Seguindo esta nomenclatura, el balance de potencias del inducido se convierte en:

$$P_i = P_m + P_{cut} + P_{esc} \quad 27$$

El par interno mecánico desarrollado por el motor será la potencia electromagnética producida por el motor dividida por la velocidad de giro (n expresada en rpm). El par también es directamente proporcional al flujo en el entrehierro y a la corriente del inducido. A partir de la curva par/velocidad propia del motor se puede considerar que la intensidad del inducido es proporcional al par.

$$J_m = \frac{P_m}{2\pi \frac{n}{60}} = K_T \Phi I_i = k_m I_i \quad 28$$

Al girar, el motor actúa también como generador desarrollando una diferencia de potencial a través del inducido proporcional a la velocidad angular del rotor $\dot{\theta}_m$. Esta diferencia de potencial es la fuerza contraelectromotriz u_m .

$$u_m = k_e \dot{\theta}_m \quad 29$$

Finalmente, la relación existente entre el subsistema electromagnético y el mecánico viene dada por la siguiente expresión, dependiente de la diferencia de potencial suministrada y la velocidad de giro de la articulación:

$$u_i(t) = R_i \cdot i_i(t) + L_i \cdot \frac{di_i(t)}{dt} + k_e \dot{\theta}_m(t) \quad 30$$

El subsistema mecánico normalmente consta de un par de engranajes reductores para aumentar el par, reducir la velocidad, permitiendo trabajar a los motores de cc en un régimen donde su eficiencia sea más alta. Se muestra en la siguiente Figura 2.10:

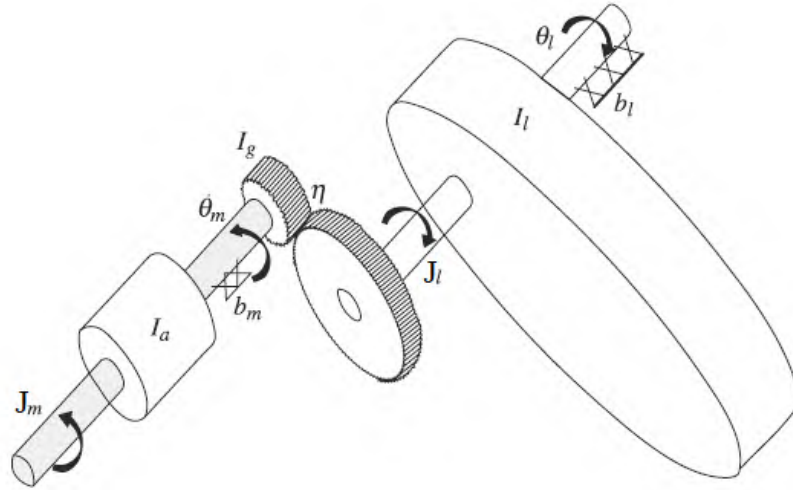


Figura 2.10. Subsistema mecánico de un motor de corriente continua

Fuente: Elaboración propia a partir de [6]

La ecuación del movimiento que rige este sistema relaciona el par motor J_m con el giro final de la articulación θ_l teniendo en cuenta el par de carga J_l , la inercia del conjunto $I_m = I_a + I_g$, la relación de transmisión η y las pérdidas por fricción viscosa b_m .

$$I_m \ddot{\theta}_m + b_m \dot{\theta}_m = J_m - \frac{J_l}{\eta} \quad 31$$

Finalmente, el giro del motor se puede definir con las expresiones que describen el sistema eléctrico y el sistema mecánico, ecuación 30 y 31 respectivamente. Se obtendrá la función de transferencia que rige el sistema completo, por lo que se necesitan en el dominio de Laplace.

$$(L_i s + R_i) \cdot i_i(s) = u_i(s) + k_e s \theta_m(s)$$

$$(I_m s^2 + b_m s) \theta_m(s) = k_m i_i(s) - \frac{J_l(s)}{\eta} \quad 32$$

El ángulo de giro θ_m depende de dos variables: la diferencial de potencial aplicada $u_i(t)$ y el par producido por el eslabón acoplado $J_l(t)$. Las funciones de transferencia serán:

$u_i(s)$ respecto de $\theta_m(s)$ para $J_l(s) = 0$

$$\frac{\theta_m(s)}{u_i(s)} = \frac{k_m}{s[(L_i s + R_i)(I_m s + b_m) + k_e k_m]} \quad 33$$

$u_i(s)$ respecto de $J_l(s)$ para $\theta_m(s) = 0$

$$\frac{\theta_m(s)}{J_l(s)} = \frac{-(L_i s + R_i)/\eta}{s[(L_i s + R_i)(I_m s + b_m) + k_e k_m]} \quad 34$$

Se pueden aplicar simplificaciones a estas funciones de transferencia teniendo en cuenta que la constante de tiempo eléctrica definida por $\tau = L_i/R_i$ es mucho más pequeña que la constante de tiempo mecánica definida por I_m/b_m . Así, dividiendo el numerador y el denominador por R_i y considerando $\tau \approx 0$:

$$\frac{\theta_m(s)}{u_i(s)} = \frac{k_m/R_i}{s(I s + b)} \quad y \quad \frac{\theta_m(s)}{J_l(s)} = \frac{-1/\eta}{s(I s + b)} \quad 35$$

Donde:

$$I \equiv I_m; \quad b \equiv b_m + \frac{k_e k_m}{R_i}$$

2.5 PLANEACIÓN DE TRAYECTORIAS

Para realizar una tarea determinada, un sistema mecatrónico usualmente debe moverse y modificar su configuración que le permita pasar de un estado inicial a otro final. Este movimiento debe ser controlado y dependerá de los rasgos cinemáticos y dinámicos del mecanismo. [19] [15]

Las trayectorias son una sucesión de puntos de control por los que debe pasar el efector final con unas características cinemáticas y temporales determinadas. El espacio recorrido por las articulaciones entre estas posiciones y la orientación puede definirse por medio de interpolación y/o con algoritmos de optimización y suavizado de curvas.

En la planeación de trayectorias se pueden diferenciar dos problemas importantes a resolver: uno es encontrar el camino geométrico que deben seguir las articulaciones y el actuador final para alcanzar el estado deseado; otro es realizar ese camino con las velocidades y aceleraciones adecuadas en un tiempo determinado. Además, se deben incluir restricciones como evitar obstáculos, zonas prohibidas de operación o capacidad física de los actuadores.

Se identifican dos tipos de espacios: el articular y el cartesiano. El espacio articular abarca las posiciones y características cinemáticas de cada una de las articulaciones, este se relaciona directamente con el vector de estados q . [5]

El espacio cartesiano define las posiciones y orientaciones de los eslabones y del efector final. El espacio cartesiano abarca a su vez otros dos: el de tarea y el de trabajo.

El espacio de tarea es aquel donde se espera que el robot realice las órdenes encomendadas. Por ejemplo, si la tarea es controlar la posición de la punta de un marcador en un tablero, entonces el espacio de la tarea es el plano euclidiano. Si la tarea es controlar la posición y orientación de un cuerpo rígido, entonces el espacio de la tarea es el espacio de 6 dimensiones de configuraciones de cuerpo rígido. Para definir el espacio de tarea solamente es necesario considerar la tarea que se va a realizar, no las características del robot.

El espacio de trabajo es una especificación de las configuraciones que puede alcanzar el efector final del robot. El espacio de trabajo a menudo se define en términos de los puntos cartesianos que puede alcanzar el efector final, pero también es posible incluir la orientación. El conjunto de posiciones que se pueden alcanzar con todas las orientaciones posibles se denomina espacio de trabajo diestro. [19]

En nuestro caso, el robot SCARA tiene un espacio de trabajo cilíndrico, la tarea que desempeñará es la escritura de caracteres en una superficie, por lo que el espacio de tarea es el plano euclidiano, una hoja de papel.

2.5.1 Cómo realizar las trayectorias

El movimiento que realiza el efector final se define por medio de puntos de control. En cada punto de control se precisan coordenadas espaciales y orientación. La trayectoria es el camino de puntos intermedios que deben seguir los elementos del manipulador para pasar por los puntos de control con una características cinemáticas determinadas.

La trayectoria que deberá seguir cada articulación entre los puntos de control se puede generar de diversas maneras. Se puede trabajar e interpolar las posiciones en el espacio tarea o en el espacio articular, de ambas maneras el movimiento será similar, pero con ligeras diferencias que se harán menos notables cuanto mayor sea que la densidad de puntos de control. [20]

Trabajando en el espacio de tarea, se interpolan los puntos de control para producir una trayectoria suave entre estos puntos. Una vez obtenidos estos puntos intermedios se calcula el vector de estados de las articulaciones para cada intervalo de tiempo, resolviendo el problema con cinemática inversa. De esta manera se obtienen las posiciones angulares de cada articulación para cada intervalo de tiempo y posición del efector final en el espacio.

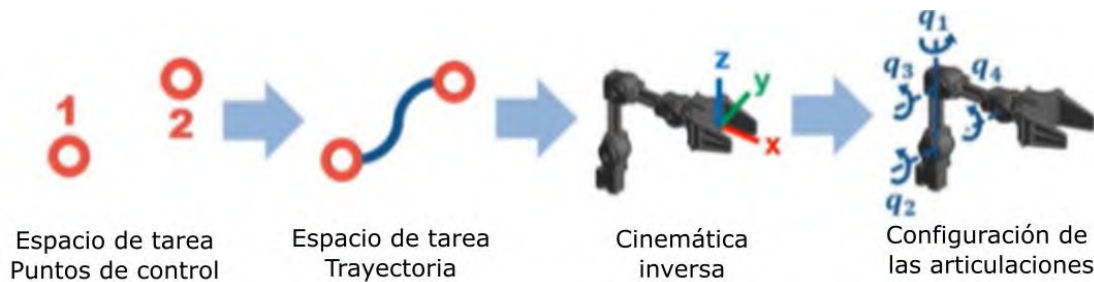


Figura 2.11. Flujo de trabajo en el espacio cartesiano

Fuente: Vídeo Matlab. Trajectory planning for Robots manipulators <https://youtu.be/Fd7wjZDoh7g>

Trabajando en el espacio articular sólo se necesita calcular mediante cinemática inversa el vector de estados para los puntos de control originales. Las posiciones articulares intermedias se calcularán por medio de interpolación; las interpolaciones intermedias normalmente son polinómicas de grado 5 o 3, ya que permite indicar propiedades en cuanto a las velocidades y/o aceleraciones en cada punto de control. Así la curva es suave en términos cinemáticos y los elementos mecánicos seguirán una trayectoria continua sin saltos en las aceleraciones, ya que es el parámetro que más afecta a la integridad del manipulador por estar directamente relacionado con la fuerza.



Figura 2.12. Flujo de trabajo en el espacio articular

Fuente: Vídeo Matlab. Trajectory planning for Robots manipulators <https://youtu.be/Fd7wjZDoh7g>

2.5.2 Trayectorias en espacio de tarea

El movimiento del manipulador necesita tres variables de posición y tres variables de orientación para quedar perfectamente definido. Además, debe ser lo más continuo posible. Lo consigue incorporando mayor cantidad de puntos a la trayectoria. Con mayor cantidad de puntos, el espacio discreto se asemeja con mayor fidelidad al espacio continuo, reduciendo también el intervalo de tiempo entre posiciones.

Hay multitud de técnicas de interpolación para aumentar la densidad de puntos intermedios. Dependiendo de las necesidades y la aplicación del manipulador, se puede suavizar la trayectoria recurriendo a una interpolación lineal, o bien haciendo uso de splines o incluso bsplines.

Con una interpolación lineal los puntos intermedios que se calculen se encontrarán en una línea recta entre las posiciones indicadas por los puntos de control. Cada dirección y orientación se calcula de la siguiente manera siendo:

$P_0 = (X_0, Y_0, Z_0, \theta_0, \psi_0, \phi_0)$ y $P_f = (X_f, Y_f, Z_f, \theta_f, \psi_f, \phi_f)$:

$$x(t) = x_0 + \frac{x_f - x_0}{t_f - t_i}(t - t_i)$$

$$y(t) = y_0 + \frac{y_f - y_0}{t_f - t_i}(t - t_i)$$

$$z(t) = z_0 + \frac{z_f - z_0}{t_f - t_i}(t - t_i)$$

$$\theta(t) = \theta_0 + \frac{\theta_f - \theta_0}{t_f - t_i}(t - t_i)$$

$$\psi(t) = \psi_0 + \frac{\psi_f - \psi_0}{t_f - t_i}(t - t_i)$$

$$\phi(t) = \phi_0 + \frac{\phi_f - \phi_0}{t_f - t_i}(t - t_i)$$

36

Los puntos intermedios de cada coordenada se pueden calcular fácilmente aplicando valores de tiempo en estas expresiones. Cuantos más valores se calculen, más se ajustará la trayectoria a una línea recta entre los puntos de control. Una vez obtenidas las posiciones y orientaciones de todos los puntos, se deberá resolver el vector de estados de las articulaciones para cada posición por medio de la cinemática inversa. Cuantos más puntos se

hayan calculado previamente, mayor gasto computacional se necesitará al resolver la cinemática inversa.

Al utilizar splines o curvas de Bezier la interpolación además de añadir puntos intermedios, genera una curva suave más favorable para la cinemática del manipulador. [21]

El spline cúbico ($k=3$) es el spline más empleado, debido a que proporciona un buen ajuste a los puntos y su cálculo no es excesivamente complejo. En cada tramo se unirán los puntos de control por medio de un polinomio cúbico que tendrá como condiciones que el punto inicial y final y sus pendientes coincidan para que la transición entre tramos sea continua. Si la derivada segunda del polinomio P se hace nula para el primer y último punto sobre el que está definido se dice que es un spline cúbico natural.

$$P_i(x) = a_i x^3 + b_i x^2 + c_i x + d_i \quad i = 0, 1, \dots, n-1 \quad 37$$

Condiciones de interpolación

$$P_i(x_i) = f(x_i) \quad i = 0, 1, \dots, n-1 \quad 38$$

Condiciones de continuidad (en nodos interiores)

$$P_i(x_{i+1}) = P_{i+1}(x_{i+1}) \quad i = 0, 1, \dots, n-2 \quad 39$$

Condiciones de suavidad (en nodos interiores)

$$\begin{aligned} P'_i(x_{i+1}) &= P'_{i+1}(x_{i+1}) & i &= 0, 1, \dots, n-2 \\ P''_i(x_{i+1}) &= P''_{i+1}(x_{i+1}) & i &= 0, 1, \dots, n-2 \end{aligned} \quad 40$$

Condición de spline cúbico natural

$$P''_0(a) = 0 \quad P''_{n-1}(b) = 0 \quad 41$$

2.5.3 Trayectorias en el espacio articular

Se trabaja en el espacio articular con el objetivo de encontrar una curva que garantice una continuidad C^2 y/o C^1 , al interpolar entre los puntos de control obteniendo así la trayectoria que deben seguir las posiciones de las articulaciones. El tipo de curva más fácil de usar y que hace más sencillos los cálculos son los polinomios. Se usarán de grado 3 y 5 porque estos se definen por sus coeficientes que estarán directamente relacionados con las condiciones cinemáticas impuestas en cada punto de control. [13] [15]

Cuando se conoce la velocidad que se debe tener en cada punto se usará es el de grado 3 por tener cuatro coeficientes que se deberán determinar para encontrar el polinomio deseado que constituya una trayectoria desde θ_i hasta θ_f en un tiempo $t_f - t_i$ con una velocidad de inicio $\dot{\theta}_i$ y una velocidad de llegada $\dot{\theta}_f$. Es decir, las velocidades y posición en los puntos de control se determinan, pero las aceleraciones no.

$$\theta(t) = c_0 + c_1 t + c_2 t^2 + c_3 t^3 \quad \dot{\theta}(t) = c_1 + 2c_2 t + 3c_3 t^2 \quad 42$$

Reescribiendo las expresiones 42 para un tiempo inicial y un tiempo final se pueden definir las condiciones frontera del polinomio.

$$\begin{aligned} \theta(t_i) &= c_0 + c_1 t_i + c_2 t_i^2 + c_3 t_i^3 = \theta_i \\ \dot{\theta}(t_i) &= c_1 + 2c_2 t_i + 3c_3 t_i^2 = \dot{\theta}_i \\ \theta(t_f) &= c_0 + c_1 t_f + c_2 t_f^2 + c_3 t_f^3 = \theta_f \\ \dot{\theta}(t_f) &= c_1 + 2c_2 t_f + 3c_3 t_f^2 = \dot{\theta}_f \end{aligned} \quad 43$$

Las ecuaciones que se deben resolver se pueden agrupar con una notación matricial donde las incógnitas serán las constantes C_0 , C_1 , C_2 y C_3 que definirán el polinomio que describirá la trayectoria que debe seguir la articulación entre un punto de control y el siguiente.

$$\begin{bmatrix} \theta_i \\ \dot{\theta}_i \\ \theta_f \\ \dot{\theta}_f \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 1 & t_f & t_f^2 & t_f^3 \\ 0 & 1 & 2t_f & 3t_f^2 \end{bmatrix} \begin{bmatrix} c_0 \\ c_1 \\ c_2 \\ c_3 \end{bmatrix} \quad 44$$

Si se conocen las aceleraciones también, en los puntos de control se necesitará un polinomio con 6 coeficientes para definir posición, velocidad y aceleración tanto en el punto de inicio como final. Este polinomio debe ser de grado 5.

$$\begin{aligned} \theta(t) &= c_0 + c_1 t + c_2 t^2 + c_3 t^3 + c_4 t^4 + c_5 t^5 \\ \dot{\theta}(t) &= c_1 + 2c_2 t + 3c_3 t^2 + 4c_4 t^3 + 5c_5 t^4 \\ \ddot{\theta}(t) &= 2c_2 + 6c_3 t + 12c_4 t^2 + 20c_5 t^3 \end{aligned} \quad 45$$

Las ecuaciones que se deben resolver se pueden agrupar con una notación matricial donde las incógnitas serán las constantes C_0 , C_1 , C_2 , C_3 , C_4 y C_5 que definirán el polinomio que describirá la trayectoria que debe seguir la articulación entre un punto de control y el siguiente.

$$\begin{bmatrix} \theta_i \\ \dot{\theta}_i \\ \ddot{\theta}_i \\ \theta_f \\ \dot{\theta}_f \\ \ddot{\theta}_f \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 2 & 0 & 0 & 0 \\ 1 & t_f & t_f^2 & t_f^3 & t_f^4 & t_f^5 \\ 0 & 1 & 2t_f & 3t_f^2 & 4t_f^3 & 5t_f^4 \\ 0 & 0 & 2 & 6t_f & 12t_f^2 & 20t_f^3 \end{bmatrix} \begin{bmatrix} C_0 \\ C_1 \\ C_2 \\ C_3 \\ C_4 \\ C_5 \end{bmatrix} \quad 46$$

2.6 SISTEMA DE CONTROL

Todo manipulador posee un sistema de control que capta continuamente información de los sensores sobre posición, velocidad, aceleración, fuerza, presión... o incluso sensores de visión o profundidad. Esta información se procesa con el fin de conocer el estado del manipulador y su interacción con el entorno para así dar órdenes precisas a los actuadores y lograr el comportamiento deseado del robot.

El sistema de control en un manipulador indica a cada actuador qué fuerza o par debe ejercer para que los elementos sigan una trayectoria específica cumpliendo con las condiciones cinemáticas deseadas. Es necesario conocer el modelo cinemático y dinámico del sistema donde estarán recogidas las leyes físicas que rigen el movimiento del robot. El modelo dinámico de un modelo con n elementos rígidos conectados en serie deberá satisfacer la siguiente expresión 23 que se aportó en el análisis dinámico.

$$\underline{Q} = I(\underline{q}) \underline{\ddot{q}} + C(\underline{q}, \underline{\dot{q}}) \underline{\dot{q}} + F(\underline{\dot{q}}) + G(\underline{q})$$

2.6.1 Esquema general

El esquema básico del sistema de control single-input/single-output (SISO) en un manipulador se muestra en la Figura 2.13. La señal de posición que se manda a los actuadores del manipulador pasa por varias fases en las que se modula y modifica. Estas etapas están definidas por el comparador, el regulador, los actuadores, transmisión y el manipulador en el lazo abierto. Para asegurar que el movimiento se está produciendo de la manera indicada por la señal de entrada se necesita una realimentación donde los sensores o transductores del robot arrojan información sobre la posición real; un comparador indica el error que se comete.

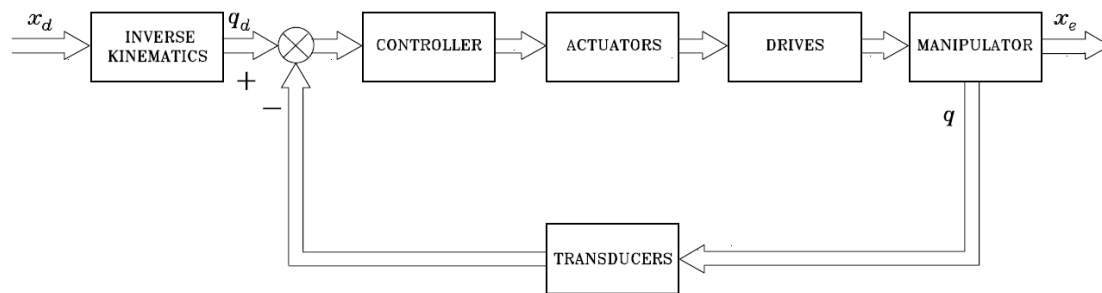


Figura 2.13. Esquema general del sistema de control

Fuente: [15]

2.6.2 Estrategia de control: desacoplo articular

El control puede realizarse utilizando técnicas lineales o no lineales. Las primeras se aplican a sistemas que realizan movimientos lentos y pueden modelarse con ecuaciones lineales de coeficientes constantes. En sistemas donde las articulaciones realizan movimientos rápidos y la dinámica toma más importancia, los métodos lineales se quedan escasos y los errores que se producen no se pueden ignorar. En estos casos se necesita el modelo dinámico del sistema cuyas ecuaciones diferenciales son no lineales y por tanto su control también. [15] [5]

Existen diversas maneras de abordar el problema de control de un sistema con múltiples grados de libertad. En general, el control de un manipulador es un problema de tipo MIMO (Multiple Input Multiple Output) ya que el número de variables de entrada vendrá dado por la cantidad de actuadores y el número de variables de salida por el cantidad de articulaciones o grados de libertad. Enfrentarse directamente a este tipo de problema es complicado y requiere de gran gasto computacional. [22]

La estrategia más sencilla de control es pensar que la totalidad del sistema está formado por sistemas independientes (n articulaciones); controlando el movimiento de cada articulación por separado, sin tener en consideración los demás elementos, se puede controlar el sistema completo. Esta simplificación se puede realizar cuando el sistema tiene múltiples engranajes, ya que estos favorecen la linealización y aíslan a la articulación de los efectos inerciales del resto de elementos. Así las ecuaciones dinámicas se linealizan permitiendo el desacople dinámico de las articulaciones. De esta manera el sistema MIMO se aproxima a n sistemas SISO modelados como una ecuación diferencial de segundo orden en el que se controla cada articulación de manera independiente. La figura 2.14 muestra la estructura de control

desacoplado de un manipulador de 4 grados de libertad. Se puede ver que cada actuador es controlado de manera independiente con su propio regulador.

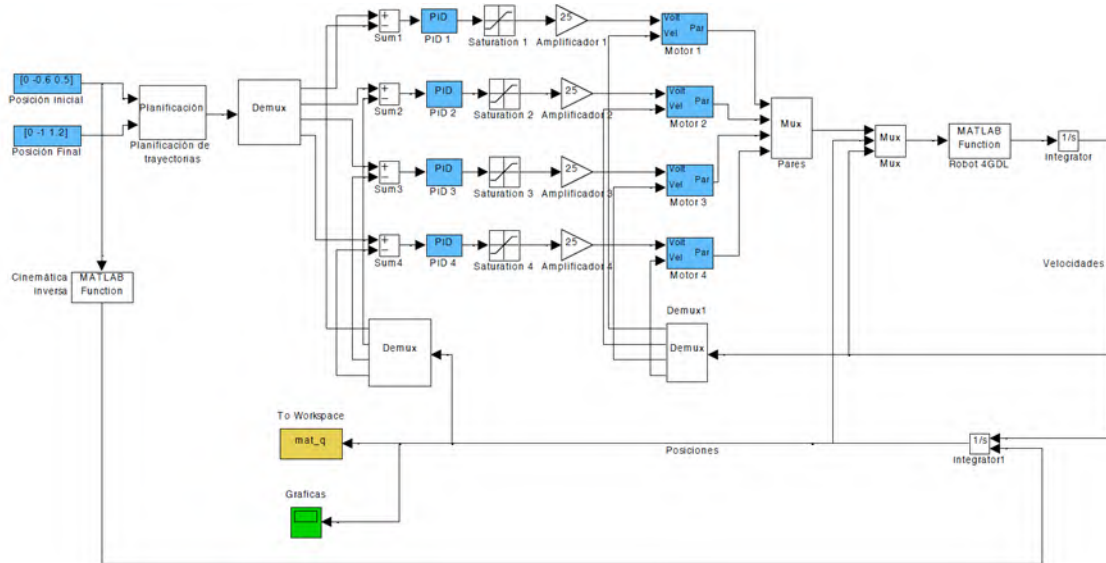


Figura 2.14. Estructura de control desacoplado de un robot de 4 grados de libertad
Fuente: <http://nbio.umh.es/files/2012/04/practica6.pdf>

Al aplicar la estrategia de aproximar el sistema como múltiples sistemas SISO cada articulación se modeliza con una ecuación diferencial ordinaria de segundo orden. En el dominio de Laplace la función de transferencia de un sistema de segundo orden en su forma canónica viene dada por la expresión 47. El modelo dinámico de un motor cc se corresponde con un sistema de segundo orden.

$$\frac{Y(s)}{U(s)} = \frac{k\omega_n^2}{s^2 + 2\zeta\omega_n s + \omega_n^2}; \quad \frac{\theta_m(s)}{u_i(s)} = \frac{k_m/R_i}{s(Is + b)} \quad y \quad \frac{\theta_m(s)}{J_l(s)} = \frac{-1/\eta}{s(Is + b)} \quad 47$$

Siendo $Y(s)$ y $U(s)$ la salida y entrada del sistema respectivamente, k la ganancia estática, ω_n la frecuencia natural y ζ el factor de amortiguamiento.

2.6.3 Respuesta del sistema

La respuesta del sistema viene dada por la naturaleza de los polos de la ecuación característica definida por el denominador de la función de transferencia. Pueden darse varios tipos de respuesta dependiendo de sus raíces y del valor que toma el factor de amortiguamiento: [23]

Sistema sobreamortiguado. Se da cuando los polos son reales. Sucede cuando la fricción predomina, hay un gran amortiguamiento ($\zeta > 1$) que disipa la energía haciendo que el sistema se comporte de manera lenta.

Sistema amortiguado crítico. Se da cuando los polos son reales e iguales. Genera una respuesta no oscilatoria lo más rápida posible. Existe el amortiguamiento justo para que el sistema no sea sobreamortiguado, pero sin llegar a oscilar ($\zeta = 1$).

Sistema subamortiguado. Se da cuando los polos son complejos conjugados. Existe amortiguamiento que disipe energía, pero no impide que el sistema oscile en un primer momento para luego estabilizarse. ($0 < \zeta < 1$).

Sistema sin pérdidas u oscilatorio. Se da cuando sus polos solo tienen parte imaginaria. El amortiguamiento es nulo ($\zeta = 0$), por lo que no se disipa energía y el sistema oscilará indefinidamente. Este caso es teórico ya que en un sistema mecánico siempre habrá rozamiento y pérdida de energía.

Las posibles respuestas se recogen en la Figura 2.15:

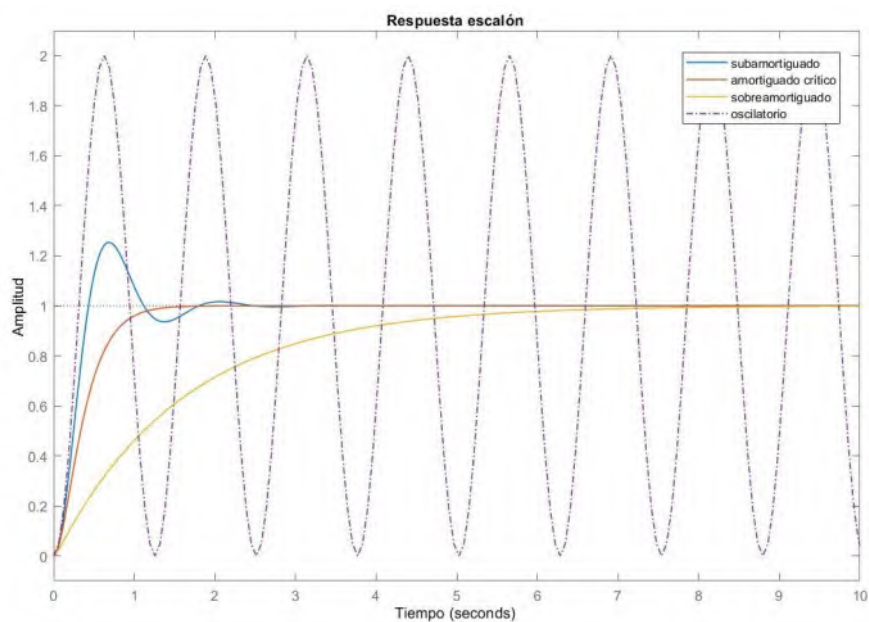


Figura 2.15. Posibles respuestas dinámicas de un sistema

Fuentes: elaboración propia

2.6.4 Reguladores

Es posible que modificar la respuesta que tiene un sistema con el uso de reguladores. Estos son dispositivos que permiten controlar el sistema en lazo cerrado para que alcance el estado de salida deseado. Hay múltiples tipos de reguladores que se usarán dependiendo del sistema; los más usuales son el controlador proporcional, el control proporcional derivativo y el control proporcional integral derivativo. [22] [24]

Regulador proporcional

Su salida es proporcional a la señal de error $e(t)$. Matemáticamente este regulador multiplica la señal de error por una constante K_p . El objetivo es minimizar el error del sistema y aumentar la acción proporcional de K_p conlleva un aumento en la velocidad de respuesta del sistema, disminuye el error en régimen permanente y aumenta la inestabilidad del sistema. Al incrementar K_p se debe encontrar un punto de equilibrio donde la inestabilidad no sea contraproducente.

El modelo matemático de este regulador será:

$$y(t) = k_p e(t) \quad 48$$

Regulador proporcional derivativo

Combina el efecto de un regulador proporcional con un regulador derivativo. El componente diferencial responde a la velocidad en la que el error cambia. Predice linealmente el valor que tendrá la señal en el futuro permitiendo mejorar la respuesta del sistema en cuanto a sobreoscilación y tiempo de respuesta sin afectar sustancialmente al error en régimen permanente. Se aumenta la estabilidad del sistema, pero se sigue manteniendo un error en régimen permanente respecto de la señal de referencia.

El modelo matemático de este regulador será:

$$y(t) = k_p e(t) + k_d \frac{de(t)}{dt} \quad 49$$

Regulador proporcional integral derivativo:

En este regulador se combina el efecto de un regulador proporcional, uno derivativo y uno integral. El componente integral se puede entender como un elemento que acumula la señal de error; cuanto más tiempo pase mayor importancia tendrá este componente y reducirá el error del sistema en régimen permanente.

El regulador PID es el controlador más completo y que más se usa; se consigue una respuesta rápida y un error en régimen permanente muy reducido.

El modelo matemático de este regulador es:

$$y(t) = k_p e(t) + k_d \frac{de(t)}{dt} + k_i \int_0^t e(\tau) d\tau \quad 50$$

El diagrama de bloques con un controlador PID se muestra en la siguiente figura 2.16:

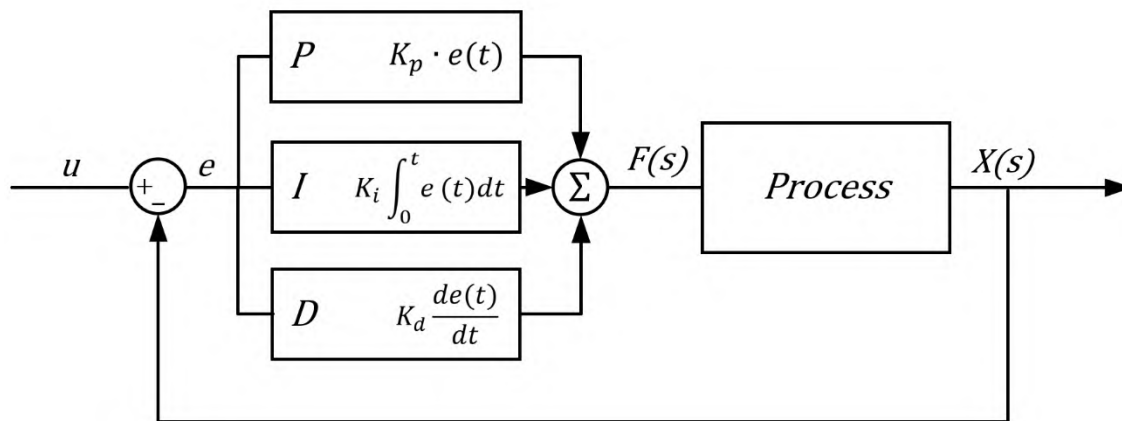


Figura 2.16. Diagrama de bloques de un regulador PID

Fuente: blog.333ohms.com

A continuación, se mencionan consideraciones del diseño para un control robusto: [22]

- a) Desempeño del seguimiento (reducir el error de seguimiento)
- b) Rechazo a perturbaciones (reducir la salida para una entrada de perturbación)
- c) Sensibilidad ante los errores en el modelado (reducir la sensibilidad)
- d) Margen de estabilidad (el modelo debe ser estable en muchas condiciones)
- e) Sensibilidad a ruido en el sensor (reducir la sensibilidad)

2.7 SOFTWARE ESPECÍFICO DE ROBÓTICA

El desarrollo de un producto es un proceso iterativo en el que a partir de una primera versión se va perfeccionando, adaptando y mejorando hasta alcanzar el resultado final. Para desarrollar un robot manipulador en la actualidad es imprescindible hacer uso de software por ordenador, el proceso iterativo se volvería interminable si cada vez que se ajustan los parámetros del modelo se tuviese que empezar casi desde el principio rehaciendo uno por uno todos los cálculos. Con el crecimiento y mejora exponencial de la electrónica y el software específico estas últimas décadas es posible abordar el problema de diseño, análisis y simulación con garantías de éxito.

2.7.1 Software de modelado

Un manipulador industrial necesita una estructura que de rigidez al conjunto con dimensiones definidas. La geometría de cada eslabón juega un papel importante y poder realizar un diseño que se adecúe a las especificaciones, eficaz y estético determinará el éxito del desarrollo.

Existen gran cantidad de software comercial de diseño y modelado tridimensional de piezas y ensamblajes. Se citan programas referentes a nivel industrial: Autodesk Inventor es un software de modelado paramétrico que permite a los usuarios producir modelos 3D precisos para ayudar en el diseño, visualización y simulación de productos antes de su construcción; SolidWorks es el otro programa de modelado paramétrico por excelencia, usado a menudo para crear piezas industriales y ensamblajes por su practicidad y detalle; CATIA se ha utilizado históricamente en aplicaciones aeroespacial pero también en diseño de productos y finalmente Fusion 360 online que ofrece un entorno de trabajo colaborativo que permite el intercambio de archivos en la nube facilitando el control de versiones modificadas por un grupo de usuarios. Todos estos programas mencionados requieren de una licencia de pago; alternativas gratuitas que incorporan algunas de las características y funcionalidades de estos son FreeCad y OpenSCAD. [25]

Al realizar el diseño del robot manipulador se deben tener en cuenta los siguientes conceptos:

Al usar un software de modelado paramétrico se puede realizar una aproximación de la geometría deseada e ir acotando más adelante los parámetros para que se ajuste a las especificaciones deseadas. Incrementando la velocidad de trabajo.

Realizar una geometría coherente, es decir, si se prevé que en una zona puede haber una concentración de tensiones se debe agrandar la superficie crítica o reforzar la zona con un aumento de grosor. Un posible análisis tensional posterior indicará si la solución adoptada es adecuada, sobredimensionada o necesita de otra solución.

Se debe definir un material a cada elemento, así el software calculará su peso y características dinámicas correctamente a partir de su geometría.

Las medidas deben ser precisas y en el momento que se realice el ensamble de los elementos que componen el manipulador deben encajar perfectamente sin solapamientos, si no puede suceder que al realizar la exportación se produzcan errores. Autodesk Inventor incorpora el módulo Solucionador de Contactos que ayuda a detectar este tipo de fenómenos.

La herramienta ensamblaje está diseñada para que se puedan unir los elementos con uniones rígidas o articuladas. Dependiendo del software tiene características más específicas como el indicar la restricción rotacional o prismática, el origen y orientación de la trama en la articulación (como sucede en SolidWorks) o uniones más sencillas donde se indica el eje

común de rotación entre dos elementos si se trata de una articulación primática y el plano que tendrán en contacto con la posibilidad de definir un desfase. Algunos programas permiten realizar animaciones una vez se han establecido la conexión de articulaciones.

Es importante que el software elegido tenga soporte con algún programa de análisis que posibilite importar el modelo creado de manera sencilla. Así la geometría, las restricciones y las características dinámicas que se le ha dado a los elementos del sistema modelado se mantienen y el software de análisis las tiene en cuenta directamente en sus cálculos sin necesidad que el usuario tenga que realizar la tediosa tarea de introducir los parámetros individualmente.

2.7.2 Software de análisis y simulación

Uno de los software de referencia en Matlab-Simulink por ser una poderosa herramienta para analizar robots en todas sus aplicaciones, incorporando módulos de diseño, análisis y simulación cuando se combina con las toolboxes de robótica, modelado de sistemas multifísicos y control que ofrece la compañía MathWorks. [26]

MATLAB se autodefine como una plataforma de programación y cálculo numérico utilizada por millones de ingenieros y científicos para analizar, desarrollar algoritmos y crear modelos. Destaca notablemente en la resolución de sistemas de ecuaciones no lineales y trabajo con matrices. Se complementa perfectamente con Simulink la herramienta de diseño y simulación de elementos dinámicos usada antes de convertir el modelo en hardware.

Simulink es un espacio de trabajo que funciona con bloques, de manera gráfica se puede añadir bloques y desarrollar el modelo físico interconectando subsistemas y componentes que lo forman, el software se encarga de las ecuaciones dinámicas que hay detrás de estos bloques permitiendo al usuario definir componentes paramétricamente a partir de sus especificaciones técnicas. Realiza simulaciones de sistemas lineales y no lineales, en tiempo continuo y discreto e híbridos de todos ellos.

Se han presentado las herramientas base que se utilizan en modelado de sistemas dinámicos de todo tipo. Estas se pueden complementar con herramientas específicas para el modelado de sistemas robóticos. Las toolboxes más relevantes son Robotics System Toolbox de Matlab, Simscape Multibody de Simulink, Simscape electrical, Simulink Control Design. Las características técnicas y posibilidades que ofrecen cada una se explicarán a continuación.

Robotics System Toolbox

Hasta hace unos años, la herramienta más utilizada el estudio de los robots en Matlab era la denominada Robotic Toolbox (RTB) creada por Peter Corke, profesor distinguido de visión robótica en la Universidad de Tecnología de Queensland. Se lanzó en 1994 como su trabajo de tesis doctoral, desde entonces no ha parado de ser ampliamente actualizada hasta el presente convirtiéndose en una referencia. [20]

Matlab a partir de 2015 incluyó la herramienta Robotics System Toolbox, es una toolbox más amplia que la creada por Peter Corke (hay que tener en cuenta que esta ha sido desarrollada por un equipo de trabajadores y la anterior prácticamente por Peter Corke personalmente) pero que incorpora muchas de las funcionalidades que se llevaban utilizando durante los años previos.

Matlab describe este módulo de la siguiente manera:

Robotics System Toolbox™ proporciona herramientas y algoritmos para diseñar, simular y probar manipuladores, robots móviles y robots humanoides. Para los manipuladores y los robots humanoides, esta toolbox incluye algoritmos para comprobación de colisiones, generación de trayectorias, cinemática de avance e inversa y dinámica mediante una representación de árbol de cuerpo rígido. [...] La toolbox ofrece ejemplos de referencia de aplicaciones robóticas industriales habituales. También incluye una librería de modelos de robots industriales disponibles en el mercado que puede importar, visualizar y simular.

En la Figura 2.17 se muestra una captura del vídeo de la empresa MathWorks presentando esta toolbox donde se enseña una simulación de un manipulador que sigue una trayectoria marcada por puntos.

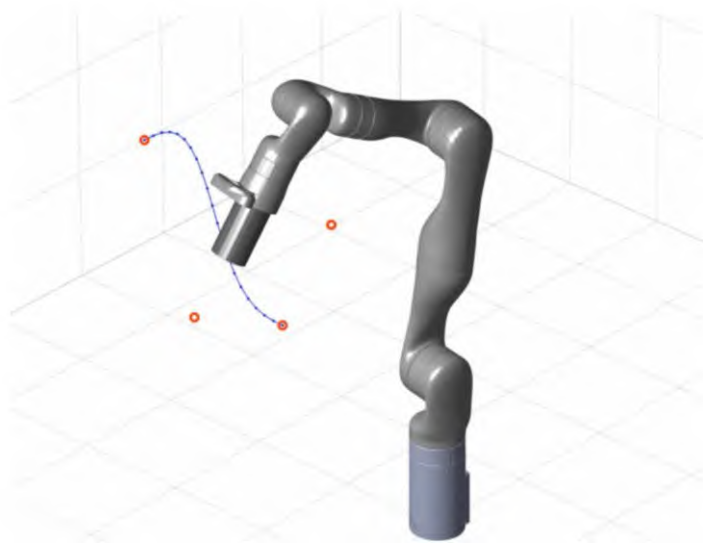


Figura 2.17. Ejemplo de robot en Robotics System Toolbox

Fuente: <https://es.mathworks.com/products/robotics.html>

Simscape Multibody

Es una herramienta que permite simular sistemas mecánicos tridimensionalmente en un entorno de simulación. Se pueden modelar sistemas de varios elementos físicos, por eso incluye 'Multibody' en el nombre. Permite añadir articulaciones, restricciones, fuerzas actuantes y sensores que describan y definan el sistema en movimiento. [27]

Con determinados software de modelado como son Autodesk Inventor, PTC Creo y Solidworks ofrece soporte para importar los modelos directamente del software de diseño manteniendo la geometría, el color, las características dinámicas y de material de los elementos y articulaciones.

Al importar un modelo en Simulink se crea un modelo de bloques de la librería Simscape automáticamente. Todo modelo mecánico necesita de los componentes que se explicarán a continuación a partir de la documentación ofrecida por Matlab:

World Frame. Proporciona la unión entre el mecanismo y el suelo. Es una trama que se predefine en cualquier sistema mecánico, siendo la referencia base de estos. Este bloque se puede duplicar y añadir a varios elementos, pero todos indicarán la referencia base. La salida W conecta la trama de referencia con los elementos que se establezcan como base.

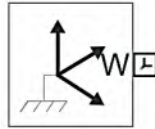


Figura 2.18. Bloque World Frame de la librería Simscape Multibody

Fuente: elaboración propia

Mechanism Configuration. Establece parámetros mecánicos que se aplican a la totalidad del sistema. Se puede definir la fuerza de gravedad a la que está sometido y el delta de linealización especifica el error que se utiliza para calcular las derivadas parciales numéricas para la linealización. La salida C conecta esta configuración con el sistema destino.

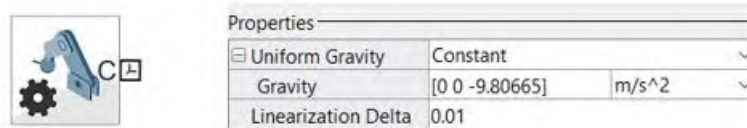


Figura 2.19. Bloque Mechanism Configuration y parámetros de la librería Simscape Multibody

Fuente: elaboración propia

Solver Configuration. En este bloque se definen parámetros de cálculo de la simulación como el método de resolución de ecuaciones diferenciales a usar, la tolerancia o el tiempo de muestra para la simulación entre otros.

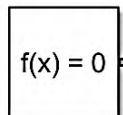


Figura 2.20. Bloque Solver Configuration de la librería Simscape Multibody

Fuente: elaboración propia

Reference Frame. Se usa al crear elementos sólidos, define una trama de referencia en el punto en el que se conecta la salida R.

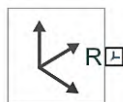


Figura 2.21. Bloque Reference Frame de la librería Simscape Multibody

Fuente: elaboración propia

Rigid transform. Define una transformación traslacional y rotacional entre dos tramas de un mismo elemento. Se puede elegir el método de transformación que mejor se adecúe a los cálculos previos realizados en el manipulador que se diseña. La entrada B se conecta con el

elemento de la trama base y la salida F se conecta con el elemento de la trama siguiente (Forward)

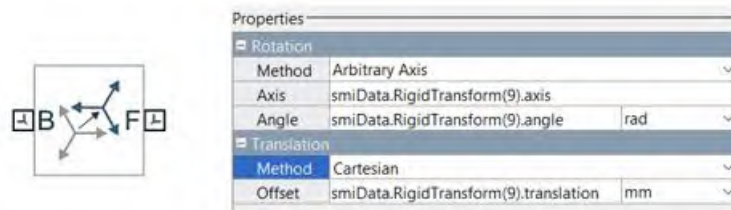


Figura 2.22. Bloque Rigid transform y parámetros de la librería Simscape Multibody
Fuente: elaboración propia

Joint. Representa una articulación rotacional o prismática según se indique (Simscape Multibody dispone de más tipos de articulaciones, pero estas son las más básicas que utilizan la mayoría de manipuladores). En Propiedades se pueden definir con expandibles características del estado inicial, el método de actuación, la mecánica interna de la articulación (posición de equilibrio, rigidez y amortiguamiento). Permite medir la posición, velocidad y aceleración de la articulación.



Figura 2.23. Bloque Joint y parámetros de la librería Simscape Multibody
Fuente: elaboración propia

Se muestra cómo Simscape Multibody importaría un manipulador de 4 grados de libertad. Se aprecian todos los elementos que se han explicado previamente:



Figura 2.24. Ejemplo de importación de modelo 4GDL a Simscape Multibody
Fuente: elaboración propia

El bloque que aparece entre cada articulación es el elemento que tiene estas dos articulaciones. Entrando dentro de este bloque se puede ver que está creado también por tipos de bloques explicados anteriormente.

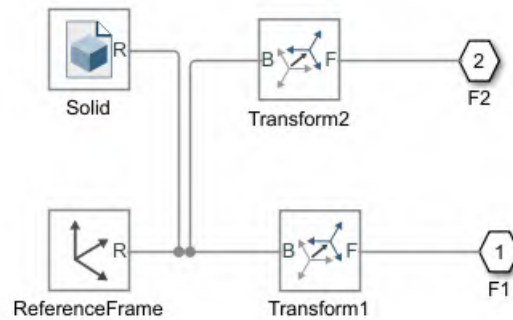


Figura 2.25. Bloque representante de una pieza en Simscape Multibody
Fuente: elaboración propia

En el bloque llamado Solid se recogen las características de la pieza importada desde el software de modelado. Tales como masa, centro de masa, inercia, color...

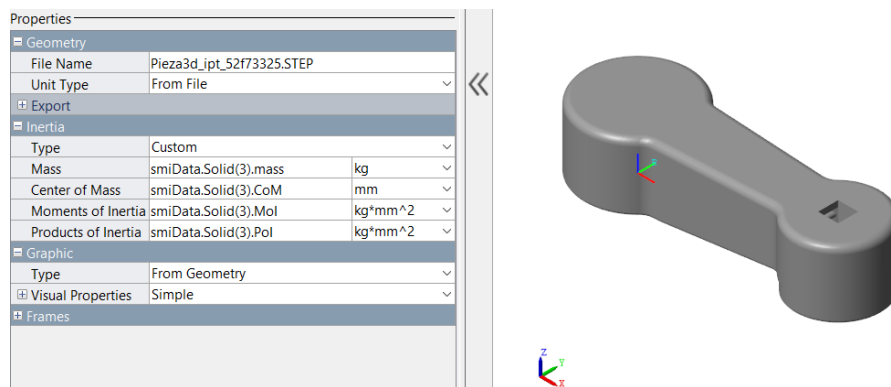


Figura 2.26. Características de la pieza importada en Simscape Multibody
Fuente: elaboración propia

Al ejecutar la simulación, el software calcula las posiciones del manipulador durante el periodo de tiempo indicado y con la herramienta 'Mechanics Explorer' el usuario puede visualizar la dinámica del sistema gracias a una animación 3D que genera.

Simscape electrical

Es posible integrar los sistemas eléctricos, hidráulicos, neumáticos y otros tipos de sistemas físicos en su modelo mediante los componentes de la familia de productos Simscape™. Se va

a explicar brevemente la toolbox que modela componentes de tipo eléctrico, se llama Simscape electrical y proporciona librerías de componentes eléctricos como motores, transistores, filtros... Ayuda a desarrollar sistemas de control y a probar el rendimiento del sistema. [28]

Simulink Control Design

Es una herramienta muy útil para determinar los parámetros de los reguladores PID, PD o PI que mejor se aplican al sistema de control del modelo. Puede ajustar automáticamente arquitecturas de control SISO y MIMO con su app llamada PID tuner app. Al iniciar la app, linealiza la planta y a continuación calcula los parámetros óptimos automáticamente. Facilita ajustar el regulador de manera cualitativa por medio de sliders atendiendo a la velocidad de respuesta y a la robustez del sistema. [29]

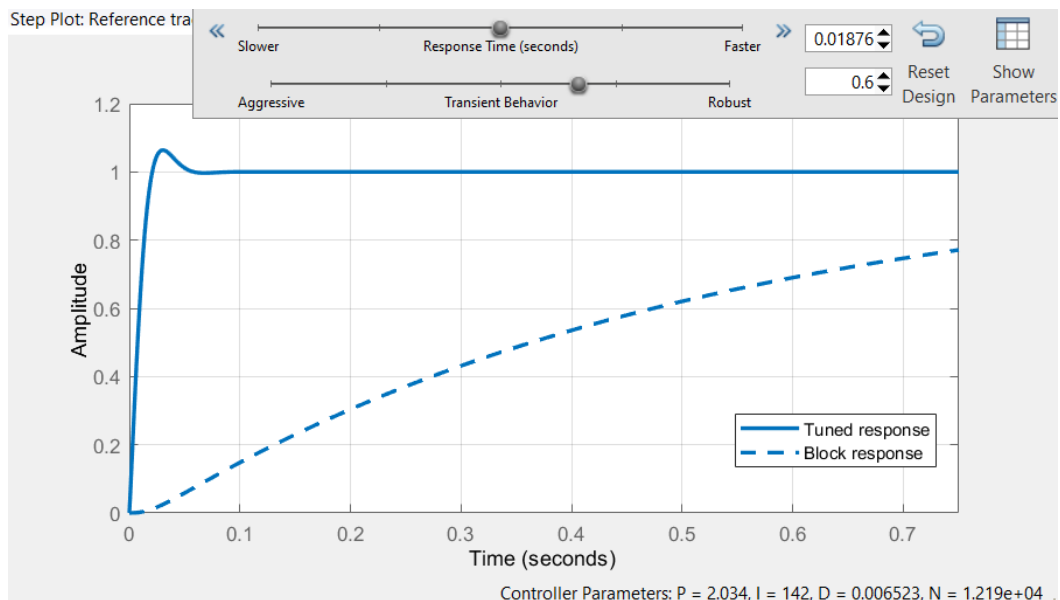


Figura 2.27. PID tuner App de Simulink Control Designer

Fuente: elaboración propia

Otras alternativas que ofrecen herramientas de simulación son los entornos virtuales Gazebo y ROS. Matlab también dispone de funciones que permiten simular el manipulador en estos entornos virtuales más parecidos a la realidad a los que se les pueden añadir obstáculos y objetos del mundo real.

3 DISEÑO Y SIMULACIÓN DEL MODELO SCARA

Una vez explicado el fundamento teórico general que hay detrás de los manipuladores se expondrá el desarrollo del robot SCARA que se ha diseñado.

Se explicarán las características técnicas que tiene el manipulador, aspectos sobre su diseño, las razones de su geometría, su cinemática y dinámica, la elección de actuadores y transmisiones por la que se ha optado. A su vez, se mostrará el razonamiento que ha habido detrás de la toma de elecciones entre diferentes opciones con las que se podía desarrollar cada fase del diseño.

Finalmente, por medio de una simulación se comprobará que el manipulador funcione de la manera deseada y pueda dibujar en el plano caracteres alfanuméricos y figuras aportadas en formato imagen.

3.1 DISEÑO

El robot diseñado es un robot SCARA clásico de 4 grados de libertad, está formado por 5 elementos principales que comprenden la base, 3 eslabones intermedios y el efector final en forma de pinza con un lápiz acoplado. La unión entre estos se realizará por medio de articulaciones rotacionales y prismática siguiendo la siguiente configuración:

Dispondrá de dos articulaciones rotacionales que ubicarán el extremo en las coordenadas deseadas en el plano, una articulación prismática cuya función es elevar la punta del rotulador o bolígrafo cuando acaba una letra o palabra y llegar al punto inicial de la siguiente, una articulación rotacional que permite el giro del efector final respecto del eje vertical.

Si bien la articulación prismática con un recorrido tan amplio o la última articulación rotacional para la tarea de escritura no son imprescindibles, el motivo de su incorporación es facilitar la versatilidad de este robot con la idea de incorporar un efector final modular que pueda sustituirse por otra herramienta que sí necesite de estas articulaciones como un taladro, o una pinza funcional que agarre, traslade y oriente objetos en el espacio.

En la siguiente Figura 3.1 se ve una primera imagen del robot diseñado en donde se aprecian sus eslabones, las articulaciones y el efector final.

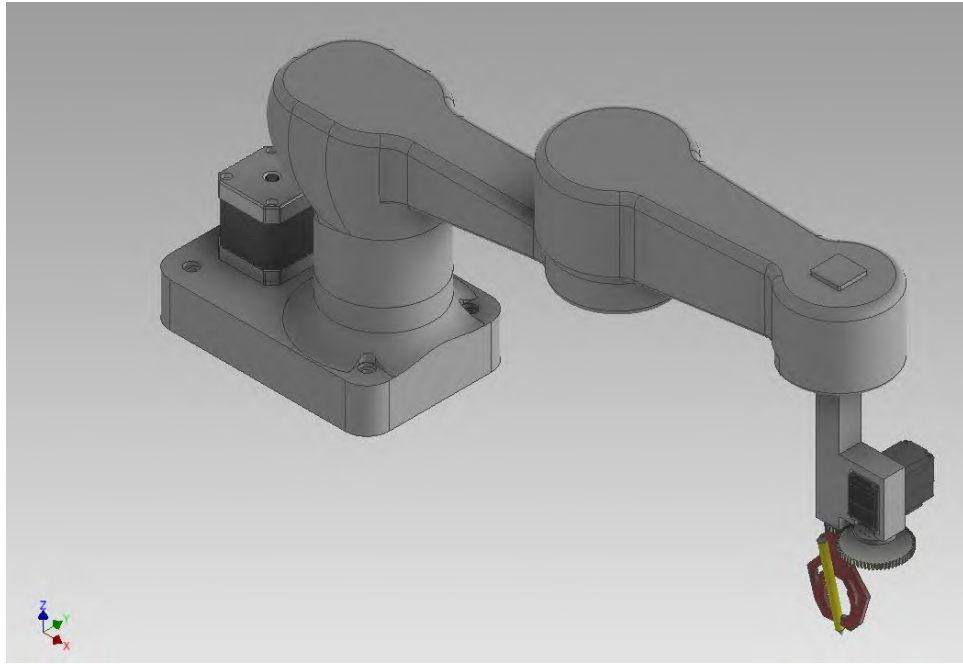


Figura 3.1. Vista en perspectiva del modelo SCARA diseñado
Fuente: elaboración propia

3.1.1 Espacio de trabajo

El espacio de trabajo por donde se podrá mover el manipulador es un toroide de sección rectangular como se muestra en la Figura 3.2. El radio externo del espacio de trabajo se calcula como la suma de las longitudes entre ejes de las articulaciones, el radio interno limita una zona de seguridad para que el manipulador no colisione con su propia base y la altura corresponde con el recorrido de la articulación prismática.

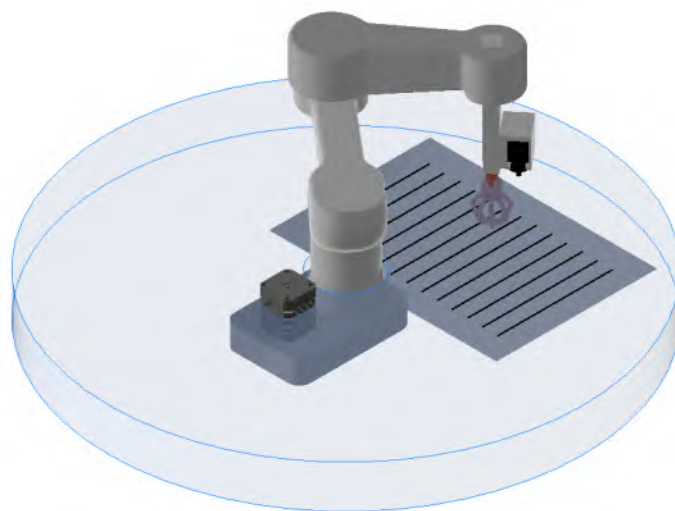


Figura 3.2 Manipulador, plano de tarea y espacio de trabajo
Fuente: elaboración propia

El plano donde escribirá los caracteres pertenece a este espacio de trabajo cilíndrico y se encuentra al mismo nivel que la base. Las dimensiones del manipulador se han calculado teniendo en cuenta que el plano objetivo tendrá las dimensiones de una hoja DIN A4 (210x297mm). Podrá alcanzar todas las coordenadas de la hoja.

La siguiente Figura 3.3 se realizó para hallar radio que debe abarcar para cumplir con las condiciones de espaciales de establecer un límite en la base de radio 40mm y un margen de 20 mm extra en la esquina más alejada para evitar que se den posiciones con singularidades. El radio óptimo calculado es de 310 mm y la esquina de la hoja se sitúa en las coordenadas ($x = 250$, $y = 148.5$) mm

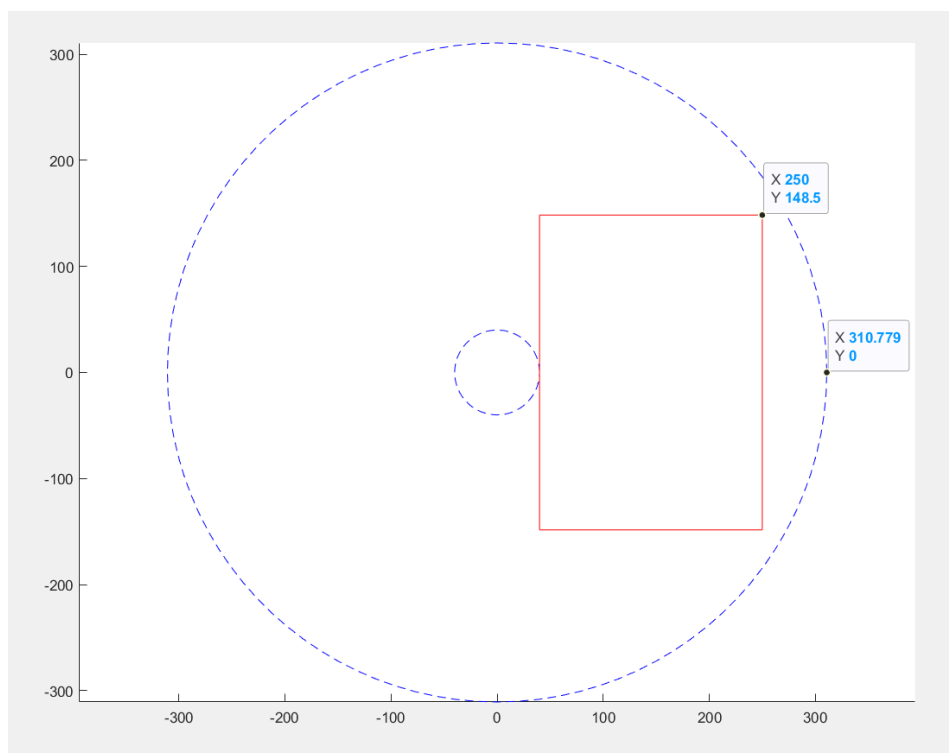


Figura 3.3. Espacio de trabajo óptimo para una hoja DIN A4

Fuente: elaboración propia con Matlab

3.1.2 Dimensiones y estructura del manipulador

Esta versión final del manipulador se ha diseñado queriendo realizar el manipulador con las dimensiones más reducidas posibles pero que pudiese abarcar de manera exitosa la totalidad de la superficie de una hoja de papel DIN A4.

Los dos primeros eslabones móviles son los encargados de posicionar el extremo del manipulador en las coordenadas planares correctas, las dimensiones de estos vienen condicionadas por la dimensión del plano de trabajo deseado. En este caso la distancia

máxima que deben abarcar es 310. Esta distancia entre ejes de articulaciones se reparte 160 mm para el primer eslabón y 150 mm para el segundo.

Las dimensiones máximas del manipulador son 465 mm de largo, 100 mm de ancho en la base y 210 mm de alto. Se han buscado unas medidas relativamente reducidas especialmente en altura para que el manipulador si en un futuro se desea desarrollar y crear físicamente no sea muy voluminoso en un laboratorio, aula o casa particular.

La forma de eslabones se ha diseñado siguiendo geometrías sencillas como rectángulos o cilindros por practicidad. A pesar de esto, no se vuelve un robot monótono, sino que por medio de radios de acuerdo y estrechamientos se consigue un diseño dinámico e incluso estético sin perder la funcionalidad.

En la siguiente Figura 3.4 se muestran las cotas generales del conjunto:

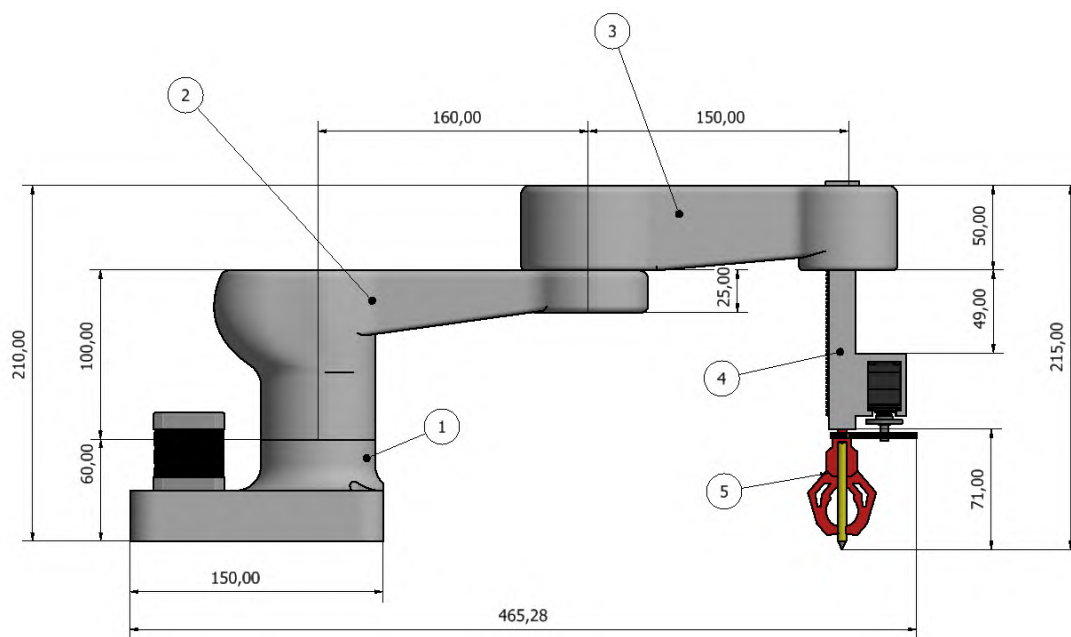


Figura 3.4. Cotas generales del manipulador

Fuente: elaboración propia

Los eslabones son huecos con un espesor de 5 mm dejando espacio para colocar en su interior los elementos que forman la transmisión y los motores. Este empaquetamiento reducido se ha podido realizar en los eslabones 2 y 3, en cambio el elemento 4 es una deslizadera estrecha y el motor se ha acoplado al exterior, así como la transmisión que orienta la pinza.

La geometría de estos eslabones tiene en cuenta la localización de los actuadores. Son elementos que aumentan el peso en zonas puntuales de la estructura. Se ha buscado una


localización lo más cercana al elemento fijo, la base. Si se encuentran alejados del eje de rotación del elemento repercuten negativamente en la dinámica del sistema; por lo que se han colocado lo más cercano a los ejes de rotación, reduciendo de esta manera las fuerzas inerciales generadas.

Todos los elementos de la estructura del manipulador están diseñados teniendo en cuenta si en algún momento se quiere llevar a la realidad este proyecto. Se pueden imprimir con impresora 3D en una cama caliente de 25x25mm puesto que ninguna pieza se supera estas medidas. La gran mayoría de las impresoras 3D del mercado tienen una superficie de trabajo de este tamaño, por lo que cualquier usuario lo podría imprimir si tiene a su disposición una.

En vista de su posible impresión material elegido para los eslabones es plástico PLA. La elección de este material en vez de otros existentes en el mercado es porque en comparación con plástico ABS (también usado en impresiones 3D) es que este último emite vapores nocivos cuando es calentado. Según el fabricante bq, el plástico PLA posee muy buenas características y las recoge de manera cualitativa:

- Rápido endurecimiento
- Mínima tensión térmica
- Mínima deformación
- Resistente a la acetona
- Es un producto biodegradable que se obtiene de azúcares derivados de vegetales.
- El precio de una bobina de 1kg en julio de 2021 es de 16,10 €

Las características técnicas en cuanto a resistencia a diferentes esfuerzos bq los recoge en la ficha técnica así:

	Módulo elástico en flexión: 3600 MPa (ISO 178)		
	Resistencia a la flexión: 108 MPa (ISO 178)		
	Dureza: 85 Sh D (ASTM D2240)		
		Prob. inyectadas	Prob. impresas ¹
			Prob. impresas ²
Resistencia a la rotura en tracción*	52 MPa	50 MPa	39 MPa
Deformación a rotura en tracción*	5%	9%	4%
Módulo elástico en tracción*	1320 MPa	1230 MPa	1120 MPa

* Estirado en dirección paralela a las capas
¹ Estirado en dirección perpendicular a las capas
² ISO 527

Figura 3.5. Plástico PLA de bq. Detalle de su ficha de datos

Fuente: <https://www.mastoner.com/images/pdfs/filamento-pla-bq-ficha-tecnica.pdf>

Los elementos no son macizos, sino que tienen un grosor de 5mm en la mayor parte de su geometría. En las zonas que se espera que puedan tener esfuerzos se ha aumentado según se ha considerado. El peso total del manipulador calculado por el software Autodesk Inventor es de aproximadamente 2.8 kg teniendo en cuenta el peso de los actuadores y de las transmisiones.

3.1.3 Características de los motores elegidos:

Los accionadores elegidos han sido 2 motores paso a paso Nema 17 (17HS19-2004S1) para las dos primeras articulaciones rotacionales y 2 servomotores MG996R para la articulación prismática y la rotacional que orienta el efector final.

Se ha optado por estos modelos por su reducido tamaño, cantidad de par aportado y ligereza en comparación con otros modelos existentes del mercado. Estos son los más utilizados en aplicaciones que no requieren gran esfuerzo o par motor, pero sí una gran precisión y velocidad. Características que los hace adecuados para el trabajo destinado.

En la siguiente Figura 3.6 se muestran los eslabones traslucidos para observar la localización de los actuadores y las transmisiones diseñadas para el movimiento del manipulador.

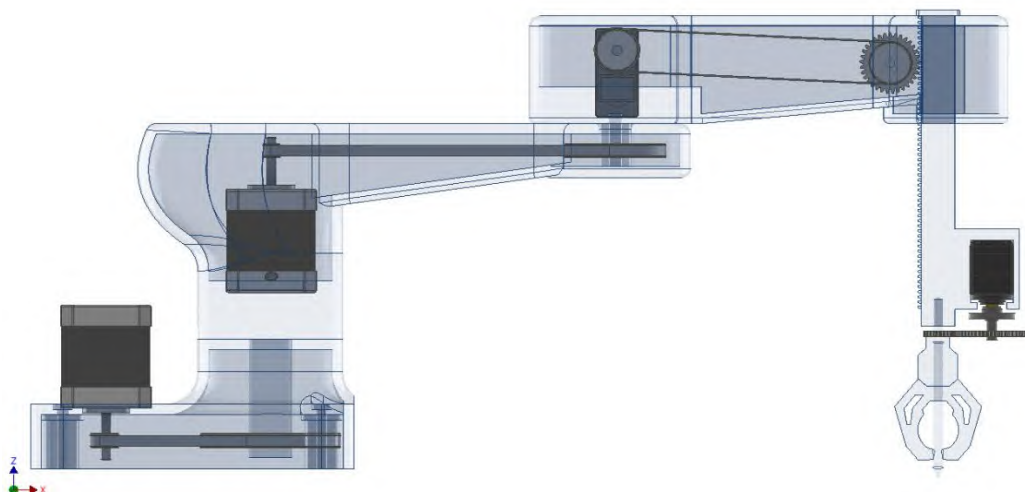


Figura 3.6. Ubicación de motores y transmisiones

Fuente: elaboración propia

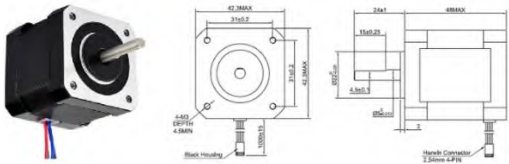
Los motores paso a paso NEMA 17 son muy precisos y realizan movimientos discretos de 1.8° . El giro efectivo del eslabón es más reducido aún; las relaciones de transmisión del motor 1 y motor 2 son 1:5 y 1:6 respectivamente. El giro discreto que se produce es de 0.36° y 0.3° en cada caso.

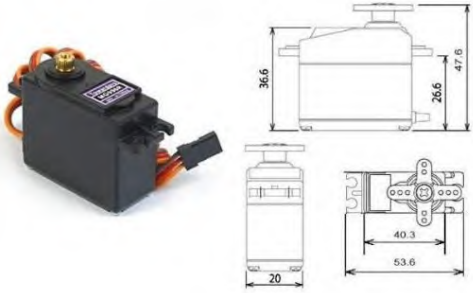
Estos motores son relativamente pesados en comparación con el conjunto (pesan 400g). Por esta razón sólo se han utilizado en las dos primeras articulaciones y se encuentran ubicados en posiciones que ayudan al equilibrio y/o no perjudican la dinámica del sistema. El motor 1 está fijo apoyado en la base; aunque se espera que la base esté atornillada a la superficie donde esté posado el manipulador este motor hace de contrapeso. El motor 2 se encuentra en el interior del primer eslabón móvil coincidiendo su eje con el eje de rotación, así cuando el eslabón rote el motor no produce desequilibrios.

Los servomotores MG996R son los encargados del movimiento del eslabón 4 y 5, deslizadera y pinza respectivamente. Tienen un peso más reducido, apenas alcanzan los 55 gramos. Sus cortas dimensiones (40.7 x 19.7 x 42.9 mm) permiten ubicarlos en los eslabones más alejados de la base de una manera que pueden quedar ocultos en el interior del eslabón 3 o acoplado en el eslabón-deslizadera 4. La principal característica de los servomotores es su velocidad y precisión. El mayor problema que se ha resuelto con este motor es que no realiza un giro continuo, sino que está limitado a 60° en cada dirección. Se ha resuelto este problema analizando el desplazamiento vertical de la deslizadera y el giro de la pinza y consiguiendo que estos realicen el recorrido completo gracias al sistema de transmisión.

En la siguiente tabla se recogen las características técnicas más relevantes de los motores elegidos.

Tabla 3.1 Características actuadores elegidos

Actuadores	Características
<p>Motor paso a paso Nema 17</p>  <p>Figura 3.7. Motor Nema 17</p> <p>Fuente: Hoja de datos Motor Nema 17</p>	<p>Peso: 0.4 kg</p> <p>Dimensiones: 42 x 42 x 48 mm</p> <p>Ángulo de paso: 1.8 °</p> <p>Par de retención: 0.59 Nm a 4.8V y 1.07Nm a 6V</p> <p>Velocidad operativa: 0.17s/60° (4.8V), 0.14s/60° (6V)</p> <p>Tensión de alimentación: 2.8V</p> <p>Corriente nominal/fase: 2A</p> <p>Resistencia de fase: 1.4 Ω</p> <p>Inductancia: 3mH</p>

<p style="text-align: center;">Servomotor MG996R</p>  <p style="text-align: center;">Figura 3.8. Motor MG996R</p> <p>Fuente: Hoja de datos Motor MG996R</p>	<p>Peso: 0.055 kg</p> <p>Dimensiones: 40.7 x 19.7 x 42.9 mm</p> <p>Capacidad de rotación: 120° aproximadamente (60° en cada dirección)</p> <p>Par de retención: 0.92 Nm a 4.8V y 1.07Nm a 6V</p> <p>Velocidad operativa: 0.17s/60° (4.8V), 0.14s/60° (6V)</p>
--	---

3.1.4 Transmisiones

Se necesitan transmisiones para que el par producido por los actuadores llegue a las articulaciones y el manipulador pueda moverse correctamente.

El sistema elegido en las dos primeras articulaciones ha sido el de poleas y correas porque el par necesario para mover la estructura no es muy elevado reduciendo la posibilidad de deslizamiento al mínimo, el movimiento con este sistema es más suave y continuo en comparación con un sistema de engranajes que la transmisión sucede por el choque entre dientes. Otro motivo que ha tenido importancia en la elección del sistema es que los motores se encuentran a una cierta distancia del eje de rotación al que están aplicados, para salvar esta separación el tren de engranajes necesario sería excesivamente largo y con gran cantidad de ruedas dentadas.

La relación de transmisión de la primera articulación es de 1:5, mientras que en la segunda articulación es de 1:6. Ambas son reductoras por lo que el par que producen los motores paso a paso se ve multiplicado por el coeficiente de transmisión. Mientras que la velocidad y el ángulo girado se reducen de manera proporcional dando como consecuencia una mayor precisión en el movimiento.

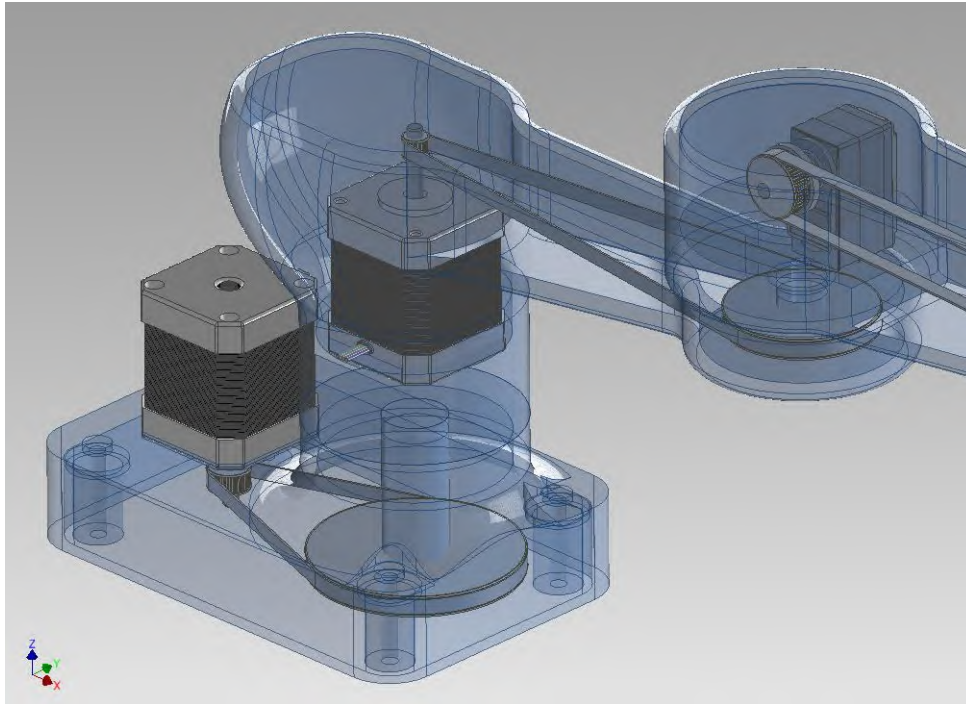


Figura 3.9. Detalle de actuadores y transmisiones de las dos primeras articulaciones
Fuente: elaboración propia

La articulación 3 es prismática, es decir realiza un movimiento lineal vertical. Se ha optado por un sistema de piñón-cremallera que permita llevarlo a cabo. El servomotor que acciona esta articulación se encuentra en el extremo opuesto del eslabón para una mayor estabilidad del manipulador. La distancia entre el servomotor y la rueda dentada se salva incorporando un sistema de polea y correa con relación de transmisión 1:1.

En el apartado anterior donde se exponen los motores utilizados se menciona que los servomotores elegidos tienen un ángulo de giro total limitado a 120° . Se desea que a pesar de este ángulo limitado el recorrido de la deslizadera sea completo (49 mm). Así, las características de la rueda dentada son:



Figura 3.10. Rueda dentada transmisión 3
Fuente: elaboración propia

Paso: 2.51 mm
Módulo: 0.8 mm
Diámetro primitivo: 26.4mm
Diámetro exterior: 28mm
Diámetro interior: 24.53
Número de dientes: 33
Espesor del diente: 5mm

La cremallera tiene unas dimensiones de dientes acordes a las de la rueda dentada para que se produzca correctamente el acople.

En la articulación 4 se ha optado por un sistema de engranajes multiplicadores de relación 3:1 que aumenta el giro del servomotor para cubrir la totalidad de grados de la circunferencia incrementándose el rango de 120° a 360°. De este modo el efector final puede orientarse en cualquier ángulo.

Las características de este par de engranajes son:

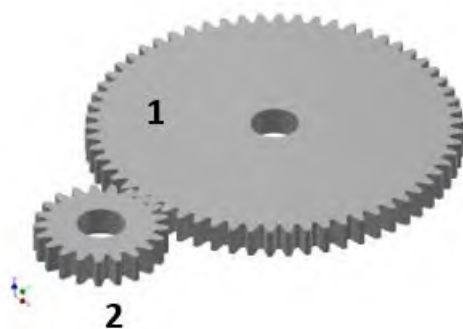


Figura 3.11. Engranajes transmisión 4
Fuente: elaboración propia

Módulo: 0.6mm

Relación de transmisión: 3:1

Distancia entre centros: 25mm

Diámetro primitivo 1: 38.7mm

Diámetro primitivo 2: 12.6mm

Z1: 63 dientes

Z2: 21dientes

Espesor del diente: 3mm

En la siguiente figura 3.12 se muestran los actuadores y transmisiones que se encargan del movimiento de los eslabones 4 y 5:

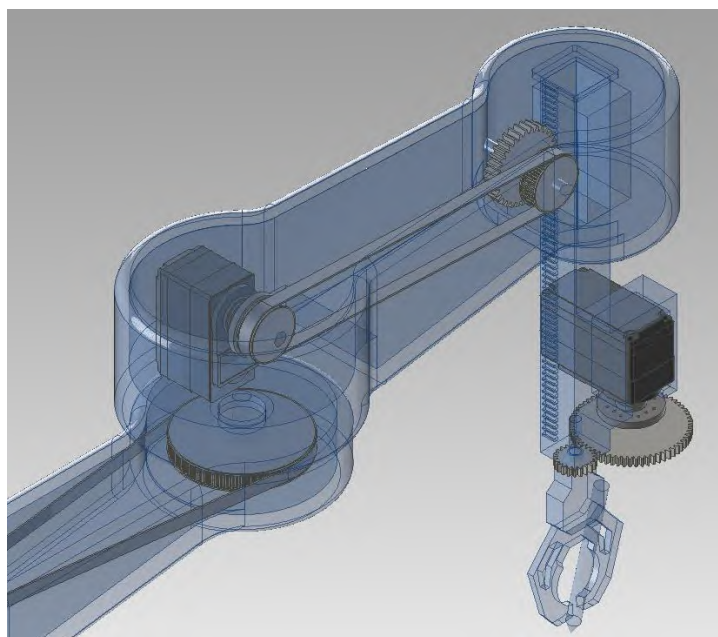


Figura 3.12. Detalle de actuadores y transmisiones de las dos últimas articulaciones
Fuente: elaboración propia

3.1.5 Cinemática del manipulador

La cinemática directa se calculará a partir de los parámetros DH que definen la configuración del manipulador. Ubicando los ejes coordenados de cada articulación de la manera en que se muestran en la Figura 3.13 se obtienen los siguientes parámetros registrados en la tabla. El software RoboAnalyzer facilita la visualización de estos parámetros creando robots con estructuras geométricas sencillas (cilindros y primas).

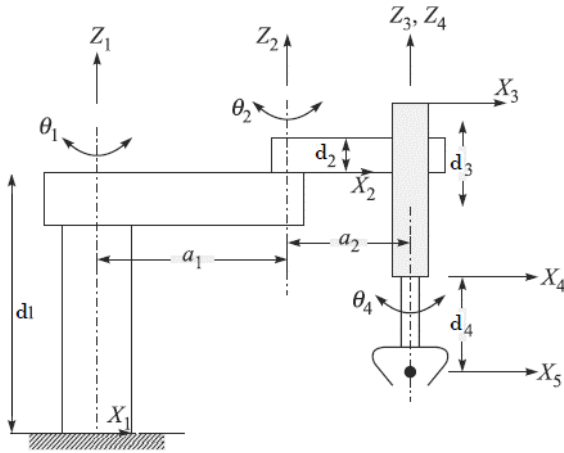


Figura 3.13. Esquema robot SCARA

Fuente: [6] pág 160

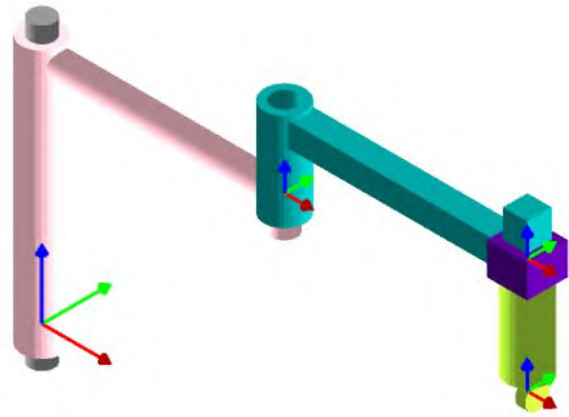


Figura 3.14. Visualización robot SCARA en RoboAnalyzer

Fuente: elaboración propia

Articulación	Tipo	α_i	a_i	θ_i	d_i
1	Rotacional	0	0.16	θ_1	0.16
2	Rotacional	0	0.15	θ_2	0.04
3	Prismática	0	0	0	d_3
4	Rotacional	0	0	θ_4	-0.071

A partir de los parámetros DH podemos obtener fácilmente las matrices de transformación homogénea entre ejes de articulaciones y la cinemática directa, recordando la expresión 51 y aplicándolo en cada articulación:

$${}^{n-1}T_n(\theta_n) = \begin{bmatrix} \cos\theta_n & -\sin\theta_n\cos\alpha_n & \sin\theta_n\sin\alpha_n & a_n\cos\theta_n \\ \sin\theta_n & \cos\theta_n\cos\alpha_n & -\cos\theta_n\sin\alpha_n & a_n\sin\theta_n \\ 0 & \sin\alpha_n & \cos\alpha_n & d_n \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad 51$$

Matrices de transformación homogénea para cada articulación:

$$\begin{aligned}
 T_{01}(\theta_1) &= \begin{bmatrix} \cos\theta_1 & -\sin\theta_1 & 0 & a_1\cos\theta_1 \\ \sin\theta_1 & \cos\theta_1 & 0 & a_1\sin\theta_1 \\ 0 & 0 & 1 & d_1 \\ 0 & 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} \cos\theta_1 & -\sin\theta_1 & 0 & 0.16\cos\theta_1 \\ \sin\theta_1 & \cos\theta_1 & 0 & 0.16\sin\theta_1 \\ 0 & 0 & 1 & 0.16 \\ 0 & 0 & 0 & 1 \end{bmatrix} \\
 T_{12}(\theta_2) &= \begin{bmatrix} \cos\theta_2 & -\sin\theta_2 & 0 & a_2\cos\theta_2 \\ \sin\theta_2 & \cos\theta_2 & 0 & a_2\sin\theta_2 \\ 0 & 0 & 1 & d_2 \\ 0 & 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} \cos\theta_2 & -\sin\theta_2 & 0 & 0.15\cos\theta_2 \\ \sin\theta_2 & \cos\theta_2 & 0 & 0.15\sin\theta_2 \\ 0 & 0 & 1 & 0.04 \\ 0 & 0 & 0 & 1 \end{bmatrix} \\
 T_{23}(d_3) &= \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & -d_3 \\ 0 & 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & -d_3 \\ 0 & 0 & 0 & 1 \end{bmatrix} \\
 T_{34}(\theta_4) &= \begin{bmatrix} \cos\theta_4 & -\sin\theta_4 & 0 & 0 \\ \sin\theta_4 & \cos\theta_4 & 0 & 0 \\ 0 & 0 & 1 & d_4 \\ 0 & 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} \cos\theta_4 & -\sin\theta_4 & 0 & 0 \\ \sin\theta_4 & \cos\theta_4 & 0 & 0 \\ 0 & 0 & 1 & -0.071 \\ 0 & 0 & 0 & 1 \end{bmatrix}
 \end{aligned}
 \tag{52}$$

La matriz de transformación completa es el producto de matrices $T_{04} = T_{01}T_{12}T_{23}T_{34}$ quedando:

$$\begin{aligned}
 T_{04} &= \begin{bmatrix} \cos(\theta_1 + \theta_2 + \theta_4) & -\sin(\theta_1 + \theta_2 + \theta_4) & 0 & a_2 \cdot \cos(\theta_1 + \theta_2) + a_1 \cdot \cos(\theta_1) \\ \sin(\theta_1 + \theta_2 + \theta_4) & \cos(\theta_1 + \theta_2 + \theta_4) & 0 & a_2 \cdot \sin(\theta_1 + \theta_2) + a_1 \cdot \sin(\theta_1) \\ 0 & 0 & 1 & d_1 + d_2 - d_3 + d_4 \\ 0 & 0 & 0 & 1 \end{bmatrix} \\
 &= \begin{bmatrix} \cos(\theta_1 + \theta_2 + \theta_4) & -\sin(\theta_1 + \theta_2 + \theta_4) & 0 & 0.15 \cdot \cos(\theta_1 + \theta_2) + 0.16 \cdot \cos(\theta_1) \\ \sin(\theta_1 + \theta_2 + \theta_4) & \cos(\theta_1 + \theta_2 + \theta_4) & 0 & 0.15 \cdot \sin(\theta_1 + \theta_2) + 0.16 \cdot \sin(\theta_1) \\ 0 & 0 & 1 & 0.145 + 0.04 - d_3 - 0.07 \\ 0 & 0 & 0 & 1 \end{bmatrix}
 \end{aligned}
 \tag{53}$$

Comparando la matriz de transformación con la matriz genérica de transformación se obtienen las posiciones y rotaciones de la trama final.

$$T_{23}(d_3) = \begin{bmatrix} n_x & o_x & a_x & p_x \\ n_y & o_y & a_y & p_y \\ n_z & o_z & a_z & p_z \\ 0 & 0 & 0 & 1 \end{bmatrix}
 \tag{54}$$

Las posiciones del extremo del efector final corresponden con p_x , p_y y p_z . Los términos que orientan están recogidos en la submatriz de rotación 3x3 superior izquierda.

$$\begin{aligned}
 p_x &= 0.15 \cdot \cos(\theta_1 + \theta_2) + 0.16 \cdot \cos(\theta_1) \\
 p_y &= 0.15 \cdot \sin(\theta_1 + \theta_2) + 0.16 \cdot \sin(\theta_1) \\
 p_z &= 0.145 + 0.04 - d_3 - 0.07
 \end{aligned}
 \tag{55}$$

$$Rot = \begin{bmatrix} \cos(\theta_1 + \theta_2 + \theta_4) & -\sin(\theta_1 + \theta_2 + \theta_4) & 0 \\ \sin(\theta_1 + \theta_2 + \theta_4) & \cos(\theta_1 + \theta_2 + \theta_4) & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad 56$$

El ángulo efectivo que girará el efector será: $\varphi = \theta_1 + \theta_2 - \theta_4$

Para resolver la cinemática inversa del manipulador se ha decidido usar el método geométrico por ser un robot común tipo SCARA. Se combina un movimiento planar de dos eslabones con articulaciones rotacionales y un movimiento vertical permitido por la articulación prismática. La función de la articulación rotacional del efector final se limita a realizar una transformación rotacional pero no traslacional del extremo del manipulador.

Las ecuaciones obtenidas se basan en la figura 3.15. El objetivo es hallar una relación entre el vector de estados q cuyas variables son $\theta_1, \theta_2, d_3, \theta_4$ con la posición del efector final P_x, P_y, P_z conocida.

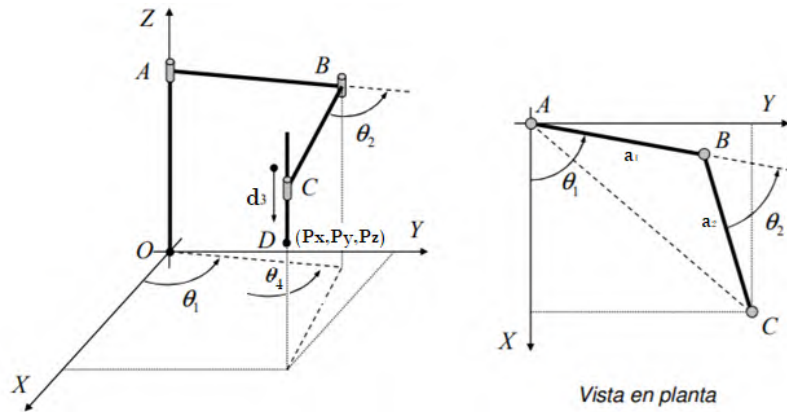


Figura 3.15. Identificación de variables robot SCARA

Fuente: [30]

$$\theta_2 = \arccos\left(\frac{x^2 + y^2 - a_1^2 - a_2^2}{2 \cdot a_1 \cdot a_2}\right) = \arccos\left(\frac{x^2 + y^2 - 0.16^2 - 0.15^2}{2 \cdot 0.16 \cdot 0.15}\right)$$

$$\theta_1 = \arctan\left(\frac{y}{x}\right) + \arctan\left(\frac{a_2 \cdot \sin(\theta_2)}{a_1 + a_2 \cdot \cos(\theta_2)}\right) = \arctan\left(\frac{y}{x}\right) + \arctan\left(\frac{0.15 \cdot \sin(\theta_2)}{0.16 + 0.15 \cdot \cos(\theta_2)}\right)$$

$$d_3 = 0.29 + 0.08 - 0.142 - p_z$$

$$\theta_4 = \theta_1 + \theta_2 - \varphi$$

57

La cinemática diferencial se calculará por medio de la matriz jacobiana. Esta representa la derivada de la cinemática directa respecto del vector de posición de las articulaciones.



La relación entre las velocidades articulares y las del extremo final: $\dot{X} = J(q) \dot{q}$

La matriz jacobiana:

$$J(q) = \frac{\partial T_{04}(q)}{\partial q} = \begin{bmatrix} \frac{\partial x}{\partial \theta_1} & \frac{\partial x}{\partial \theta_2} & \frac{\partial x}{\partial d_3} & \frac{\partial x}{\partial \theta_4} \\ \frac{\partial y}{\partial \theta_1} & \frac{\partial y}{\partial \theta_2} & \frac{\partial y}{\partial d_3} & \frac{\partial y}{\partial \theta_4} \\ \frac{\partial z}{\partial \theta_1} & \frac{\partial z}{\partial \theta_2} & \frac{\partial z}{\partial d_3} & \frac{\partial z}{\partial \theta_4} \\ \frac{\partial \varphi}{\partial \theta_1} & \frac{\partial \varphi}{\partial \theta_2} & \frac{\partial \varphi}{\partial d_3} & \frac{\partial \varphi}{\partial \theta_4} \end{bmatrix} = \begin{bmatrix} -a_2\theta_1\sin(\theta_1 + \theta_2) - a_1\theta_1\sin(\theta_1) & -a_2\theta_2\sin(\theta_1 + \theta_2) & 0 & 0 \\ a_2\theta_1\cos(\theta_1 + \theta_2) + a_1\theta_1\cos(\theta_1) & a_2\theta_2\cos(\theta_1 + \theta_2) & 0 & 0 \\ 0 & 0 & -1 & 0 \\ 1 & 1 & 0 & -1 \end{bmatrix} \tag{58}$$

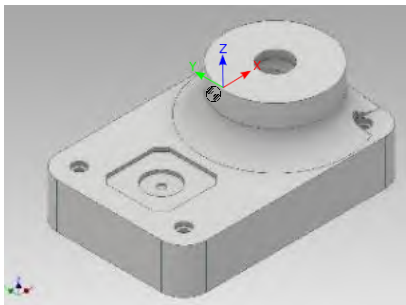
3.1.6 Características dinámicas de los eslabones

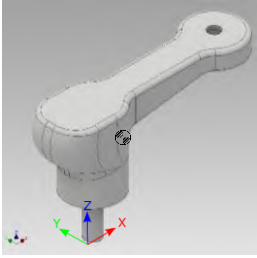
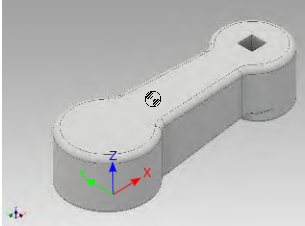

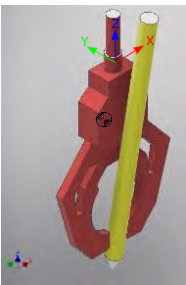
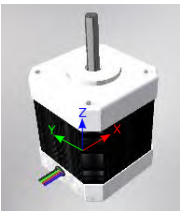
En la siguiente tabla 3.2 se recogen las características dinámicas de los cuerpos que componen el sistema. Se han incluido tanto la estructura mecánica, es decir, los eslabones como los elementos motrices y de transmisión.

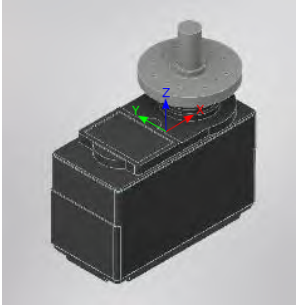
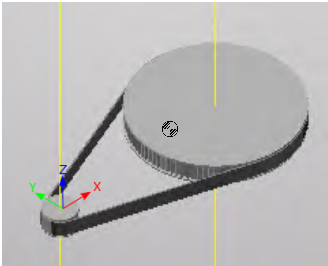
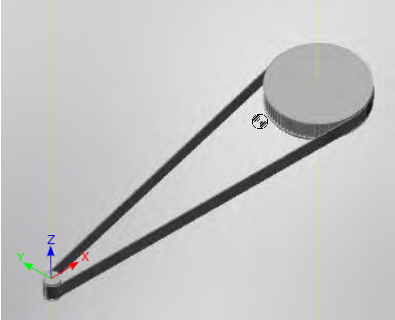
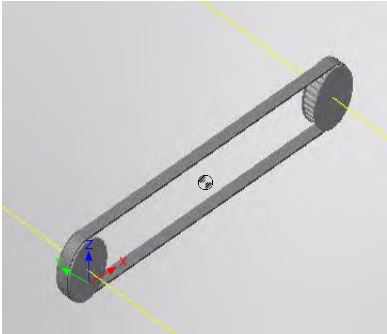
El centro de gravedad está calculado respecto del origen que se ha elegido al crear el elemento. En los eslabones el origen elegido es el mismo que se utilizó al realizar el análisis cinemático con los parámetros DH. En los elementos de transmisión y motores el origen se toma donde confluyen los planos de simetría del objeto. En las figuras se muestra con el símbolo  y el centro de gravedad .

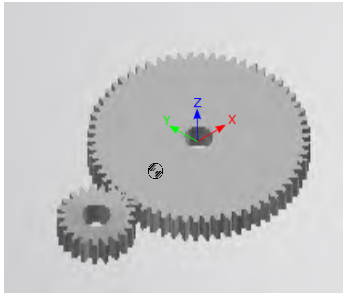
Los cálculos de masa y matriz de inercia los ha calculado teniendo en cuenta que el material usado en los eslabones es PLA con densidad 1.24g/cm³ y en el resto de los elementos se ha dejado el material que indica la librería de Autodesk Inventor por defecto.

Tabla 3.2. Parámetros dinámicos de los componentes del modelo
Fuente: elaboración propia.

Elemento	Características dinámicas
<p>Base:</p>  <p>Figura 3.16. Base del robot en Autodesk Inventor</p>	<p>Masa: 0.206 kg</p> <p>cdg: [-30 0 26.1]mm</p> <p>Elemento fijo</p>

<p>Pieza1:</p>  <p>Figura 3.17. Pieza 1 del robot en Autodesk Inventor</p>	<p>Masa: 0.397 kg</p> <p>cdg: [34.3 0 98.8] mm</p> $I_{CDG} = \begin{bmatrix} 0.73 & 0 & 0 \\ 0 & 2.37 & 0 \\ -0.45 & 0 & 1.92 \end{bmatrix} \cdot 10^{-3} \text{ kgm}^2$
<p>Pieza2:</p>  <p>Figura 3.18. Pieza 2 del robot en Autodesk Inventor</p>	<p>Masa: 0.116 kg</p> <p>cdg: [56.1 0 21.9] mm</p> $I_{CDG} = \begin{bmatrix} 0.22 & 0 & 0 \\ 0 & 1.60 & 0 \\ -0.06 & 0 & 1.63 \end{bmatrix} \cdot 10^{-3} \text{ kgm}^2$
<p>Pieza3:</p>  <p>Figura 3.19. Pieza 3 del robot en Autodesk Inventor</p>	<p>Masa: 0.054 kg</p> <p>cdg: [3.3 0 -53.0] mm</p> $I_{CDG} = \begin{bmatrix} 0.10 & 0 & 0 \\ 0 & 0.10 & 0 \\ 0 & 0 & 0 \end{bmatrix} \cdot 10^{-3} \text{ kgm}^2$
<p>Efactor final:</p>  <p>Figura 3.20. Eslabón final del robot en Autodesk Inventor</p>	<p>Masa: 0.12 kg</p> <p>cdg: [0 -1.8 -25] mm</p> $I_{CDG} = \begin{bmatrix} 0.028 & 0 & 0 \\ 0 & 0.031 & 0 \\ 0 & 0 & 0.046 \end{bmatrix} \cdot 10^{-3} \text{ kgm}^2$
<p>Motor Nema 17</p>  <p>Figura 3.21. Motor Nema 17 en Autodesk Inventor</p>	<p>Masa: 0.400kg</p> <p>cdg: [0 0 26] mm</p> $I_{CDG} = \begin{bmatrix} 0.134 & 0 & 0 \\ 0 & 0.134 & 0 \\ 0 & 0 & 0.100 \end{bmatrix} \cdot 10^{-3} \text{ kgm}^2$

<p>Servomotor MG</p>  <p>Figura 3.22. Motor MG996R en Autodesk Inventor</p>	<p>Masa: 0.055 kg</p> <p>cdg: [12.2 0 21.1] mm</p> $I_{CDG} = \begin{bmatrix} 0.01 & 0 & 0 \\ 0 & 0.008 & 0 \\ 0 & 0 & 0.005 \end{bmatrix} \cdot 10^{-3} \text{ kgm}^2$
<p>Transmisión 1</p>  <p>Figura 3.23. Transmisión 1 del robot en Autodesk Inventor</p>	<p>Masa: 0.17 kg</p> <p>cdg: [52.3 0 0] mm</p> $I_{CDG} = \begin{bmatrix} 0.042 & 0 & 0 \\ 0 & 0.076 & 0 \\ 0 & 0 & 0.117 \end{bmatrix} \cdot 10^{-3} \text{ kgm}^2$
<p>Transmisión 2</p>  <p>Figura 3.24. Transmisión 2 del robot en Autodesk Inventor</p>	<p>Masa: 0.086 kg</p> <p>cdg: [98.4 0 0] mm</p> $I_{CDG} = \begin{bmatrix} 0.011 & 0 & 0 \\ 0 & 0.065 & 0 \\ 0 & 0 & 0.076 \end{bmatrix} \cdot 10^{-3} \text{ kgm}^2$
<p>Transmisión 3</p>  <p>Figura 3.25. Transmisión 3 del robot en Autodesk Inventor</p>	<p>Masa: 0.028 kg</p> <p>cdg: [64.2 0 0] mm</p> $I_{CDG} = \begin{bmatrix} 0.001 & 0 & 0 \\ 0 & 0.113 & 0 \\ 0.005 & 0 & 0.112 \end{bmatrix} \cdot 10^{-3} \text{ kgm}^2$

<p style="text-align: center;">Transmisión 4</p>  <p style="text-align: center;">Figura 3.26. Transmisión 4 del robot en Autodesk Inventor</p>	<p style="text-align: center;">Masa: 0.021 kg</p> <p style="text-align: center;">cdg: [-4 0 0] mm</p> $I_{CDG} = \begin{bmatrix} 0.002 & 0 & 0 \\ 0 & 0.005 & 0 \\ 0 & 0 & 0.003 \end{bmatrix} \cdot 10^{-3} \text{ kgm}^2$
--	---

3.1.7 Efector final

El diseño del efector final se ha optado por no darle funcionalidad. Aunque se haya pintado de colores diferentes la pinza y el lápiz, ambos forman parte del mismo elemento rígido. Es decir, la pinza no tiene mecanismo accionador que la permita abrirse o cerrarse; el papel que juega en el diseño es el de tener en cuenta el peso del efector final y que su extremo sea la posición objetivo.

El motivo por el que no se ha diseñado un pinza funcional es que en el momento de la simulación se debería tener en cuenta el contacto entre varios elementos, definir la superficie de contacto, la fuerza necesaria para agarrar el lápiz, el engranaje que debería para realizar el movimiento de apertura y cierre, el posible desplazamiento que pueda existir en la zona de agarre de la pinza... Son muchas variables para tener en cuenta en la simulación dinámica y se piensa que se escapa del objetivo de este TFG. En la simulación se dará por aceptable que el extremo de este elemento, es decir la punta del lápiz, siga la trayectoria indicada.

Existen multitud de proyectos de diseño de pinzas, el mecanismo de actuación generalmente es un servomotor acoplado a una rueda dentada incompleta que engrana con otra rueda dentada consiguiendo que giren en sentidos opuestos y ejerciendo un par de compresión simétrico. Es un mecanismo sencillo pero eficaz. En la siguiente Figura 3.27 se muestra la pinza metálica tomada como referencia para realizar el diseño propio y el diseño realizado junto con el mecanismo de transmisión que la orienta.

Diseño creado en Autodesk Inventor

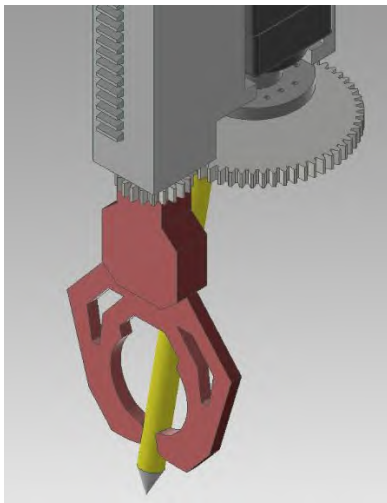


Figura 3.27. Detalle Efector final en Autodesk Inventor

Fuente: elaboración propia

Pinza metálica



Figura 3.28. Referencia efector final

Fuente: [Pinza](#)

3.2 SIMULACIÓN

En este apartado se explicará cómo se ha llevado a cabo la simulación. Se mostrará paso a paso el trabajo realizado desde el diseño del modelo desde Autodesk Inventor, la exportación hacia el entorno de Simulink y cómo se ha elaborado los scripts de creación de trayectorias, el modelo cinemático, dinámico y su control para su simulación haciendo uso de múltiples librerías y recursos que ofrece la versión R2021a de Matlab para estudiantes de la Universidad de Cantabria.

3.2.1 Generación de trayectorias/puntos de control

Curve fitting using polygonal approximation for a robotic writing task

El objetivo inicial del manipulador era poder representar gráficamente una palabra o una secuencia de caracteres alfanuméricos que son introducidos por teclado. A esta primera idea se le añadió que a partir de una imagen el robot pueda dibujar el contenido que aparezca en esta.

El problema general que se debe resolver es la generación de una trayectoria continua que pueda seguir el extremo del manipulador y que el resultado final se adecúe a lo esperado; es decir si se trata de un texto que este sea lo más legible posible o si se trata de una imagen que esta sea lo más fiel a la imagen de referencia. [31]

Esta trayectoria debe quedar definida por los puntos de control, estos son coordenadas por donde debe pasar obligatoriamente el extremo del manipulador. Al control del robot se le manda la secuencia ordenada de puntos de control. En la siguiente Figura 3.29 se muestra la secuencia de puntos que resulta al escribir “ab1a”. Es una trayectoria continua desde la primera ‘a’ hasta la última ‘a’. El manipulador se eleva al pasar de un caracter al siguiente, se pueden ver puntos negros más elevados.

Cada caracter está definido por una serie de coordenadas (en la figura 3.29 aparecen de color rojo). Estos puntos han sido introducidos uno por uno, cuando el programa requiere un caracter este obtiene la secuencia de puntos correspondiente. Por ejemplo, la letra ‘a’ queda definida por las siguientes coordenadas:

```
a--> [5  0;4  0;2  1;0  5;3  9;5  10;8  9;8  5;6.5  2;5  0;6.5  2;8  5;  8
2;10  0];
```

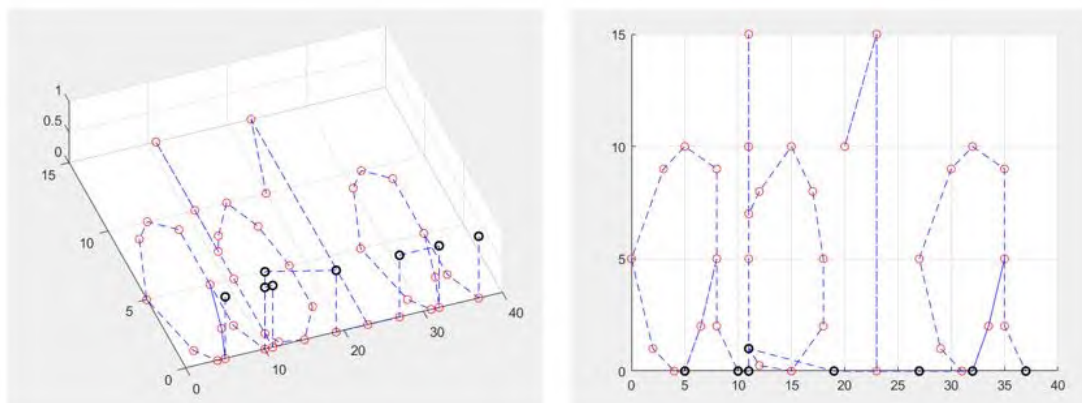


Figura 3.29. Puntos de control de caracteres alfanuméricos

Fuente: elaboración propia

La forma que resulta es angulosa, para suavizarla se han aplicado métodos de interpolación que se explicarán más adelante.

A partir de esta idea básica surge la pregunta de si se podrá realizar una obtención de puntos similar introduciendo como referencia una imagen. Es decir, aportar una imagen y obtener una secuencia de puntos de control que corresponda con la forma del objeto que contiene.

Esta idea se ha podido llevar a cabo y se ha probado con éxito en imágenes en las que el contenido tiene un contorno bien definido. La imagen se procesa con una serie de filtros de la librería de Matlab Image Processing Toolbox. La página oficial de Matlab describe de esta manera la librería:

Image Processing Toolbox™ proporciona un conjunto completo de algoritmos estándar de referencia y apps de flujo de trabajo para el procesamiento, el análisis y la visualización de imágenes, así como para el desarrollo de algoritmos. Puede llevar a cabo segmentación de imágenes, mejora de imágenes, reducción de ruido, transformaciones geométricas, registro de imágenes y procesamiento de imágenes 3D.

Los algoritmos usados son sencillos pero eficaces cuando en la imagen de referencia se puede detectar fácilmente la forma. Los pasos que se han seguido son:

1. Importar la imagen y recoger todos los datos de color de píxeles en una matriz. Esta será la matriz original de donde parte.
2. Convertir la imagen a escala de grises, así el programa detecta mejor los bordes.
3. Binarizar la imagen con un filtro threshold. Los píxeles con valores mayores a 140 los considera negros y el resto blancos. Así se consigue un contorno definido.
4. Escalar la imagen. El teléfono móvil con el que se han realizado las fotografías captura imágenes de 3472x3472 píxeles. Obtener un contorno de una imagen así implica que la secuencia de puntos final es demasiado grande afectando significativamente al tiempo de cálculo empleado. Una mayor densidad de píxeles en la imagen no significa un mejor resultado si luego se va a interpolar.
5. Se aplica un filtro que detecta los bordes. Image Processing Toolbox™ incorpora una función llamada `edge` que detecta los bordes de las figuras que aparecen en una imagen. Se ha especificado el método Sobel puesto que es el más sencillo e intuitivo; este a partir de una imagen binarizada detecta los píxeles donde el gradiente de intensidad es máximo y los marca como bordes.

```
edge (Imagen_binarizada, 'Sobel')
```

6. Se aplica una función que simplifica el trazo creado. Se impone la condición de que cada píxel negro solo puede estar en contacto con otros 2 píxeles negros; así se define un trazo simplificado de anchura 1px.

7. Se crea la secuencia ordenada de puntos que el manipulador tendrá como trayectoria a seguir. La separación entre puntos es de 1px, por lo que la cantidad de puntos resulta excesiva.
8. Se reduce la cantidad de puntos con el fin de reducir el tiempo y gasto computacional que se necesitará en el cálculo de la cinemática inversa.
9. Escalar y centrar acorde con el plano de trabajo.

Paso 1. Importar la imagen



Figura 3.30. Paso 1 Imagen a puntos de control

Paso 2. Aplicar escala de grises



Figura 3.31. Paso 2 Imagen a puntos de control

Paso 3. Binarizar la imagen

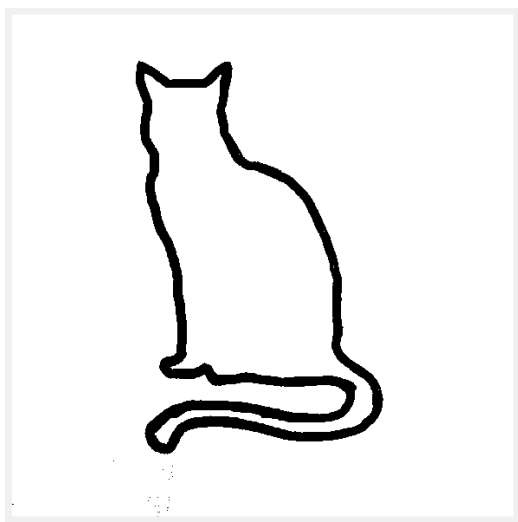


Figura 3.32. Paso 3 Imagen a puntos de control

Paso 4. Escalar la imagen

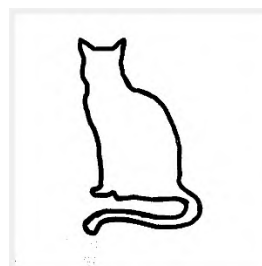
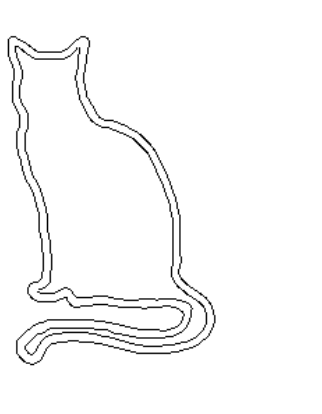


Figura 3.33. Paso 3 Imagen a puntos de control

Paso 5. Aplicar filtro detector de bordes



Paso 6. Simplificar el trazo

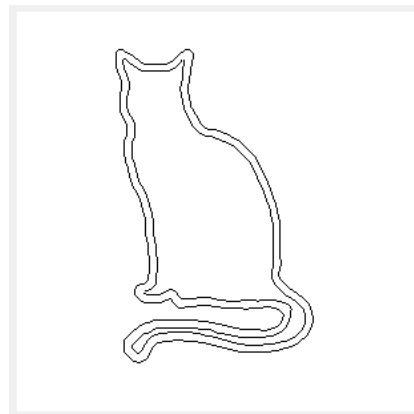


Figura 3.34. Paso 5 Imagen a puntos de control

Figura 3.35. Paso 6 Imagen a puntos de control

Paso 7: Crear la trayectoria

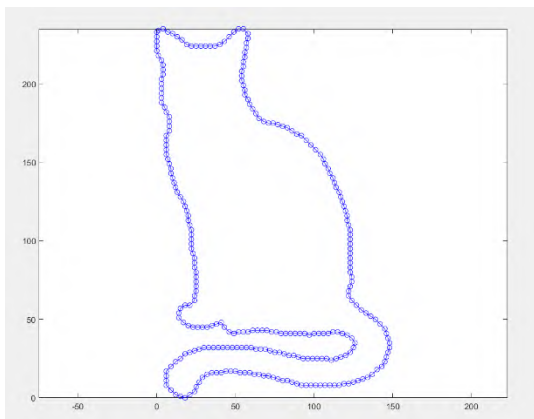


Figura 3.36. Paso 7 Imagen a puntos de control

Paso 8. Reducir puntos de la trayectoria

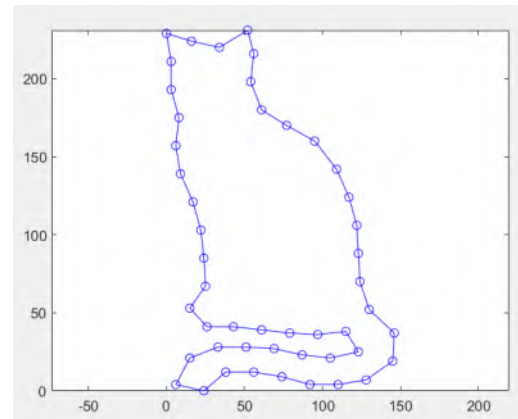


Figura 3.37. Paso 8 Imagen a puntos de control

Paso 9. Escalar y centrar en el origen (ejemplo origen = [0,0,0])

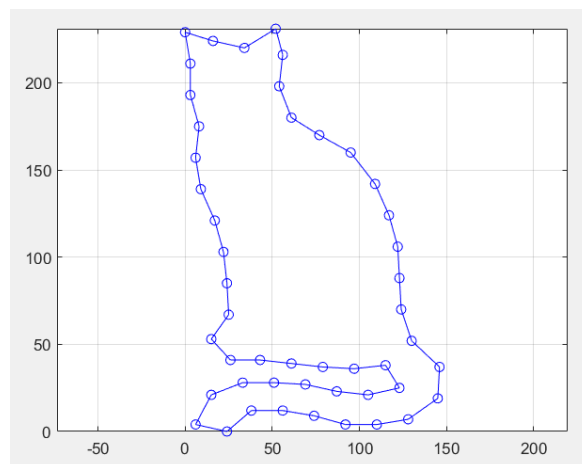


Figura 3.38. Paso 9 Imagen a puntos de control

3.2.2 Simulación cinemática

Realizar una simulación cinemática tiene como objetivo conocer las posiciones, velocidades y aceleraciones de las articulaciones cuando el manipulador realiza una trayectoria dada. Así mismo se comprueba de manera visual el resultado obtenido usando diferentes técnicas de creación de trayectorias, el robot es capaz de moverse de manera continua, el efector final sigue los puntos indicados correctamente, no se alcancen puntos de singularidad y la cantidad de puntos de control correcta encontrando un equilibrio entre cantidad de puntos y calidad de la trayectoria.

La simulación cinemática se llevará a cabo utilizando la librería Robotics System Toolbox™. Esta contiene gran cantidad de herramientas que facilitan el análisis de los robots.

El código desarrollado para realizar la simulación toma como referencia un trabajo realizado por el equipo de Matlab en el que simulaba y comparaba las diferentes maneras de crear trayectorias en un manipulador de 7 grados de libertad. [32] [20]

La toolbox contiene múltiples funciones y herramientas que han sido usadas para realizar el análisis cinemático numérico, para generar trayectorias utilizando los métodos más extendidos en la actualidad, para visualizar el movimiento que realiza el manipulador al seguir la trayectoria indicada y los resultados cinemáticos de los elementos.

Se han realizado diferentes simulaciones trabajando en el espacio articular, en el espacio cartesiano y utilizando varios métodos de interpolación con el fin de encontrar aquel que diese buen resultado en términos que la trayectoria sea suave, alcance los puntos correctamente, las velocidades y aceleraciones articulares no sean muy pronunciadas y a su vez no requiriese un gasto computacional excesivo.

Trabajando en el espacio articular se determinan las posiciones de las articulaciones por medio de cinemática inversa para cada punto de control y se interpolan las posiciones articulares obtenidas para conocer el movimiento entre los puntos de control. Las características cinemáticas de velocidad y aceleración se calcularán con una derivación numérica.

En Matlab, la herramienta Robotics System Toolbox proporciona funciones que realizan el cálculo de la cinemática inversa de manera general y optimizada para un manipulador dado. A continuación, se muestran las diferentes etapas que tiene el código es realmente sencillo y

no se ha necesitado introducir ecuaciones del movimiento, sino que definiendo el manipulador con sus parámetros DH es suficiente.

1. **Creación del manipulador.** Se debe generar un objeto denominado 'RigidBodyTree' al que se le añaden 'cuerpos', que son cada eslabón que forma el manipulador. El manipulador se puede definir de manera sencilla a partir de sus parámetros DH, se indican las articulaciones que enlazan los eslabones y se le puede añadir una malla de los elementos en formato .stl para que al visualizar la simulación sea más fiel a la realidad. Existe la posibilidad de indicar parámetros másicos para hacer un análisis dinámico.

Se muestra un fragmento de código de cómo se crean este tipo de objetos:

```
body1 = rigidBody('body1'); % Creación del cuerpo 1
jnt1 = rigidBodyJoint('jnt1','revolute'); %Creación de la articulación
jnt1.HomePosition = 0; %COnfiguración de la posición inicial de la
articulación
setFixedTransform(jnt1,dhparams(1,:),'dh'); % Define el lugar de la
articulación a partir de los parámetros DH
body1.Joint = jnt1; %Se añade la articulación al cuerpo
addVisual(body1,"Mesh",'Elementos/Pieza2d.stl')%Se añade la malla visual
addBody(robot,body1,'body0') % Se conecta al cuerpo anterior (base)
```

2. **Cálculo de la cinemática inversa.** Con la función 'inverseKinematics' se inicializa el robot que queramos analizar y podrá calcular la cinemática inversa del elemento que le indiquemos, en este caso el efecto final 'endeffector' proporcionando información de la matriz de posición 'tgtPosicion', Pesos y un punto de partida inicial 'ikInitGuess'.

Se tienen las coordenadas de los puntos de control en la variable `PuntosControl`, pero para resolver la cinemática inversa la función requiere que se pase estos datos en formatos matriz de posición. Esta conversión la realiza la función `trvec2tform`. Las posiciones articulares que debe tener el manipulador para que el efector final esté en la posición indicada por 'tgtPosicion'.

Los pesos indican la importancia que tiene cada grado de libertad en el efector final, en el robot SCARA se deberán resolver con prioridad las traslaciones en los ejes x,y,z

.La rotación respecto de z no tiene tanta prioridad y el resto de rotaciones deberían ser nulas en un sistema ideal.

Al ser la resolución un método iterativo este necesita un punto de partida en cada iteración. El primer ciclo toma como posición inicial `homeConfiguration`, para el resto de intervalos toma como referencia la configuración calculada anterior.

Introduciendo correctamente las variables en la función que resuelve la cinemática inversa, esta devuelve en la variable 'config' las posiciones articulares calculadas y en la variable 'info' las iteraciones necesarias que ha necesitado para converger.

Se muestra el fragmento de código donde se realiza la cinemática inversa del manipulador a partir de los puntos de control.

```
ik = inverseKinematics('RigidBodyTree',robot); % Se inicia el solver
Pesos = [1 1 1 0 0 0.2]; % Se asignan los pesos a cada grado de
                        libertad

ikInitGuess=homeConfiguration(robot); % Inicia a resolver desde la
                        posición de home
for idx = 1:numPuntosControl % Bucle para calcular las posiciones en
                        cada punto de control
    tgtPosicion = trvec2tform(PuntosControl(:,idx)');
    [config,info] = ik('endeffector',tgtPosicion,Pesos,ikInitGuess);
    ikInitGuess=config;
    PosJuntas(:,idx) = config';
end
```

- 3. Interpolación de los puntos de control.** Una vez obtenidas las posiciones articulares en cada punto de control (`PosJuntas`) se necesita calcular los puntos intermedios de la trayectoria. Matlab ofrece diferentes métodos de interpolación como son lineal, cúbica, quintica e incluso curvas suaves como bsplines.

La función de interpolación necesitará que se indique una matriz donde se recogen todas las posiciones de cada articulación para cada punto de control, un vector de tiempos de los puntos de control y un vector de intervalos de tiempos intermedios. Si la interpolación no es lineal, sino que es cúbica, quíntica o con bsplines y se desea definir la velocidad y/o aceleración articular en cada punto se deberán proporcionar

en una matriz similar a la de posiciones. Una manera de hacer que el movimiento sea continuo es derivando la matriz `PosJuntas`, también se puede especificar uno concreto adecuado a las características pertinentes.

Se extrae un fragmento de código para mostrar que es relativamente sencillo realizar las interpolaciones con esta toolbox. Se resuelve gracias a la función `cubicpolytraj`. Los datos que genera esta función son posiciones, velocidades y aceleraciones (q , q_d , q_{dd}) en todos los puntos de la trayectoria.

```
n_inter=3; %Número de intervalos intermedios entre puntos de control
tpts = 0:(numPuntosControl-1); %Vector de tiempos en puntos de control
tvec=0:1/n_inter:(numPuntosControl-1); %Vector de tiempos intermedios
VelJuntas=horzcat(zeros(4,1),diff(PosJuntas,1,2)); %Velocidades

[q,qd,qdd] = cubicpolytraj(PosJuntas,tpts,tvec, , ...
    'VelocityBoundaryCondition',VelJuntas); %Función interpolación, se
obtienen las posiciones, velocidades y aceleraciones articulares de cada
articulación
```

4. **Animación.** Una vez obtenidas todas las variables cinemáticas que determinan el movimiento del manipulador a lo largo de la trayectoria a seguir solo queda representar la animación del manipulador. Se realiza mediante un bucle en el que frame a frame se muestra el manipulador en cada posición calculada y se va dibujando el recorrido que realizado.

En la siguiente Figura 3.39 realizada durante la simulación del modelo cinemático se ve que el robot está dibujando la silueta de un gato. El extremo del efector final va siguiendo la trayectoria marcada por los puntos de control. En la captura no se aprecia, pero entre cada punto de control marcado en rojo hay 2 puntos intermedios de color azul calculados mediante interpolación cúbica.

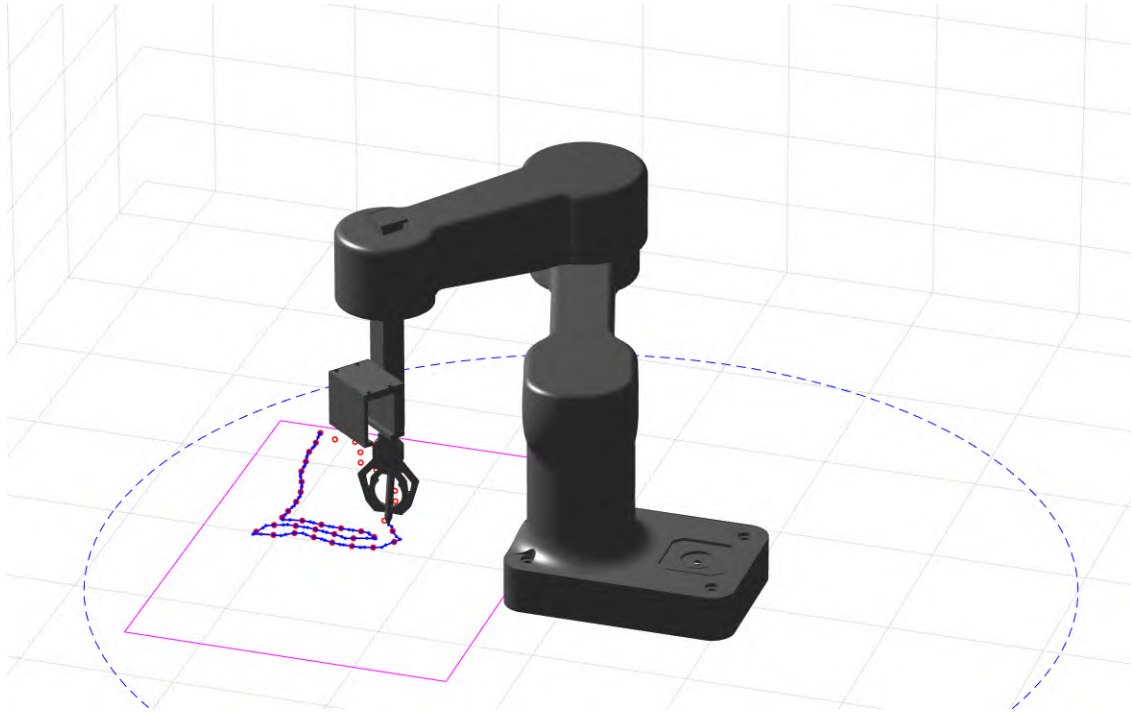


Figura 3.39. Animación SCARA dibujante en Robotics System Toolbox

Fuente: elaboración propia

Trabajar en el espacio cartesiano es muy parecido a trabajar en el espacio articular. La diferencia es que se realiza el cálculo de la cinemática inversa para todos los puntos de la trayectoria y no solo en los puntos de control.

Por ejemplo, si la forma de una imagen se ha definido con 20 puntos de control y al interpolar se consideran 4 intervalos entre cada punto de control para que quede una forma suave, el cálculo de la cinemática inversa se debe realizar para 100 puntos. Mientras que trabajando en el espacio articular se calcula la cinemática para 20 puntos de control y el resto de los puntos se obtienen por interpolación. Una operación de interpolación de tipo polinomio consume mucho menos gasto computacional que el cálculo de la cinemática inversa que involucra resolver sistemas de ecuaciones no lineales.

La interpolación se realiza en el espacio cartesiano, es decir se suaviza la trayectoria espacial final que seguirá el efector final. Esto trae consigo indirectamente que las posiciones, velocidades y aceleraciones en el espacio articular sean algo más suaves en comparación si no se realizase ningún tipo de interpolación. Los pasos a seguir trabajando en el espacio articular son:

1. Creación del manipulador.
2. Interpolación de los puntos de control.
3. Cálculo de la cinemática inversa.
4. Animación.

3.3 SIMULACIÓN DINÁMICA COMPLETA

Realizar una simulación dinámica completa es uno de los objetivos de este TFG. Se pretende que esta simulación integre desde los sistemas motrices accionados eléctricamente, las transmisiones y la estructura mecánica del manipulador. Simulink, el entorno de simulación de Matlab dispone de una herramienta muy potente llamada Simscape.

Existe un complemento que se puede instalar en Matlab que conecta el software de modelado Autodesk Inventor con el entorno de Simulink y Simscape Multibody pudiendo exportar el ensamblaje manteniendo las características dinámicas de los elementos y sus restricciones de movimiento impuestas en las articulaciones y en los motores.

El modelo que se desea realizar en Simulink incorporará los elementos necesarios para realizar un control simulado del robot SCARA diseñado para que siga una trayectoria de puntos que representan una palabra o imagen. Los sistemas para llevarlo a cabo son:

Unas entradas como las posiciones articulares variables que toma de referencia, el modelo del manipulador que estará formado por su estructura mecánica y por sus cuatro actuadores, un sistema de control específico para cada actuador que reduzca el error en el seguimiento de la trayectoria, un sistema de sensores que arrojen los resultados obtenidos permitan conectarse con un lazo de realimentación. El esquema del sistema completo se puede ver en la figura 3.40.

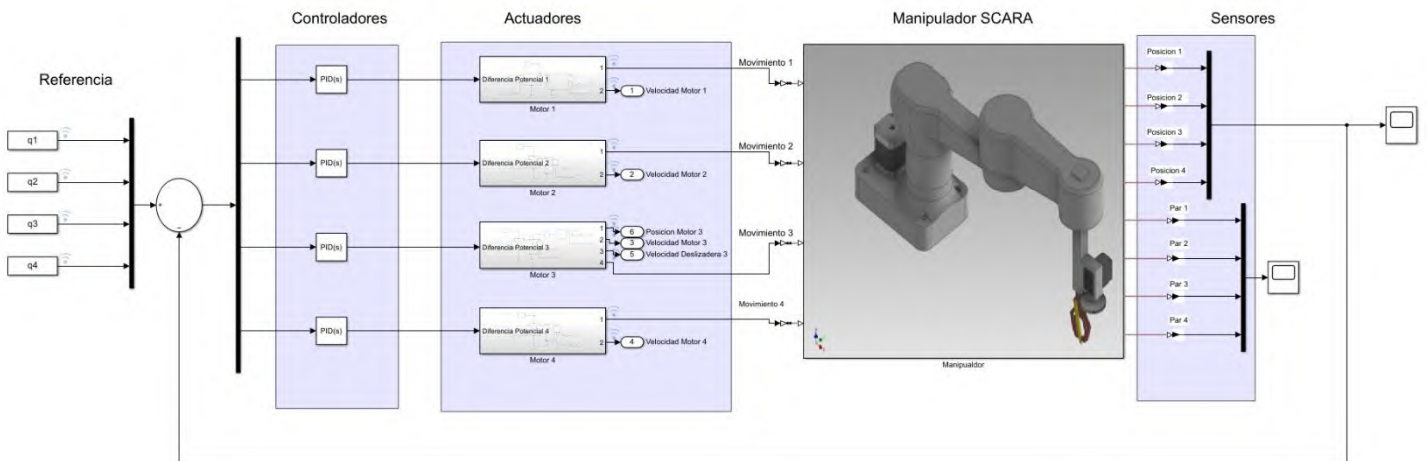


Figura 3.40. Esquema del modelo dinámico completo en Simulink

Fuentes: elaboración propia

A continuación, se explicarán los subsistemas tanto del modelo mecánico como los actuadores y el sistema de transmisión. Los bloques de la librería de Simscape representan componentes físicos reales; es decir, el software realiza los cálculos basándose en las ecuaciones diferenciales que se expusieron en el apartado teórico. Así el usuario no las tiene que introducir, sino que configura los bloques a partir de parámetros. Es una manera relativamente sencilla de reproducir el modelo; sin embargo, hay que tener una buena noción de qué unidades y/o tipo de variables permite cada bloque.

Se ha aproximado el funcionamiento de los servomotores y los motores paso a paso que se eligieron al diseñar el modelo como si fuesen motores CC convencionales con el objeto de simplificar la simulación. El modelo que se ha creado está formado por una serie de bloques con significado físico tanto eléctrico como mecánico. En la figura se puede ver el subsistema eléctrico en color azul y el subsistema mecánico en color verde. [33]

El bloque principal que transforma la energía eléctrica en mecánica se denomina DC Motor, se ha configurado con los parámetros especificados en las fichas de características de los motores Nema 17. Los parámetros del motor MG996R se toman los medidos por medio de una obtención experimental en un estudio previo. [34]

El rango de valores obtenidos por este estudio es acorde con lo esperado. Si resultasen ser erróneos la simulación creada en nuestro estudio se debe tomar como una idealización y aproximación. Se debe tener en cuenta que partimos de una aproximación al considerar los actuadores como motores de corriente continua convencionales.

Los datos introducidos están recogidos en la siguiente Tabla 3.3.

Tabla 3.3 Parámetros característicos de los actuadores

	Motor paso a paso Nema 17	Servomotor MG996R
Constante de par motor (K_t)	0.35 Nm/A	0.0879 Nm/A
Constante electromotriz (K_e)	0.04 V/rad s	0.0876 V/rad s
Resistencia de armadura (R_a)	1.4 Ω	8.3 Ω
Inductancia de armadura (L_a)	3 mH	1.5 mH
Momento de inercia (J_m)	82 g cm ²	80 g cm ²
Constante de fricción (B)	0.0141 N m s /rad	0.0143 N m s /rad

El motor deberá tener una fuente de energía, el bloque que la simula es Controlled Voltage Source. La diferencial de potencial que suministra dependerá de la entrada 1 Diferencia Potencial 1. Otro elemento imprescindible es la referencia, o tierra.

El sistema mecánico está formado por la relación de transmisión, inercia del conjunto, sensores de posición y velocidad. En el bloque de relación de transmisión se modela la reducción de velocidad que producen las poleas y engranajes que se diseñaron. La inercia modela cómo afecta la dinámica del resto de eslabones. El sensor de velocidad y posición indica cuántos radianes se ha desplazado la articulación y su velocidad. El sensor detecta la diferencia de posición y velocidad angular respecto de una referencia mecánica que se debe indicar con el bloque Mechanical Rotational Reference.

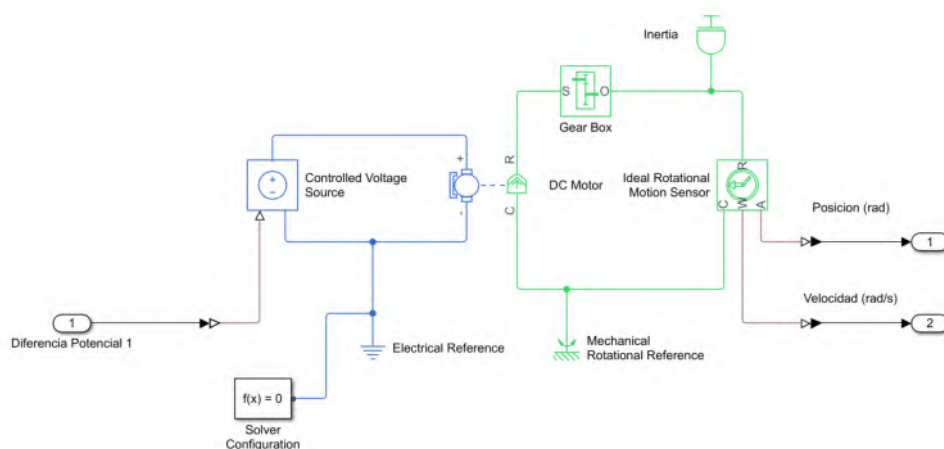


Figura 3.41. Modelo en Simulink de motor de corriente continua

Cada actuador estará definido por unos parámetros y relación de transmisión propios. El diagrama de bloques es similar en todas las articulaciones salvo en la articulación prismática. Esta necesitará de un bloque que transforme el movimiento rotacional suministrado por el motor a un movimiento traslacional que tendrá la deslizadera. El bloque que realiza esta transformación se llama Wheel and Axle. Además, si se desea medir la posición y velocidad de las deslizadera se necesita añadir un sensor llamado Ideal Translational Motion Sensor. El modelo de la tercera articulación quedará así:

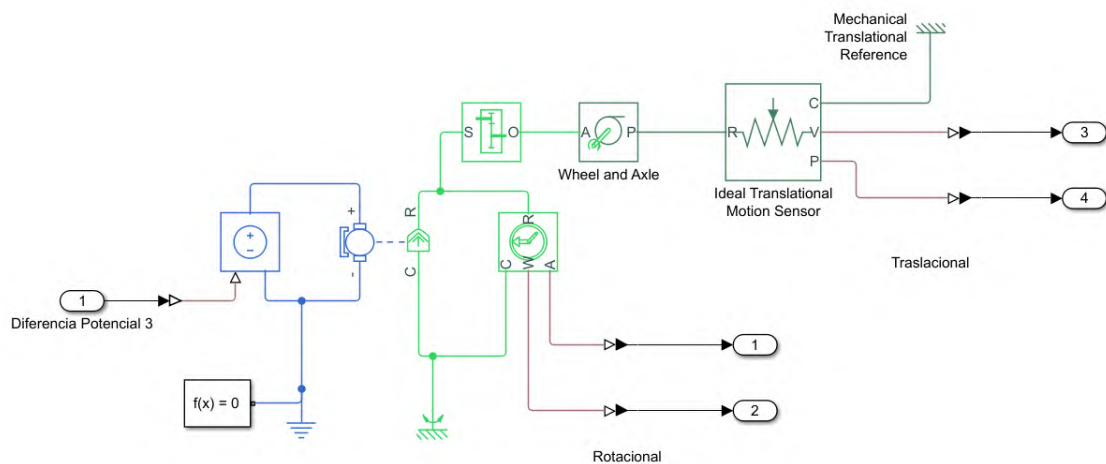


Figura 3.42 Modelo en Simulink de motor corriente continua acoplado a articulación prismática

Fuente: elaboración propia

La salida del sistema motriz son las posiciones angulares o lineales de las articulaciones rotacionales o prismáticas respectivamente que se moverán como causa de la diferencia de potencial suministrada. El sistema mecánico simula tomando estos desplazamientos como entrada para cada articulación.

Cada motor está controlado de manera individual por un regulador PID. Estos reguladores ayudan a tener un tiempo de respuesta más corto y previenen de inestabilidades en el sistema. Matlab dispone de una herramienta para obtener los parámetros K_p , K_i , K_d y N que definen un regulador PID.

La herramienta se llama PID Tuner app, linealiza el sistema entorno a un tiempo indicado y propone diferentes combinaciones de parámetros a la vez que muestra la respuesta del sistema. Permite que el usuario de manera cualitativa por medio de dos sliders varíe la respuesta en términos de velocidad y robustez.

En la siguiente tabla 3.4 se muestran los parámetros que se obtuvieron después de utilizar la aplicación:

Tabla 3.4 Parámetros reguladores PID

	Kp	Ki	Kd	N
1	0.657	59.05	0	100
2	0.132	30.02	0	100
3	0.548	47.33	0	100
4	0.698	53.87	-0.003	204.06

Se ve que la constante derivativa en la mayoría de los casos es nula, es decir, el sistema sólo necesita de las constantes Kp y Ki siendo un regulador de tipo PI el más apropiado.

En las siguiente figuras se muestra con línea de puntos cómo era la respuesta antes de incorporar el regulador PID y con trazo continuo la respuesta obtenida al modificar los parámetros Kp, Ki, Kd, N.

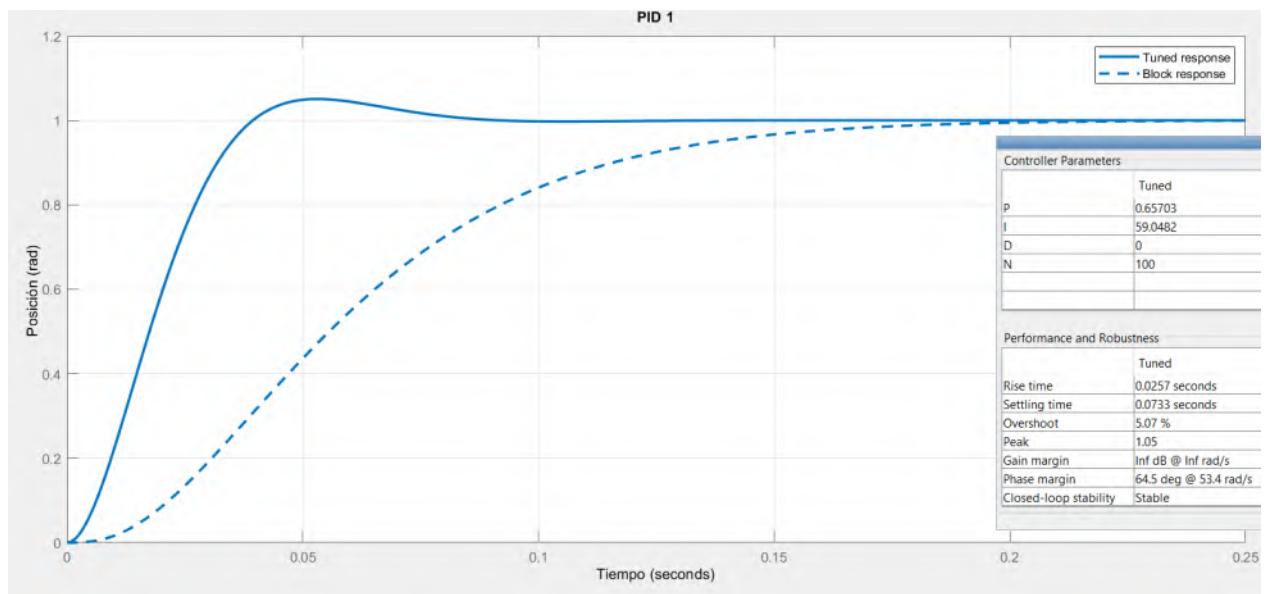


Figura 3.43. Regular PID articulación 1

Fuente: elaboración propia

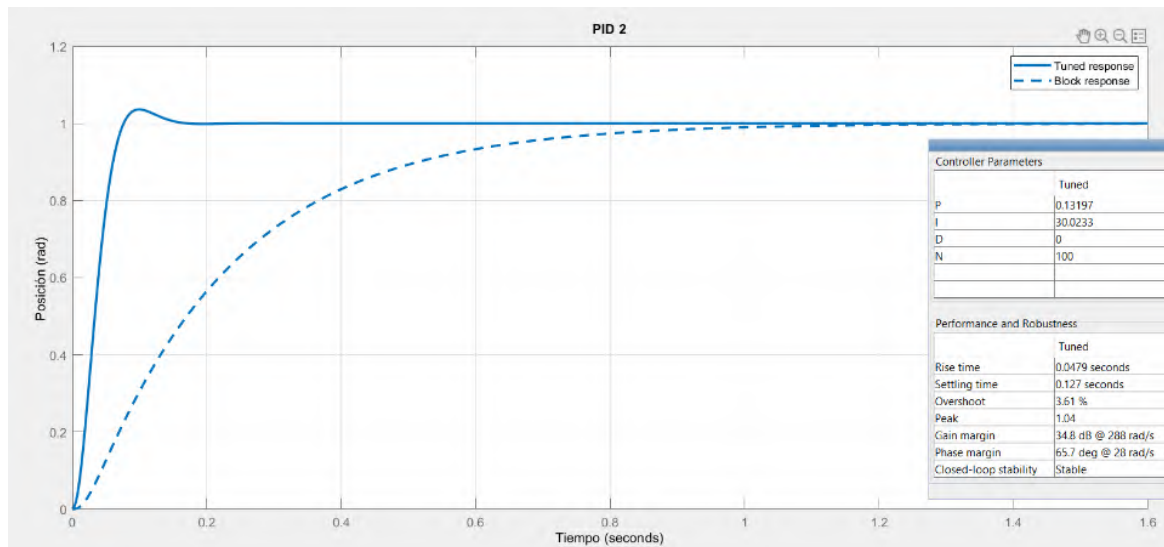


Figura 3.44 Regulador PID articulación 2

Fuente: elaboración propia

La respuesta se da ante una entrada escalón genérica, aunque la articulación prismática por límites geométricos no puede desplazarse 1m este regulador se diseña en un contexto teórico.

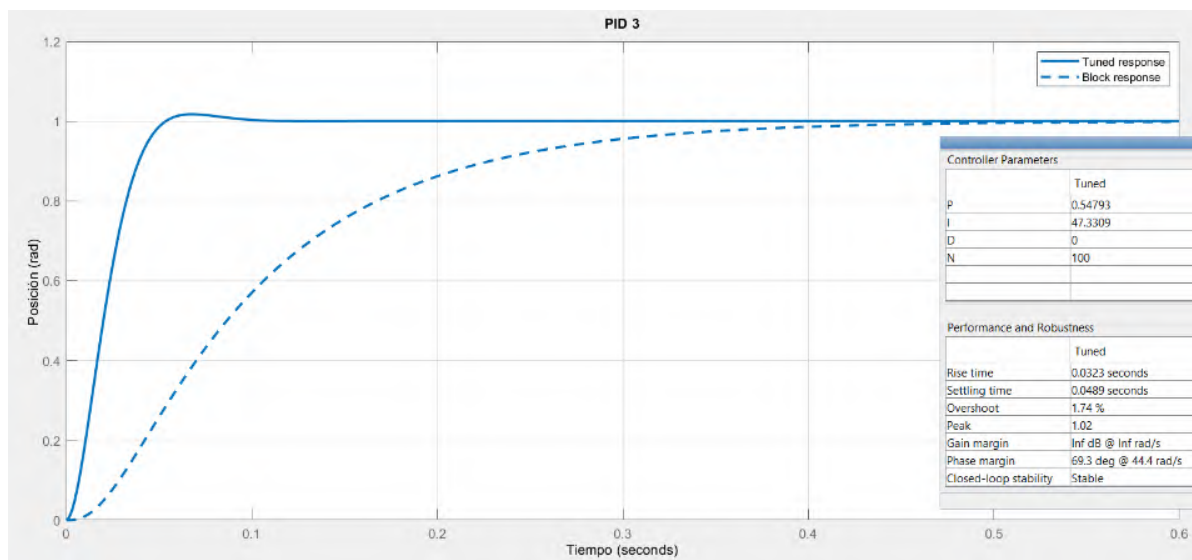


Figura 3.45 Regulador PID articulación 3

Fuente: elaboración propia

En el entorno de Simulink se modelarán los motores a partir de sus parámetros característicos y serán los actuadores del movimiento que se produzca.

Todos los elementos sólidos en Simscape tiene una trama base y otra final, la relación entre estas se realiza por medio de una transformación homogénea. Entre la trama final de un elemento y la trama base del siguiente se ubican las articulaciones. Los motores se han acoplado a los elementos con uniones rígidas sin posibilidad de movimiento.

Cada articulación tiene multitud de parámetros modificables como son la posición de equilibrio, la rigidez y amortiguamiento cuando entra en contacto con otros elementos, límites de actuación, la manera en que se accionará el movimiento y un sistema de sensores que miden la posición, velocidad, aceleración y par suministrado. En la siguiente figura se muestran las tramas de los dos eslabones y sus dos articulaciones.

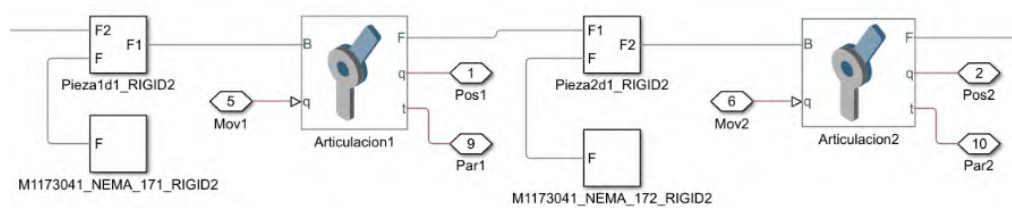


Figura 3.48 Detalle de tramas, elementos y articulaciones en Simulink

Fuente: elaboración propia

En cada articulación se ha configurado dos sensores, uno indica el par aplicado y otro la posición. La señal de posición se mandará por un lazo de realimentación hasta un comparador y de este a los controladores individuales para cada uno.

Los sensores de posición se conectan a un `scope` que muestra gráficamente la variación de la señal a los largo del tiempo de simulación. Además, los datos resultantes de la simulación se envían al `Workspace` y así se puede comprar la trayectoria referencia creada por cinemática inversa y la trayectoria que sigue el manipulador cuando se analiza el sistema completo.

4 RESULTADOS

En este apartado se expondrán los resultados obtenidos en las diferentes simulaciones cinemáticas que se han realizado trabajando en el espacio cartesiano y articular con las diferentes técnicas de interpolación expuestas.

Se compararán los resultados con el fin de determinar el método que mejores resultados proporcione en términos de calidad gráfica y rendimiento computacional. La simulación cinemática que mejores resultados ofrezca se utilizará para realizar la simulación dinámica.

4.1 SIMULACIÓN CINEMÁTICA

El manipulador seguirá la trayectoria definida por dos imágenes que toma como referencia, la silueta de una estrella y la silueta de un gato. Las fotografías se han capturado con el móvil y tienen una resolución original de 3096 x 3096 px.



Figura 4.1 Imagen original Gato.jpeg



Figura 4.2 Imagen original Estrella.jpeg

Las siluetas de estas imágenes se han definido con 100, 50 y 25 puntos de control para realizar una comparativa. En la siguiente Figura 4.3 se puede apreciar que la calidad de la silueta depende directamente de la cantidad de puntos de control cuando la imagen analizada tiene geometrías más complejas. Con 100 puntos ambas imágenes quedan definidas completamente, pero siempre se buscará un número de puntos lo más bajo posible pero que el resultado siga siendo aceptable. En este caso, teniendo en cuenta que en las siguientes fases se interpolarán los puntos de control, se considera que la imagen del gato queda lo suficientemente definida con 50 puntos y la imagen de la estrella con 25 puntos.

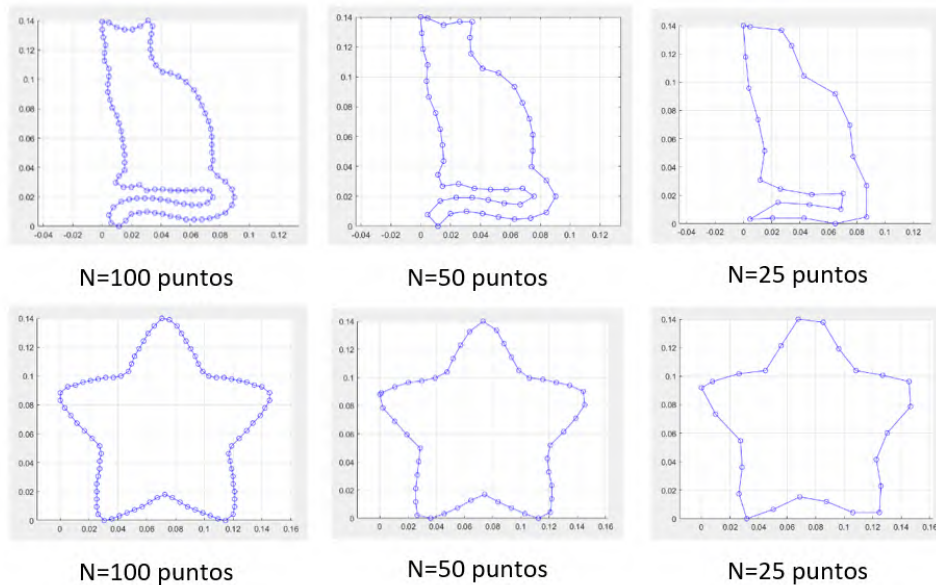


Figura 4.3 Comparativa densidad de puntos de control

Fuente: elaboración propia

Los resultados de las simulaciones cinemáticas se realizarán solo con la imagen del gato, puesto que si se mostrase de ambos sería muy repetitivo. Se ha optado por el gato ya que implica una geometría más complicada. Los resultados que se mostrarán a continuación han sido determinados interpolando los datos de los puntos de control cuando se ha trabajado en el espacio cartesiano y articular. Los resultados se enfocarán en mostrar la trayectoria cartesiana seguida por el efector final, cómo se han comportado las variables cinemáticas de las articulaciones (posición, velocidad y aceleración angular) y el tiempo que tarda el equipo en realizar la simulación.

En las simulaciones se ha elegido la opción de definir la figura con 50 puntos de control y entre cada uno hay 3 intervalos definidos por los puntos intermedios interpolados. En total la trayectoria final estará definida por 150 puntos.

La secuencia de simulaciones ejecutadas sigue la siguiente organización:

Trabajando en el espacio cartesiano

Interpolación lineal

Interpolación cúbica

Trabajando en el espacio articular

Interpolación lineal

Interpolación cúbica con velocidad en punto de control distinta de cero.


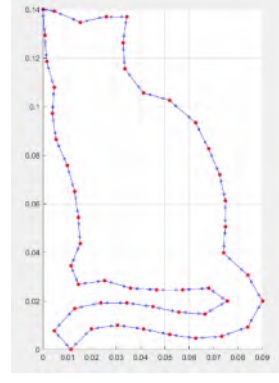
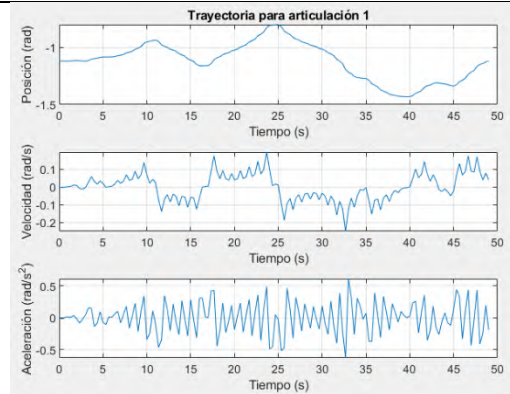
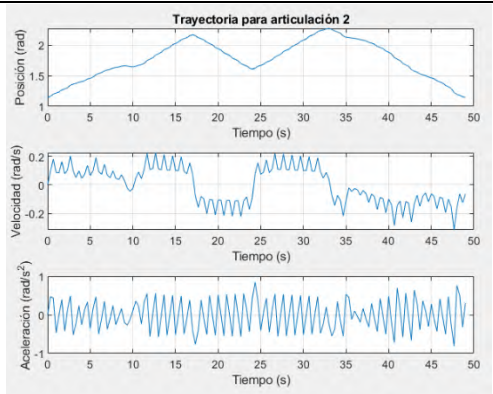
Interpolación quíntica con velocidad y aceleración en puntos de control distintas de cero.

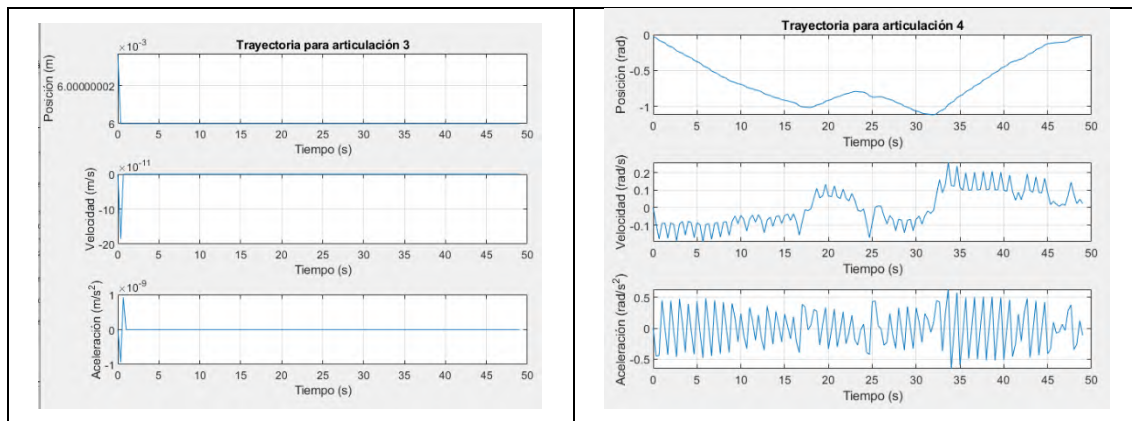
4.1.1 Trabajando en el espacio cartesiano

Se recuerda que se interpolarán los puntos de control en el espacio cartesiano y la figura se suaviza. Esta interpolación no implica que las variables cinemáticas articulares se suavicen también. Se estudiarán dos casos, interpolación lineal e interpolación cúbica.

Al utilizar **interpolación lineal** el efector final seguirá la trayectoria de una recta que une cada punto de control. Los puntos intermedios calculados se encontrarán en esa recta.

Tabla 4.1 Resultados interpolación lineal en espacio cartesiano

Interpolación lineal Gato (N=50) en espacio cartesiano	
 <p>Manipulador dibujando la figura</p>	 <p>Trayectoria resultante obtenida</p>
Se muestran los resultados de posición, velocidad y aceleración angulares de cada articulación durante la simulación	
	



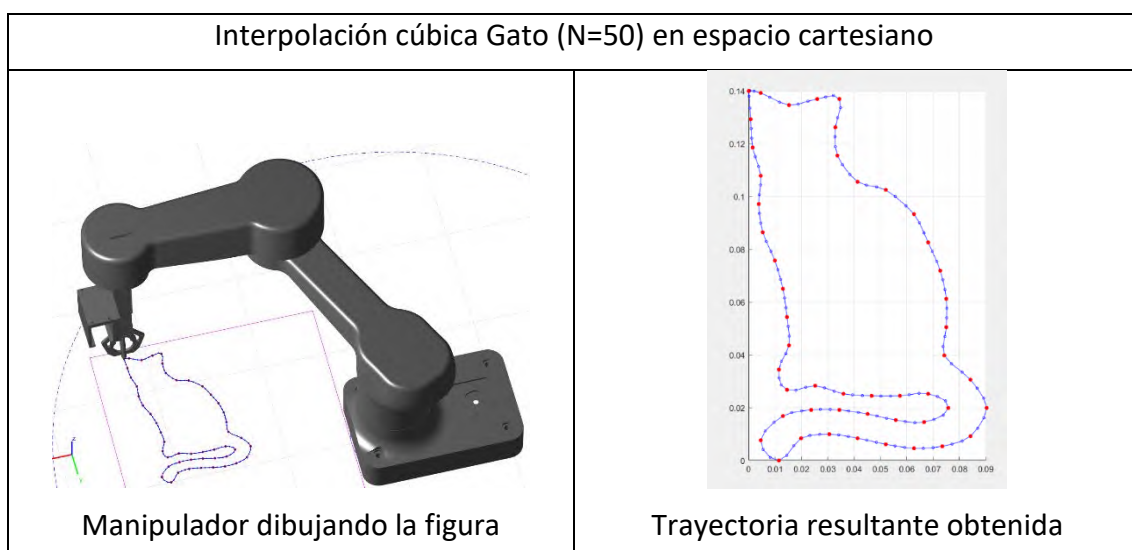
La trayectoria resultante, como era de esperar, es muy angulosa. En las zonas de la cola y de las orejas donde la geometría se curva más la interpolación lineal se queda insuficiente para dibujar la figura con calidad.

Las velocidades y aceleraciones que sufren las articulaciones son muy irregulares. Se ve que la articulación 2, es decir, el segundo eslabón rotacional es el que se desplaza una mayor cantidad de radianes. Además, es el que sufre unos cambios en las aceleraciones más bruscos. Hay momentos en los que pasa de -0.54 rad/s^2 a 0.87 rad/s^2 en un intervalo de tiempo muy corto. Este suceso conllevará una respuesta dinámica deficiente.

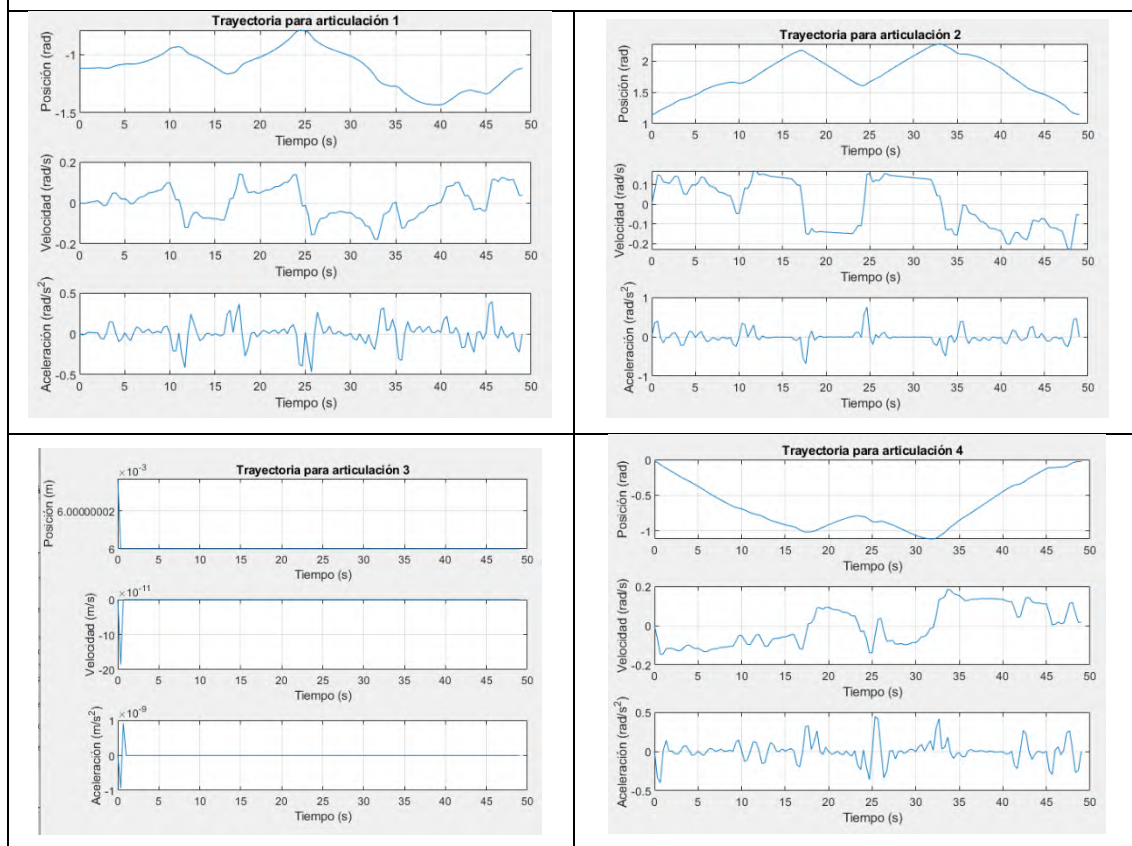
Tiempo de cálculo que ha tardado el ordenador en calcular los puntos necesarios para la simulación ha sido de 6.25 segundos.

Utilizando **interpolación cúbica** se espera que la trayectoria espacial sea suave y continua. En cambio, en el espacio articular se espera que se produzcan momentos de gran aceleración y desaceleración.

Tabla 4.2 Resultados interpolación cúbica en espacio cartesiano



Se muestran los resultados de posición, velocidad y aceleración angular de las articulaciones:



La trayectoria resultante, como se esperaba, es muy suave. En las zonas de la cola y de las orejas donde la geometría se curva más la interpolación cúbica da muy buenos resultados.

Las velocidades y aceleraciones que sufren las articulaciones tienen picos de irregularidad. Se ve que la articulación 2 nuevamente es la que hace un mayor recorrido. Además, es la que sufre unos picos mayores en aceleración llegando hasta 0.82 rad/s^2 .

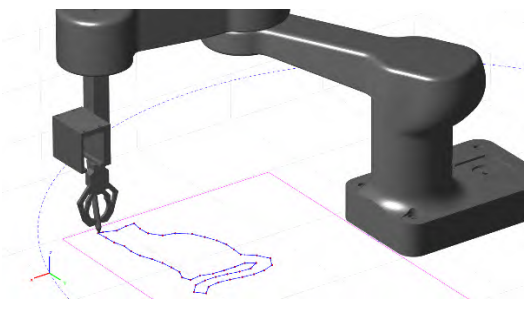
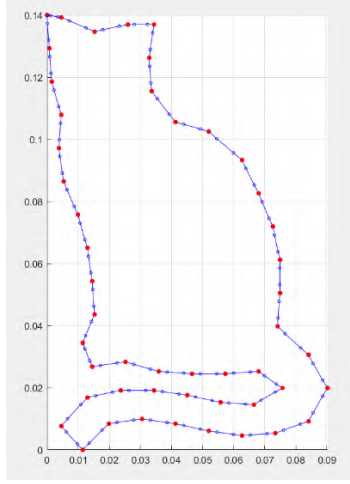
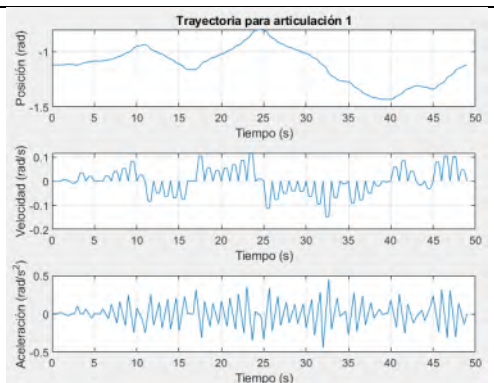
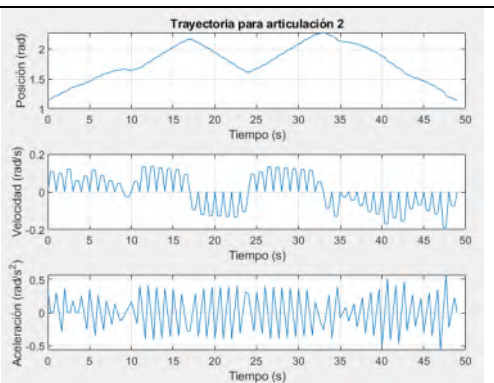
Tiempo de cálculo que ha tardado el ordenador en calcular los puntos necesarios para la simulación ha sido de 6.86 segundos.

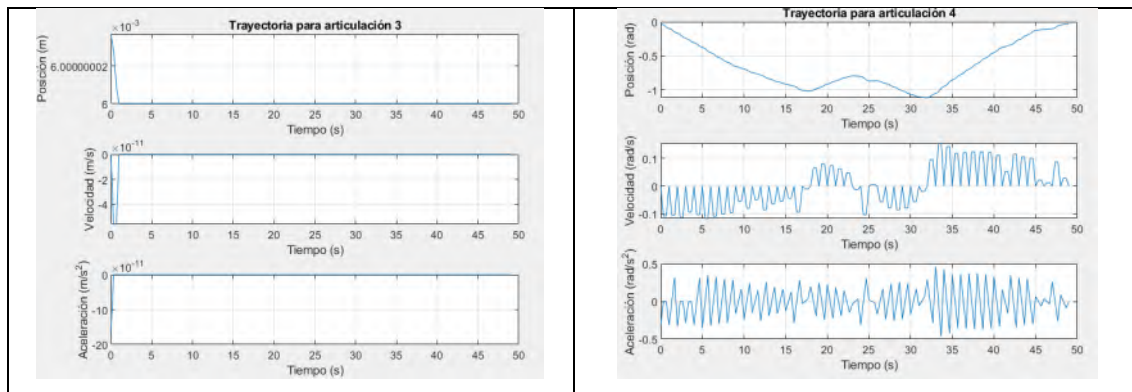
4.1.2 Trabajando en el espacio articular

Al trabajar en el espacio articular las variables que se controlan son las características cinemáticas de las articulaciones y no la trayectoria que realiza el efector final. Este hecho conlleva ventajas ya que las articulaciones sufren menos y el par que deben suministrar los actuadores es más continuo.

Utilizando una **interpolación lineal** las configuraciones articulares intermedias entre cada punto de control se encontrarán en la línea recta que une los puntos de control. Las velocidades están restringidas en cada punto de control, siendo estas nulas. De esta manera el modelo no realizará un movimiento continuo, sino que parará en cada punto de la trayectoria.

Tabla 4.3 Resultados interpolación lineal en espacio articular

Interpolación lineal Gato (N=50) en espacio articular	
 <p>Manipulador dibujando la figura</p>	 <p>Trayectoria resultante obtenida</p>
Se muestran los resultados de posición, velocidad y aceleración angular de las articulaciones:	
	



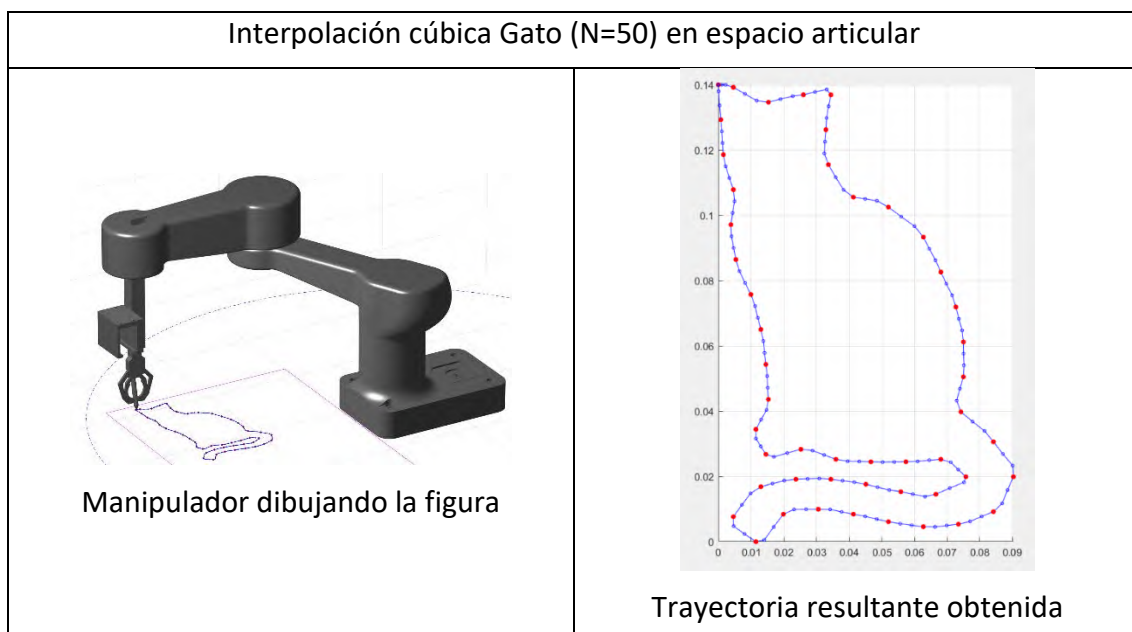
La trayectoria resultante, como era de esperar, es bastante angulosa. Al tener la figura puntos de control muy juntos no se aprecia del todo que la trayectoria entre cada punto no es rectilínea, sino que es una ligera curva. La calidad de la representación no es aceptable.

Las velocidades y aceleraciones que sufren las articulaciones son muy irregulares con gran cantidad de picos. Estas perturbaciones afectarían negativamente al sistema provocando muchas vibraciones.

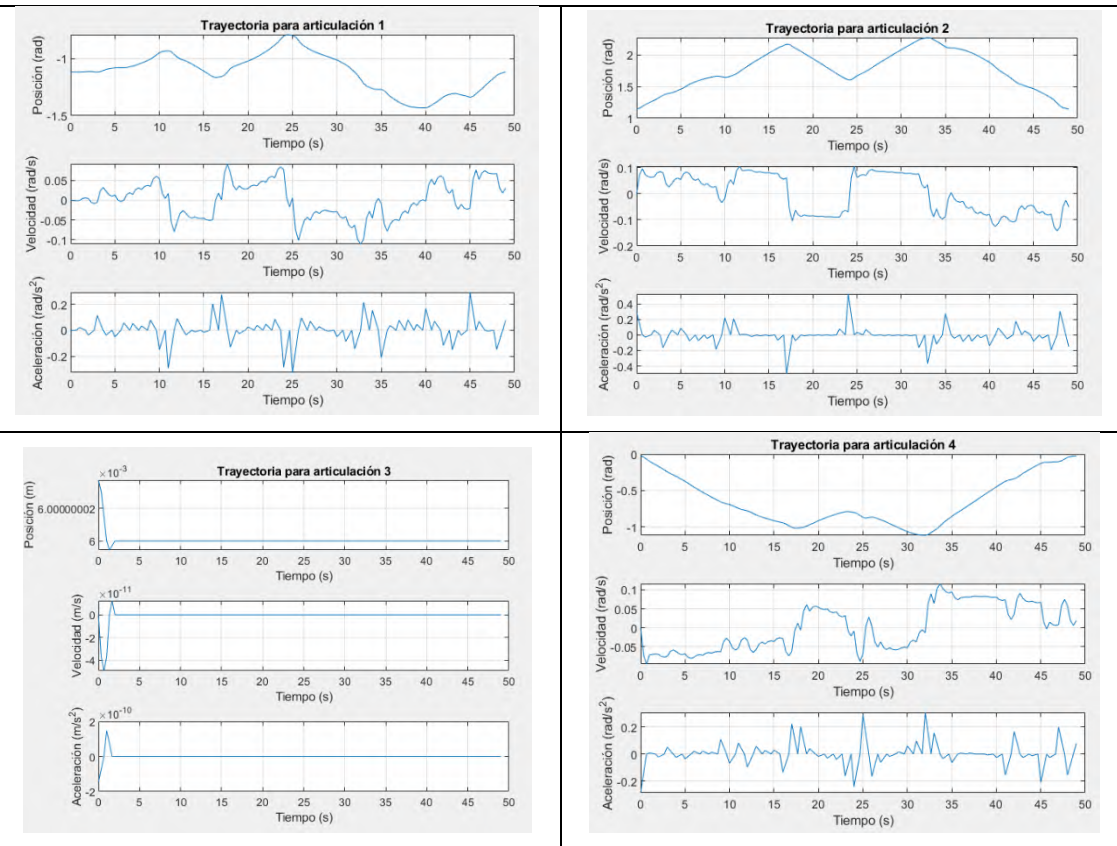
El tiempo de cálculo que ha tardado el ordenador en calcular los puntos necesarios para la simulación ha sido de 2.14 segundos.

Al emplear una **interpolación cúbica** las velocidades en los puntos de control son definidas y se obliga al manipulador a que pase por estos puntos con esa característica cinemática. La trayectoria espacial que realiza el efector final se determina aplicando cinemática directa a las posiciones articulares calculadas.

Tabla 4.4 Resultados interpolación cúbica en espacio articular



Se muestran los resultados de posición, velocidad y aceleración angular de las articulaciones:



La trayectoria cartesiana resultante es algo angulosa pero menos que al utilizar la interpolación lineal. Este fenómeno se da porque las velocidades en los puntos de control no son nulas, sino que se han pasado como variables a la función de interpolación. De esta manera, el extremo final mantiene la dirección con la que llega y sale de un punto de control.

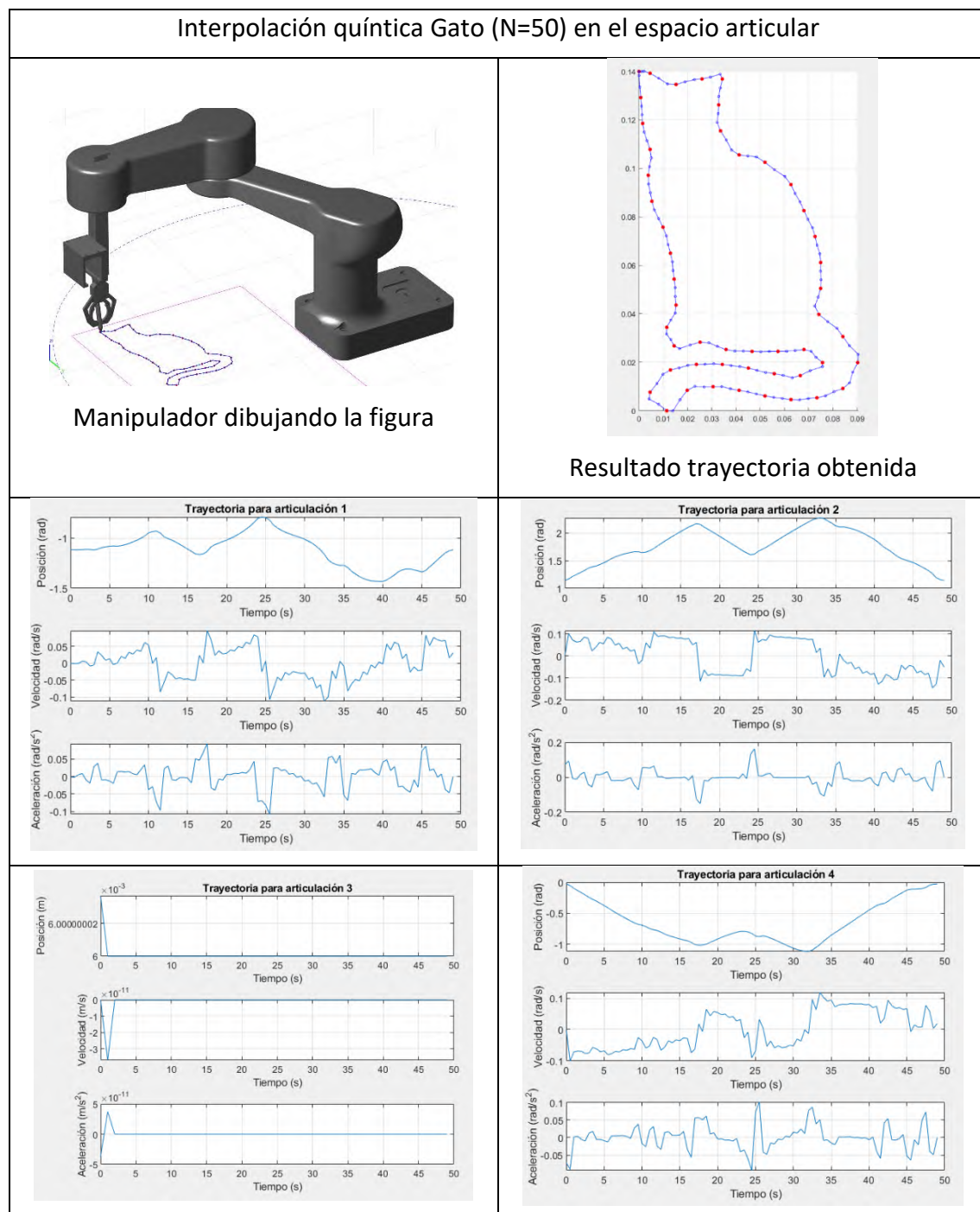
Como sucede en la interpolación lineal articular, al tener la figura una gran densidad de puntos de control no se aprecia del todo que la trayectoria entre cada punto no es rectilínea, sino que es una ligera curva. La calidad de la representación se puede dar por aceptable al ser esta curva en las zonas más complicadas.

Las velocidades articulares son algo irregulares y las aceleraciones presentan picos de gran amplitud en determinados momentos. La aceleración es la variable que más afecta a la respuesta dinámica, se desea que esta sea lo más suave posible puesto que se pueden producir vibraciones que afecten negativamente al sistema.

El tiempo de cálculo que ha tardado el ordenador en calcular los puntos necesarios para la simulación ha sido de 2.67 segundos.

Al emplear una **interpolación quíntica** las velocidades y aceleraciones angulares en los puntos de control son definidas y se obliga al manipulador a que pase por estos puntos con esa característica cinemática. La trayectoria espacial que realiza el efector final se determina aplicando cinemática directa a las posiciones articulares calculadas.

Tabla 4.5 Resultados interpolación quíntica en espacio articular



La trayectoria cartesiana resultante es algo angulosa en determinadas zonas, pero en general se ve con buena suavidad. Se consigue de la misma manera que sucedía con al interpolación cúbica en el espacio articular.

Como sucede en las otras interpolación articular, al tener la figura una gran densidad de puntos de control no se aprecia del todo que la trayectoria entre cada punto no es rectilínea, sino que es una ligera curva. La calidad de la representación se puede dar por aceptable por mantener la curva en las zonas más complicadas.

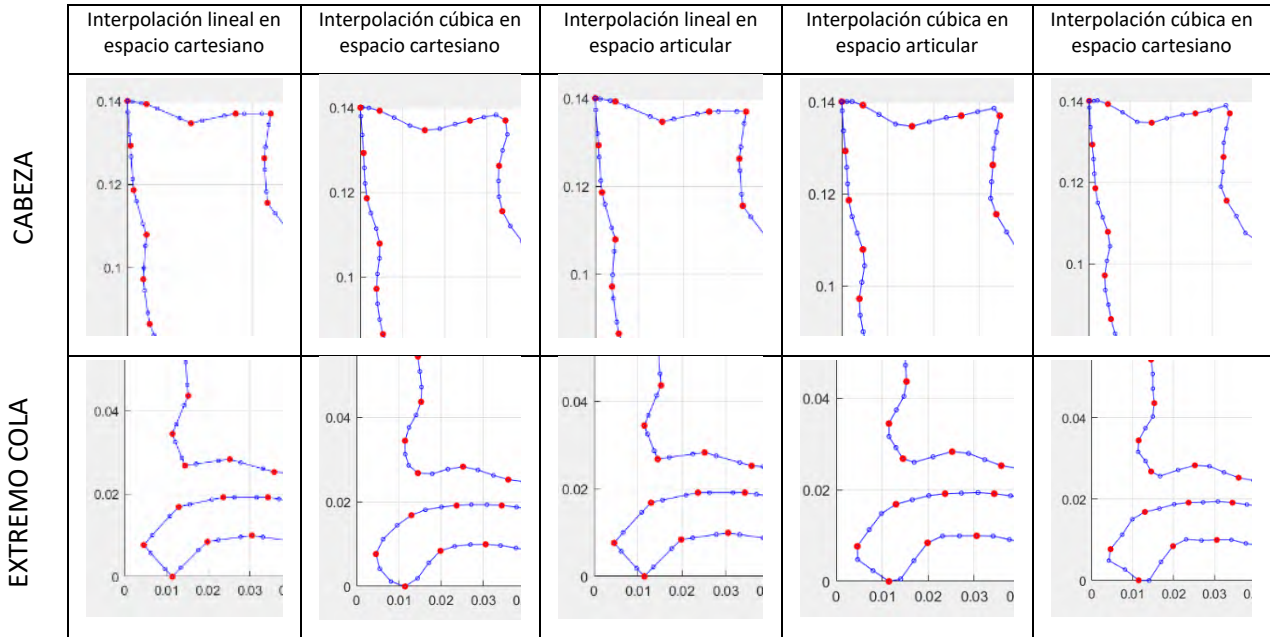
Las velocidades articulares son algo irregulares y las aceleraciones presentan picos, pero de menos amplitud que al aplicar una interpolación cúbica. En términos cinemáticos aplicar esta interpolación conlleva un control sobre las velocidades y aceleraciones angulares casi total.

El tiempo de cálculo que ha tardado el ordenador en calcular los puntos necesarios para la simulación ha sido de 3.04 segundos.

4.2 COMPARACIÓN DE RESULTADOS DE LA SIMULACIÓN CINEMÁTICA

A continuación, se mostrarán imágenes comparativas entre las diferentes simulaciones donde se verán con mayor claridad las diferencias en la trayectoria espacial y articular al utilizar los diversos métodos. Se mostrará la trayectoria articular de la unión 2 por ser la que mayor recorrido efectúa y en la que se han producido las mayores aceleraciones.

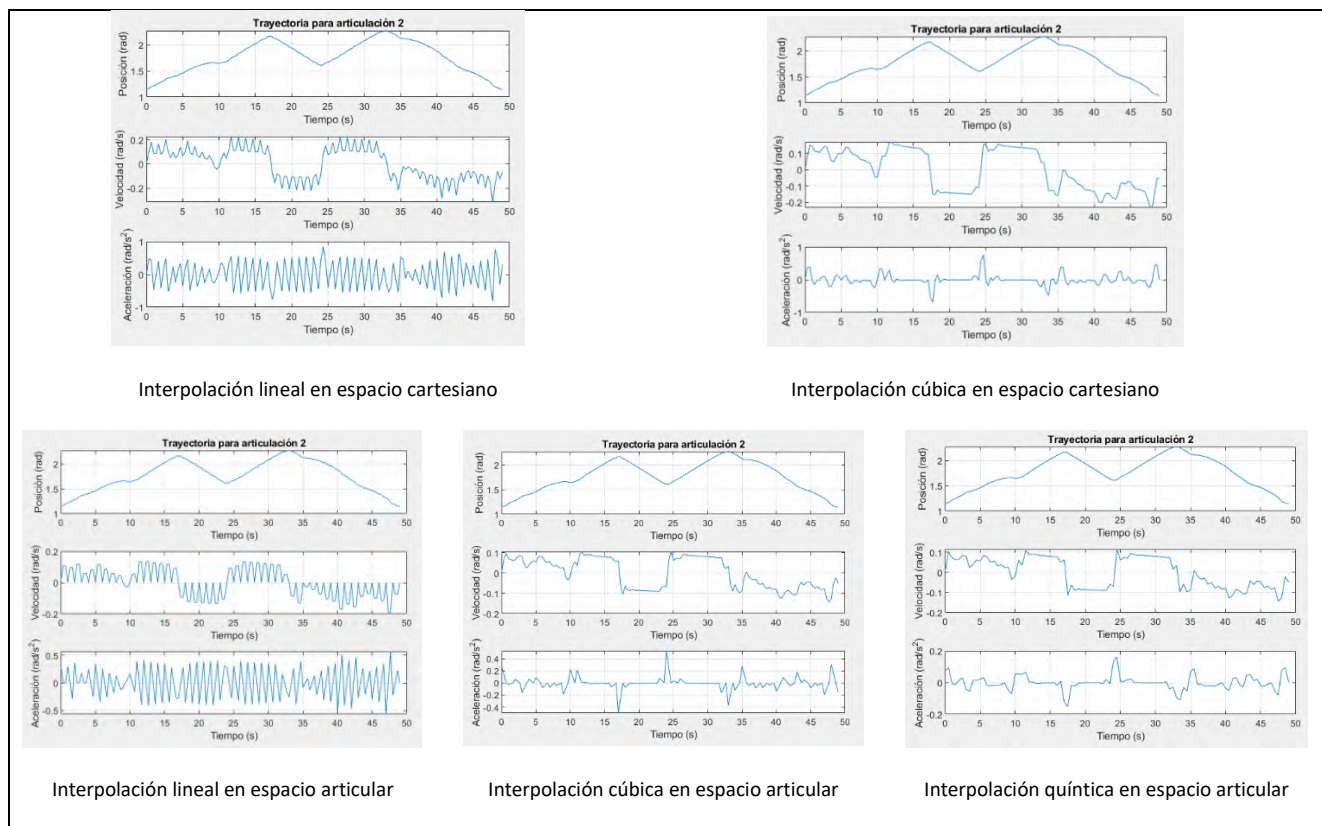
Tabla 4.6 Comparación de resultados gráficos



En esta recopilación de imágenes se determina que el método que da mejor resultado gráfico es la interpolación cúbica trabajando en el espacio cartesiano, seguido de las interpolaciones cúbica y quíntica trabajando en el espacio articular. Los resultados, usando el resto de los métodos, son más angulosos y el cambio entre intervalos se percibe en ellos.

Las características cinemáticas de la segunda articulación al utilizar los diferentes métodos de interpolación se exponen en la siguiente Tabla 4.7 para facilitar la visualización de la comparativa.

Tabla 4.7 Comparativa cinemática de los métodos de interpolación en la articulación 2



Las interpolaciones lineales tanto en el espacio articular como en el espacio cartesiano dejan resultados con multitud de picos en las variables de velocidad y aceleración angular. La amplitud de estas variables es similar a los resultados obtenidos con la interpolación cúbica en ambos espacios.

Los resultados obtenidos al utilizar interpolaciones cúbicas presentan una forma similar. Las velocidades son bastante estables y no presentan una gran cantidad de picos como sucedía con la interpolación lineal. La amplitud de la velocidad tanto en el espacio cartesiano como articular es similar, llegando a un máximo absoluto de 0.22 rad/s y 0.16 rad/s respectivamente. En las aceleraciones es donde se nota más diferencia y se evidencia la razón

por la que se trabaja en el espacio articular. El máximo de aceleración angular en el espacio cartesiano es cercano a 1 rad/s^2 , mientras que la interpolación cúbica en el espacio articular no llega a 0.5 rad/s^2 .

La interpolación quíntica ofrece unos mejores resultados atendiendo a las características cinemáticas, siendo el máximo de aceleración angular menor a 0.2 rad/s^2 .

Finalmente se mostrarán el tiempo que ha tardado el equipo en realizar las simulaciones. Así se puede hacer una idea de cuáles son las que más gasto computacional exigen. Para un mismo manipulador, este tiempo depende directamente del número de puntos de control que tenga la figura, del número de puntos intermedios que se calcularán las interpolaciones, el tipo de interpolación elegida y principalmente del espacio de trabajo donde se interpolen los puntos. Trabajar en el espacio articular consume menos gasto computacional que trabajar en el espacio cartesiano.

Cabe mencionar que, en el caso más desfavorable, calcular la cinemática inversa de 150 puntos de un manipulador de 4 grados de libertad, como el diseñado, no es una tarea muy compleja para un software tan potente como es Matlab. Cuantos más grados de libertad tenga el manipulador, las cantidad de ecuaciones no lineales a resolver aumenta significativamente y por tanto el tiempo de cálculo.

Tabla 4.8 Comparativa de tiempo de cálculo empleado con los diferentes métodos de interpolación

Interpolación lineal en espacio cartesiano	Interpolación cúbica en espacio cartesiano	Interpolación lineal en espacio articular	Interpolación cúbica en espacio articular	Interpolación quíntica en espacio articular
6.25 segundos	6.86 segundos	2.14 segundos	2.67 segundos	3.04 segundos

En resumen:

Se optará por el método de la interpolación con polinomios cúbicos en el espacio cartesiano cuando se priorice el resultado gráfico por encima del rendimiento cinemático y del gasto computacional requerido.

Se optará por una interpolación con polinomios quínticos o cúbicos en el espacio articular cuando se priorice el rendimiento cinemático y sea suficiente un resultado gráfico aceptable. Con preferencia a la interpolación quíntica cuando se desea limitar las aceleraciones de las articulaciones.

4.3 SIMULACIÓN DINÁMICA

La simulación dinámica toma como base la simulación cinemática previamente explicada. Utiliza como entrada o puntos de referencia la trayectoria interpolada calculada. En este caso, se ha elegido una interpolación en el espacio articular de tipo quíntica.

El análisis de resultados que se expondrá a continuación es una comparativa entre la trayectoria ideal calculada por la cinemática y la trayectoria más fiel a la realidad que seguirá el manipulador cuando en la simulación se tenga en cuenta la masa de los eslabones, sus inercias, las características eléctricas y mecánicas de los motores elegidos y el sistema de control.

Cuando se ejecuta la simulación, el programa tarda 10.2 segundos en calcular todos los parámetros del sistema con este ejemplo. A continuación, inicia una animación (Figura 4.4) donde se muestra el movimiento que realiza manipulador. Desgraciadamente el entorno de Simscape no permite visualizar `plot` y no se puede configurar que ‘dibuje’ la trayectoria. Es un aspecto meramente visual, puesto que las coordenadas sí se recogen las coordenadas del movimiento de la traza final y a posteriori se consigue dibujar la trayectoria.

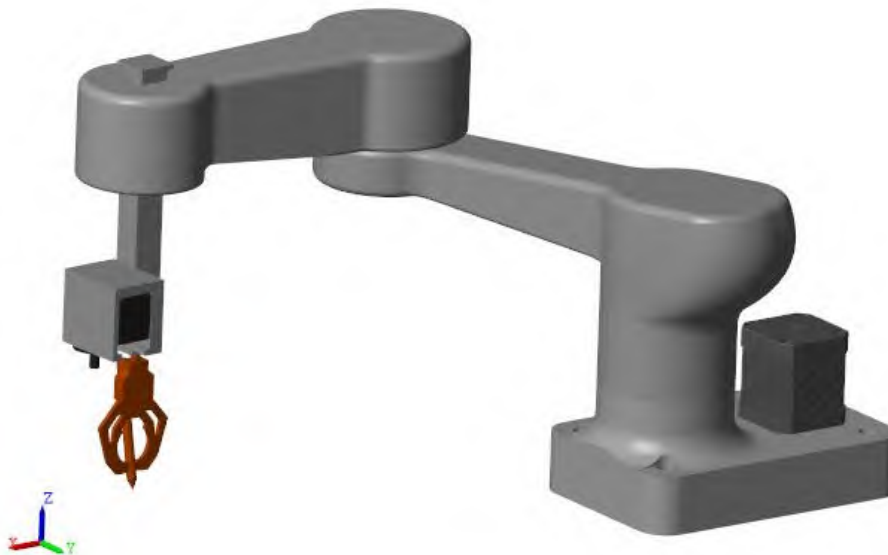


Figura 4.4 Animación del modelo dinámico con Mechanics Explorer
Fuente: elaboración propia

Los resultados que se muestran indican cómo se comporta el sistema dinámicamente con y sin etapa de control ajustada con el fin de evidenciar la importancia de este subsistema. En las siguientes Figuras 4.5 y 4.6 la trayectoria que realiza el manipulador en azul con la simulación dinámica y la que debería realizar en naranja.

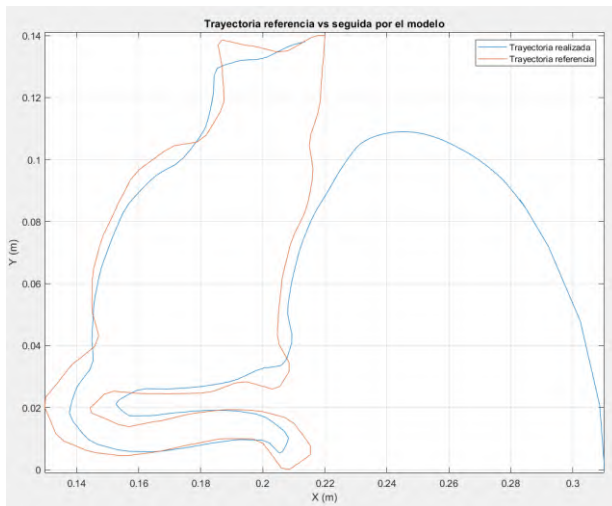


Figura 4.5 Trayectoria resultante en modelo dinámico sin etapa de control ajustada

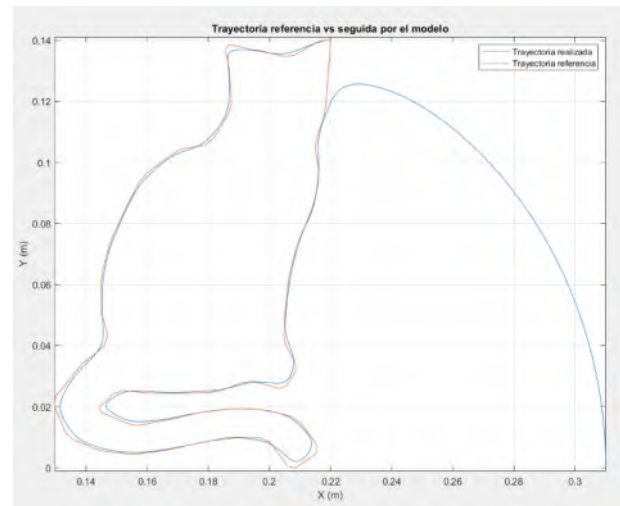


Figura 4.6 Trayectoria resultante en modelo dinámico con etapa de control ajustada

Como se esperaba, el sistema de control es eficaz y necesario para que el manipulador pueda seguir correctamente la trayectoria. En la imagen derecha una vez que se alcanza la silueta copia muy bien la trayectoria a seguir. Sin embargo, en la simulación donde no se definió correctamente la etapa de control el manipulador siguió ineficazmente la silueta.

El manipulador comienza la trayectoria en su posición preestablecida, en este caso la configuración que se importó desde Autodesk Inventor con todos sus eslabones a 0° y la articulación prismática a 0m. Esta configuración implica que está completamente extendido y el efector final se encuentra en (0.31, 0, -0.006) m. El arco que llega al cuerpo de la figura es el camino que sigue desde su posición de reposo hasta alcanzar la silueta.

El resultado al realizar la simulación dinámica en cuanto a posición, velocidad y aceleración articulares se asemeja a los resultados obtenidos en la simulación cinemática, se muestran los resultados de la articulación 1, 2, 3 y 4 en las figuras 4.7, 4.8, 4.9, 4.10 respectivamente. Las gráficas resultantes son muy similares excepto alguna salvedad que se enunciarán más adelante.

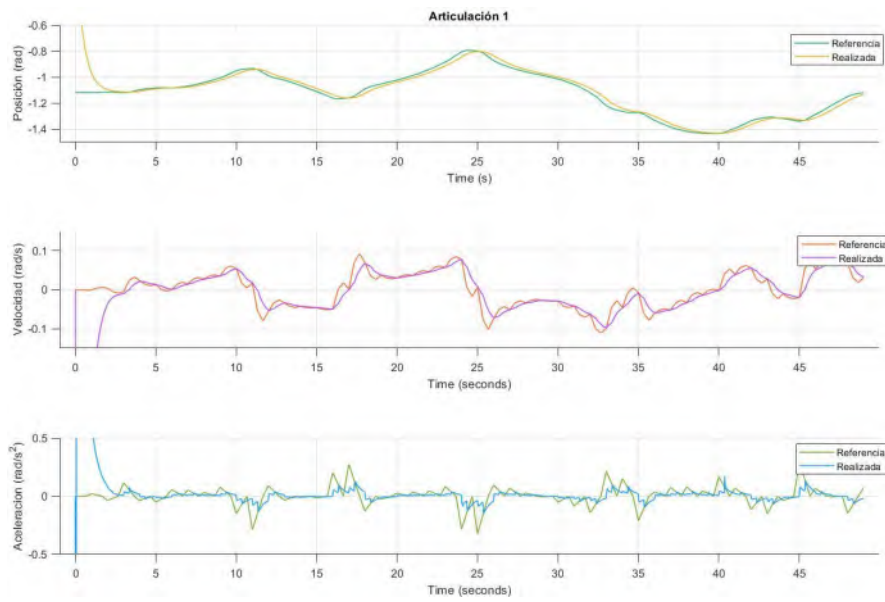


Figura 4.7 Resultados simulación dinámica articulación 1

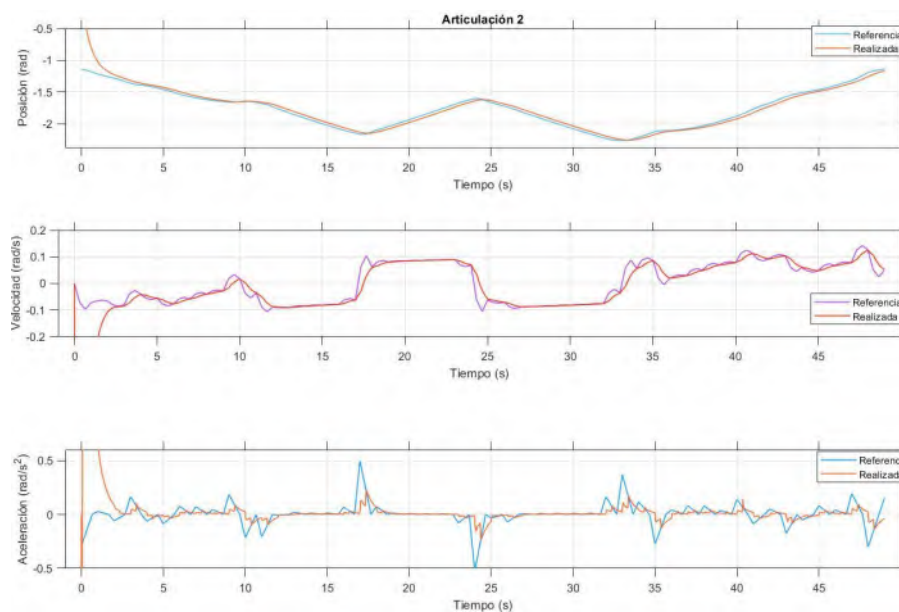


Figura 4.8 Resultados simulación dinámica articulación 2

Durante los primeros segundos de simulación el manipulador debe desplazarse desde la posición de reposo hasta la figura que debe dibujar. Se ve que tarda 3 segundos aproximadamente en realizar esta tarea, ya que en las articulaciones 1 y 2 hay una gran diferencia entre las gráficas hasta ese momento.

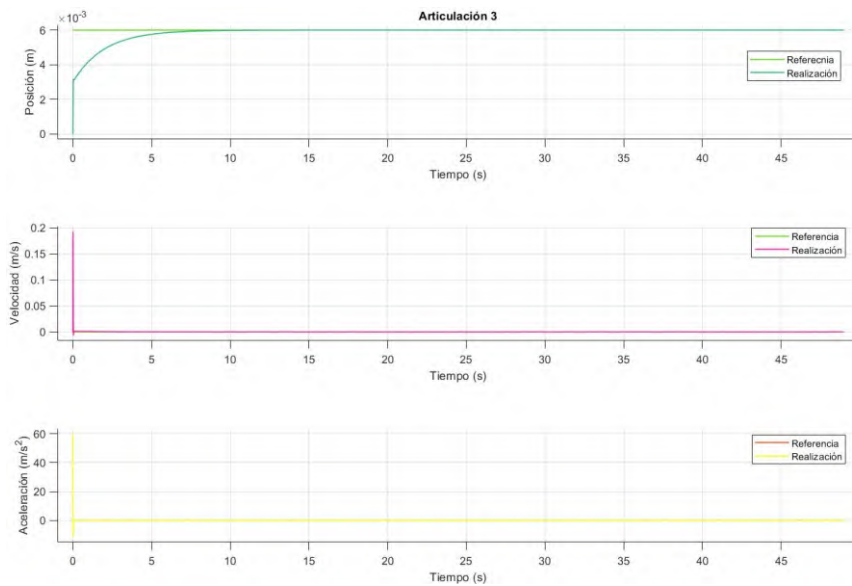


Figura 4.9 Resultados simulación dinámica articulación 3

La articulación 3 no tiene gran interés, ya que el extremo final del manipulador se encuentra a la misma altura que el plano de trabajo y por tanto el desplazamiento vertical es prácticamente nulo a excepción de los segundos iniciales.

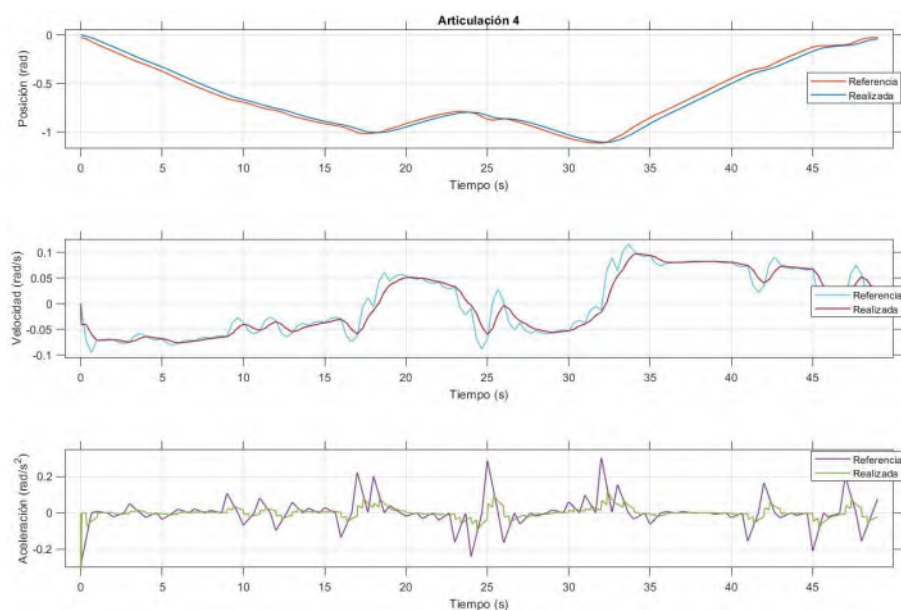


Figura 4.10 Resultados simulación dinámica articulación 4

Un detalle que se quiere remarcar es que el manipulador realiza un movimiento más suave en términos de velocidad, esto se debe a que las inercias y masas de los elementos entran en juego en esta simulación, haciendo más difícil modificar la velocidad del manipulador de un intervalo al siguiente. Como consecuencia de este efecto, las amplitudes de las aceleraciones angulares de los elementos son más reducidas.

De esta manera concluye la simulación dinámica del manipulador dibujante SCARA. Se considera un resultado satisfactorio los resultados obtenidos ya que este consigue mantenerse por la trayectoria definida. En el anexo I se recogen más ejemplos con resultados de las trayectorias con forma de estrella y alfanuméricas.

5 CONCLUSIONES

Cuando iniciamos nuestra investigación sobre este campo de la robótica teníamos la sensación de que entrábamos en un mundo apasionante, de plena actualidad y con una evolución tan rápida que cualquier investigación sobre el tema quedaría obsoleta en poco tiempo. Pero, al adentrarnos en el tema, nos hemos dado cuenta de que el avance tecnológico permite ensayar nuevas posibilidades teniendo en cuenta planteamientos teóricos de siempre. Por eso, el resultado final resulta novedoso, pero con sólidas bases tradicionales, lo que permite augurar un futuro para nuestra investigación con una duración más larga en el tiempo y con múltiples aplicaciones en la vida social.

Como ya hemos indicado más arriba, entendemos que la robótica es un ámbito que permite una perfecta integración de todos los campos que debe manejar un ingeniero industrial (mecánica, electrónica, dinámica, cinemática, automática, electrotecnia, computación, etc.) según hemos podido adquirir en nuestra formación académica. Una de nuestras inquietudes profesionales es especializarnos en Diseño Gráfico Industrial y creemos que con el diseño de este robot manipulador hemos iniciado nuestro camino hacia la especialización en el ámbito industrial.

Los mencionados conocimientos multidisciplinarios de la Ingeniería en Tecnologías Industriales, así como el manejo de herramientas informáticas como Matlab y técnicas de modelado (Inventor) adquiridas en diferentes asignaturas a lo largo de nuestra carrera nos han permitido cumplir el desafío inicial de diseñar un modelo de robot manipulador dibujante que tuviera aplicación en la vida cotidiana (escribir, dibujar...), a la vez que en un futuro, implementando algunas de sus características, también pudiera aplicarse en el mundo industrial.

A la vista de nuestro análisis y los resultados de nuestro estudio, creemos haber cumplido con los objetivos que inicialmente nos propusimos.

Para la fundamentación teórica tuvimos que leer diversas fuentes bibliográficas que nos permitieron organizar las características morfológicas y tipología de diferentes manipuladores, a partir de cuya información nos centramos en el modelo Drawer SCARA, diseñando con precisión sus elementos físicos, actuadores y transmisión.

Así mismo destacamos como interesantes los apartados dedicados a la cinemática y dinámica (control) del modelo, consiguiendo un análisis detenido de los parámetros cinemáticos articulares. Igualmente hemos atendido a la planeación de trayectorias, persiguiendo la mejor manera de crear la trayectoria que debe seguir el extremo del efector final al proporcionarle como entrada una imagen.

Para ello contamos con el soporte de un software muy apropiado como es Matlab como programa potente en la resolución de ecuaciones matemáticas, dentro del cual encontramos muy útiles sus múltiples herramientas y librerías para modelar, simular y animar sistemas físicos, en especial, la versión que arrancó en 2015, actualizada hasta 2021, conocida como *Toolbox Robotic System*, en cinemática y dinámica. Al realizar la simulación completa multifísica donde entraban en juego también variables eléctricas de los motores se recurrió a Simscape utilizando sus módulos dedicados a modelos mecánicos (Simscape Multibody) y eléctricos (Simscape electrical) .

Estos recursos de análisis matemático se complementaron con técnicas de modelado aprendidas también en nuestra formación académica, nos referimos al programa Inventor, que nos permite aprovechar las ventajas de modelos parametrizados, con una modificación rápida y sencilla proporcionando agilidad en la fase de diseño, así como del diseño del sistema de control con Simscape Control Design.

Con todo el aporte teórico, software informático y tecnología que hemos reseñado, elaboramos la aportación esencial de nuestra investigación, consistente en el diseño y simulación de un robot manipulador dibujante SCARA, usando el mejor método de interpolación entre los estudiados, que ofrece un equilibrio entre calidad gráfica y gasto computacional.

Como se incluye en el título de nuestro trabajo, el robot manipulador es dibujante y no solo escritor. Podrán existir códigos más completos, pero el aquí utilizado ha sido diseñado personalmente, utilizando las librerías de Matlab, que nos ha ayudado a comprender completamente el desafío de generación y trazado de trayectorias. Especial mención a Image Processing Toolbox y Robotics System Toolbox). Es un reto conseguido el hecho de que nuestro robot manipulador en la simulación puede dibujar y, por tanto, también puede reproducir símbolos gráficos, esto es, letras (mayúsculas, minúsculas) o incluso, números.

Pretende ser un modelo económico, de uso particular o en el ámbito educativo, factible de conseguir, con una aplicación versátil en la vida cotidiana pues, además de dibujar, al robot manipulador se le podrá incorporar una pinza funcional que pueda sujetar otras cosas, permitiendo así otras múltiples posibilidades de aplicación industrial.

Como todo primer trabajo de investigación, aunque se hayan cumplido los objetivos previstos inicialmente, siempre quedan aspectos abiertos a seguir investigando, como futuras vías abiertas a profundizar en ellas, en concreto, en el modelado y control de los manipuladores industriales, pudiendo realizar tareas más elaboradas de utilidad en la industria.

Se propondría implementar este modelo en la realidad, construyendo sus elementos con impresión 3D. Igualmente podríamos prever la adición al robot de accesorios como incorporar una cámara que detecte una imagen del entorno y la dibuje en tiempo real o iluminación propia para trabajar en entornos oscuros. Pero eso ya es otro trabajo de investigación.

6 REFERENCIAS Y BIBLIOGRAFÍA

- [1] IFR, «Executive Summaty Worl Robotics 2020 Industrial Robots,» 2020.
- [2] L. García, «Valdecilla contará con un nuevo robot quirúrgico Da Vinci tras una inversión de 3.8 millones,» *elDiario.es*, pp. https://www.eldiario.es/cantabria/politica/valdecilla-da-vinci-inversion-millones_1_1146665.html, 6 2 2020.
- [3] BostonDynamics, «BostonDynamics,» 8 7 2021. [En línea]. Available: www.bostondynamics.com/atlas.
- [4] IFR, [En línea]. Available: <https://ifr.org/>.
- [5] Antonio Barrientos, Fundamentos de robótica, Mc Graw Hill, 1997.
- [6] S. K. Saha, Introduction to Robotics, Second Edition, Nueva Delhi: McGraw Hill Education, 2014.
- [7] R. Franz, The Kinematics of Machiney: Outlines of a Theory of Machines, London: MacMillan, 1876.
- [8] K. M. L. a. F. C. Park, MODERN ROBOTICS, Cambridge University Press, 2019.
- [9] V. R. González, «Fundamentos de robótica,» 2012. [En línea]. Available: http://platea.pntic.mec.es/vgonzale/cyr_0204/cyr_01/robotica/index.htm.
- [10] R. N. Jazar, Theory of Applied Robotics, Springer, 2010.
]
- [11] IFR, «Chapter 1. Reviews definitions and classifications of industrial robots and service robots,»
] 2020.
- [12] «Revista de robots,» 27 4 2021. [En línea]. Available: [.https://revistaderobots.com/robots-y-robotica/robot-scara-articulados-caracteristicas-y-marcas/](https://revistaderobots.com/robots-y-robotica/robot-scara-articulados-caracteristicas-y-marcas/). [Último acceso: 2021 8 17].
]
- [13] M. Pérez-Cisneros, Fundamentos de robótica y mecatrónica con Matlab y Simulink, Madrid: Ra-
] Da, 2014.
- [14] R. S. H. J Denavit, A kinematic Notation for Lower-Pair Mechanised Based on Matrices, ASME
] Journal of Applied Mechanism, 1951.
- [15] L. S. L. V. G. O. Bruno Siciliano, Robotics: Modelling, Planning and Control, Springer, 2010.
]
- [16] A. O. Baturone, Robótica, manipuladores y robots móviles, Marcombo, 2001.
]
- [17] P. Corke, Robotics, Vision and Control. Fundamental algorithms in MATLAB, Springer, 2017.
]

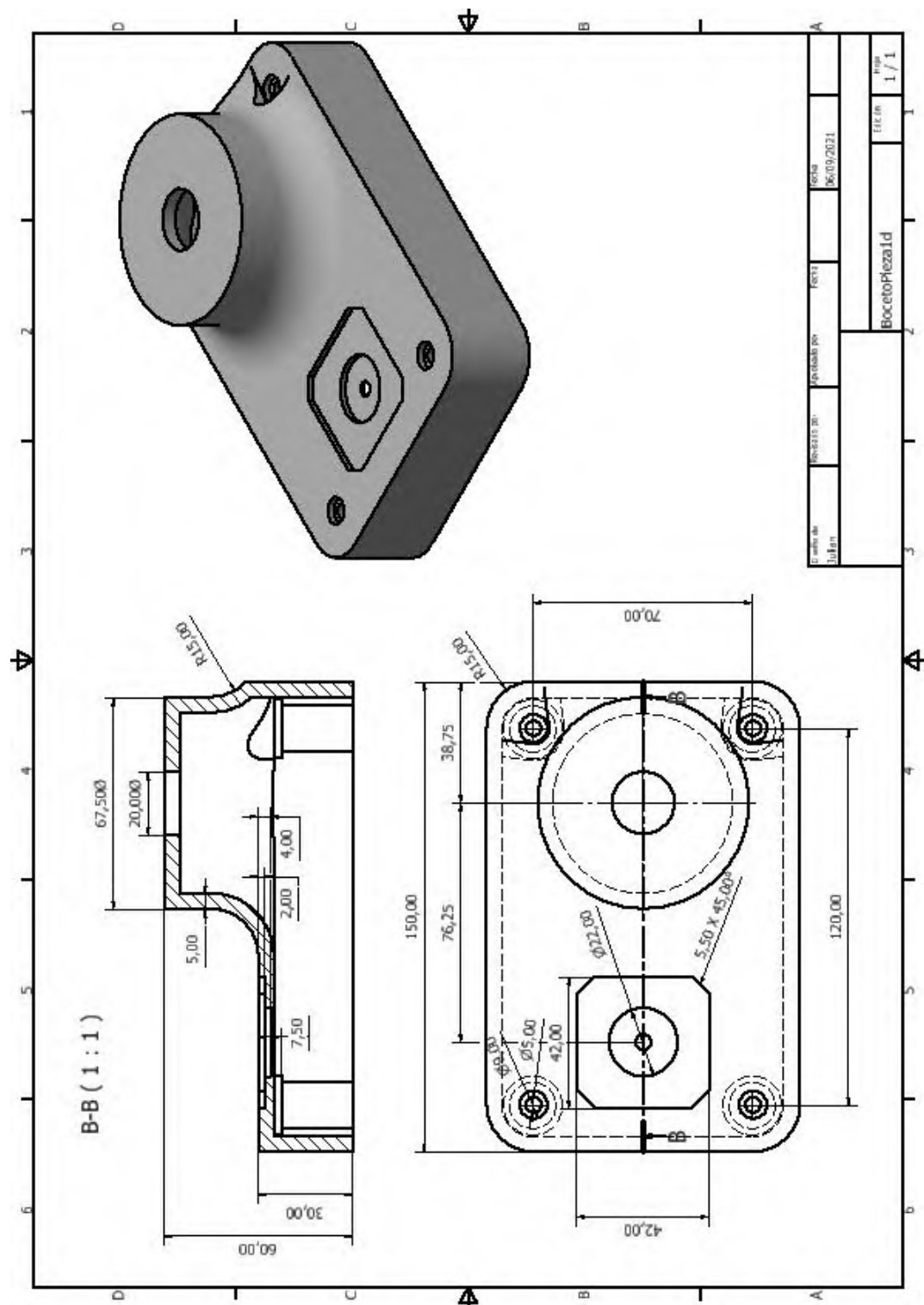
- [18 F. Mora, Máquinas eléctricas 6a ed, McGraw Hill, 2008.
]
- [19 K. M. L. a. F. C. Park, «Modern Robotics,» 8 4 2020. [En línea]. Available:
] <https://modernrobotics.northwestern.edu/nu-gm-book-resource/2-5-task-space-and-workspace/>. [Último acceso: 24 7 2021].
- [20 MathWorks, Robotics System Toolbox™ User's Guide, The MathWorks, 2021.
]
- [21 «Interpolación con funciones splines,» [En línea]. Available:
] http://www4.ujaen.es/~angelcid/Archivos/An_Met_Num_INFORMATICA/Splines.pdf.
- [22 K. Ogata, Ingeniería de control moderna 5ed, Pearson, 2010.
]
- [23 V. S. Rafael Kelly, Control de movimiento de robots manipuladores, Madrid: Prentice Hall,
] 2003.
- [24 C. Pardo, «Picuino,» 7 2021. [En línea]. Available: picuino.com/es/arduprog/control-pid.html.
] [Último acceso: 17 8 2021].
- [25 L. C, «3Dnatives,» 18 3 2019. [En línea]. Available: 3dnatives.com/es/mejores-sofware-cad-programa-180320192/. [Último acceso: 21 8 2021].
- [26 MathWorks, Simulink User's Guide, The MathWorks, 2021.
]
- [27 MathWorks, Simscape Multibody, The MathWorks, 2021.
]
- [28 MathWorks, Simscape electrical User's Guide, The MathWorks, 2021.
]
- [29 MathWorks, Simulink Control Design User's Guide, The MathWorks, 2021.
]
- [30 A. G. M. M. C. A. Noriega González, «Resolución del problema cinemático inverso en un
] robot SCARA mediante grupos de Assur,» Oviedo, 2014.
- [31 F. T.-R. S. C.-M. F. R.-L. L.A. Pérez-Gaspar, «Curve fitting using polygonal approximation for a
] robotic writing task,» Oaxaca, México, 2015.
- [32 M. S. C. Team, «Trajectory Planning for Robot Manipulator,» 3 2021. [En línea]. Available:
] MathWorks Student Competitions Team (2021). Traj<https://github.com/mathworks-robotics/trajectory-planning-robot-manipulators>. [Último acceso: 30 7 2021].
- [33 «Control tutorials for Matlab & Simulink,» [En línea]. Available:
] <https://ctms.engin.umich.edu/CTMS/index.php?example=MotorPosition§ion=SimulinkModeling>. [Último acceso: 24 8 2021].

- [34 C. G. Manuel Guillermo, «Obtención experimental de los parámetros del motor que se utilizará en el sistema de locomoción de una esfera rodante,» 2009.
- [35 «women.es,» 4 Agosto 2021. [En línea]. Available: <https://www.womens.es/wp-content/uploads/El-sistema-robotico-Da-Vinci.jpg.webp>.
- [36 Alvy, «Microsiervos,» 14 7 2020. [En línea]. Available: <https://www.microsiervos.com/archivo/robots/quinzcy-robot-artista-dibujar-escribir-matematicas.html>.

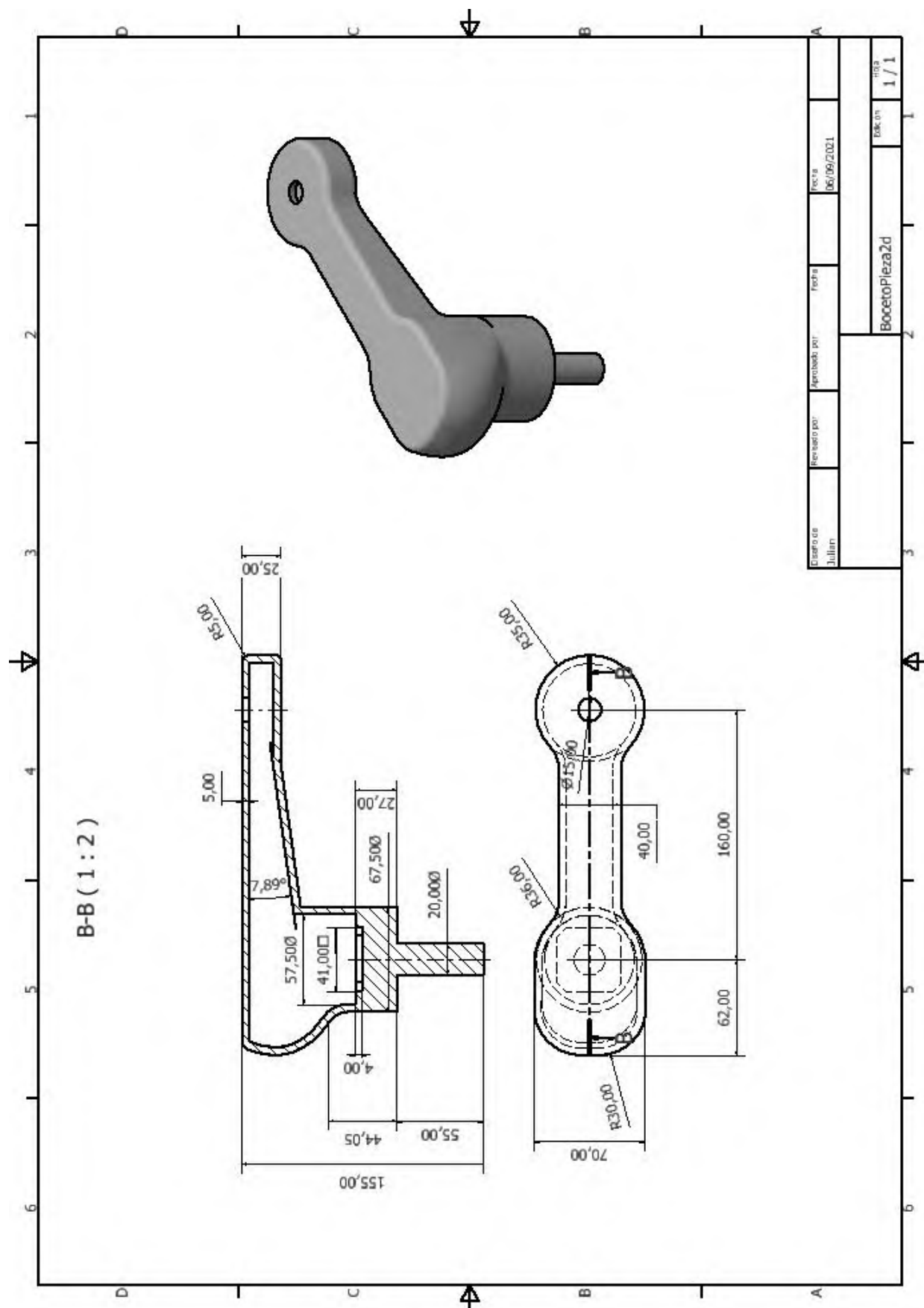
7 ANEXOS

7.1 PLANOS DE LOS ELEMENTOS DISEÑADOS

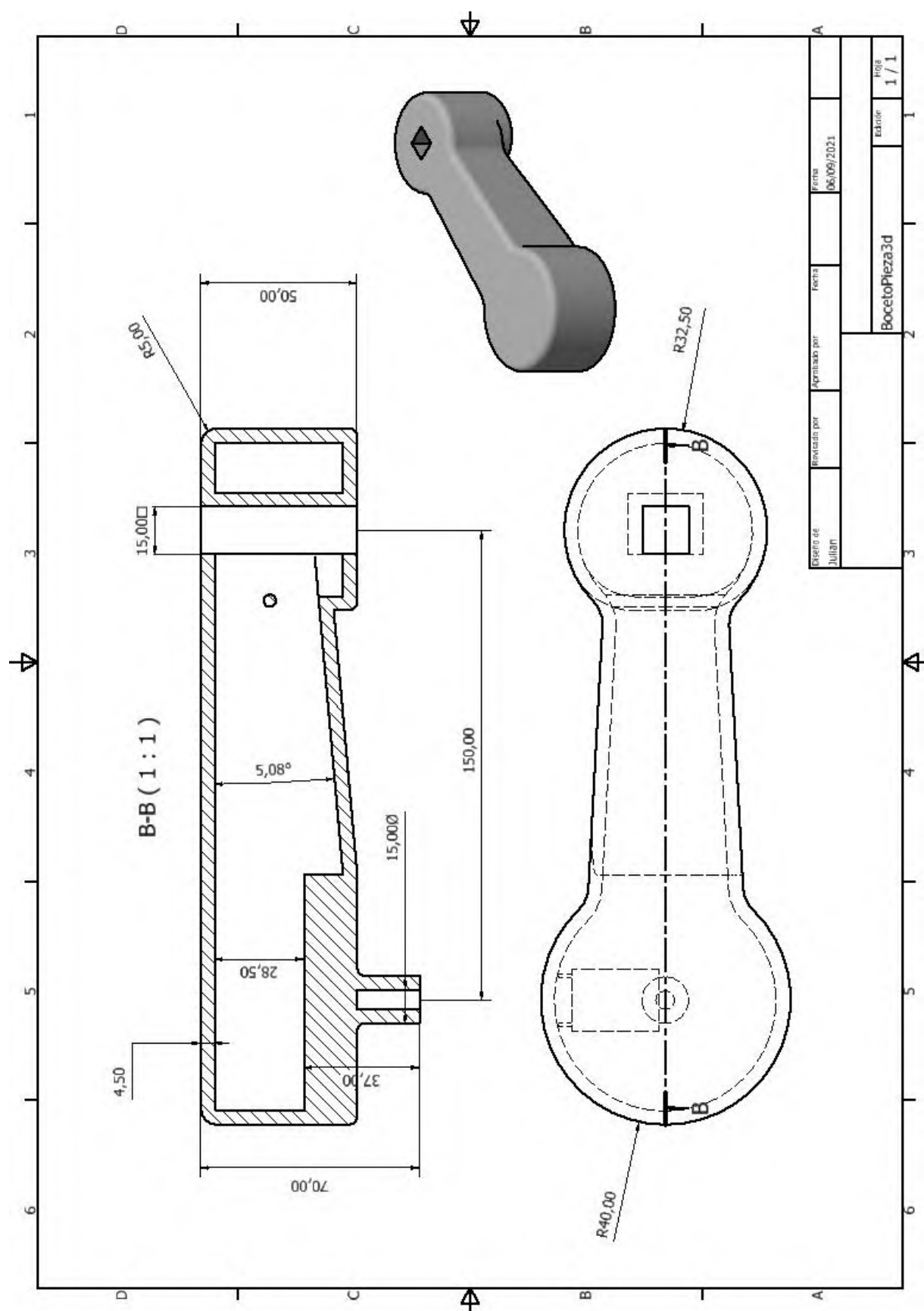
7.1.1 Base



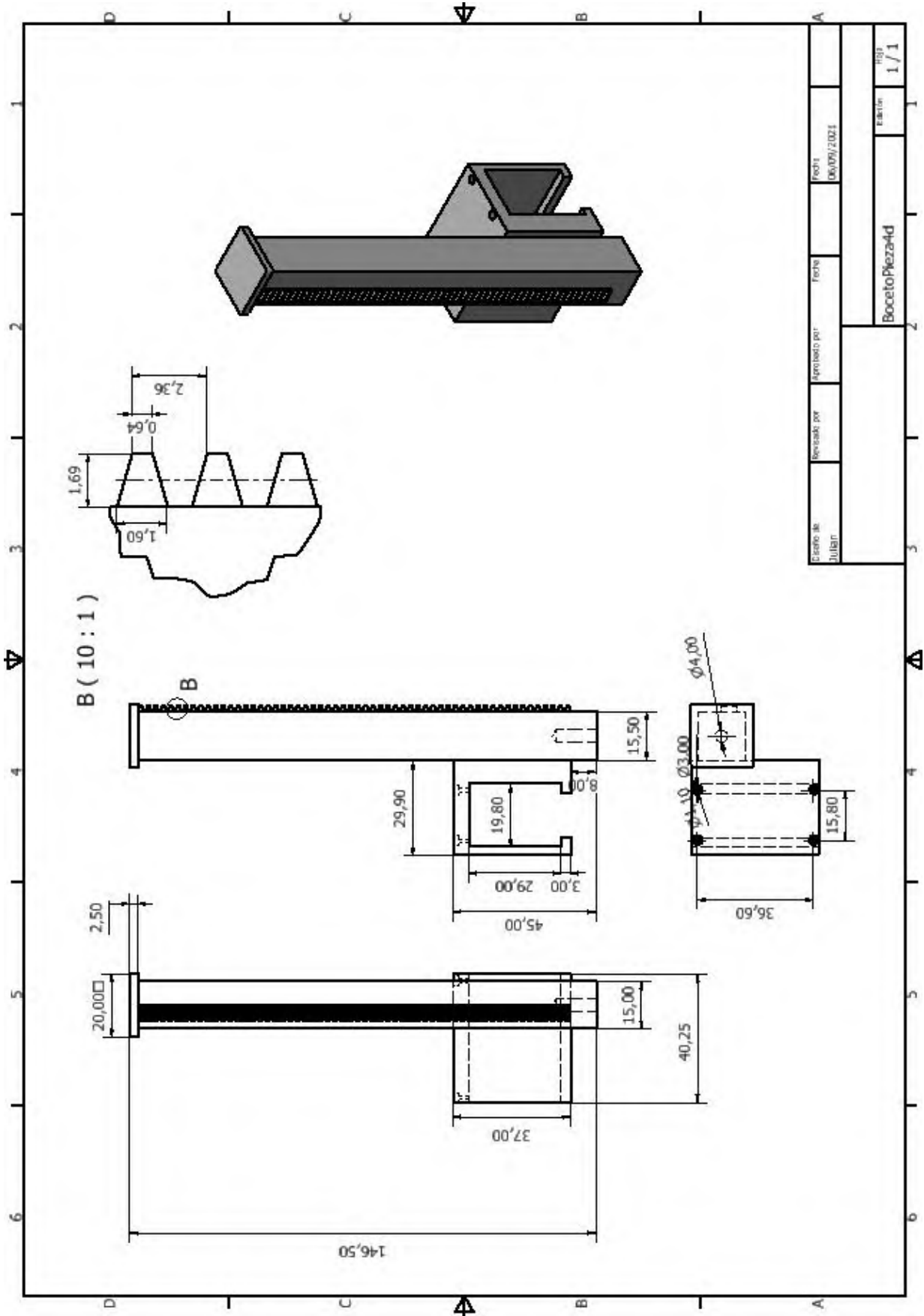
7.1.2 Elemento 2



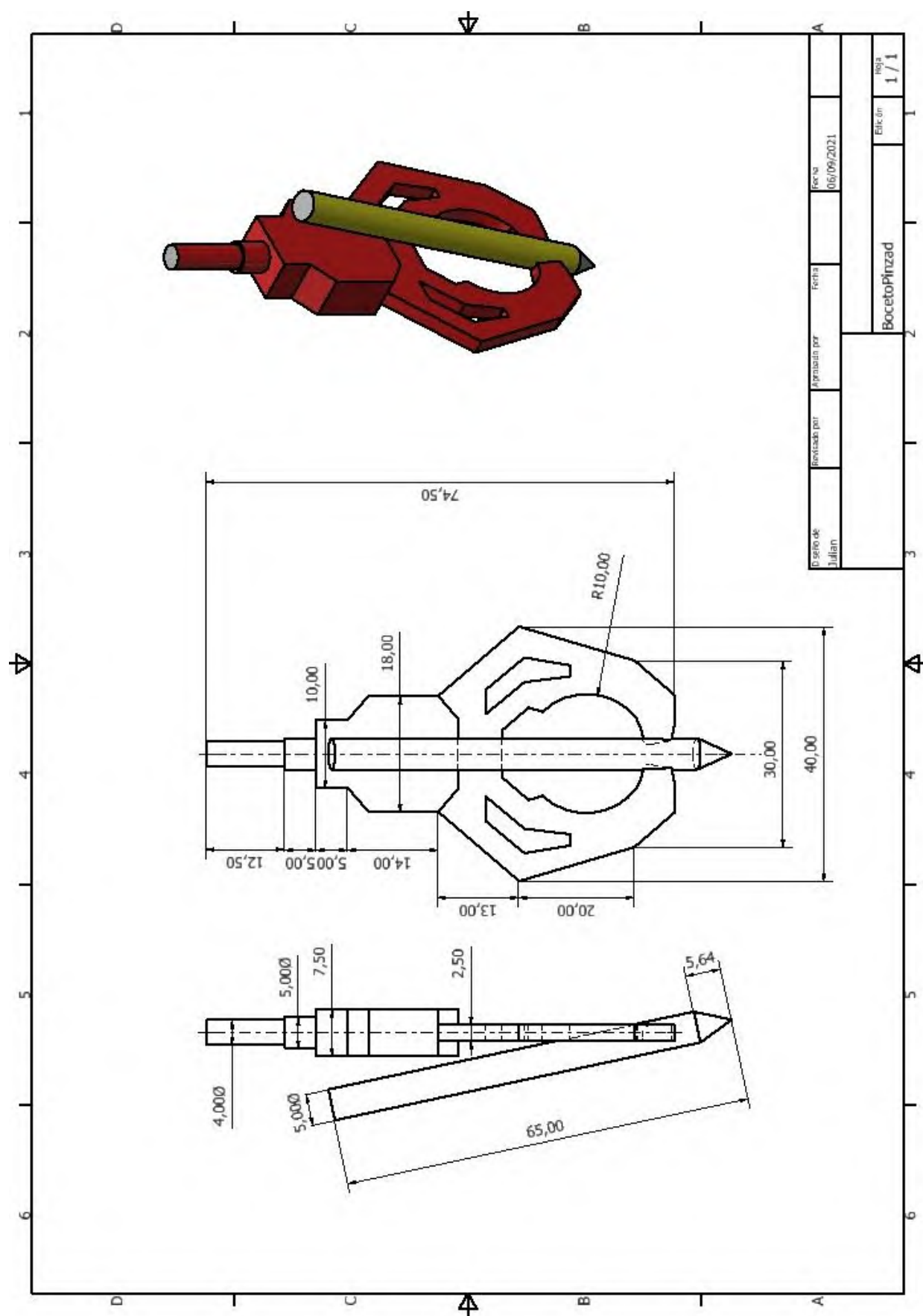
7.1.3 Elemento 3



7.1.4 Elemento 4



7.1.5 Pinza

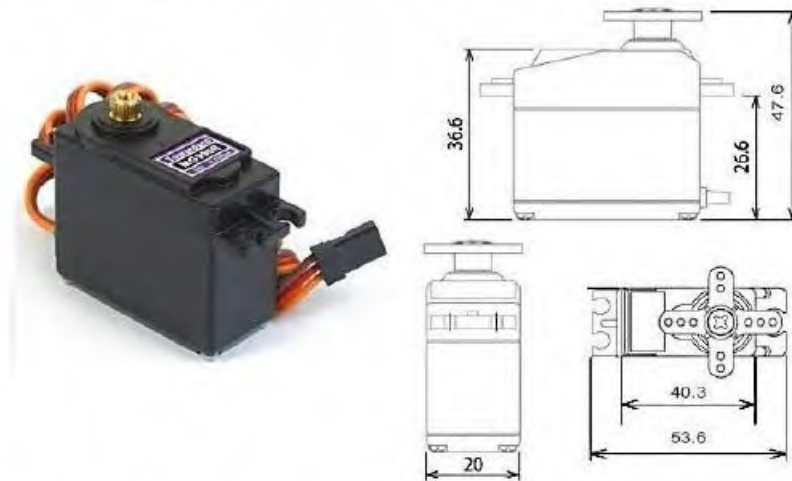


Nema 17:



MG996R:

MG996R High Torque Metal Gear Dual Ball Bearing Servo



This High-Torque MG996R Digital Servo features metal gearing resulting in extra high 10kg stalling torque in a tiny package. The MG996R is essentially an upgraded version of the famous MG995 servo, and features upgraded shock-proofing and a redesigned PCB and IC control system that make it much more accurate than its predecessor. The gearing and motor have also been upgraded to improve dead bandwith and centering. The unit comes complete with 30cm wire and 3 pin 'S' type female header connector that fits most receivers, including Futaba, JR, GWS, Cirrus, Blue Bird, Blue Arrow, Corona, Berg, Spektrum and Hitec.

This high-torque standard servo can rotate approximately 120 degrees (60 in each direction). You can use any servo code, hardware or library to control these servos, so it's great for beginners who want to make stuff move without building a motor controller with feedback & gear box, especially since it will fit in small places. The MG996R Metal Gear Servo also comes with a selection of arms and hardware to get you set up nice and fast!

Specifications

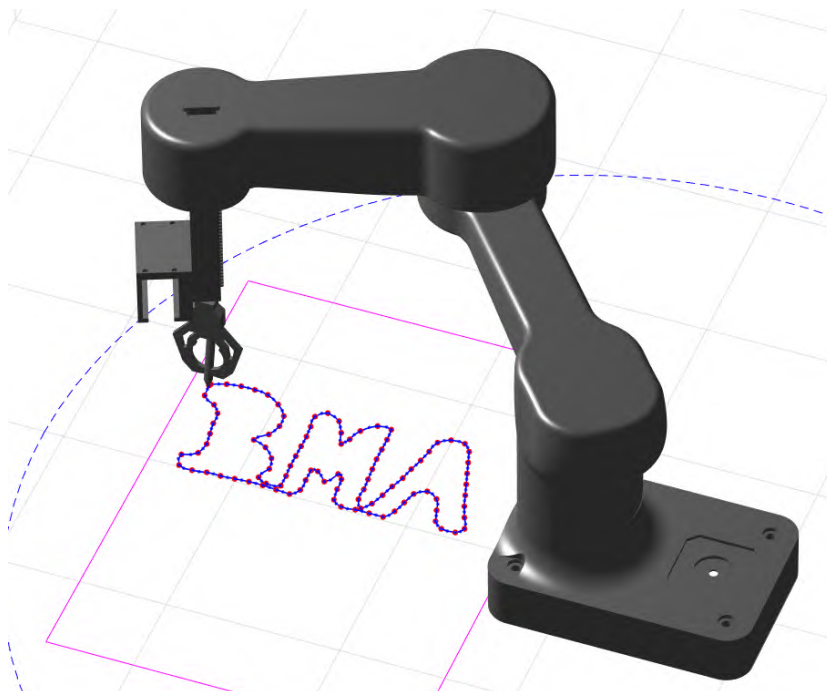
- Weight: 55 g
- Dimension: 40.7 x 19.7 x 42.9 mm approx.
- Stall torque: 9.4 kgf·cm (4.8 V), 11 kgf·cm (6 V)
- Operating speed: 0.17 s/60° (4.8 V), 0.14 s/60° (6 V)

7.3 MÁS EJEMPLOS DE TRAYECTORIAS REALIZADAS

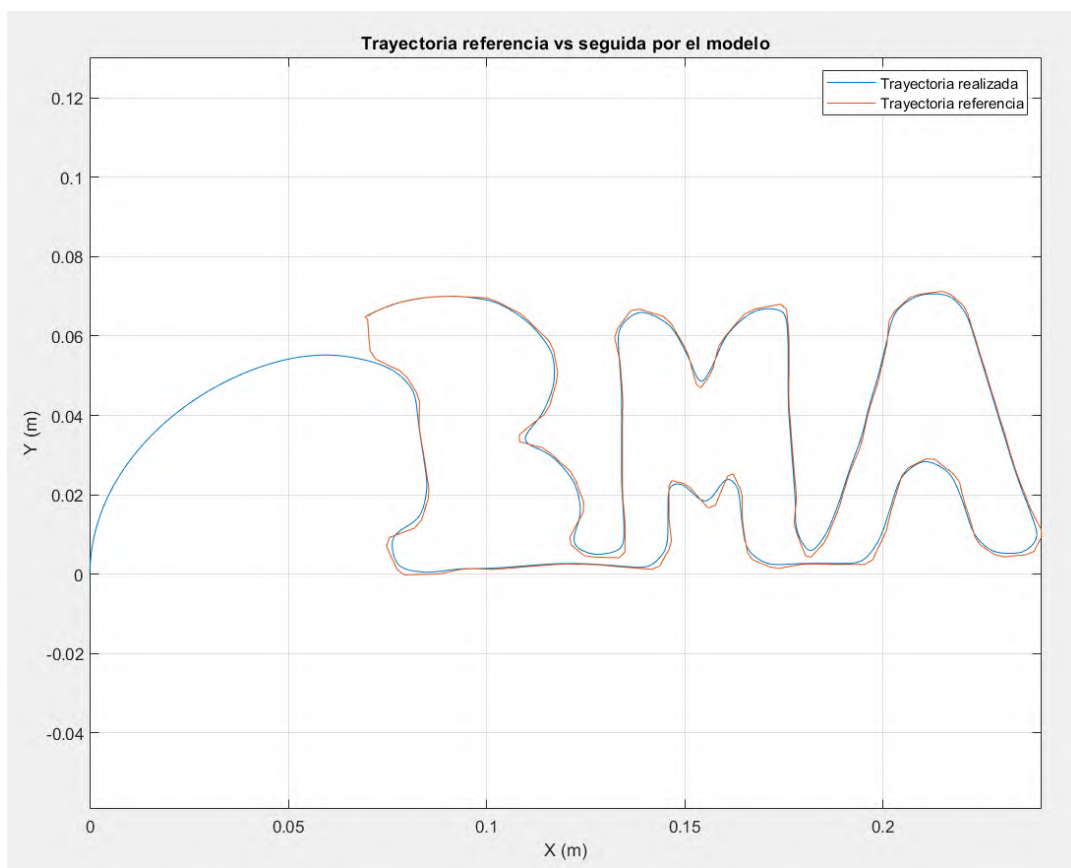
Imagen referencia BMA.jpeg:



Simulación cinemática manipulador:

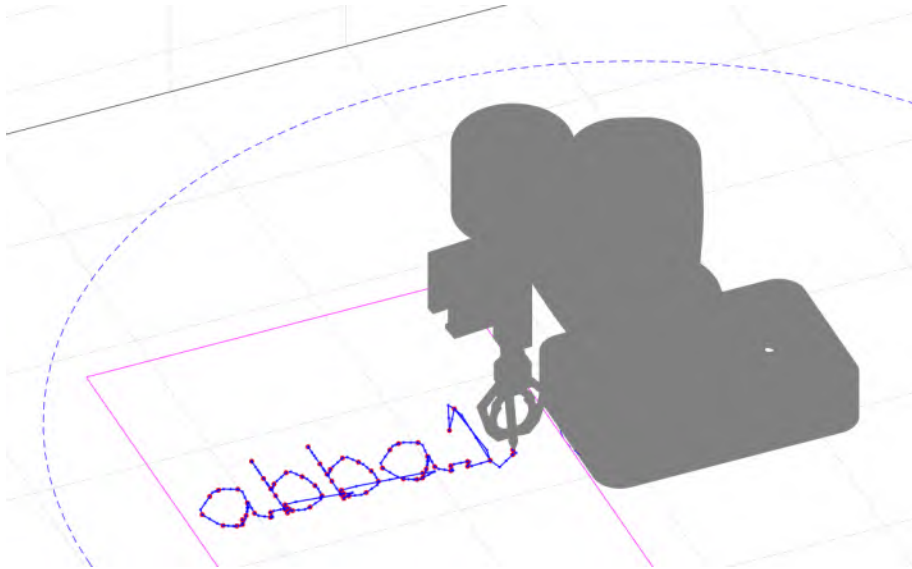


Simulación dinámica manipulador:



Caracteres de referencia: 'abba1'

Simulación cinemática manipulador.



Simulación dinámica manipulador.

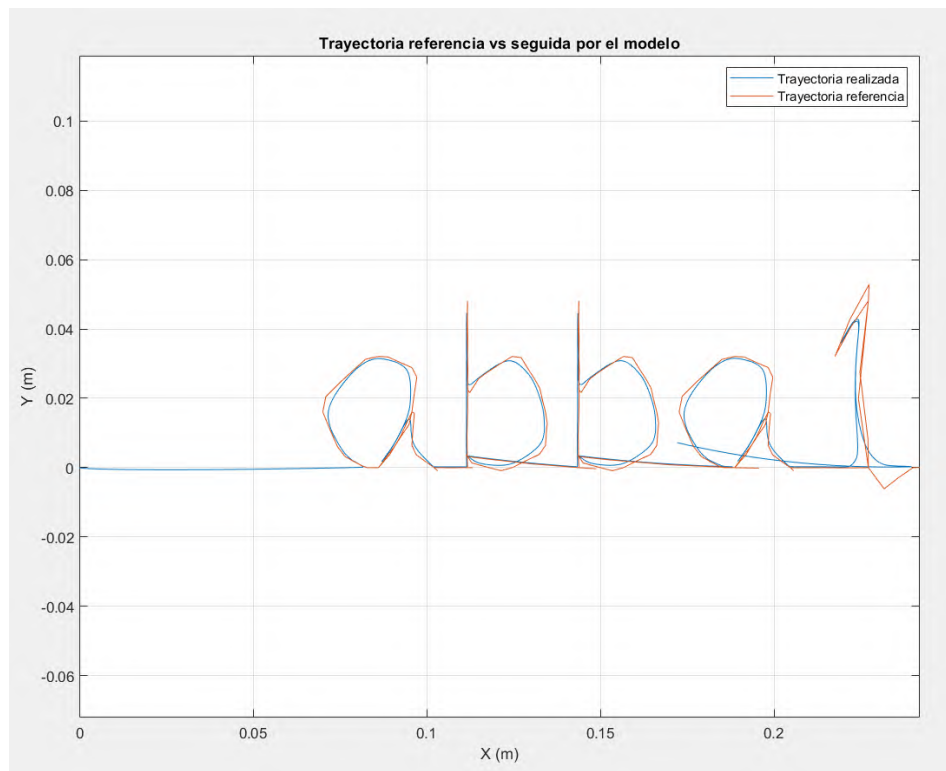
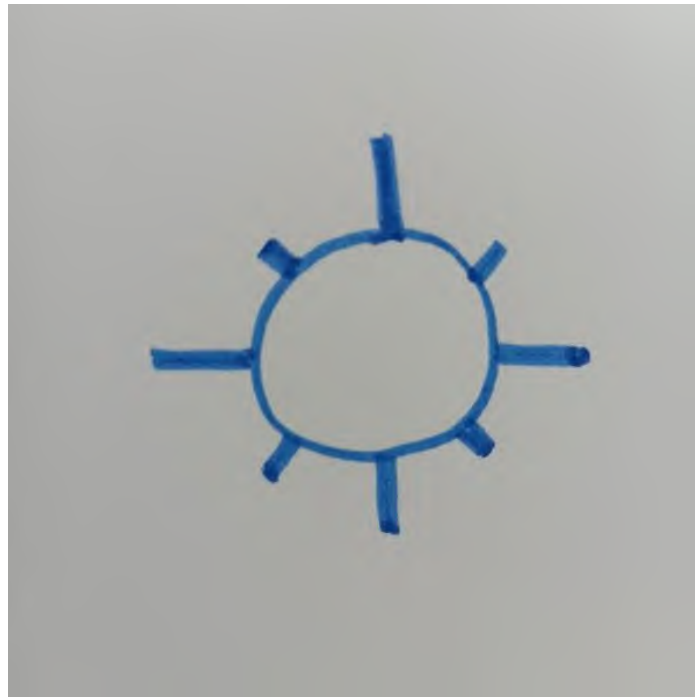
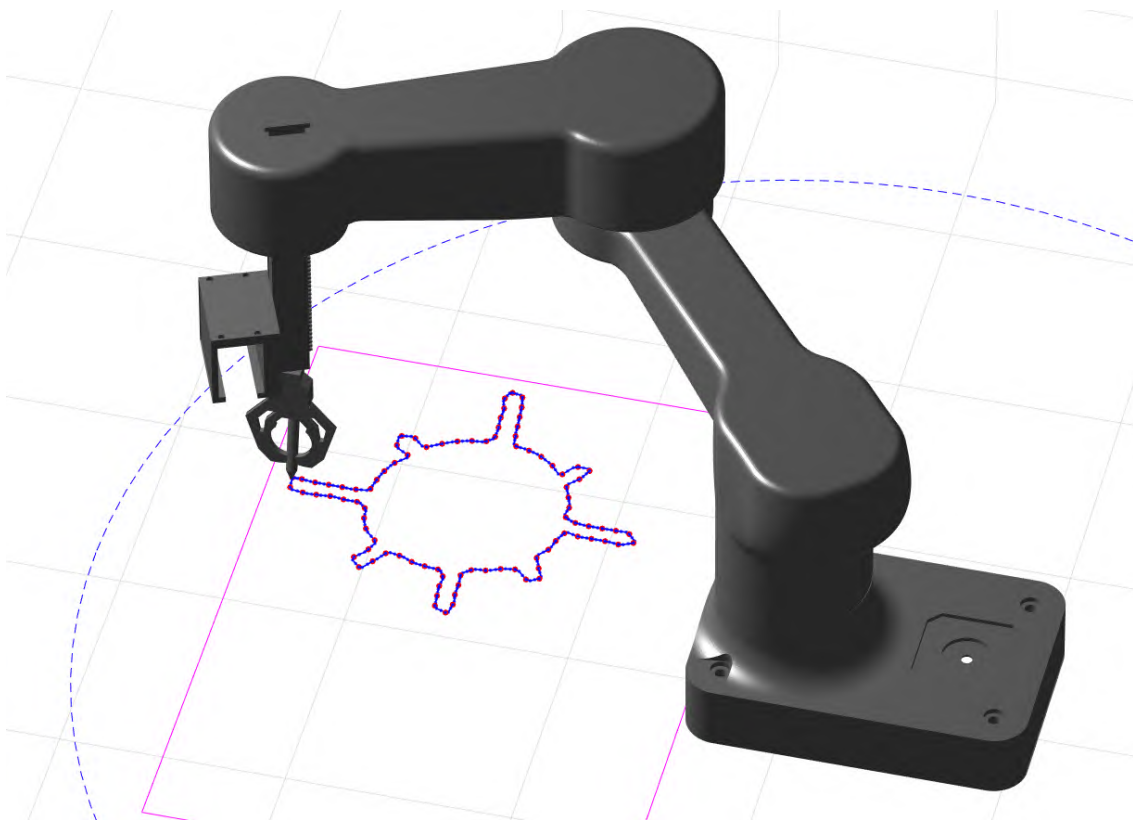


Imagen de referencia Sol.jpg:



Simulación cinemática del manipulador.



Simulación dinámica del manipulador.

