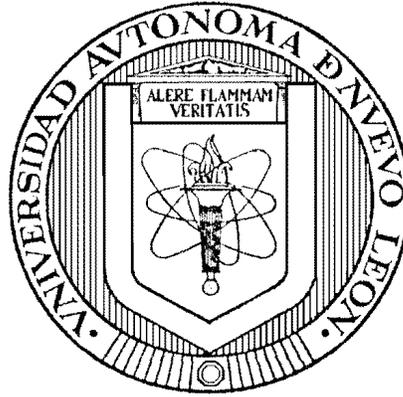


UNIVERSIDAD AUTÓNOMA DE NUEVO LEÓN
FACULTAD DE INGENIERÍA MECÁNICA Y ELÉCTRICA
DIVISIÓN DE ESTUDIOS DE POSGRADO



DISEÑO DE MANIPULADORES BASADO EN MÉTODOS ITERATIVOS E
ÍNDICES DE DESEMPEÑO

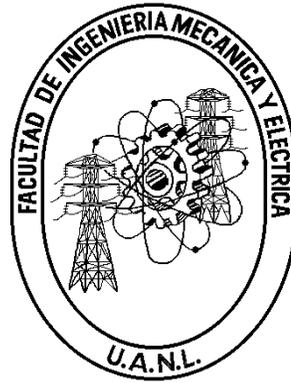
POR
GUSTAVO GONZÁLEZ SANMIGUEL

TESIS
EN OPCIÓN AL GRADO DE MAESTRO EN CIENCIAS DE LA
INGENIERÍA ELÉCTRICA CON ORIENTACIÓN
EN CONTROL AUTOMÁTICO

CD. UNIVERSITARIA

FEBRERO 2014

UNIVERSIDAD AUTÓNOMA DE NUEVO LEÓN
FACULTAD DE INGENIERÍA MECÁNICA Y ELÉCTRICA
DIVISIÓN DE ESTUDIOS DE POSGRADO



DISEÑO DE MANIPULADORES BASADO EN MÉTODOS ITERATIVOS E
ÍNDICES DE DESEMPEÑO

POR
GUSTAVO GONZÁLEZ SANMIGUEL

TESIS
EN OPCIÓN AL GRADO DE MAESTRO EN CIENCIAS DE LA
INGENIERÍA ELÉCTRICA CON ORIENTACIÓN
EN CONTROL AUTOMÁTICO

CD. UNIVERSITARIA

FEBRERO 2014

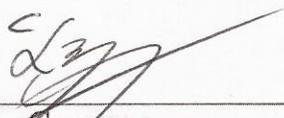
UNIVERSIDAD AUTÓNOMA DE NUEVO LEÓN
FACULTAD DE INGENIERÍA MECÁNICA Y ELÉCTRICA
DIVISIÓN DE ESTUDIOS DE POSGRADO

Los miembros del comité de tesis recomendamos que la tesis "Diseño de manipuladores basado en métodos iterativos e índices de desempeño" realizada por el alumno Gustavo González Sanmiguel, con matrícula 1609594, sea aceptada por su defensa como opción al grado de Maestro en Ciencias de la Ingeniería Eléctrica con Orientación en Control Automático.

El Comité de Tesis



Asesor
Dr. Juan Angel Rodríguez Liñán



Coasesor
Dr. Luis Martín Torres Treviño



Coasesor
Dr. Ismael López Juárez

Vo. Bo.
Dr. Mosiés Hinojosa Rivera
Subdirector de Posgrado

Cd. Universitaria, Febrero 2014

Agradezco a Dios, por darme la oportunidad de vivir y por estar conmigo en cada paso que doy. Por haberme dado sabiduría, fortaleza, salud, coraje, y no dejarme solo en los momentos difíciles, y haberme permitido llegar a la meta en este gran proyecto

Mi madre y padre por darme la vida, quererme mucho, creer siempre en mi y apoyarme. Por ser los pilares más importantes y por demostrarme siempre su cariño y apoyo incondicional sin importar nuestras diferencias de opiniones.

A mi hermana por siempre ser un gran ejemplo, al esforzarse siempre al máximo en todo lo que hace.

A mi novia por siempre apoyarme en el transcurso de este camino y brindarme su cariño incondicional.

A mi asesor por su amabilidad, buena disposición, paciencia, por el tiempo que me dedicó para que este trabajo culminara exitosamente.

Gustavo González Sanmiguel

Agradecimientos

Al CONACYT por su apoyo económico con su beca así como a la Universidad Autónoma de Nuevo León y a la Facultad de Ingeniería Mecánica y Eléctrica por darme la oportunidad de estudiar y ser un profesional.

Al Dr. Juan Ángel Rodríguez, Dr. César Guerra, Dr. Luis Torres y Dr. David A. Díaz por sus valiosos comentarios que enriquecieron el contenido de mi tesis, así como mi crecimiento como persona en el transcurso de estos dos años, por la amabilidad con la que me ayudaron y orientaron.

A todos los profesores del Programa Doctoral por compartir abiertamente sus conocimientos durante mis estudios, especialmente al Dr. César Elizondo, Dr. Marco Tulio Mata, Dr. Antonio de la O, Dr. Cornelio Posadas, Dr. Ignacio Ayala.

A todos mis compañeros del Doctorado de Eléctrica, Luis Lauro González, Shih Meng Yen, Miguel Angel Flores, Allan Soriano, Daniel Enrique Rivas, Omar Donjuan, Rubén Hernández, Jorge Eduardo Valdez.

A todos mis maestros de carrera, especialmente a Carlos Lara, Sergio Valderrabano, Hiram Cabrera, Juan Antonio Rojas, Federico Hernández, Raúl Alvarado Coiffier.

Resumen

En la actualidad existen diversas tareas que realizan robots manipuladores en la industria, tales como: ensamble, soldadura, pintura, movimiento de sustancias químicas o radioactivas, e incluso en áreas más delicadas como la industria alimentaria y cirugías quirúrgicas.

Sin embargo, a pesar de existir una gran diversidad de tareas industriales, en la mayoría de éstas sólo se usa una estructura de robot, la cual es el robot serial de 6 grados de libertad (tipo angular o antropomórfico). Existen otras estructuras de robots, así como una gran cantidad de dimensionamientos de cada estructura que hipotéticamente podrían ejecutar las diferentes tareas de manera más eficiente.

Debido a lo anterior en este trabajo se presenta una metodología de diseño en la cual se consideran índices de desempeño cinetostáticos y dinámicos, con el objetivo de analizar la estructura del robot (cinemática, dinámica y dimensiones) que realiza mejor una actividad específica.

Este enfoque da la ventaja que se optimice el diseño de un robot para una tarea específica, especializándolo con el fin de que realice mejor la tarea que algún otro arbitrariamente elegido, como la configuración angular, SCARA, cilíndrico, etcétera, así como un dimensionamiento óptimo lo optimizan para la realización de la tarea.

El objetivo de este trabajo es la generación de dicha metodología en un algoritmo que pueda realizar todo el proceso, y que al ejecutarlo no sea necesario realizar excesivos cálculos analíticos o numéricos, los cuales pueden ser muy áridos y lentos.

Símbolos y notación

\dot{x}, \dot{y}	El punto es utilizado para denotar la derivada con respecto del tiempo.
A^T, x^T	La transpuesta de la matriz A y el vector x .
\mathbb{R}	El campo de los números reales.
\mathbb{C}	El campo de los números complejos.
\in	Elemento
∞	Infinito.
\triangleq	Igual por definición.
\approx	Aproximadamente igual.
\equiv	Idéntico.
$dist\{x, y\}$	La distancia entre los puntos x y y , en algún espacio definido.
$\ x\ $	Una norma de x .
A^{-1}	Inversa de A .
j	Unidad de los números imaginarios, $j = \sqrt{-1}$.
$\partial F / \partial x$	Derivada parcial de la función F con respecto a la variable x .
$tr(A)$	Traza de A
P	Vector de Traslación
R	Matriz de rotación
q_i	Variable de desplazamiento o rotación de la articulación i
μ	Rotación ejercida sobre el eje x
β	Rotación ejercida sobre el eje y
γ	Rotación ejercida sobre el eje z
A_{i-1}^1	Matriz de transformacion homogénea
θ_i	Variable de rotación de metodología DH
d_i	Variable de traslación de metodología DH
a_i	Variable de traslación de metodología DH
α_i	Variable de rotación de metodología DH
J	Matriz Jacobiana
τ	Vector de fuerzas o torques
M (q)	Matriz de inercias
C (q, \dot{q})	Matriz de coriolis
G (q)	Matriz de efectos por gravedad
CN	Índice de desempeño: número de condición
FM	Índice de desempeño: manipulabilidad de fuerza
KM	Índice de desempeño: manipulabilidad cinemática
SV	Índice de desempeño: valores singulares
Wd	Índice de desempeño: manipulabilidad dinámica
WI	Índice de desempeño: isotropía dinámica

Índice General

Agradecimientos	iv
Resumen	v
Símbolos y notación	vi
Índice General	vii
Índice de Tablas	ix
Índice de Figuras	x
1 Introducción	1
1.1 Antecedentes	1
1.2 Motivación	2
1.3 Planteamiento del problema	3
1.4 Organización y alcances del trabajo	4
2 Análisis de manipuladores	5
2.1 Objetivo	5
2.2 Cinemática de manipuladores	5
2.2.1 Metodología Denavit - Hartenberg	6
2.2.2 Cinemática inversa	11
2.2.3 Modelo cinemático de velocidad	12
2.2.4 Modelo cinemático de velocidad inverso	13
2.2.5 Homogenización de la matriz jacobiana	16
2.3 Dinámica de manipuladores	19
2.3.1 Metodología Newton-Euler	20
2.4 Índices de desempeño	23
2.4.1 Índices de desempeño cinetostático	24
2.4.2 Índices de desempeño dinámico	29
2.5 Conclusiones del capítulo	31
3 Diseño iterativo de manipuladores	32
3.1 Introducción	32
3.2 Algoritmos evolutivos multiobjetivo	33

3.2.1	Clasificación de algoritmos evolutivos	34
3.2.2	EvoNorm	36
3.3	Algoritmo generador de parámetros para robots	38
3.3.1	Generación de población	38
3.3.2	Función de evaluación	39
3.3.3	Proceso de reproducción	41
3.4	Conclusiones de Capítulo	42
4	Casos de estudio	43
4.1	Introducción	43
4.2	Comparación y selección de manipuladores predeterminados	44
4.2.1	Desempeño de robot angular de 3 grados de libertad	46
4.2.2	Desempeño de robot serial SCARA de 4 grados de libertad	49
4.2.3	Desempeño de robot angular de 6 grados de libertad	51
4.2.4	Comparación y selección de manipuladores	53
4.3	Manipulador diseñado para optimizar los índices de desempeño en una tarea propuesta	54
4.3.1	Dimensionamiento de manipulador angular de 3 grados de libertad generado por algoritmo evolutivo	54
4.3.2	Dimensionamiento de robot SCARA generado por algoritmo evolutivo	56
4.3.3	Dimensionamiento de robot 6DOF generado por algoritmo evolutivo	57
4.3.4	Comparación y selección de manipuladores	58
4.4	Manipulador para soldadura en superficies de tubulares	59
4.5	Conclusiones de Capítulo	64
	Conclusiones y problemas abiertos	65
	Bibliografía	67
A	Modelo dinámico y cinemático de manipuladores utilizados en experimentos	69
A.1	Manipulador angular serial de 3 grados de libertad	70
A.2	Manipulador SCARA de 4 grados de libertad	73
A.3	Manipulador angular de 6 grados de libertad	75

Índice de Tablas

4.2.1	Tabla de posiciones cartesianas por las que debe pasar el efector final del robot.	44
4.2.2	Parámetros del manipulador angular 3 GDL.	45
4.2.3	Parámetros del manipulador SCARA.	45
4.2.4	Parámetros del manipulador de 6 GDL.	46
4.2.5	Cinemática inversa del manipulador angular de 3 GDL.	48
4.2.6	Resultados de índices de desempeño del manipulador angular de 3 GDL.	49
4.2.7	Cinemática inversa de manipulador SCARA.	50
4.2.8	Resultados de índices de desempeño del manipulador SCARA.	51
4.2.9	Matriz con resultados de cinemática inversa de la tarea.	52
4.2.10	Matriz con resultados de índices de desempeño del manipulador angular de 6 GDL.	53
4.2.11	Tabla comparando el desempeño de los robots en las posiciones asignadas de la tarea por el usuario.	53
4.3.12	Parámetros del manipulador de 3 GDL Evolutivo.	55
4.3.13	Resultados de cinemática inversa manipulador angular de 3DOF.	55
4.3.14	Resultados de índices de desempeño del manipulador angular de 3 GDL.	55
4.3.15	Matriz de datos robot SCARA.	56
4.3.16	Resultados de cinemática inversa manipulador SCARA.	56
4.3.17	Resultados de índices de desempeño del manipulador SCARA.	57
4.3.18	Parámetros del manipulador 6 GDL.	57
4.3.19	Resultados de cinemática inversa manipulador 6 GDL.	58
4.3.20	Resultado de índices de desempeño manipulador angular 6 GDL.	58
4.3.21	Tabla comparativa de índices de desempeño entre manipuladores.	59
4.4.22	Tabla de posiciones en el plano cartesiano por las que debe pasar el efector final del robot.	60
4.4.23	Parámetros del manipulador 6 GDL Fanuc.	61
4.4.24	Parámetros del manipulador 6 GDL generado por algoritmo.	61
4.4.25	Resultados de cinemática inversa manipulador 6 GDL.	62
4.4.26	Resultados de cinemática inversa manipulador 6 GDL generado por algoritmo.	62
4.4.27	Resultado de índices de desempeño manipulador angular 6 GDL Fanuc.	62
4.4.28	Resultado de índices de desempeño manipulador angular 6 GDL generado por algoritmo.	63
4.4.29	Tabla comparativa de índices de desempeño entre manipuladores.	63

Índice de Figuras

2.1	Parámetros DH para un eslabón giratorio.	10
4.1	Caso de soldadora sobre placa cuadrada con orientación libre del extremo final del manipulador.	45
4.2	Caso de soldadora sobre placa cuadrada con robot de 6GDL.	46
4.3	Caso de soldadora sobre placa cuadrada con robot SCARA.	47
4.4	Caso de robot soldador en superficie tubular.	60

Capítulo 1

Introducción

1.1 Antecedentes

En la actualidad existen diversas tareas que realizan los robots manipuladores en la industria, tales como ensamble de componentes electrónicos, ensamble automotriz, soldadura, pintado de piezas, trabajo de metales e industria alimentaria, y en años recientes más allá del sector industrial en cirugías quirúrgicas, lo cual requiere de un alto nivel de precisión y exactitud en la tarea que realiza.

En la industria, las principales tareas que realizan los manipuladores se pueden clasificar en los siguientes puntos [4]:

- Manipulación de objetos (transporte, paletizado, operaciones para el mecanizado de metal y moldeo de plástico).
- Operaciones con máquina y herramienta (corte, perforaciones, pulido). Soldadura por puntos y soldadura de arco).
- Ensamble (fijación, prensa de montaje, insertando, desmontaje).
- Actividades de dispensación (pintura, encolado, la aplicación de adhesivo, pulverización).

Para realizar estas actividades, las estructuras de manipuladores que se utilizan en la actualidad están distribuidas de la siguiente manera [2]:

- Angular o articulado (89.7 %)
- SCARA (7.7 %)
- Paralela (1.7 %)
- Cartesiano (0.7 %)
- Otros (0.2 %)

Sin embargo, para ciertas tareas es intuitivo que el tipo de robot utilizado no es el más adecuado, esto debido a que no existe una metodología firme de diseño que determine claramente que estructura de robot seleccionar, no poder seleccionar claramente la estructura del robot trae como consecuencia que la selección del tipo de manipuladores en la industria se base solamente en la demanda de ventas, procesos empíricos, en indicadores de producción, seguridad industrial; esto significa que no se explota un método de diseño o selección basado en indicadores del desempeño para realizar tareas más eficientemente, lo cual generaría estructuras mecánicas de manipuladores para mejorar su desempeño.

1.2 Motivación

La causa principal por la que se decidió realizar este tema, es poder contar con una metodología de selección de manipulador el cual pueda resolver la tarea que se busca que desempeñe de forma mas eficiente.

La evaluación sería mediante índices de desempeño cinetostáticos y dinámicos, los cuales miden la calidad con que el robot ejecuta la tarea. Además un algoritmo que proponga el

"mejor" robot que realiza la actividad, y obtener los parámetros de diseño del robot. De esta manera, el robot estaría diseñado optimamente para la realización de la tarea.

1.3 Planteamiento del problema

El objetivo de este trabajo es obtener un método de diseño que mejore el desempeño de los manipuladores, con esta finalidad, se propone que cada manipulador analizado sea representado por modelos cinemáticos y dinámicos, los cuales relacionan el comportamiento del robot. Sin embargo dichas ecuaciones son difíciles de manejar ya que son muy extensas cuando se incrementan los grados de libertad del robot. Esto complicando aún más los criterios de selección del manipulador mas adecuado para la actividad.

Denavit y Hartenberg [2] propusieron en 1955 un método matricial que permite establecer de manera sistemática un sistema de coordenadas ligado a cada eslabón de una cadena articulada, pudiéndose determinar a continuación las ecuaciones cinemáticas de la cadena completa. El método Newton-Euler propone la determinación de las fuerzas que deben ser aplicadas en las articulaciones para conseguir determinadas posiciones velocidades y aceleraciones [3].

Entre los índices de desempeño, están:

- Índices de desempeño cinetostático (Elipsoide de velocidad, número de condición, mínimo valor singular, manipulabilidad),
- Índices de desempeño dinámico (Matriz de inercia generalizada, manipulabilidad dinámica, maximización del volumen y uniformidad del hiperparalelepípedo de aceleración, condicionamiento dinámico) ,
- Índices de límites articulares (Límite de la articulación, evitación de límites articulares), disponibilidad de movilidad articular, índices definidos sobre el espacio de trabajo,

- Índices de espacio de trabajo (Tipos de espacios de trabajo, global de condicionamiento, global de manipulabilidad, radio de aceleración).

1.4 Organización y alcances del trabajo

El contenido de este trabajo está organizado de la siguiente forma: En el capítulo 2 se introduce el análisis cinemático y los índices de desempeño cinetostáticos. Después se presenta el análisis dinámico del robot, así como los índices de desempeño dinámico.

En el capítulo 3 se presenta la metodología que se utilizó para la generación del robot mediante un algoritmo de optimización, la cual se basó en algoritmos evolutivos. Así como el algoritmo de evaluación de manipuladores.

Posteriormente en el capítulo 4 se presentan los resultados obtenidos mediante casos de estudio, con los cuales se muestra cómo se seleccionaron y generaron robots para actividades específicas, así como el desempeño que mostraron, y las conclusiones de comparación.

Por último en el capítulo 5, se dan las conclusiones de este trabajo, recomendaciones, así como se plantean trabajos futuros.

Capítulo 2

Análisis de manipuladores

2.1 Objetivo

El análisis del robot manipulador consiste en la caracterización del movimiento de manipuladores seriales mediante sus modelos cinemáticos y dinámicos. Aunado a esto el análisis mediante índices de desempeño cinemáticos y dinámicos los cuales están basados en los modelos cinemáticos y dinámicos.

En base al resultado que muestran los índices de desempeño que evalúan al robot realizando una tarea, se podrá saber que tan calificado es el robot para realizar dicha tarea.

2.2 Cinemática de manipuladores

Para localizar un cuerpo rígido en el espacio es necesario contar con una herramienta matemática con la cual se pueda representar tanto la posición espacial, así como su orientación con respecto a un sistema de referencia.

Para poder resolver el problema de representar tanto la posición y la orientación conjuntamente por un sólo método[3] se introdujeron las coordenadas homogéneas. Esta se define como

matriz de transformación \mathbf{T} a la matriz de 4x4 que representa la transformación de un vector de coordenadas de un sistema a otro.

Estas matrices constan de cuatro elementos los cuales son:

$$\mathbf{T} = \begin{bmatrix} \mathbf{R}_{3 \times 3} & \mathbf{p}_{3 \times 1} \\ \mathbf{f}_{1 \times 3} & \mathbf{w}_{1 \times 1} \end{bmatrix}, = \begin{bmatrix} \textit{Rotación} & \textit{Traslación} \\ \textit{Perspectiva} & \textit{Escalado} \end{bmatrix}$$

Una matriz homogénea se encuentra compuesta por cuatro submatrices de distinto tamaño: $\mathbf{R}_{3 \times 3}$ Matriz de rotación, $\mathbf{p}_{3 \times 1}$ Matriz de traslación, $\mathbf{f}_{1 \times 3}$ Matriz de perspectiva, $\mathbf{w}_{1 \times 1}$ Matriz de escalamiento.

Estos son elementos de la matriz de transformación homogénea:

$$\mathbf{P} = \begin{bmatrix} x \\ y \\ z \end{bmatrix}, \quad \mathbf{R} = \begin{bmatrix} x_1 \cdot x_2 & x_1 \cdot y_2 & x_1 \cdot z_2 \\ y_1 \cdot x_2 & y_1 \cdot y_2 & y_1 \cdot z_2 \\ z_1 \cdot x_2 & z_1 \cdot y_2 & z_1 \cdot z_2 \end{bmatrix}, \quad \mathbf{f} = [f_1 \quad f_2 \quad f_3], \quad \mathbf{w} = [w],$$

En robótica generalmente sólo interesa conocer el valor de $\mathbf{R}_{3 \times 3}$ y de $\mathbf{f}_{1 \times 3}$, considerando $\mathbf{f}_{1 \times 3} = \mathbf{0}$ y $\mathbf{w}_{1 \times 1} = 1$.

2.2.1 Metodología Denavit - Hartenberg

La resolución del problema cinemático directo consiste en encontrar las relaciones que permiten conocer la localización espacial del extremo del robot a partir de los valores de sus coordenadas articulares q_i [3].

Se han escogido coordenadas cartesianas $[P_x \ P_y \ P_z]^T \in \mathbb{R}^3$ y ángulos RPY (Roll-Pitch-Yaw) de Euler $[\mu \ \beta \ \gamma]^T \in \mathbb{R}^3$ para representar la posición y orientación del extremo de un robot de n

grados de libertad, la solución al problema cinemático directo vendrá dada por las relaciones:

$$\begin{aligned} P_x &= f_x(q_1, q_2, \dots, q_n) \\ P_y &= f_y(q_1, q_2, \dots, q_n) \\ P_z &= f_z(q_1, q_2, \dots, q_n) \\ \mu &= f_\alpha(q_1, q_2, \dots, q_n) \\ \beta &= f_\beta(q_1, q_2, \dots, q_n) \\ \gamma &= f_\gamma(q_1, q_2, \dots, q_n) \end{aligned}$$

En general, un robot manipulador serial de n grados de libertad está formado por n eslabones unidos por n articulaciones, de forma que cada par de articulación - eslabón constituye un grado de libertad. A cada eslabón se le puede asociar un sistema de referencia solidario a él y, utilizando transformaciones homogéneas, es posible representar las rotaciones y traslaciones relativas entre los distintos eslabones que componen al robot. Denomínese A_{i-1}^i a la matriz de transformación homogénea que representa la posición (P_x, P_y, P_z) y orientación (μ, β, γ) relativa entre los sistemas asociados a dos eslabones consecutivos del robot. De tal modo que la transformación homogénea del origen al sistema coordenado solidario al efector final se obtiene por la composición de cada transformación entre dos sistemas coordenados de eslabones consecutivos, es decir:

$$A_0^n = A_0^1 A_1^2 \dots A_{n-1}^n$$

Para describir la relación que existe entre dos elementos contiguos se suele utilizar en robótica la representación de Denavit - Hartenberg (DH) que consiste en un método matricial que permite establecer de manera sistemática un sistema de coordenadas ligado a cada eslabón i de una cadena articulada, pudiéndose determinar a continuación las ecuaciones cinemáticas de la cadena completa. Según la representación de DH, escogiendo adecuadamente los sistemas de coordenadas asociados a cada eslabón, es posible pasar de uno al siguiente mediante 4 transformaciones básicas que dependen exclusivamente de las características geométricas del eslabón. Estas transformaciones básicas consisten en una sucesión de rotaciones y traslaciones que permiten relacionar el sistema de referencia del elemento i con el sistema del elemento $i - 1$.

Algoritmo de Denavit - Hartenberg [3]

El algoritmo Denavit - Hartenberg se utiliza para la determinación de los sistemas coordenados asociados a cada eslabón, el cual depende de 4 transformaciones básicas las cuales se calculan con el siguiente algoritmo:

1. Numerar los eslabones comenzando con 1 (primer eslabón móvil de la cadena) y acabando con n (último eslabón móvil). Se enumerará como eslabón 0 a la base fija del robot.
2. Numerar cada articulación comenzando por 1 (la correspondiente al primer grado de libertad) y acabando en n .
3. Localizar el eje de cada articulación. Si ésta es rotativa, el eje será su propio eje de giro. Si es prismática, será el eje a lo largo del cual se produce el desplazamiento.
4. Para i de 0 a $n - 1$ situar el eje z_i sobre el eje de la articulación $i + 1$.
5. Situar el origen del sistema de la base S_0 en cualquier punto del eje z_0 . Los ejes x_0 e y_0 se situarán de modo que formen un sistema dextrógiro con x_0 .
6. Para i de 1 a $n - 1$, situar el sistema S_i (solidario al eslabón i) en la intersección del eje z_i con la línea normal común a z_{i-1} y z_i . Si ambos ejes se cortasen se situaría S_i en el punto de corte. Si fuesen paralelos S_i se situaría en la articulación $i + 1$.
7. Situar x_i en la línea normal común a x_{i-1} y z_i .
8. Situar y_i de modo que forme un sistema dextrógiro con x_i con x_i y z_i .
9. Situar el sistema S_n en el extremo del robot de modo que z_n coincida con la dirección de z_{n-1} y x_n sea normal a z_{n-1} y z_n .
10. Obtener θ_i como el ángulo que hay que girar en torno a z_{i-1} , para que x_{i-1} y x_i queden paralelos.

11. Obtener d_i como la distancia, medida a lo largo de z_{i-1} , que habría que desplazar S_{i-1} para que x_i y x_{i-1} quedasen alineados.
12. Obtener a_i como la distancia medida a lo largo de x_i (que ahora coincidiría con x_{i-1}) que habría que desplazar el nuevo S_{i-1} .
13. Obtener α_i como el ángulo que habría que girar entorno a x_i (que ahora coincidiría con x_{i-1}), para que el nuevo S_{i-1} coincidiese totalmente con S_i .
14. Obtener las matrices de transformación \mathbf{A}_{i-1}^i :

$$\mathbf{A}_{i-1}^i = \mathbf{T}(z, \theta_i) \mathbf{T}(0, 0, d) \mathbf{T}(a_i, 0, 0) \mathbf{T}(x, \alpha_i)$$

Lo cual realizando el producto entre matrices resulta:

$$\begin{aligned} \mathbf{A}_{i-1}^i &= \begin{bmatrix} C\theta_i & -S\theta_i & 0 & 0 \\ S\theta_i & C\theta_i & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & d_i \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & a_i \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & C\alpha_i & -S\alpha_i & 0 \\ 0 & S\alpha_i & C\alpha_i & 1 \\ 0 & 0 & 0 & 1 \end{bmatrix} = \\ &= \begin{bmatrix} C\theta_i & -C\alpha_i S\theta_i & S\alpha_i S\theta_i & a_i C\theta_i \\ S\theta_i & C\alpha_i C\theta_i & -S\alpha_i C\theta_i & a_i S\theta_i \\ 0 & S\alpha_i & C\alpha_i & d_i \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (2.2.1) \end{aligned}$$

Donde Cw y Sw representan la función $\cos(w)$ y $\sin(w)$ respectivamente.

15. Obtener la matriz de transformación que relaciona el sistema de la base con el del extremo del robot $T = A_0^1 A_1^2 \dots A_{n-1}^n$.
16. La matriz T define la orientación (submatriz de rotación) y posición (submatriz de traslación) del extremo referido a la base en función de las n coordenadas articulares.

Los cuatro parámetros de DH ($\theta_i, d_i, a_i, \alpha_i$) dependen únicamente de las características geométricas de cada eslabón y de las articulaciones que le unen con el anterior y siguiente. En concreto y como se aprecia en la figura 2.1, estos parámetros representan:

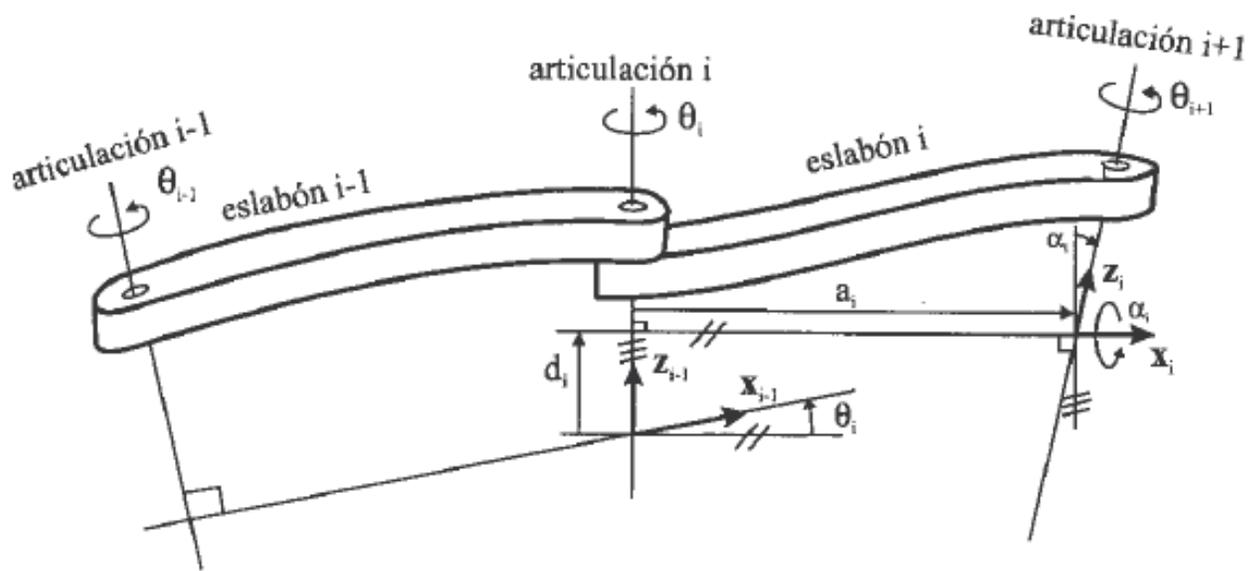


Figura 2.1: Parámetros DH para un eslabón giratorio.

- θ_i El ángulo que forman los ejes x_{i-1} y x_i medido en un plano perpendicular al eje z_{i-1} , utilizando la regla de la mano derecha. Se trata de un parámetro variable en articulaciones giratorias.
- d_i La distancia a lo largo del eje z_{i-1} desde el origen del sistema de coordenadas $(i-1)$ -ésimo hasta la intersección del eje z_{i-1} con el eje x_i . Se trata de un parámetro variable en articulaciones prismáticas.
- a_i La distancia a lo largo del eje x_i que va desde la intersección del eje z_{i-1} hasta el origen del sistema i -ésimo, en el caso de articulaciones giratorias. En el caso de articulaciones prismáticas, se calcula como la distancia más corta entre los ejes z_{i-1} y z_i .
- α_i El ángulo de separación del eje z_{i-1} y el eje z_i , medido en un plano perpendicular al eje x_i , utilizando la regla de la mano derecha.

Finalmente, con la matriz de transformación obtenemos las posición y rotación del efector final del robot.

$$T = \begin{bmatrix} T_{1,1} & T_{1,2} & T_{1,3} & T_{1,4} \\ T_{2,1} & T_{2,2} & T_{2,3} & T_{2,4} \\ T_{3,1} & T_{3,2} & T_{3,3} & T_{3,4} \\ T_{4,1} & T_{4,2} & T_{4,3} & T_{4,4} \end{bmatrix}$$

$$P_x = T_{1,4}, \quad P_y = T_{2,4}, \quad P_z = T_{3,4},$$

$$\mu = \text{atan} \left(\frac{T_{3,2} \cos(\beta)}{T_{3,3} \cos(\beta)} \right), \quad \beta = \text{atan} \left(\frac{-T_{3,1}}{\sqrt{T_{1,1}^2 + T_{2,1}^2}} \right), \quad \gamma = \text{atan} \left(\frac{T_{2,1} \cos(\mu)}{(T_{1,1} \cos(\mu))} \right)$$

2.2.2 Cinemática inversa

La cinemática inversa consiste en encontrar los valores que deben adoptar las coordenadas articulares del robot $q = q_1, q_2, \dots, q_n^T$ para que su extremo se posicione y oriente según una determinada localización espacial.

Existen muchos métodos para la solución de la cinemática inversa, entre ellos:

- Métodos geométricos
- Solución a partir de matriz de transformación homogénea
- Desacoplo cinemático
- Métodos numéricos
- Algoritmos evolutivos

En el caso de este trabajo se seleccionó la utilización de algoritmos evolutivos como solución al problema inverso, ya que este presenta la posibilidad de poder encontrar múltiples soluciones al problema inverso a demás de que son completamente estructurables al grado de que se puede programar completamente, caso que no sucede con los otros métodos no numéricos. Se utilizó específicamente el algoritmo EvoNorm el cual es explicado en el capítulo 3. Lo que se realizó fue la optimización del error de posición del robot utilizando el algoritmo evolutivo, utilizando cinemática directa.

2.2.3 Modelo cinemático de velocidad

El modelado cinemático de un robot busca las relaciones entre las variables articulares: La posición y orientación del extremo del robot, sin embargo, también se requiere conocer la relación entre las velocidades articulares y las velocidad del extremo del robot en el espacio cartesiano, a lo cual se denomina modelo cinemático diferencial o de velocidad.

El modelo cinemático diferencial se obtiene [3] a través de la denominada matriz jacobiana, la cual permite conocer las velocidades del extremo del robot a partir de los valores de las velocidades de cada articulación, y a su vez la matriz jacobiana inversa permitirá conocer las velocidades articulares necesarias para obtener unas velocidades determinadas en el extremo del robot.

El método más directo para obtener la relación entre velocidades articulares y del extremo del robot consiste en diferenciar las ecuaciones correspondientes al modelo cinemático directo. Así dadas las ecuaciones que resuelven el problema cinemático directo de un robot de n GDL

$$\begin{aligned}
 x &= f_x(q_1, \dots, q_n) \\
 y &= f_y(q_1, \dots, q_n) \\
 z &= f_z(q_1, \dots, q_n) \\
 \mu &= f_\mu(q_1, \dots, q_n) \\
 \beta &= f_\beta(q_1, \dots, q_n) \\
 \gamma &= f_\gamma(q_1, \dots, q_n)
 \end{aligned}
 \tag{2.2.2}$$

Si se derivan con respecto al tiempo ambos miembros del conjunto de ecuaciones anteriores, se tendrá

$$\begin{aligned}
\dot{x} &= \sum_1^n \frac{\partial f_x}{\partial q_i} \dot{q}_i \\
\dot{y} &= \sum_1^n \frac{\partial f_y}{\partial q_i} \dot{q}_i \\
\dot{z} &= \sum_1^n \frac{\partial f_z}{\partial q_i} \dot{q}_i \\
\dot{\mu} &= \sum_1^n \frac{\partial f_\mu}{\partial q_i} \dot{q}_i \\
\dot{\beta} &= \sum_1^n \frac{\partial f_\beta}{\partial q_i} \dot{q}_i \\
\dot{\gamma} &= \sum_1^n \frac{\partial f_\gamma}{\partial q_i} \dot{q}_i
\end{aligned} \tag{2.2.3}$$

El modelo cinemático diferencial 2.2.3 se expresa en forma matricial como

$$\begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{z} \\ \dot{\mu} \\ \dot{\beta} \\ \dot{\gamma} \end{bmatrix} = \mathbf{J} \begin{bmatrix} \dot{q}_1 \\ \dot{q}_2 \\ \vdots \\ \dot{q}_n \end{bmatrix}, \quad \text{con } \mathbf{J} = \begin{bmatrix} \frac{\partial f_x}{\partial q_1} & \frac{\partial f_x}{\partial q_2} & \cdots & \frac{\partial f_x}{\partial q_n} \\ \frac{\partial f_y}{\partial q_1} & \frac{\partial f_y}{\partial q_2} & \cdots & \frac{\partial f_y}{\partial q_n} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{\partial f_\gamma}{\partial q_1} & \frac{\partial f_\gamma}{\partial q_2} & \cdots & \frac{\partial f_\gamma}{\partial q_n} \end{bmatrix} \tag{2.2.4}$$

Donde \mathbf{J} se denomina matriz Jacobiana. Puesto que el valor numérico de cada uno de los elementos del Jacobiano dependerá de los valores instantáneos de las coordenadas articulares q_i , el valor del Jacobiano será diferente en cada uno de los puntos del espacio articular.

2.2.4 Modelo cinemático de velocidad inverso

Así como es posible obtener la relación directa que calcula las velocidades del extremo a partir de las velocidades articulares [3], puede obtenerse la relación inversa que permite calcular las velocidades articulares en función de las del extremo. Esto debido a la relación que tiene uno con el otro, dicha relación se encuentra representada en el jacobiano, aplicando algebra lineal se puede llegar a la obtención de las velocidades articulares:

$$\begin{bmatrix} \dot{q}_1 \\ \dot{q}_2 \\ \vdots \\ \dot{q}_n \end{bmatrix} = \mathbf{J}^{-1} \begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{z} \\ \dot{\mu} \\ \dot{\beta} \\ \dot{\gamma} \end{bmatrix} \quad (2.2.5)$$

Existen distintas formas de calcular la matriz Jacobiana inversa, entre las que se encuentran:

1. Cálculo de la inversa simbólicamente a partir de la matriz Jacobiana (La desventaja de este método que es difícil realizarlo analíticamente debido a que cuenta con funciones trigonométricas; la ventaja es que una vez calculado se puede aplicar cuantas veces se desee).
2. Evaluación numérica de la matriz \mathbf{J} para una postura concreta del robot, e invirtiendo numéricamente esta matriz se encuentra la relación inversa válida de la misma (La desventaja de este método es que tiene que evaluarse e invertirse para cada posición del robot; la ventaja es que el proceso no es muy complicado de realizar, sin embargo se depende del método numérico que se este utilizando).
3. Repitiendo el procedimiento seguido para la obtención del Jacobiano directo, pero ahora partiendo del modelo cinemático inverso (La desventaja es que este procedimiento es algebraicamente complicado debido a que se cuenta con funciones trigonométricas).
4. Si la matriz jacobiana no sea cuadrada, en general se calcula la pseudoinversa (La desventaja es que en caso de que el robot se encuentre en una configuración singular, no se puede percatar de lo mismo por este método).

Extendiendo un poco el procedimiento donde se presenta una configuración singular (aquella en que el determinante de la jacobiana se hace cero), lo que se realiza en ese caso es:

1. Identificar la articulación correspondiente al grado de libertad perdido (causante de que el determinante se anule).

2. Eliminar la fila de la Jacobiana correspondiente al grado de libertad perdido y la columna correspondiente a la articulación causante.
3. Con el nuevo Jacobiano reducido, calcular las velocidades de todas las articulaciones, a excepción de la eliminada, necesarias para conseguir las velocidades cartesianas deseadas. La velocidad de la articulación eliminada se mantendrá en cero.

2.2.5 Homogenización de la matriz jacobiana

Los índices de desempeño cinetostático en general, y algunos de los índices de desempeño dinámico, están definidos en función de la matriz Jacobiana y la matriz de inercia. Cuando el manipulador es de más de tres grados de libertad o sus articulaciones no son del mismo tipo, las unidades de los elementos de estas matrices no son homogéneas, y esta situación dificulta la evaluación de los índices de desempeño que están definidos en función de ellas para propósitos de comparación.

Para que los índices de desempeño basados en la matriz Jacobiana puedan ser evaluados de manera consistente, los elementos de la matriz Jacobiana deben ser dimensionalmente homogéneos [8]. En la literatura científica se ha advertido que el uso de índices de desempeño de una matriz Jacobiana cuyos elementos no sean dimensionalmente homogéneos puede causar problemas importantes en el diseño y el control de un manipulador [5].

Algunos autores han sugerido que se divida el estudio de los índices de desempeño para manipuladores con Jacobianas homogéneas y no homogéneas. Para los manipuladores cuya Jacobiana es homogénea, es decir, aquellos manipuladores de 2 o 3 grados de libertad cuyas articulaciones son del mismo tipo, los índices presentados anteriormente tiene aplicación directa y ofrecen información a nivel teórico y práctico. Por otro lado, en el caso de manipuladores cuya matriz no es homogénea, es decir aquellos manipuladores con articulaciones prismáticas y rotacionales de 3 o más grados de libertad, se han propuesto diversas formas de solucionar este problema, una de ellas se basa en la aplicación de la longitud característica.

Longitud característica

La longitud característica definida por Angeles[1], tiene la función de adimensionar los valores de longitud y a su vez normalizar la matriz Jacobiana dado un índice de desempeño específico. La longitud característica es aquella que genera un valor más isotrópico (o uno muy cercano) en

la matriz Jacobiana.

$$\bar{\mathbf{J}} \equiv \begin{bmatrix} \frac{\partial f_x}{\partial q_1} \frac{1}{L} & \cdots & \frac{\partial f_x}{\partial q_n} \frac{1}{L} \\ \frac{\partial f_y}{\partial q_1} \frac{1}{L} & \cdots & \frac{\partial f_y}{\partial q_n} \frac{1}{L} \\ \frac{\partial f_z}{\partial q_1} \frac{1}{L} & \cdots & \frac{\partial f_z}{\partial q_n} \frac{1}{L} \\ \frac{\partial f_\mu}{\partial q_1} & \cdots & \frac{\partial f_\mu}{\partial q_n} \\ \frac{\partial f_\beta}{\partial q_1} & \cdots & \frac{\partial f_\beta}{\partial q_n} \\ \frac{\partial f_\gamma}{\partial q_1} & \cdots & \frac{\partial f_\gamma}{\partial q_n} \end{bmatrix}$$

La obtención de la longitud característica es un problema de minimización muy complicado ya que cada uno de los elementos del jacobiano son sumamente no lineales y con funciones trascendentales. Sin embargo existen varios métodos de obtenerla, entre éstos se encuentran:

- Obtención de forma simbólica de la longitud característica. Este método sólo puede ser usado con robots manipuladores de pocos grados de libertad, y a pesar de esto, es complicado obtenerla y en algunos casos prácticamente imposible su obtención.
- La evaluación numérica de la matriz \mathbf{J} para todos los posibles valores de posición del robot, utilizando un rango de longitudes características, y de estas soluciones escoger la que minimice el índice de desempeño deseado. La desventaja de este método es que necesita mucho tiempo, debido a que la cantidad de evaluaciones aumenta de forma exponencial a medida que aumentan los grados de libertad del robot.
- Utilización de algoritmos de minimización numéricos, los cuales buscan minimizar el valor en base a un criterio deseado. Este método tiene la desventaja de que muchos algoritmos de minimización presentan convergencia prematura o son engañados con mínimos locales de las funciones a minimizar.

- Algoritmos de inteligencia artificial (softcomputing), estos algoritmos funcionan de manera similar a los algoritmos de minimización numéricos, sin embargo presentan resistencia a mínimos locales de algunas funciones, y normalmente encuentran un valor muy cercano al mínimo real.

2.3 Dinámica de manipuladores

La dinámica se ocupa de la relación entre las fuerzas que actúan sobre un cuerpo y el movimiento que en él se origina. Por lo tanto el modelo dinámico de un robot tiene por objeto conocer la relación entre el movimiento de las articulaciones del robot y las fuerzas implicadas en ellas matemáticamente:

1. La localización del robot definida por sus variables articulares o por las coordenadas de localización de su extremo, y sus derivadas, velocidad y aceleración.
2. Las fuerzas y pares aplicados en las articulaciones (o en el extremo del robot).
3. Los parámetros dimensionales del robot, como longitud, masas e inercias de sus elementos.

El problema de obtener el modelo dinámico de un robot se complica a medida que se incrementan los grados de libertad del mismo, sin embargo el modelo dinámico es imprescindible para conseguir:

1. Simulación del movimiento del robot.
2. Diseño y evaluación de la estructura mecánica del robot.
3. Dimensionamiento de los actuadores.
4. Diseño y evaluación del control dinámico del robot.

Este último fin es evidentemente de gran importancia, pues de la calidad del control dinámico del robot depende la precisión y velocidad de sus movimientos. Entre los métodos que existen destacan:

- Metodología Euler-Lagrange

- Metodología Newton-Euler

El modelo que se utiliza es limitado, ya que no se toma en cuenta fricción, incertidumbres paramétricas y actuadores.

La obtención del modelo dinámico de un robot a partir de la formulación lagrangiana conduce a un algoritmo con un costo computacional de orden $O(n^4)$, es decir, el número de operaciones a realizar crece con la potencia cuarta del número de grados de libertad, mientras que el algoritmo Newton-Euler tiene un costo computacional de orden $O(n)$. En nuestro caso utilizaremos la metodología Newton-Euler debido a que este requiere de menos operaciones para obtener el modelo dinámico.

2.3.1 Metodología Newton-Euler

La formulación de Newton-Euler parte del equilibrio de fuerzas y pares:

$$\sum \mathbf{F} = m\dot{\mathbf{v}}, \quad \sum \tau = \mathbf{I} \cdot \omega + \omega \times (\mathbf{I} \cdot \omega)$$

Un adecuado desarrollo de estas ecuaciones conduce a una formulación recursiva en la que se obtienen la posición, velocidad y aceleración del eslabón i referidos a la base del robot a partir de los correspondientes del eslabón $i - 1$ y del movimiento relativo de la articulación i . Este algoritmo se basa en operaciones vectoriales (con productos escalares y vectoriales entre magnitudes vectoriales, y productos de matrices con vectores) siendo más eficiente en comparación con las operaciones matriciales asociadas a la formulación Lagrangiana. De hecho, el orden de complejidad computacional de la formulación recursiva Newton-Euler es $O(n)$ lo que indica que depende directamente del número de grados de libertad [3].

Algoritmo para el modelado dinámico por Newton-Euler [3]

1. Asignar a cada eslabón un sistema de referencia de acuerdo con las reglas de Denavit-Hartenberg

2. Obtener las matrices de rotación \mathbf{R}_{i-1}^i y sus inversas \mathbf{R}_i^{i-1} , siendo:

$$\mathbf{R}_{i-1}^i = \begin{bmatrix} \cos \theta_i & -\cos \alpha_i \sin \theta_i & \sin \alpha_i \sin \theta_i \\ \sin \theta_i & \cos \alpha_i \cos \theta_i & -\sin \alpha_i \cos \theta_i \\ 0 & \sin \alpha_i & \cos \alpha_i \end{bmatrix}$$

3. Establecer las condiciones iniciales. Para el sistema de la base $\{S_0\}$: ${}^0\omega_0$: velocidad angular = $[0, 0, 0]^T$

$${}^0\dot{\omega}_0 : \text{aceleración angular} = [0, 0, 0]^T$$

$${}^0\mathbf{V}_0 : \text{velocidad lineal} = [0, 0, 0]^T$$

$${}^0\dot{\mathbf{V}}_0 : \text{velocidad lineal} = [0, 0, 0]^T$$

${}^0\omega_0, {}^0\dot{\omega}_0, {}^0\mathbf{V}_0$ son típicamente nulos salvo que la base del robot esté en movimiento. Para el extremo del robot se conocerá la fuerza y el par ejercidos externamente ${}^{n+1}\mathbf{F}_{n+1}$ y ${}^{n+1}\mathbf{n}_{n+1}$

$$\mathbf{z}_0 = [0, 0, 0]^T$$

$${}^i\mathbf{p}_i = \text{Coordenadas del origen del sistema } \{S_i\} \text{ respecto a } \{S_{i-1}\} = [a_i, d_i S_i, d_i C_i]$$

$${}^i\mathbf{s}_i = \text{Coordenadas del centro de masas del eslabón } i \text{ respecto del sistema } \{S_i\}$$

$${}^i\mathbf{I}_i = \text{Matriz de inercia del eslabón } i \text{ respecto de su centro de masas expresado en } \{S_i\}$$

Para $i = 1, \dots, n$ realizar los pasos 4 a 7:

4. Obtener la velocidad angular del sistema $\{S_i\}$.

$${}^i\omega_i = \begin{cases} \mathbf{R}_i^{i-1} ({}^{i-1}\omega_{i-1} + \mathbf{z}_0 \dot{q}_i) & \text{si el eslabón } i \text{ es de rotación} \\ \mathbf{R}_i^{i-1} ({}^{i-1}\omega_{i-1}) & \text{si el eslabón } i \text{ es de traslación} \end{cases}$$

5. Obtener la aceleración angular del sistema $\{S_i\}$.

$${}^i\dot{\omega}_i = \begin{cases} \mathbf{R}_i^{i-1} ({}^{i-1}\dot{\omega}_{i-1} + \mathbf{z}_0 \ddot{q}_i) + {}^{i-1}\omega_{i-1} \times \mathbf{z}_0 \dot{q}_i & \text{si el eslabón } i \text{ es de rotación} \\ \mathbf{R}_i^{i-1} ({}^{i-1}\dot{\omega}_{i-1}) & \text{si el eslabón } i \text{ es de traslación} \end{cases}$$

6. Obtener la aceleración lineal del sistema i :

$${}^i\dot{\mathbf{v}}_i = \begin{cases} {}^i\dot{\omega}_i \times {}^i\mathbf{p}_i + {}^i\omega_i \times ({}^i\omega_i \times {}^i\mathbf{p}_i) + \mathbf{R}_i^{i-1}({}^{i-1}\dot{\mathbf{v}}_{i-1}) & \text{si el eslabón } i \text{ es de rotación} \\ \mathbf{R}_i^{i-1}(\mathbf{z}_0\ddot{q}_i + {}^{i-1}\dot{\mathbf{v}}_{i-1}) + {}^i\dot{\omega}_i \times {}^i\mathbf{p}_i + 2{}^i\omega_i \times \mathbf{R}_i^{i-1}\mathbf{z}_0\dot{q}_i + & \text{si el eslabón } i \text{ es de traslación} \\ {}^i\omega_i \times ({}^i\omega_i \times {}^i\mathbf{p}_i) & \end{cases}$$

7. Obtener la aceleración lineal del centro de gravedad del eslabón i :

$${}^i\mathbf{a}_i = {}^i\dot{\omega}_i \times {}^i\mathbf{S}_i + {}^i\omega_i \times ({}^i\omega_i \times {}^i\mathbf{s}_i) + {}^i\dot{\mathbf{v}}_i$$

Para $i = n \dots 1$ realizar los pasos 8 a 10

8. Obtener la fuerza ejercida sobre el eslabón i :

$${}^i\mathbf{f}_i = \mathbf{R}_i^{i-1}({}^{i+1}\mathbf{f}_{i+1}) + m_i {}^i\mathbf{a}_i$$

9. Obtener la fuerza o par aplicado a la articulación i :

$${}^i\mathbf{n}_i = \mathbf{R}_i^{i-1}[{}^{i+1}\mathbf{n}_i + (\mathbf{R}_{i-1}^i {}^i\mathbf{p}) \times {}^{i+1}\mathbf{f}_{i+1}] + ({}^i\mathbf{p}_i + {}^i\mathbf{s}_i) \times m_i {}^i\mathbf{a}_i + {}^i\mathbf{I}_i {}^i\dot{\omega}_i + {}^i\omega_i \times ({}^i\mathbf{I}_i {}^i\omega_i)$$

10. Obtener la fuerza o par aplicado a la articulación i :

$$\tau_i = \begin{cases} {}^i\mathbf{n}_i^T \mathbf{R}_i^{i-1} \mathbf{z}_0 & \text{si el eslabón } i \text{ es de rotación} \\ {}^i\mathbf{f}_i^T \mathbf{R}_i^{i-1} \mathbf{z}_0 & \text{si el eslabón } i \text{ es de traslación} \end{cases}$$

Donde τ es el par o fuerza efectivo (par motor menos pares de rozamiento o perturbación).

Ya obtenido este modelo, se puede cambiar a su representación en forma compacta:

$$\tau = M(q)\ddot{q} + C(q, \dot{q})\dot{q} + G(q)$$

donde las matrices son conocidas como:

- $M(q)$ = Matriz de inercias.
- $C(q, \dot{q})$ = Matriz de coriolis .
- $G(q)$ = Matriz de efectos por gravedad.

2.4 Índices de desempeño

Un índice de desempeño de un robot es una cantidad escalar que permite evaluar el funcionamiento de un robot a partir de un criterio definido. Los índices de desempeño son importantes instrumentos para la planificación de movimientos y el diseño de robots manipuladores. En la planificación de movimientos un índice de desempeño ayuda a optimizar la operación del robot (en este caso se habla de índices dependientes de la postura).

En la etapa de diseño, un índice de desempeño permite orientar el proceso de cálculo de los parámetros geométricos e inerciales que definen la arquitectura del robot (en este caso se requiere de índices independientes de la postura).

Se han propuesto varios índices de desempeño en las últimas décadas [11]. El interés por estos índices se generó debido a la necesidad de criterios o herramientas para evaluar el diseño de robots manipuladores, una tarea que dependía de la experiencia de los diseñadores y el uso de herramientas CAD.

Este interés por los índices de desempeño ha continuado en las décadas posteriores, en las cuales diversos autores han propuesto criterios que permiten evaluar la capacidad del robot para realizar tareas de manipulación desde una perspectiva global, hasta otros que permiten evaluar algún aspecto en particular de la tarea de interés.

Entre los índices de desempeño [11], los principales son:

- Índices de desempeño cinetostático (Número de condición, manipulabilidad, mínimo valor singular),
- Índices de desempeño dinámico (Matriz de inercia generalizada, manipulabilidad dinámica, maximización del volumen y uniformidad del hiperparalelepípedo de aceleración, condicionamiento dinámico),

- Índices de límites articulares (Límite de la articulación, disponibilidad de movilidad articular, evitación de límites articulares), índices definidos sobre el espacio de trabajo,
- Índices de espacio de trabajo (Tipos de espacios de trabajo, global de condicionamiento, global de manipulabilidad, radio de aceleración).

2.4.1 Índices de desempeño cinetostático

Los índices de desempeño cinetostático permiten evaluar la capacidad del manipulador para transformar las velocidades o fuerzas en los actuadores, en velocidades o fuerzas en el efector final. La mayoría de estos índices están definidos en función de la matriz Jacobiana del manipulador, la cual establece las relaciones de velocidad y fuerza entre las articulaciones y el efector final.

Los elementos de la matriz Jacobiana dependen únicamente de la geometría del manipulador y su configuración. Los índices presentados en esta sección son dependientes de la postura y pueden ser utilizados en la planificación de trayectorias de tal manera que el robot acoja configuraciones en las cuales su capacidad para generar velocidades y fuerzas, ó la exactitud en el control, sea optimizada.

Elipsoide de Velocidad

Un concepto que ayuda a visualizar la definición de los índices de desempeño presentados en esta sección es el elipsoide de velocidad [17]. Para comenzar considérese el modelo de velocidad de un robot manipulador:

$$\dot{\mathbf{x}} = \mathbf{J}\dot{\mathbf{q}}$$

donde $\dot{\mathbf{x}}$ es el vector de velocidades del efector final, $\dot{\mathbf{q}}$ es el vector de velocidades articulares, y \mathbf{J} es la matriz Jacobiana. El elipsoide de velocidad está dado por el conjunto de todas las velocidades del efector final que son realizables para las velocidades articulares que satisfacen

que $\|\dot{\mathbf{q}}\| \leq 1$. Dado que

$$\|\dot{\mathbf{q}}\|^2 = \dot{\mathbf{q}}^T \dot{\mathbf{q}} = \dot{\mathbf{x}}^T \mathbf{J}^T \mathbf{J} \dot{\mathbf{x}}$$

se tiene que las velocidades que forman parte del elipsoide de velocidad son aquellas que satisfacen la siguiente condición:

$$\dot{\mathbf{x}}^T \mathbf{J}^T \mathbf{J} \dot{\mathbf{x}} \leq 1$$

Del elipsoide de velocidad se puede observar que el robot tiene la capacidad de moverse a mayor velocidad cuanto mayor sea el volumen del elipsoide. Para las velocidades en el espacio articular que cumplan con $\|\dot{\mathbf{q}}\| \leq 1$, las mayores velocidades que efector final puede alcanzar, se obtienen sobre el semieje mayor del elipsoide.

Por el contrario, el efector final del robot se moverá con más baja velocidad en dirección del semieje menor del elipsoide. Por otro lado, si la forma del elipsoide se aproxima a la esfera, entonces el efector final se puede mover uniformemente en todas direcciones.

Mediante la descomposición de la matriz Jacobiana en valores singulares se puede demostrar que los valores singulares $\sigma_1, \sigma_2, \dots, \sigma_n$, corresponden a las magnitudes de los ejes principales del elipsoide de velocidad. Los valores singulares, las raíces cuadradas de los valores propios de la matriz $\mathbf{J}^T \mathbf{J}$.

Cuando la matriz Jacobiana es singular, al menos uno de los valores singulares es igual a cero. Aquellas configuraciones en las cuales la matriz \mathbf{J} es singular, se llaman configuraciones singulares. Cuando un manipulador serial se encuentra en una configuración singular, pierde al menos un grado de libertad, es decir que pierde la capacidad de generar movimiento en ciertas direcciones.

A continuación se presentan algunos índices de desempeño que permiten evaluar las configuraciones desde el punto de vista cinetostático. Estos índices se pueden calcular directa o indirectamente a partir de los valores singulares de la matriz Jacobiana. Por este motivo, además de sus características propias, se han considerado como medidas de alejamiento a configuraciones

singulares.

Número de condición

Este índice fue propuesto por Salisbury y Craig [12] para el diseño de los dedos de una mano robótica. El número de condición k de una matriz Jacobiana \mathbf{J} esta definido de la siguiente manera:

$$CN = k(\mathbf{J}) = \|\mathbf{J}\| \times \|\mathbf{J}^{-1}\|$$

donde las normas que normalmente se utilizan son:

- La norma 2, definida como la raíz cuadrada del mayor eigenvalor de la matriz $\mathbf{J}^{-T}\mathbf{J}^{-1}$.
- La norma euclidiana o Frobenius definida como $\|\mathbf{A}\| = \sqrt{\sum_i^m \sum_i^n |a_{ij}|^2}$. Donde n es la cantidad de renglones que cuenta la matriz A , y m la cantidad de columnas.

Las principales características que denota el valor del número de condición de un robot son:

- La precisión / destreza de un robot.
- La cercanía de una posición singular. En general, no es posible definir una distancia matemática a una singularidad para los robots cuyos GDL es una mezcla, por lo que el uso del número de condición es válida como un índice cuando se utilizan articulaciones prismáticas y rotacionales.
- Como un criterio de rendimiento para optimizar el diseño y la comparación robots.
- Como un criterio para determinar el área de trabajo útil de un robot.

La definición del número de condición deja claro que puede calcular su forma analítica como una función del robot de forma muy simple. Pero evaluando el Jacobiano en un punto específico se puede obtener numéricamente el valor de dicho índice.

Sin embargo, para un robot que tiene tanto la traslación y la orientación, hay un inconveniente importante del número de condición: La matriz involucrada en su cálculo no es homogénea en términos de unidades. Por lo tanto, el valor del número de condición para un robot dado cambiará de acuerdo con la elección unidad, mientras que claramente la precisión cinemática es constante.

Para hacer frente a este problema se ha sugerido para definir una normalización de matriz Jacobiana inversa dividiendo los elementos de traslación de la matriz de una longitud tal como la longitud de los enlaces en una posición isotropica, la longitud característica se define como aquella que minimiza la condición en toda su área de trabajo.

Manipulabilidad cinemática

El índice de la manipulabilidad w fue propuesto por Yoshikawa [17]. El propósito de este índice es medir la capacidad de un robot, en cierta configuración, para generar velocidades en el efector final. Este índice de desempeño es proporcional al volumen del elipsoide de velocidad. Para el caso general (incluyendo robots redundantes) la manipulabilidad esta definida de la siguiente manera:

$$KM = w = \sqrt{\det(\mathbf{J}\mathbf{J}^T)}$$

A mayores valores de la manipulabilidad w el robot tiene una mayor capacidad de realizar movimientos en el efector final. En el caso de que la manipulabilidad sea igual a cero entonces el robot se encuentra en una configuración singular. En esta configuración el robot pierde la capacidad de realizar movimientos en ciertas direcciones.

Por otro lado, Yoshikawa presentó además el concepto de la manipulabilidad de fuerza w_f . Este índice trata de medir la capacidad del robot, en una configuración dada, para ejercer fuerzas sobre un objeto en reposo a través de su efector final.

Al igual que en el caso anterior, se puede generar un elipsoide de fuerza. Para generar el

elipsoide de fuerza se utiliza el modelo estático del robot, i.e., la relación que existe entre el vector de fuerzas en los actuadores, τ , y el vector de fuerzas generadas en el efector final, f . El modelo estático de un manipulador es el siguiente:

$$\tau = \mathbf{J}^T \mathbf{f}$$

De esta forma, el conjunto de todos los vectores f que son realizables por una τ tal que $\|\tau\| \leq 1$ forman un elipsoide en el espacio euclidiano m dimensional, dado por

$$\mathbf{f}^T \mathbf{J} \mathbf{J}^T \mathbf{f} \leq 1$$

El volumen del elipsoide de fuerza es inversamente proporcional a la manipulabilidad (de velocidad). Por lo tanto la manipulabilidad de fuerza es:

$$FM = w_f = \frac{1}{\sqrt{\det(\mathbf{J} \mathbf{J}^T)}}$$

Esto significa que la dirección en la que el efector final puede ejercer mayor fuerza, es aquella en donde se puede generar menos velocidad.

Mínimo Valor Singular

El mínimo valor singular fue propuesto por Klein [7], como índice de desempeño. Naturalmente este valor permite medir la cercanía de la configuración del robot a una configuración singular, ya que directamente indica si la magnitud de alguno de los ejes principales del elipsoide esta a punto de anularse. En el caso del número de condición y la manipulabilidad, esta situación (la cercanía a una singularidad) en algunos casos puede ser imprecisa ya que su valor también depende de otros valores singulares y la disminución en un valor singular puede ser compensada por un incremento o decremento en otro valor singular. El volumen del elipsoide de velocidad será mayor cuando el valor de este índice sea mayor.

2.4.2 Índices de desempeño dinámico

Los robots manipuladores están sujetos a fuerzas y pares no lineales como lo son: Las fuerzas inerciales, fuerzas centrífugas y de coriolis, y las fuerzas de la gravedad. Los índices de desempeño dinámico sirven para evaluar la capacidad de efectuar fuerzas de un robot manipulador, el cual se encuentra en movimiento.

En esta sección se presentan, algunos conceptos e índices que se han propuesto para evaluar la dinámica de un manipulador.

Entre los distintos índices de desempeño que se conocen, en este desarrollo se mencionan dos índices:

- Manipulabilidad Dinámica
- Índice de Condicionamiento Dinámico

Manipulabilidad Dinámica

Este índice, definido por Yoshikawa [16], mide la capacidad de un robot para generar aceleraciones en el efector final, dado un conjunto de fuerzas en los actuadores, todo esto considerando la dinámica del robot.

Esto se logra a partir de:

$$v = J(q)\dot{q}$$

Se deriva y se obtiene:

$$\dot{v} = J(\ddot{q}) + a_r(q, \dot{q}) \quad a_r(q, \dot{q}) = \dot{J}\dot{q}$$

Introduciendo los vectores:

$$\tilde{\tau} = \tau - h(q, \dot{q}) - g(q) \tag{2.4.1}$$

$$\tilde{v} = \dot{v} - a_r(q, \dot{q}) \quad (2.4.2)$$

A partir de la ecuación dinámica generalizada (2.4.1) y de la ecuación (2.4.2), se obtiene:

$$\tilde{v} = JM^{-1}\tilde{J}$$

La idea en la que se fundamenta este índice es en cuantificar el grado de arbitrariedad con la que puede cambiar la aceleración bajo ciertas restricciones de la fuerza motriz de las articulaciones. Obteniendo la definición [16]:

$$W_d = \sqrt{\det[J(M^T M)^{-1} J^T]}$$

Otra forma de obtenerlo es utilizando descomposición de valores singulares, considerando:

$$JM^{-1} = U\Sigma V^T$$

Obteniendo los valores singulares de Σ los cuales son dados por el vector valores singulares $\sigma_1 \geq \sigma_2 \geq \dots \geq \sigma_m \geq 0$ con los cuales se obtiene el valor del índice:

$$W_d = \sigma_1 \sigma_2 \cdots \sigma_m$$

Isotropía dinámica

Este índice fue definido por Angeles [9] y según sus autores su propósito es servir como medida de la distancia que existe entre la matriz de inercia del manipulador y una matriz de inercia isotrópica.

Este índice se basa en el hecho de que si la matriz de inercia generalizada de un manipulador es isotrópica, entonces, el manipulador se dice que esta en una configuración dinámica isotropica. Y lo que se busca saber, es qué tan lejos se encuentra cierta posición del manipulador de una posición isotrópica.

El índice de isotropía dinámica se define como:

$$WI = \frac{1}{2}e^T W e$$

donde e está definido como un vector compuesto de la matriz triangular superior de la siguiente matriz de error:

$$E(q) = \frac{1}{\sigma}(I(q) - \sigma 1)$$

y a su vez σ está definido como:

$$\sigma = \frac{\text{tr}(I^2)}{\text{tr}(I)}$$

2.5 Conclusiones del capítulo

En este capítulo se presentaron las metodologías de análisis de robots cinemáticos seriales, así como todo el proceso que implica la obtención de los modelos mediante dichas metodologías.

Entre las metodologías que se estudiaron fueron la de Denavit-Hartenberg, la cual se usa para la obtención del modelo cinemático; Newton-Euler para la obtención del modelo dinámico; así como la obtención de los índices de desempeño y sus respectivas ecuaciones.

Capítulo 3

Diseño iterativo de manipuladores

3.1 Introducción

Una amplia variedad de casos en la ingeniería, la industria, y otros campos, implican la optimización simultánea de varios objetivos. En muchos casos, los objetivos se definen en unidades incomparables, y presentan compromisos de diseño, con un cierto grado de conflicto entre ellos (es decir, uno de los objetivos no se puede mejorar sin deterioro de al menos otro de los objetivos). Estos problemas se llaman problemas de optimización multiobjetivo (MOP).

En la optimización de un sólo objetivo, es posible determinar entre cualquier par dado de soluciones si una es mejor que el otra. Como resultado de ello, por lo general obtenemos una única solución óptima. Sin embargo, en la optimización multiobjetivo no existe un método sencillo para determinar si una solución es mejor que otra. El método de optimización multiobjetivo más utilizado para comparar soluciones es la llamada relación de dominancia de Pareto, que, en lugar de una única solución óptima, da lugar a un conjunto de alternativas con diferentes soluciones de compromiso entre los objetivos. Sin embargo presenta la desventaja de que existen múltiples soluciones óptimas de Pareto, en la práctica, sólo una solución tiene que ser seleccionada para la aplicación.

Se pueden distinguir dos enfoques principales para resolver problemas de optimización multiobjetivo. El primero se llama la Decisión Multi-Criterios (MCDM), en la mayoría de los métodos del MCDM la toma de decisiones juega un papel importante en el suministro de información para construir un modelo de referencia que es explotado por el método de programación matemática para encontrar las soluciones que mejor se ajustan al modelo de referencia del DM. Optimización Multiobjetivo Evolutiva (EMO) [6] es otro método útil para resolver problemas de optimización multiobjetivo. Dado que los algoritmos evolutivos utilizan un enfoque basado en la población, generalmente encuentran una aproximación a la solución el cual respeta la función objetivo que se desee. Aunque en los EMO la tarea de toma de decisiones no ha recibido mucha atención en el pasado, en recientes años un número considerable de obras se han ocupado de la incorporación de las preferencias en Algoritmos Evolutivos Multi-Objetivo (MOEAs).

Sin embargo, las técnicas de programación matemática presentan la desventaja que son problemas de optimización muy no lineales, generalmente tienden a obtener soluciones locales, las cuales no se desean si se busca resolver un problema de la forma mas cercana a la solución mínima global.

Considerando las desventajas mencionadas, se utiliza el uso de algoritmos evolutivos multiobjetivo, para resolver el diseño de robots para que realice tareas en específico, optimizándolo de acuerdo a los índices de desempeño cinetostáticos y dinámicos presentados en el capítulo 2.

3.2 Algoritmos evolutivos multiobjetivo

Los Algoritmos Evolutivos (AE) [15] son algoritmos iterativos de poblaciones de soluciones basados en métodos de optimización metaheurísticos que utilizan mecanismos inspirados en la biología, como mutación, cruce, selección natural y supervivencia del más fuerte, a fin de perfeccionar un conjunto de soluciones candidatas.

La característica de los algoritmos evolutivos en comparación con otros métodos de optimización es su "caja negra", carácter que hace sólo unas suposiciones sobre las funciones objetivo subyacentes. Por otra parte, la definición de funciones objetivo por lo general requiere menor visión de la estructura del espacio del problema para la construcción de una heurística admisible. Por lo tanto, AE se desempeña consistentemente bien en diferentes categorías de problemas.

Lo que se busca en un algoritmo evolutivo multiobjetivo es la optimización global, la cual consiste en encontrar los mejores elementos posibles \mathbf{x}^* de acuerdo a un criterio $F = \{f_1, f_2, \dots, f_n\}$. Estos criterios se expresan como funciones matemáticas, las cuales denominamos funciones objetivo.

Una función objetivo (función de evaluación) $f : X \mapsto Y$ con $Y \subseteq \mathbb{R}$ es la función que se busca optimizar. El dominio X de f se llama espacio del problema y puede representar cualquier tipo de elementos como números, listas, planos de construcción, y así sucesivamente. Se elige de acuerdo con el problema a optimizar. Las funciones objetivo no son necesariamente expresiones matemáticas simples, sino que pueden ser algoritmos complejos que, por ejemplo, implican múltiples simulaciones. La Optimización Global comprende las técnicas para encontrar los mejores elementos \mathbf{x}^* en todo X con respecto a los criterios $f \in F$.

3.2.1 Clasificación de algoritmos evolutivos

Existe una gran cantidad de algoritmos evolutivos, sin embargo todos ellos tienen en común que buscan (o se encuentran basados en) imitar los mecanismos biológicos de la teoría de la evolución, los cuales constan de:

- Selección
- Evaluación
- Reproducción

Sin embargo, a pesar de estar basados en la misma estructura, existen distintos algoritmos evolutivos [10] dentro de la familia de computación evolutiva, entre los más conocidos se encuentran:

- Algoritmos genéticos simples
 - Genes reales
 - Programación genética
 - Dominancia
 - Premutación
 - Multimodal
- Inteligencia colectiva
 - Colonia de hormigas
 - Cúmulo de partículas (PSO)
- Estrategias evolutivas
 - Evolución diferenciada
- Estimación de distribuciones
 - EvoNorm
 - Adaptación de la matriz de covarianza (CAMES)

Cada uno de los algoritmos mostrados, tienen características inherentes al funcionamiento que lo define. En el caso estudiado en este trabajo se requiere un algoritmo que cumpla con las siguientes características:

- Tiempo de ejecución corto,

- Robusto ante un gran número de variables,
- Basado en metodologías concretas (Esto quiere decir que se basa en criterios matemáticos o estadísticos),
- Se garantice o al menos se tengan antecedentes de optimización multiobjetivo con tal algoritmo.

Teniendo en cuenta estas características o requisitos, se seleccionó al algoritmo EvoNorm, el cual tiene un tiempo de ejecución muy corto, funciona muy bien con grandes (y pequeñas) cantidades de variables, está basado en criterios estadísticos (estimación de distribuciones), ya se han realizado estudios multiobjetivo con este algoritmo.

3.2.2 EvoNorm

EvoNorm [14][13] utiliza variables aleatorias con distribución normal, con las cuales se describen muchos de los fenómenos que ocurren en la vida diaria. En este algoritmo es simulado usando dos criterios estadísticos: La media (la cual es la medida numérica de la tendencia central en la que ocurre una variable), y la desviación estándar (la cual mide la dispersión con la que ocurre una variable de acuerdo a su media). La distribución normal puede usarse para representar un conjunto de posibles valores a evaluar en las variables de una ecuación

$$N(\mu, \sigma) = \mu + \sigma \sum_i U \quad (3.2.1)$$

donde, μ es la media, σ es la desviación estándar, y U es el valor aleatorio generado por una distribución uniforme. El algoritmo EvoNorm tiene el mismo funcionamiento que los otros algoritmos evolutivos, ya que consta de los procesos de evaluación, selección y reproducción o mutación, pero este último es substituido por nuevos procedimientos en los cuales se calculan

los parámetros en base al resultado de la función de evaluación de acuerdo a los valores de las variables.

EvoNorm evoluciona los valores generando un nuevo vector de números reales como valores a utilizar. Estos valores son evaluados y el mejor de ellos es seleccionado para calcular los nuevos parámetros de distribución con los cuales se genera una nueva población . Este proceso se repite por un número establecido de generaciones.

El pseudocódigo de EvoNorm es:

1. Generación aleatoria de la población,
2. Evaluación de los individuos (NTI),
3. Selección de los n mejores individuos ,
4. Calculo de la media μ y desviación estándar σ dados por:

$$\mu(pr) = \sum_i^n \frac{PS(pr, i)}{n} \quad (3.2.2)$$

y

$$\sigma(pr) = \sqrt{\sum_i^n (PS(pr, i) - \mu(pr))^2} \quad (3.2.3)$$

donde pr representa la variable de decisión, PS representa el individuo seleccionado, n representa el número de individuos seleccionados. PS es una matriz de tamaño de $NTPr \times NTI$ (donde $NTPr$ es el número total de parámetros y NTI es el número total de individuos).

5. Generación de una nueva población de tamaño m utilizando los parámetros obtenidos de la media y desviación estándar, sin embargo se agrega un paso extra el cual consiste en utilizar la μ y σ del mejor individuo en el 50 % de los casos.

$$P_i = \begin{cases} N(\mu_{pr}, \sigma_{pr}) & U() > 0.5 \\ N(\mu_{best}, \sigma_{best}) & U() \leq 0.5 \end{cases}$$

6. Reevaluar si el criterio (función objetivo a optimizar) no a sido establecido.

3.3 Algoritmo generador de parámetros para robots

Como se explicó al inicio de este capítulo los algoritmos evolutivos constan de tres mecanismos principales: Generación, Evaluación y Reproducción. En las siguientes secciones se explicará más a detalle cómo se elabora cada uno de estos procesos, específicamente para el algoritmo EvoNorm con el fin de generar robots para tareas específicas.

3.3.1 Generación de población

A cada uno de los segmentos del conjunto de parámetros generados con los cuales se evaluará o generará un robot se le llama individuo, los cuales son considerados cada uno como una solución candidata, recordando que para modelar un robot se necesitan sus valores de Denavit-Hartenberg así como los valores de masa y centros de masa, en el algoritmo cada parámetro generado para el individuo representa un valor de la matriz de Denavit-Hartenberg, así como los valores de masa del robot.

A todo el conjunto de individuos se le llama población la cual es todo el conjunto de posibles soluciones, de cantidad finita y acotada de individuos. Los individuos de la población inicial, se generan por la función aleatoria uniformemente distribuida U , considerando cotas máximas y mínimas definidas. Después de este proceso se efectúa la generación de la tabla de Denavit-Hartenberg y las distancias al centro de masa y sus respectivas masas, con los valores generados.

El avance de una generación es el proceso de evaluar los individuos de una generación y generar una nueva población, el total de generaciones es un valor asignado n .

Para generar la nueva población, el algoritmo necesita los siguientes datos:

- Tipo de articulación (Rotación o traslación)
- Cotas máximas y mínimas de las dimensiones y masas del eslabón.
- Tamaño de población de individuos NTI
- Generaciones totales a realizar NTG

El tipo de articulación es necesario para definir la estructura del robot serial, y en base a esta estructura se generan los valores de cada parámetro considerando las cotas máximas y mínimas. Este proceso se repite hasta generar el total de individuos dado por el tamaño de la población asignada. Para cada individuo, el algoritmo calcula simbólicamente su modelo cinemático directo, matriz jacobiana y modelo dinámico.

Sembrado

El sembrado es un proceso que se agrega al algoritmo, que este consiste en asignar las dimensiones y pesos de un individuo conocido previamente como una buena solución y con esto el algoritmo evolutivo rápidamente distinguiría si este individuo es uno adecuado para la tarea y empezará a evolucionar a partir de los valores de este robot, para optimizarlos.

3.3.2 Función de evaluación

Una parte crítica de un algoritmo evolutivo es una correcta función de evaluación, ya que ésta mide o califica el desempeño del individuo. En este trabajo se seleccionó una función multiobjetivo por considerar simultáneamente varios criterios de evaluación, dados por los índices de desempeño cinetostáticos y dinámicos de la sección 2.4

1. Error cuadrático de posición y rotación del efector final (QEP)
2. Número de condición (NC_{freu})

3. Manipulabilidad cinemática (MC)
4. Manipulabilidad de fuerza (MF)
5. Mínimo valor singular (MSV)
6. Manipulabilidad dinámica (DM)
7. Isotropía dinámica (DI)

Método de discriminantes

El método de discriminantes es muy utilizado en los algoritmos evolutivos multiobjetivos, este proceso consiste en guardar los valores de cada criterio de evaluación del mejor individuo hasta el momento, y en base a esto realizar una comparación con otro individuo, si el siguiente individuo consta de peor índice, se le añade un valor de error específico al individuo. Realizando esto para todos los criterios que se definieron. La ecuación para el método de discriminantes se define:

$$MetDisc(ErrorValue, BV, AV) = \begin{cases} ErrorValue & \text{si } BV \geq AV \\ 0 & \text{si } BV < AV \end{cases} \quad (3.3.1)$$

donde BV es el mejor valor hasta el momento (desde población en generación inicial hasta actual); AV el valor actual; y $ErrorValue$ el valor de error en caso que no se cumpla la condición que $BV \geq AV$.

Tomando como base el método de discriminantes se plantea la función, la cual se seleccionó con el concepto de que se mejore la posición final así como los índices de desempeño de estas posiciones. Esta ecuación esta definida por:

$$\begin{aligned}
fe = QEP + MetDisc\left(\frac{QEP}{6}, NC_{freuBV}, NC_{freuAV}\right) + MetDisc\left(\frac{QEP}{6}, MC_{BV}, MC_{AV}\right) + \\
MetDisc\left(\frac{QEP}{6}, MF_{BV}, MF_{AV}\right) + MetDisc\left(\frac{QEP}{6}, MSV_{BV}, MSV_{AV}\right) + \\
MetDisc\left(\frac{QEP}{6}, MD_{BV}, MD_{AV}\right) + MetDisc\left(\frac{QEP}{6}, DI_{BV}, DI_{AV}\right)
\end{aligned} \tag{3.3.2}$$

donde QEP es el error cuadrático de posición del efector final, el cual se obtiene a partir de la posición actual del robot y por la tarea que se desea realizar.

En la función de evaluación se le dió la misma prioridad a los índices que a la posición final debido a que si se evalúa de otra manera los índices de desempeño mejora pero comprometían la posición de la tarea a realizar encontrada por el algoritmo.

3.3.3 Proceso de reproducción

El proceso de reproducción del algoritmo se divide en cuatro etapas:

- Selección de los mejores individuos,
- Obtención de la media μ y desviación estándar σ de los mejores individuos,
- Búsqueda del mejor individuo de la población I_{best} ,
- Generación de nuevos individuos.

La selección de mejores individuos consiste en comparar los valores de la función de evaluación de cada individuo de la población de robots, y en seleccionar los mejores de la población. La cantidad a seleccionar es un porcentaje asignado, de la población este caso utilizamos el 25%.

La media y desviación estándar de cada parámetro calculan el conjunto valores que contiene cada individuo.

La búsqueda del mejor individuo de la población consiste en buscar el individuo que obtuvo mejor resultado de la función de evaluación. A este individuo se le dará mas prioridad para la generación de nuevos individuos, esto no quiere decir que todos los individuos van a ser generados en base al mejor individuo, sino que tambien se utilizarán a los demás del 25% de los mejores individuos, sólo que se le dará más al mejor individuo.

En caso que se generara toda la población en base a este mejor individuo el algoritmo muy probablemente obtenga una convergencia prematura, es decir que obtenga un mínimo local, lo cual no se desea. Sino que se busca el mínimo global, o en otras palabras el robot que realice mejor la tarea que se especificó en el problema a solucionar.

La generación de una nueva población se realiza generando un valor aleatorio entre 0 y 1; y seleccionando a un individuo de la poblacion de mejores, si queda abajo de 0.5 o al mejor si queda arriba del valor 0.5 (una moneda virtual), cada individuo generado se crea a partir de la distribución normal de cada parámetro y con esto se generan los individuos "hijos" de los mejores.

3.4 Conclusiones de Capítulo

En este capítulo se explicó brevemente los diferentes tipos de algoritmos de optimización multiobjetivo, dentro de todos ellos, se enfocó especialmente al algoritmo Evonorm el cual es un algoritmo evolutivo, y se programó en Matlab para nuestro caso, la optimización de los parámetros de diseño del robot, para optimizar la realización o ejecución de una tarea. Calificando al robot mediante los índices de desempeño que fueron explicados en el capítulo anterior, los cuales a su vez requieren parcialmente del modelo cinemático y dinámico del robot.

Capítulo 4

Casos de estudio

4.1 Introducción

En este capítulo se muestran experimentos en casos de estudio para verificar los resultados obtenidos en este trabajo. Estos experimentos consisten en asignar tareas representadas por las posiciones cartesianas del efector final del robot, luego se calcula el modelo cinemático, Jacobiano de velocidad, y modelo dinámico, mediante el algoritmo propuesto en este trabajo, de manipuladores que realicen las tareas asignadas. Posteriormente se evalúa su desempeño bajo los índices descritos en la Sección 2.4 al realizar la tarea asignada.

Por otro lado, se realizan experimentos en los que el algoritmo evolutivo modifica las características del manipulador generando parametrización de nuevos manipuladores.

A continuación se presentan los casos de estudio planteados, estos son:

- Caso de comparación y selección de manipuladores predeterminados.
- Caso de evolución de manipuladores para obtener un mejor desempeño para alguna tarea.

4.2 Comparación y selección de manipuladores predeterminados

En este caso de estudio lo que se busca es que el extremo del manipulador pase por la línea trazada en una placa horizontal que mide $2,000mm \times 2,000mm$, la cual está separada por cada lado a $1,000mm$ de los ejes x y y por cada lado y a una altura de $2,000mm$ sobre el suelo, tal y como se muestra en la Figura 4.1.

Las coordenadas cartesianas de las posiciones del extremo del manipulador para realizar la tarea están dadas en la tabla 4.2.1.

x (mm)	y (mm)	z (mm)
3,000	1,000	2,000
2,500	1,500	2,000
2,000	2,000	2,000
1,500	2,500	2,000
1,000	3,000	2,000

Tabla 4.2.1: Tabla de posiciones cartesianas por las que debe pasar el efector final del robot.

El algoritmo propuesto en este trabajo para modelado y evaluación de desempeño sólo requiere la matriz de parámetros de Denavit-Hartenberg, así como el valor de las masas de eslabones y distancias de centros de masa.

Se comparan tres manipuladores seriales: El primero de configuración angular con orientación libre del extremo final del manipulador mostrado en la Figura 4.1 de tres grados de libertad, cuyos parámetros de Denavit-Hartenberg de cada eslabón están dados en la Tabla 4.2.2, el segundo de configuración angular de 6 GDL, mostrado en la Figura 4.2, cuyos parámetros de Denavit-Hartenberg de cada eslabón están dados en la Tabla 4.2.4, el tercero de configuración SCARA, mostrado en la Figura 4.3, cuyos parámetros de Denavit-Hartenberg de cada eslabón están dados en la Tabla 4.2.3.

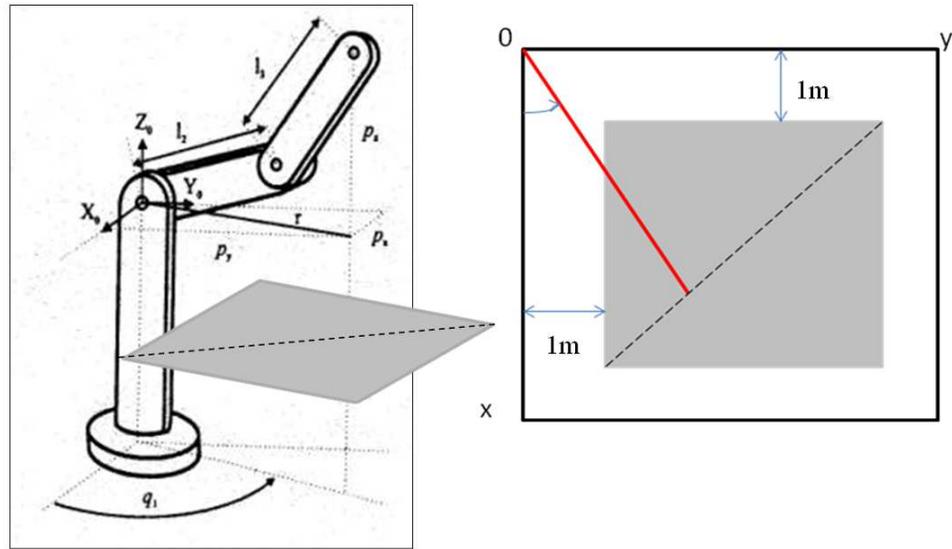


Figura 4.1: Caso de soldadora sobre placa cuadrada con orientación libre del extremo final del manipulador.

$\theta(rad)$	$d(mm)$	$a(mm)$	$\alpha(rad)$	$m_x(mm)$	$m_y(mm)$	$m_z(mm)$	$m(kg)$
q_1	0	0	$-\pi/2$	0	0	0	0
q_2	0	3,000	0	3,000/2	0	0	1
q_3	0	3,000	0	3,000/2	0	0	1

Tabla 4.2.2: Parámetros del manipulador angular 3 GDL.

$\theta(rad)$	$d(mm)$	$a(mm)$	$\alpha(rad)$	$m_x(mm)$	$m_y(mm)$	$m_z(mm)$	$m(kg)$
q_1	0	3,000	0	3,000/2	0	0	1
q_2	0	3,000	π	3,000/2	0	0	1
$-q_3$	0	0	0	0	0	0	0
0	q_4	0	0	0	0	$q_4/2$	1

Tabla 4.2.3: Parámetros del manipulador SCARA.

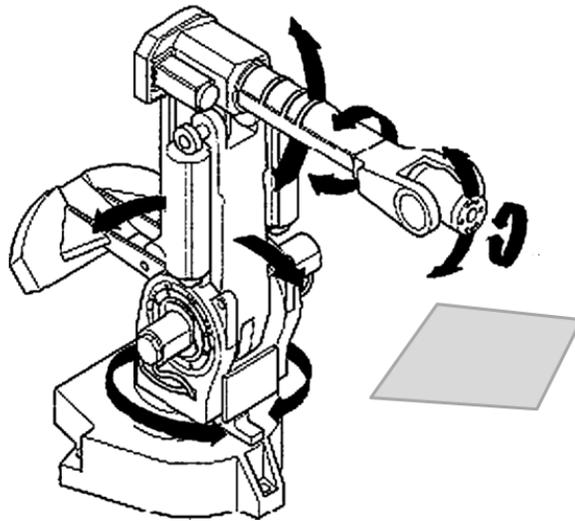


Figura 4.2: Caso de soldadora sobre placa cuadrada con robot de 6GDL.

$\theta(rad)$	$d(mm)$	$a(mm)$	$\alpha(rad)$	$m_x(mm)$	$m_y(mm)$	$m_z(mm)$	$m(kg)$
q_1	0	0	$-\pi/2$	0	0	0	1
q_2	3,000	0	$\pi/2$	0	0	350	1
$q_3 - \pi/2$	-100	$\pi/2$	0	0	0	-50	1
q_4	3,000	0	$-\pi/2$	0	0	320	1
q_5	0	0	$\pi/2$	0	0	0	1
q_6	200	0	0	0	0	50	1

Tabla 4.2.4: Parámetros del manipulador de 6 GDL.

4.2.1 Desempeño de robot angular de 3 grados de libertad

A continuación se muestran los modelos cinemático y dinámico, así como los índices de desempeño calculados por el algoritmo para el manipulador angular serial de 3 GDL de la Figura 4.1:

Modelo cinemático directo de posición y orientación del extremo final del manipulador, tal como se mostró en la Sección 2.2.2:

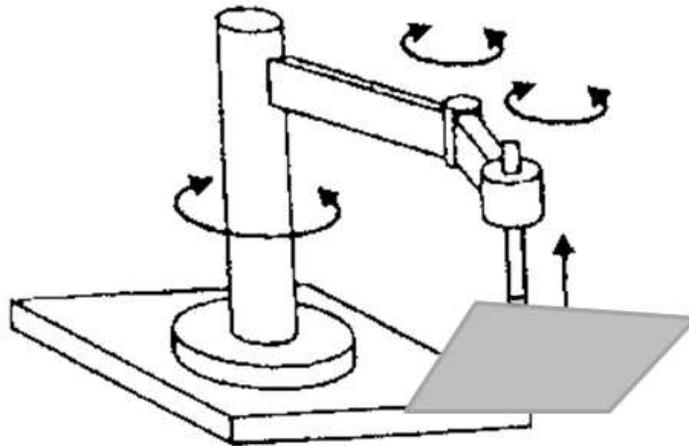


Figura 4.3: Caso de soldadora sobre placa cuadrada con robot SCARA.

$$\begin{aligned}
 Px &= 3 * \cos(q1) * \cos(q2) + 3 * \cos(q1) * \cos(q2) * \cos(q3) - 3 * \cos(q1) * \sin(q2) * \sin(q3) \\
 Py &= 3 * \cos(q2) * \sin(q1) - 3 * \sin(q1) * \sin(q2) * \sin(q3) + 3 * \cos(q2) * \cos(q3) * \sin(q1) \\
 Pz &= -3 * \sin(q2) - 3 * \cos(q2) * \sin(q3) - 3 * \cos(q3) * \sin(q2) \\
 \mu &= \pi/2 \\
 \beta &= \text{atan2}((\cos(q2) * \sin(q3) + \cos(q3) * \sin(q2)) / ((\cos(q1) * \cos(q2) * \cos(q3) - \cos(q1) * \\
 &\quad \sin(q2) * \sin(q3))^2 + (\sin(q1) * \sin(q2) * \sin(q3) - \cos(q2) * \cos(q3) * \sin(q1))^2)^{1/2}) \\
 \gamma &= -\text{atan2}((\sin(q1) * \sin(q2) * \sin(q3) - \cos(q2) * \cos(q3) * \sin(q1)) / (\cos(q1) * \cos(q2) * \\
 &\quad \cos(q3) - \cos(q1) * \sin(q2) * \sin(q3)))
 \end{aligned}$$

Elementos de la matriz jacobiana:

$$\begin{aligned}
 J(1,1) &= 3 * \sin(q1) * \sin(q2) * \sin(q3) - 3 * \cos(q2) * \sin(q1) - 3 * \cos(q2) * \cos(q3) * \sin(q1) \\
 J(1,2) &= -3 * \cos(q1) * \sin(q2) - 3 * \cos(q1) * \cos(q2) * \sin(q3) - 3 * \cos(q1) * \cos(q3) * \sin(q2) \\
 J(1,3) &= -3 * \cos(q1) * \cos(q2) * \sin(q3) - 3 * \cos(q1) * \cos(q3) * \sin(q2) \\
 J(2,1) &= 3 * \cos(q1) * \cos(q2) + 3 * \cos(q1) * \cos(q2) * \cos(q3) - 3 * \cos(q1) * \sin(q2) * \sin(q3) \\
 J(2,2) &= -3 * \sin(q1) * \sin(q2) - 3 * \cos(q2) * \sin(q1) * \sin(q3) - 3 * \cos(q3) * \sin(q1) * \sin(q2) \\
 J(2,3) &= -3 * \cos(q2) * \sin(q1) * \sin(q3) - 3 * \cos(q3) * \sin(q1) * \sin(q2) \\
 J(3,1) &= J(4,1) = J(4,2) = J(4,3) = J(5,1) = J(6,2) = J(6,3) = 0 \\
 J(3,2) &= 3 * \sin(q2) * \sin(q3) - 3 * \cos(q2) * \cos(q3) - 3 * \cos(q2) \\
 J(3,3) &= 3 * \sin(q2) * \sin(q3) - 3 * \cos(q2) * \cos(q3) \\
 J(5,2) &= (3 * \cos(q2/2 + q3/2)^2 + \cos((3 * q2)/2 + (3 * q3)/2)^2 - 2) / (2 * (\cos(q2 + q3)^2)^{3/2}) \\
 J(5,3) &= (3 * \cos(q2/2 + q3/2)^2 + \cos((3 * q2)/2 + (3 * q3)/2)^2 - 2) / (2 * (\cos(q2 + q3)^2)^{3/2}) \\
 J(6,1) &= 1
 \end{aligned}$$

Matriz de inercia de su modelo dinámico:

$$\begin{aligned}
 M(1,1) &= (81 * m2 * \cos(q2)^2)/4 - 27 * m3 + 9 * m3 * \cos(q2)^2 + 27 * m3 * \cos(q2 + q3/2)^2 + \\
 &\quad 27 * m3 * \cos(q3/2)^2 + (81 * m3 * \cos(q2 + q3)^2)/4 \\
 M(1,2) &= M(1,3) = M(2,1) = M(3,1) = 0 \\
 M(2,2) &= (81 * m2)/4 + (117 * m3)/4 + 27 * m3 * \cos(q3) \\
 M(2,3) &= (27 * m3 * (2 * \cos(q3) + 3))/4 \\
 M(3,2) &= (27 * m3 * (2 * \cos(q3) + 3))/4 \\
 M(3,3) &= (81 * m3)/4
 \end{aligned}$$

El modelo dinámico completo se muestra en el Apéndice A.1.

La cinemática inversa se obtienen numéricamente para cada posición cartesiana del extremo final, los resultados se muestran en la tabla 4.2.5. Estos obtenidos utilizando el algoritmo evolutivo EVONORM, con el cual se puede ver que el error de posición es muy pequeño.

i	$q1$	$q2$	$q3$	Error Pos
1	18.435	321.1057	112.8854	$3.1020e^{-7}$
2	30.96	319.84	118.17	$2.351e^{-7}$
3	45	319.47	120	$1.5758e^{-7}$
4	59.0362	319.8425	118.1786	$4.0029e^{-7}$
5	71.565	73.9911	247.1146	$3.1259e^{-7}$

Tabla 4.2.5: Cinemática inversa del manipulador angular de 3 GDL.

Luego de la obtención de los modelos cinemáticos y dinámicos mostrados, se calcula el índice de desempeño del manipulador para cada posición cartesiana indicada en la Tabla 4.2.1 . Los resultados de la evaluación se muestran en la tabla 4.2.6.

i	CN_{min}	FM_{max}	KM_{max}	SV_{max}	Wd_{max}	WI_{min}
1	0.5708	$2.027e^{-7}$	0.584	0.3372	0.0021	1.0928
2	0.5556	$2.2985e^{-7}$	0.5711	0.3376	0.0021	1.1024
3	0.5516	$2.4115e^{-7}$	0.5666	0.3376	0.0021	1.1055
4	0.5556	$2.2985e^{-7}$	0.5711	0.3376	0.0021	1.1024
5	0.5708	$2.027e^{-7}$	0.5841	0.3372	0.0026	1.0414

Tabla 4.2.6: Resultados de índices de desempeño del manipulador angular de 3 GDL.

4.2.2 Desempeño de robot serial SCARA de 4 grados de libertad

El mismo proceso es repetido con el robot SCARA, cuyos parámetros son mostrados en la Tabla 4.2, del cual se obtienen del algoritmo los siguientes resultados :

Modelo cinemático directo de posición y orientación del extremo final del manipulador:

$$Px = 3 * \cos(q1) + 3 * \cos(q1) * \cos(q2) - 3 * \sin(q1) * \sin(q2)$$

$$Py = 3 * \sin(q1) + 3 * \cos(q1) * \sin(q2) + 3 * \cos(q2) * \sin(q1)$$

$$Pz = -q4$$

$$Rx = Ry = 0$$

$$Rz = \text{atan2}((\cos(q3) * (\cos(q1) * \sin(q2) + \cos(q2) * \sin(q1)) + \sin(q3) * (\cos(q1) * \cos(q2) - \sin(q1) * \sin(q2)))/(\cos(q3) * (\cos(q1) * \cos(q2) - \sin(q1) * \sin(q2)) - \sin(q3) * (\cos(q1) * \sin(q2) + \cos(q2) * \sin(q1))))$$

Elementos de la matriz Jacobiana:

$$J(1, 1) = -3 * \sin(q1) - 3 * \cos(q1) * \sin(q2) - 3 * \cos(q2) * \sin(q1)$$

$$J(1, 2) = -3 * \cos(q1) * \sin(q2) - 3 * \cos(q2) * \sin(q1)$$

$$J(2, 1) = 3 * \cos(q1) + 3 * \cos(q1) * \cos(q2) - 3 * \sin(q1) * \sin(q2)$$

$$J(2, 2) = 3 * \cos(q1) * \cos(q2) - 3 * \sin(q1) * \sin(q2)$$

$$J(3, 4) = -1$$

$$J(6, 1) = J(6, 2) = J(6, 3) = 1$$

$$J(1, 3) = J(1, 4) = J(2, 3) = J(2, 4) = J(3, 1) = J(3, 2) = J(3, 3) = J(4, 1) = J(4, 2) = 0$$

$$J(4, 3) = J(4, 4) = J(5, 1) = J(5, 2) = J(5, 3) = J(5, 4) = J(6, 4) = 0$$

Matriz de inercia de su modelo dinámico:

$$\begin{aligned}
 M(1, 1) &= (81 * m1)/4 + (9 * m2)/4 + 54 * m2 * \cos(q2/2)^2 + 36 * m3 * \cos(q2/2)^2 + 36 * \\
 &\quad m4 * \cos(q2/2)^2 \\
 M(1, 2) &= (27 * m2)/4 + 27 * m2 * \cos(q2/2)^2 + 18 * m3 * \cos(q2/2)^2 + 18 * m4 * \cos(q2/2)^2 \\
 M(2, 1) &= (27 * m2)/4 + 27 * m2 * \cos(q2/2)^2 + 18 * m3 * \cos(q2/2)^2 + 18 * m4 * \cos(q2/2)^2 \\
 M(2, 2) &= (81 * m2)/4 + 9 * m3 + 9 * m4 \\
 M(4, 4) &= m4 \\
 M(1, 3) &= M(1, 4) = M(2, 3) = M(2, 4) = M(3, 1) = M(3, 2) = M(3, 3) = 0 \\
 M(3, 4) &= M(4, 1) = M(4, 2) = M(4, 3) = 0
 \end{aligned}$$

El modelo dinámico completo se muestra en el Apéndice A.2.

La cinemática inversa se obtienen numéricamente para cada posición cartesiana del extremo final, los resultados se muestran en la Tabla 4.2.7. Estos obtenidos utilizando el algoritmo evolutivo EVONORM, con el cual se puede ver que el error de posición es muy pequeño.

i	$q1$	$q2$	$q3$	$q4$	Error Pos
1	76.628	243.6122	63.054	1m	$2.137e^{-7}$
2	91.89	238.14	217.45	1m	$2.52e^{-7}$
3	106.87	236.25	352.31	1m	$1.941e^{-7}$
4	119.96	238.14	349.16	1m	$1.3347e^{-7}$
5	129.75	243.61	215.03	1m	$2.434e^{-7}$

Tabla 4.2.7: Cinemática inversa de manipulador SCARA.

Luego de la obtención de los modelos cinemáticos y dinámicos mostrados, se calcula el índice de desempeño del manipulador para cada posición cartesiana indicada en la Tabla 4.2.1 . Los resultados de la evaluación se muestran en la Tabla 4.2.8.

i	CN_{min}	FM_{max}	KM_{max}	SV_{max}	Wd_{max}	WI_{min}
1	1.1878	$6.594e^{-7}$	0.6057	0.763	0.0084	0.7089
2	1.158	$9.954e^{-7}$	0.0636	0.198	0.0084	0.738
3	1.148	$7.104e^{-7}$	0.0719	0.289	0.0084	0.749
4	1.158	$6.954e^{-7}$	0.433	0.1592	0.0084	0.738
5	1.187	$6.594e^{-7}$	7.798	0.3042	0.0084	0.7089

Tabla 4.2.8: Resultados de índices de desempeño del manipulador SCARA.

4.2.3 Desempeño de robot angular de 6 grados de libertad

El mismo proceso es repetido con el robot angular de 6 GDL, cuyos parámetros son mostrados en la Tabla 4.2, del cual se obtienen del algoritmo los siguientes resultados :

Modelo cinemático directo de posición y orientación del extremo final del manipulador:

$$\begin{aligned}
Px &= 3 * \sin(q1) * \sin(q3) - (\cos(q3) * \sin(q1))/10 - 3 * \sin(q1) - 3 * \cos(q1) * \cos(q2) * \\
&\quad \cos(q3) - (\cos(q1) * \cos(q2) * \sin(q3))/10 + (\cos(q5) * \sin(q1) * \sin(q3))/5 - (\cos(q1) * \\
&\quad \cos(q2) * \cos(q3) * \cos(q5))/5 + (\cos(q3) * \cos(q4) * \sin(q1) * \sin(q5))/5 + (\cos(q1) * \\
&\quad \sin(q2) * \sin(q4) * \sin(q5))/5 + (\cos(q1) * \cos(q2) * \cos(q4) * \sin(q3) * \sin(q5))/5 \\
Py &= 3 * \cos(q1) + (\cos(q1) * \cos(q3))/10 - 3 * \cos(q1) * \sin(q3) - 3 * \cos(q2) * \cos(q3) * \\
&\quad \sin(q1) - (\cos(q1) * \cos(q5) * \sin(q3))/5 - (\cos(q2) * \sin(q1) * \sin(q3))/10 - (\cos(q2) * \\
&\quad \cos(q3) * \cos(q5) * \sin(q1))/5 - (\cos(q1) * \cos(q3) * \cos(q4) * \sin(q5))/5 + (\sin(q1) * \\
&\quad \sin(q2) * \sin(q4) * \sin(q5))/5 + (\cos(q2) * \cos(q4) * \sin(q1) * \sin(q3) * \sin(q5))/5 \\
Pz &= 3 * \cos(q3) * \sin(q2) + (\sin(q2) * \sin(q3))/10 + (\cos(q3) * \cos(q5) * \sin(q2))/5 + (\cos(q2) * \\
&\quad \sin(q4) * \sin(q5))/5 - (\cos(q4) * \sin(q2) * \sin(q3) * \sin(q5))/5 \\
\mu &= -atan((\sin(q6) * (\cos(q5) * (\cos(q2) * \sin(q4) - \cos(q4) * \cos(q3 - pi/2) * \sin(q2)) + \\
&\quad \sin(q2) * \sin(q5) * \sin(q3 - pi/2)) - \cos(q6) * (\cos(q2) * \cos(q4) + \cos(q3 - pi/2) * \\
&\quad \sin(q2) * \sin(q4)))/(\sin(q5) * (\cos(q2) * \sin(q4) - \cos(q4) * \cos(q3 - pi/2) * \sin(q2)) - \\
&\quad \cos(q5) * \sin(q2) * \sin(q3 - pi/2))) \\
\beta &= -atan((\cos(q6) * (\cos(q5) * (\cos(q2) * \sin(q4) - \cos(q4) * \cos(q3 - pi/2) * \sin(q2)) \\
&\quad + \sin(q2) * \sin(q5) * \sin(q3 - pi/2)) + \sin(q6) * (\cos(q2) * \cos(q4) + \cos(q3 - pi/2) * \\
&\quad \sin(q2) * \sin(q4)))/((\sin(q6) * (\sin(q4) * (\sin(q1) * \sin(q3 - pi/2) - \cos(q1) * \\
&\quad \cos(q2) * \cos(q3 - pi/2)) + \cos(q1) * \cos(q4) * \sin(q2)) - \cos(q6) * (\cos(q5) * (\cos(q4) * \\
&\quad (\sin(q1) * \sin(q3 - pi/2) - \cos(q1) * \cos(q2) * \cos(q3 - pi/2)) - \cos(q1) * \sin(q2) * \sin(q4)) + \\
&\quad \sin(q5) * (\cos(q3 - pi/2) * \sin(q1) + \cos(q1) * \cos(q2) * \sin(q3 - pi/2))))^2 + (\sin(q6) * \\
&\quad (\sin(q4) * (\cos(q1) * \sin(q3 - pi/2) + \cos(q2) * \cos(q3 - pi/2) * \sin(q1)) - \cos(q4) * \sin(q1) * \\
&\quad \sin(q2)) - \cos(q6) * (\cos(q5) * (\cos(q4) * (\cos(q1) * \sin(q3 - pi/2) + \cos(q2) * \cos(q3 - \\
&\quad pi/2) * \sin(q1)) + \sin(q1) * \sin(q2) * \sin(q4)) + \sin(q5) * (\cos(q1) * \cos(q3 - pi/2) - \\
&\quad \cos(q2) * \sin(q1) * \sin(q3 - pi/2))))^2(1/2))
\end{aligned}$$

$$\gamma = -atan((sin(q6) * (sin(q4) * (cos(q1) * sin(q3 - pi/2) + cos(q2) * cos(q3 - pi/2) * sin(q1)) - cos(q4) * sin(q1) * sin(q2)) - cos(q6) * (cos(q5) * (cos(q4) * (cos(q1) * sin(q3 - pi/2) + cos(q2) * cos(q3 - pi/2) * sin(q1)) + sin(q1) * sin(q2) * sin(q4)) + sin(q5) * (cos(q1) * cos(q3 - pi/2) - cos(q2) * sin(q1) * sin(q3 - pi/2)))))) / (sin(q6) * (sin(q4) * (sin(q1) * sin(q3 - pi/2) - cos(q1) * cos(q2) * cos(q3 - pi/2)) + cos(q1) * cos(q4) * sin(q2)) - cos(q6) * (cos(q5) * (cos(q4) * (sin(q1) * sin(q3 - pi/2) - cos(q1) * cos(q2) * cos(q3 - pi/2)) - cos(q1) * sin(q2) * sin(q4)) + sin(q5) * (cos(q3 - pi/2) * sin(q1) + cos(q1) * cos(q2) * sin(q3 - pi/2)))))))$$

La matriz jacobiana se muestra en el Apéndice A.3.

La cinemática inversa se obtiene numéricamente para cada posición cartesiana del extremo final, los resultados se muestran en la Tabla 4.2.9. Estos obtenidos utilizando el algoritmo evolutivo EVONORM, con el cual se puede ver que el error de posición es muy pequeño.

i	q_1	q_2	q_3	q_4	q_5	q_6	Error Pos
1	237.071	337.103	20.412	16.82	136.49	0	$5.519e^{-7}$
2	353.4604	23.378	156.13	188.58	179.707	334.599	$7.191e^{-7}$
3	252.34	158.48	150.7	171.6	17.74	359.8	$8.86e^{-7}$
4	267.39	158.59	152.61	4.46	340.05	2.01	$6.6081e^{-7}$
5	285.12	155.70	156.69	290.09	66.78	122.8	$5.909e^{-7}$

Tabla 4.2.9: Matriz con resultados de cinemática inversa de la tarea.

Luego de la obtención de los modelos cinemáticos y dinámicos mostrados, se calcula el índice de desempeño del manipulador para cada posición cartesiana indicada en la Tabla 4.2.1 . Los resultados de la evaluación se muestran en la Tabla 4.2.10.

i	CN	FM	KM	SV	Wd	WI
1	1.7743	$3.4464e^{-7}$	0.0622	0.1876	$2.9127e^{-21}$	1.017
2	179.98	$4.7765e^{-5}$	$4.48e^{-4}$	0.0015	$4.7497e^{-21}$	0.9558
3	3.691	$9.3985e^{-7}$	0.0235	0.0926	$5.62e^{-21}$	0.9214
4	3.519	$7.571e^{-7}$	0.0409	0.2221	$8.557e^{-21}$	0.9534
5	1.88	$1.89e^{-7}$	0.203	0.2615	$4.57e^{-22}$	1.117

Tabla 4.2.10: Matriz con resultados de índices de desempeño del manipulador angular de 6 GDL.

4.2.4 Comparación y selección de manipuladores

A continuación se muestra la tabla comparativa de los índices de desempeño de los robots mostrados anteriormente, en la cual se muestran con números resaltados los resultados del mejor robot para cada posición de la tarea. Los subíndices *min* y *max* representan que dicho índice se debe de minimizar o maximizar respectivamente.

R	Pos	$NC_{(Min)}$	$MF_{(Max)}$	$MC_{(Max)}$	$MVS_{(Max)}$	$DM_{(Max)}$	$DI_{(Min)}$
A	1	0.570	$2.02e^{-7}$	0.584	0.337	0.002	1.092
N	2	0.555	$2.29e^{-7}$	0.571	0.337	0.002	1.102
G	3	0.551	$2.41e^{-7}$	0.566	0.337	0.002	1.105
3	4	0.555	$2.29e^{-7}$	0.571	0.337	0.002	1.102
D	5	0.570	$2.02e^{-7}$	0.584	0.337	0.002	1.041
S	1	1.187	$6.59e^{-7}$	0.605	0.763	0.008	0.708
C	2	1.158	$9.95e^{-7}$	0.063	0.198	0.008	0.738
A	3	1.148	$7.10e^{-7}$	0.071	0.289	0.008	0.749
R	4	1.158	$6.95e^{-7}$	0.433	0.159	0.008	0.738
A	5	1.187	$6.59e^{-7}$	7.798	0.304	0.008	0.708
A	1	1.774	$3.44e^{-7}$	0.062	0.187	$2.912e^{-21}$	1.017
N	2	179.9	$4.77e^{-5}$	$4.48e^{-4}$	0.001	$4.749e^{-21}$	0.9558
G	3	3.691	$9.39e^{-7}$	0.0235	0.0926	$5.62e^{-21}$	0.9214
6	4	3.519	$7.57e^{-7}$	0.0409	0.2221	$8.557e^{-21}$	0.9534
D	5	1.88	$1.89e^{-7}$	0.203	0.2615	$4.57e^{-22}$	1.117

Tabla 4.2.11: Tabla comparando el desempeño de los robots en las posiciones asignadas de la tarea por el usuario.

En esta tabla se puede apreciar que el robot angular de 3 grados de libertad tiene mejores valores en la mayoría de sus índices cinetostáticos, y en cambio el robot SCARA presenta mejor valor en los índices dinámicos.

Con estos resultados, si en la tarea a realizar, no es necesaria el ejercer fuerzas en el efector final, es recomendable el robot SCARA, sin embargo, si no es necesario aplicar esfuerzos, se recomienda el robot angular de 3 grados de libertad.

4.3 Manipulador diseñado para optimizar los índices de desempeño en una tarea propuesta

En esta sección se plantea un escenario en el que el algoritmo evolutivo mutiobjetivo modifica o genera los parámetros d_i, θ, α, a , que corresponden a nuevos manipuladores con características o parámetros hipotéticos y se evalúa su desempeño.

Este algoritmo realiza un proceso evolutivo hasta encontrar los parámetros d_i, θ, α, a que maximizan u optimizan los índices de la sección anterior, y por tanto esos son los parámetros que corresponden a un nuevo manipulador diseño para realizar optimamente la tarea específica analizada bajo los índices de desempeño considerados.

4.3.1 Dimensionamiento de manipulador angular de 3 grados de libertad generado por algoritmo evolutivo

Después de probar los robots predeterminados conocidos por el usuario, se generaron los parámetros del robot, los cuales optimizan el índice de desempeño multiobjetivo de la Ecuación 3.3.2. En la Tabla 4.3.12 se muestran los parámetros generados por el algoritmo:

θ	$d(mm)$	$a(mm)$	α	$m_x(mm)$	$m_y(mm)$	$m_z(mm)$	$m(kg)$
q_1	1,000	1,316.9	1.9951	650	0	500	2
q_2	2,604.8	1,547.5	2.4079	770	0	1,300	3
$-q_3$	2,047	1,896	0	2,394.4	940	1,000	1.76

Tabla 4.3.12: Parámetros del manipulador de 3 GDL Evolutivo.

Correspondiente a los parámetros anteriores se obtuvo la cinemática inversa con los mejores índices de desempeño, estos se muestran en la Tabla 4.3.13.

i	$q1$	$q2$	$q3$	Error Pos
1	38.17	50.78	145.89	$1.4454e^{-4}$
2	49.7786	48.71	152.01	$1.3542e^{-4}$
3	26.7686	306.2058	302.7681	$1.275e^{-4}$
4	41.8145	308.53	304.3838	$2.4298e^{-4}$
5	57.11	314.98	309.471	$1.3487e^{-4}$

Tabla 4.3.13: Resultados de cinemática inversa manipulador angular de 3DOF.

Así como sus respectivos índices de desempeño que se obtienen para cada posición:

i	CN	FM	KM	SV	Wd	WI
1	0.5224	$5.101e^{-7}$	3.1498	0.7656	$2.4394e^{-4}$	0.5638
2	0.5307	$5.6427e^{-7}$	1.8641	0.6994	$1.5803e^{-4}$	0.5638
3	0.6150	$4.7368e^{-7}$	0.6956	0.4453	$1.805e^{-4}$	0.5638
4	0.6243	$4.5438e^{-7}$	0.6858	0.4349	$1.7874e^{-4}$	0.5638
5	0.6548	$4.0812e^{-7}$	0.6551	0.4019	$1.7199e^{-4}$	0.5638

Tabla 4.3.14: Resultados de índices de desempeño del manipulador angular de 3 GDL.

4.3.2 Dimensionamiento de robot SCARA generado por algoritmo evolutivo

Ya obtenidos los resultados del manipulador angular de 3 GDL, se prosiguió a realizar el mismo procedimiento con el manipulador SCARA, en la Tabla 4.3.15 se muestran los parámetros generados por el algoritmo:

θ	$d(mm)$	$a(mm)$	α	$m_x(mm)$	$m_y(mm)$	$m_z(mm)$	$m(kg)$
q_1	1,542.6	2,412	2.06	1,200	0	770	2.56
q_2	2,460	1,302.6	1.667	650	0	1,230	3
$-q_3$	1,049	1,542.7	2.16	770	0	770	1.72
3	q_4	1,125.2	2.82	560	0	$q_4/2$	3

Tabla 4.3.15: Matriz de datos robot SCARA.

Correspondiente a los parámetros anteriores se obtuvo la cinemática inversa con los mejores índices de desempeño, estos se muestran en la Tabla 4.3.16.

i	q_1	q_2	q_3	q_4	Error Pos
1	76.628	243.6122	63.054	1m	$2.137e^{-7}$
2	91.89	238.14	217.45	1m	$2.52e^{-7}$
3	106.87	236.25	352.31	1m	$1.941e^{-7}$
4	119.96	238.14	349.16	1m	$1.3347e^{-7}$
5	129.75	243.61	215.03	1m	$2.434e^{-7}$

Tabla 4.3.16: Resultados de cinemática inversa manipulador SCARA.

Así como sus respectivos índices de desempeño que se obtienen para cada posición:

i	CN	FM	KM	SV	Wd	WI
1	1.653	$9.129e^{-7}$	0.1325	0.763	0.0084	0.7089
2	1.158	$9.954e^{-7}$	0.0636	0.198	0.0084	0.738
3	1.148	$7.104e^{-7}$	0.0719	0.289	0.0084	0.749
4	1.158	$6.954e^{-7}$	0.433	0.1592	0.0084	0.738
5	1.187	$6.594e^{-7}$	7.798	0.3042	0.0084	0.7089

Tabla 4.3.17: Resultados de índices de desempeño del manipulador SCARA.

4.3.3 Dimensionamiento de robot 6DOF generado por algoritmo evolutivo

Ya obtenidos los resultados del robot angular de 3 GDL y SCARA, se prosiguió a realizar el mismo procedimiento con el manipulador de 6 GDL, en la Tabla 4.3.18 se muestran los parámetros generados por el algoritmo:

θ	$d(mm)$	$a(mm)$	α	$m_x(mm)$	$m_y(mm)$	$m_z(mm)$	$m(kg)$
q_1	1,206	2,270	2.26	1,110	0	600	1.6
q_2	2,752	1,887	1.986	950	0	1,350	2.25
q_3	2,260	1,496	1.02	740	0	1,130	1.22
q_4	1,000	1,000	1.25	500	0	500	2.25
q_5	1,677	1,995	1.558	990	0	830	1.22
q_6	3,000	1,119	1.540	550	0	1,500	2.01

Tabla 4.3.18: Parámetros del manipulador 6 GDL.

Correspondiente a los parámetros anteriores se obtuvo la cinemática inversa con los mejores índices de desempeño, estos se muestran en la Tabla 4.3.19.

i	q_1	q_2	q_3	q_4	q_5	q_6	Error Pos
1	97.1794	112.1164	3.0023	355.79	284.4907	4.2628	$6.1173e^{-4}$
2	55.4050	357.4053	250.3138	51.0162	254.7313	272.9686	$5.7294e^{-4}$
3	89.7366	353.0394	152.4412	353.9791	316.6394	287.0347	$5.820e^{-4}$
4	49.4405	306.1600	1.0256	359.5196	280.3368	203.4860	$5.96e^{-4}$
3	94.6297	37.9241	336.9794	0.3037	251.4483	336.3033	$5.3480e^{-4}$

Tabla 4.3.19: Resultados de cinemática inversa manipulador 6 GDL.

Así como sus respectivos índices de desempeño que se obtienen para cada posición:

i	CN	FM	KM	SV	Wd	WI
1	3.0209	$3.5133e^{-7}$	0.0610	0.1991	$1.1618e^{-9}$	0.5821
2	4.397	$7.8033e^{-7}$	4.7379	0.2104	$1.304e^{-9}$	0.5821
3	4.0072	$1.081e^{-6}$	24.14	0.0983	$1.06e^{-9}$	0.5821
4	1.7735	$2.2462e^{-7}$	0.0954	0.5891	$1.5101e^{-9}$	0.5821
5	2.7449	$1.9311e^{-7}$	0.111	0.6849	$1.6307e^{-9}$	0.5821

Tabla 4.3.20: Resultado de índices de desempeño manipulador angular 6 GDL.

4.3.4 Comparación y selección de manipuladores

A continuación se muestra la tabla comparativa de los índices de desempeño de los robots mostrados anteriormente, en la cual se muestran con números resaltados los resultados del mejor robot para cada posición de la tarea. Los subíndices *min* y *max* representan que dicho índice se debe de minimizar o maximizar respectivamente.

R	Pos	$NC_{(Min)}$	$MC_{(Max)}$	$MF_{(Max)}$	$MVS_{(Max)}$	$DM_{(Max)}$	$DI_{(Min)}$
A	1	0.674	1.98e⁻⁶	0.727	0.486	0.065	0.552
N	2	0.712	2.1e⁻⁶	0.612	0.409	0.080	0.552
G	3	0.693	2.16e⁻⁶	0.641	0.432	0.082	0.552
3	4	0.702	2.11e⁻⁶	0.623	0.418	0.060	0.552
D	5	0.672	1.98e⁻⁶	0.727	0.486	0.065	0.552
S	1	1.2572	6.534e ⁻⁷	0.057	0.156	0.005	0.518
C	2	1.2793	7.841e ⁻⁷	0.057	0.156	0.005	0.518
A	3	1.0997	4.875e ⁻⁷	0.076	0.157	0.005	0.518
R	4	1.2302	7.093e ⁻⁷	0.055	0.156	0.005	0.518
A	5	1.2551	6.497e ⁻⁷	0.058	0.156	0.002	0.518
A	1	3.0209	3.5133e ⁻⁷	0.0610	0.1991	1.1618e ⁻⁹	0.5821
N	2	4.397	7.8033e ⁻⁷	4.7379	0.2104	1.304e ⁻⁹	0.5821
G	3	4.0072	1.081e ⁻⁶	24.14	0.0983	1.06e ⁻⁹	0.5821
6	4	1.7735	2.2462e ⁻⁷	0.0954	0.5891	1.5101e ⁻⁹	0.5821
D	5	2.7449	1.9311e ⁻⁷	0.111	0.6849	1.6307e ⁻⁹	0.5821

Tabla 4.3.21: Tabla comparativa de índices de desempeño entre manipuladores.

Como se puede apreciar en la tabla mostrada anterior, el mejor manipulador generado por el algoritmo es el manipulador angular 3 GDL, el cual gana en prácticamente todos los índices. Este diseño cabe recordar que fue realizado de forma automatizada por el algoritmo, solo se muestran en la tabla anterior sus resultados.

Este robot resultante sobresale tanto en actividades que no involucran mucha fuerza, así como en tareas que también involucran fuerzas.

4.4 Manipulador para soldadura en superficies de tubulares

En este caso se presenta un manipulador soldador que suelda sobre un tubo metálico, dicho robot se compara con uno generado por el algoritmo generador de robots.

La tarea a realizar consiste en que los robots lleguen a las siguientes posiciones coordenadas

(Tabla 4.4.22) las cuales representan la posición final en el efector final del robot, y se pueden ver más claramente en la Figura 4.4.

$x(mm)$	$y(mm)$	$z(mm)$	μ	β	γ
600.00	0	302.50	0°	89°	0°
598.23	0	301.76	0°	45°	0°
597.50	0	300.00	0°	0°	0°
598.23	0	298.23	0°	-45°	0°
600.00	0	297.50	0°	-89°	0°

Tabla 4.4.22: Tabla de posiciones en el plano cartesiano por las que debe pasar el efector final del robot.

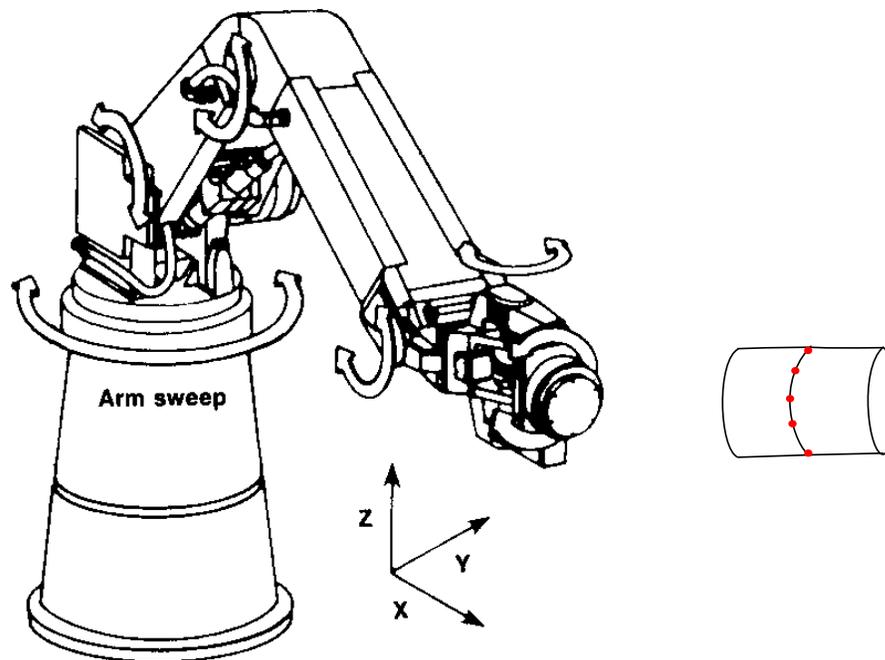


Figura 4.4: Caso de robot soldador en superficie tubular.

En esta tarea se comparan los resultados entre un robot FANUC Arc Mate 100; y un robot el cual fue generado por el algoritmo, el cual fue diseñado con el objetivo de tener mejor desempeño en esta tarea.

Los parámetros Denavit-Hartenberg así como los centros de masa de cada robot evaluado se muestran en la Tabla 4.4.23.

θ	$d(mm)$	$a(mm)$	α	$m_x(mm)$	$m_y(mm)$	$m_z(mm)$	$m(kg)$
q_1	450	150	$\pi/2$	50	0	150	7.23
$q_2 + \pi/2$	0	600	0	200	0	0	7.23
$q_3 - \pi/2$	0	50	$-\pi/2$	17	0	0	2.4
q_4	0	640	$\pi/2$	215	0	0	7.1
q_5	0	100	0	35	0	0	1.2
0	0	1	$q_6 - \pi/2$	0	0	0.05	1.2

Tabla 4.4.23: Parámetros del manipulador 6 GDL Fanuc.

Así como los parámetros del manipulador de 6 GDL generado por el algoritmo desarrollado, los datos son mostrados en la Tabla 4.4.24.

θ	$d(mm)$	$a(mm)$	α	$m_x(mm)$	$m_y(mm)$	$m_z(mm)$	$m(kg)$
q_1	0	300	$\pi/2$	0	0	150	1
q_2	0	250	0	83.3	0	0	5
q_3	0	20	$-\pi/2$	17	0	6.6	2.2
q_4	0	220	$\pi/2$	215	0	73.3	3
q_5	0	10	0	35	0	3.33	0.5
0	0	0	q_6	0	0	0	0.5

Tabla 4.4.24: Parámetros del manipulador 6 GDL generado por algoritmo.

Posteriormente se muestran los valores de la cinemática inversa con los cuales se obtienen los mejores índices de desempeño para el robot Fanuc en la Tabla 4.4.25 y para el robot generado por el algoritmo en la Tabla 4.4.26.

i	q_1	q_2	q_3	q_4	q_5	q_6	Error Pos
1	360.0008	183.2147	229.8535	360.0008	217.936	360.0008	0.0213
2	360.0008	172.4145	236.1388	360.0008	266.4426	360.0008	0.0275
3	360.0008	333.2265	303.1978	360.0008	83.5716	360.0008	0.0249
4	360.0008	321.1887	308.9331	360.0008	134.8800	360.0008	0.0215
5	360.0008	164.5363	223.6369	360.0008	60.8424	359.9951	0.0278

Tabla 4.4.25: Resultados de cinemática inversa manipulador 6 GDL.

i	q_1	q_2	q_3	q_4	q_5	q_6	Error Pos
1	360.0008	18.9101	74.6567	0.0012	89.0057	359.9965	0.1889
2	360.015	17.3756	77.3480	359.9995	45.0036	359.9938	$1.2170e^{-4}$
3	360.0073	14.9375	78.9626	0.0012	359.9998	359.9991	0.0017
4	0.00087	12.9216	78.5874	0.00122	315.0001	359.9943	$2.2087e^{-4}$
5	359.9965	12.4893	76.5701	0.0012	271.9935	359.99936	$6.8311e^{-5}$

Tabla 4.4.26: Resultados de cinemática inversa manipulador 6 GDL generado por algoritmo.

Así como sus respectivos índices de desempeño que se para obtienen dichas posiciones en la Tabla 4.4.27 para el robot Fanuc, y para el robot generado por el algoritmo en la Tabla 4.4.28.

i	CN	FM	KM	SV	Wd	WI
1	55.5151	$7.2261e^{-16}$	$2.9661e^7$	0.7095	$3.6194e^{-33}$	1.1855
2	72.0705	$3.3583e^{-14}$	$8.7694e^5$	0.5859	$1.2295e^{-33}$	1.1867
3	79.6260	$4.0141e^{-14}$	$5.3396e^5$	0.5197	$1.8644e^{-33}$	1.3629
4	53.7017	$1.7520e^{-14}$	$1.3404e^6$	0.7583	$3.0149e^{-33}$	1.4126
5	59.5371	$1.0869e^{-15}$	$1.9720e^7$	0.7033	$3.7201e^{-33}$	1.1313

Tabla 4.4.27: Resultado de índices de desempeño manipulador angular 6 GDL Fanuc.

i	<i>CN</i>	<i>FM</i>	<i>KM</i>	<i>SV</i>	<i>Wd</i>	<i>WI</i>
1	42.4552	$3.2280e^{-15}$	$6.6399e^6$	0.7186	$2.0870e^{-13}$	1.0650
2	44.3262	$1.5337e^{-13}$	$1.5641e^5$	0.5935	$3.8987e^{-15}$	1.0797
3	74.0444	$9.1670e^{-13}$	$2.3381e^4$	0.5237	$6.1000e^{-16}$	1.0968
4	52.1371	$2.0412e^{-13}$	$1.1824e^5$	0.4261	$2.8767e^{-15}$	1.1055
5	44.3514	$2.9547e^{-14}$	$2.7150e^7$	0.7060	$2.0381e^{-13}$	1.1010

Tabla 4.4.28: Resultado de índices de desempeño manipulador angular 6 GDL generado por algoritmo.

Ya teniendo las dos evaluaciones, pasamos a comparar cada uno de los dos manipuladores en la Tabla 4.4.29 .

<i>Pos</i>	<i>NC_(Min)</i>	<i>MC_(Max)</i>	<i>MF_(Max)</i>	<i>MVS_(Max)</i>	<i>DM_(Max)</i>	<i>DI_(Min)</i>
1	55.5151	$7.2261e^{-16}$	2.9661e⁷	0.7095	$3.6194e^{-33}$	1.1855
2	72.0705	$3.3583e^{-14}$	8.7694e⁵	0.5859	$1.2295e^{-33}$	1.1867
3	79.6260	$4.0141e^{-14}$	5.3396e⁵	0.5197	$1.8644e^{-33}$	1.3629
4	53.7017	$1.7520e^{-14}$	1.3404e⁶	0.7583	$3.0149e^{-33}$	1.4126
5	59.5371	$1.0869e^{-15}$	1.9720e⁷	0.7033	$3.7201e^{-33}$	1.1313
1	42.4552	3.2280e⁻¹⁵	$6.6399e^6$	0.7186	2.0870e⁻¹³	1.0650
2	44.3262	1.5337e⁻¹³	$1.5641e^5$	0.5935	3.8987e⁻¹⁵	1.0797
3	74.0444	9.1670e⁻¹³	$2.3381e^4$	0.5237	6.1000e⁻¹⁶	1.0968
4	52.1371	2.0412e⁻¹³	$1.1824e^5$	0.4261	2.8767e⁻¹⁵	1.1055
5	44.3514	2.9547e⁻¹⁴	$2.7150e^7$	0.7060	2.0381e⁻¹³	1.1010

Tabla 4.4.29: Tabla comparativa de índices de desempeño entre manipuladores.

En la Tabla 4.4.29 se puede observar que el robot que mejores índices de desempeño obtuvo es el manipulador generado por el algoritmo evolutivo, el cual sobresale en cinco de los seis índices que evaluamos.

4.5 Conclusiones de Capítulo

En este capítulo se presentaron 2 casos de estudio, en el cual se revisa que tan bien realizan la tarea una serie de robots asignados, y de los cuales se realizan las observaciones de cual es el robot que se considera mejor a utilizar, de acuerdo a los resultados obtenidos.

Así mismo también se muestra que con el algoritmo obtenido, se genera las funciones simbólicas del modelo cinemático y dinámico de los robots. Las cuales se generan de forma automática por el algoritmo.

Entre los principales resultados destacan que en los dos casos de estudio los manipuladores generados por el algoritmo superan en la mayoría de los índices de desempeño, lo cual comprueba que el algoritmo realmente optimiza para la tarea que se esta realizando.

Conclusiones y problemas abiertos

En este trabajo de investigación se ha propuesto un esquema de diseño de manipuladores seriales basado en índices de desempeño cinetostáticos y dinámicos así como en un algoritmo evolutivo, con los cuales se busca optimizar sus parámetros de diseño para que estos valores sean los idóneos para realizar dicha tarea.

La principal ventaja es que el algoritmo sólo requiere como datos de entrada el total de grados de libertad, y especificación entre articulación rotativa o prismática, así como las posiciones del efector final del robot en la tarea que se desea realizar.

El algoritmo genera la siguiente información:

- Modelo cinemático directo de posición y orientación (simbólico).
- Modelo cinemático inverso (numérico).
- Modelo cinemático diferencial o de velocidad (simbólico).
- Modelo dinámico inverso (simbólico).
- Valores de parámetros de diseño del manipulador óptimo.

La metodología y algoritmo propuestos en este trabajo puede usarse o modificarse para aplicaciones, tales como: Selección de robots, diseño de robots, análisis de movimiento y dinámica, logística en manufactura robótica, entre otros.

Como trabajos futuros, será conveniente estudiar los siguientes casos no resueltos en este trabajo:

- *Generación de los modelos cinemáticos y dinámicos de robots paralelos.*
- *Optimización del código .-* Mejora del tiempo de ejecución.
- *Diseño del robot sin una estructura específica.-* Omitir la necesidad de que se seleccione manualmente que tipo de estructura se desee que se genere.
- *Migración de código a un lenguaje libre (open source).-* Migrar código a lenguaje de programación que sea abierto y gratuito, el cual mantenga la propiedad de ser multiplataforma.
- *Utilizar índices de desempeño de espacio de trabajo.-* Incluir más índices de desempeño en el algoritmo con los que se puedan analizar otras propiedades de la tarea.
- *Incluir los efectos del controlador en el análisis presentado.*

Bibliografía

- [1] Jorge Angeles, *Fundamentals of robotic mechanical systems*, Springer, (2007).
- [2] British Automation and Robot association, *Robot facts 2012*, Self-Published (2012).
- [3] A. Barrientos and C. Balaguer, *Fundamentos de robótica, 2a ed.*, McGraw-Hill Interamericana de España S.L., (2007).
- [4] IFR Statistical Department, *Statistics, market analysis, forecasts, case studies and profitability of robot investment*, Self-Published (2011).
- [5] C. y. B. C. Doty, K. Melchiorri, *A theory of generalized inverse a to robotics.*, Int. J. Robot. (1993).
- [6] Randy L Haupt and Sue Ellen Haupt, *Practical genetic algorithms*, Wiley-Interscience, 2004.
- [7] Charles A. Klein and Bruce E. Blaho, *Dexterity measures for the design and control of kinematically redundant manipulators*, The International Journal of Robotics Research **6** (1987), no. 2, 72–83.
- [8] Duffy JJ. Lipkin HH, *Hybrid twist and wrench control for a robotic manipulator.*, J. of Mech., Trans, and Automation. **110** (1988), no. 2, 138–144.
- [9] O. Ma and J. Angeles, *The concept of dynamic isotropy and its applications to inverse kinematics and trajectory planning*, Robotics and Automation, 1990. Proceedings., 1990 IEEE International Conference on, (1990), pp. 481–486 vol.1.
- [10] Mitchell Melanie, *An introduction to genetic algorithms*, Cambridge, Massachusetts London, England, Fifth printing **3** (1999).
- [11] Héctor A. Moreno, Roque Saltaren, Isela Carrera, Lisandro Puglisi, and Rafael Aracil, *Índices de desempeño de robots manipuladores: una revisión del estado del arte*, Revista Iberoamericana de Automática e Informática Industrial {RIAI} **9** (2012), no. 2, 111 – 122.
- [12] J. Kenneth Salisbury and John J. Craig, *Articulated hands: Force control and kinematic issues*, The International Journal of Robotics Research **1** (1982), no. 1, 4–17.
- [13] Luis Torres, *Evonorm, a new evolutionary algorithm to continuous optimization*, Genetics and Evolutionary Computation 2006, (2006).

- [14] Luis M. Torres-Treviño, Felipe A. Reyes-Valdes, Victor López, and Rolando Praga-Alejo, *Multi-objective optimization of a welding process by the estimation of the pareto optimal set*, Expert Systems with Applications **38** (2011), no. 7, 8045 – 8053.
- [15] Thomas Weise, *Global optimization algorithms–theory and application*, Self-Published (2009).
- [16] T. Yoshikawa, *Dynamic manipulability of robot manipulators*, Robotics and Automation. Proceedings. 1985 IEEE International Conference on, vol. 2, 1985, pp. 1033–1038.
- [17] Tsuneo Yoshikawa, *Manipulability of robotic mechanisms*, The International Journal of Robotics Research **4** ((1985)), no. 2, 3–9.

Apéndice A

Modelo dinámico y cinemático de manipuladores utilizados en experimentos

A.1 Manipulador angular serial de 3 grados de libertad

Ecuaciones dinámicas del robot:

$$\begin{aligned}
 \tau(1) = & (45 * m2 * \text{diff}(q1(t), t, t))/8 + (63 * m3 * \text{diff}(q1(t), t, t))/8 + (27 * m3 * \cos(q2 - q3) \\
 & * \text{diff}(q1(t), t, t))/4 + (9 * m3 * \cos(q2 - 2 * q3) * \text{diff}(q1(t), t, t))/2 + (27 * m3 * \cos(3 * q2 \\
 & + q3) * \text{diff}(q1(t), t, t))/4 + (9 * m2 * \cos(2 * q2) * \text{diff}(q1(t), t, t))/8 + (9 * m2 * \cos(3 * q2) * \\
 & \text{diff}(q1(t), t, t))/2 + (9 * m2 * \cos(4 * q2) * \text{diff}(q1(t), t, t))/2 + (27 * m3 * \cos(2 * q2 + 2 * \\
 & q3) * \text{diff}(q1(t), t, t))/8 + 9 * m3 * \cos(2 * q2 + 3 * q3) * \text{diff}(q1(t), t, t) + (9 * m3 * \cos(3 * \\
 & q2 + 2 * q3) * \text{diff}(q1(t), t, t))/2 + (9 * m3 * \cos(2 * q2 + 4 * q3) * \text{diff}(q1(t), t, t))/2 + (9 * \\
 & m2 * \cos(q2) * \text{diff}(q1(t), t, t))/2 + 9 * m3 * \cos(q3) * \text{diff}(q1(t), t, t) + (9 * m2 * \sin(q2) * \\
 & \text{diff}(q1(t), t) * \text{diff}(q2(t), t))/2 + (63 * m3 * \sin(q2) * \text{diff}(q1(t), t) * \text{diff}(q2(t), t))/8 + (9 * \\
 & m3 * \sin(q3) * \text{diff}(q1(t), t) * \text{diff}(q2(t), t))/4 + (27 * m3 * \sin(q3) * \text{diff}(q1(t), t) * \text{diff}(q3(t), \\
 & t))/8 - (9 * m3 * \sin(q2 - 2 * q3) * \text{diff}(q1(t), t) * \text{diff}(q2(t), t))/2 - (27 * m3 * \sin(q2 + 2 * \\
 & q3) * \text{diff}(q1(t), t) * \text{diff}(q2(t), t))/8 - (27 * m3 * \sin(2 * q2 + q3) * \text{diff}(q1(t), t) * \text{diff}(q2(t), \\
 & t))/4 - (27 * m3 * \sin(2 * q2 + q3) * \text{diff}(q1(t), t) * \text{diff}(q3(t), t))/8 - 9 * m3 * \sin(q2 + 3 * q3) * \\
 & \text{diff}(q1(t), t) * \text{diff}(q2(t), t) - (27 * m3 * \sin(3 * q2 + q3) * \text{diff}(q1(t), t) * \text{diff}(q2(t), t))/4 - \\
 & (9 * m3 * \sin(q2 + 4 * q3) * \text{diff}(q1(t), t) * \text{diff}(q2(t), t))/2 - (27 * m2 * \sin(2 * q2) * \text{diff} \\
 & (q1(t), t) * \text{diff}(q2(t), t))/8 - 9 * m2 * \sin(3 * q2) * \text{diff}(q1(t), t) * \text{diff}(q2(t), t) - (9 * m2 * \\
 & \sin(4 * q2) * \text{diff}(q1(t), t) * \text{diff}(q2(t), t))/2 + (9 * m3 * \sin(2 * q3) * \text{diff}(q1(t), t) * \text{diff} \\
 & (q2(t), t))/2 + (9 * m3 * \sin(2 * q3) * \text{diff}(q1(t), t) * \text{diff}(q3(t), t))/4 - (63 * m3 * \sin(2 * \\
 & q2 + 2 * q3) * \text{diff}(q1(t), t) * \text{diff}(q2(t), t))/8 - (99 * m3 * \sin(2 * q2 + 2 * q3) * \text{diff}(q1(t), \\
 & t) * \text{diff}(q3(t), t))/8 - 9 * m3 * \sin(2 * q2 + 3 * q3) * \text{diff}(q1(t), t) * \text{diff}(q2(t), t) - (9 * m3 * \\
 & \sin(3 * q2 + 2 * q3) * \text{diff}(q1(t), t) * \text{diff}(q2(t), t))/2 - (27 * m3 * \sin(2 * q2 + 3 * q3) *
 \end{aligned}$$

$$\begin{aligned} & diff(q1(t), t) * diff(q3(t), t))/2 - (9 * m3 * sin(2 * q2 + 4 * q3) * diff(q1(t), t) * diff(q2(t), t))/2 - \\ & (9 * m3 * sin(2 * q2 + 4q3) * diff(q1(t), t) * diff(q3(t), t))/2 + (9 * m3 * sin(q2 + q3) * \\ & diff(q1(t), t) * diff(q2(t), t))/4 \end{aligned}$$

$$\begin{aligned} \tau(2) = & (45 * m2 * diff(q1(t), t, t))/8 + (63 * m3 * diff(q1(t), t, t))/8 + (27 * m3 * cos(q2 - q3) * \\ & diff(q1(t), t, t))/4 + (9 * m3 * cos(q2 - 2 * q3) * diff(q1(t), t, t))/2 + (27 * m3 * cos(3 * q2 + \\ & q3) * diff(q1(t), t, t))/4 + (9 * m2 * cos(2 * q2) * diff(q1(t), t, t))/8 + (9 * m2 * cos(3 * q2) * \\ & diff(q1(t), t, t))/2 + (9 * m2 * cos(4 * q2) * diff(q1(t), t, t))/2 + (27 * m3 * cos(2 * q2 + 2 * q3) \\ & * diff(q1(t), t, t))/8 + 9 * m3 * cos(2 * q2 + 3 * q3) * diff(q1(t), t, t) + (9 * m3 * cos(3 * q2 + 2 * \\ & q3) * diff(q1(t), t, t))/2 + (9 * m3 * cos(2 * q2 + 4 * q3) * diff(q1(t), t, t))/2 + (9 * m2 * cos(q2) * \\ & diff(q1(t), t, t))/2 + 9 * m3 * cos(q3) * diff(q1(t), t, t) + (9 * m2 * sin(q2) * diff(q1(t), t) * \\ & diff(q2(t), t))/2 + (63 * m3 * sin(q2) * diff(q1(t), t) * diff(q2(t), t))/8 + (9 * m3 * sin(q3) * \\ & diff(q1(t), t) * diff(q2(t), t))/4 + (27 * m3 * sin(q3) * diff(q1(t), t) * diff(q3(t), t))/8 - (9 * \\ & m3 * sin(q2 - 2 * q3) * diff(q1(t), t) * diff(q2(t), t))/2 - (27 * m3 * sin(q2 + 2 * q3) * diff \\ & (q1(t), t) * diff(q2(t), t))/8 - (27 * m3 * sin(2 * q2 + q3) * diff(q1(t), t) * diff(q2(t), t))/4 - \\ & (27 * m3 * sin(2 * q2 + q3) * diff(q1(t), t) * diff(q3(t), t))/8 - 9 * m3 * sin(q2 + 3 * q3) * \\ & diff(q1(t), t) * diff(q2(t), t) - (27 * m3 * sin(3 * q2 + q3) * diff(q1(t), t) * diff(q2(t), t))/ \\ & 4 - (9 * m3 * sin(q2 + 4 * q3) * diff(q1(t), t) * diff(q2(t), t))/2 - (27 * m2 * sin(2 * q2) * \\ & diff(q1(t), t) * diff(q2(t), t))/8 - 9 * m2 * sin(3 * q2) * diff(q1(t), t) * diff(q2(t), t) - (9 * m2 \\ & * sin(4 * q2) * diff(q1(t), t) * diff(q2(t), t))/2 + (9 * m3 * sin(2 * q3) * diff(q1(t), t) * \\ & diff(q2(t), t))/2 + (9 * m3 * sin(2 * q3) * diff(q1(t), t) * diff(q3(t), t))/4 - (63 * m3 * sin \\ & (2 * q2 + 2 * q3) * diff(q1(t), t) * diff(q2(t), t))/8 - (99 * m3 * sin(2 * q2 + 2 * q3) * diff \\ & (q1(t), t) * diff(q3(t), t))/8 - 9 * m3 * sin(2 * q2 + 3 * q3) * diff(q1(t), t) * diff(q2(t), t) - \\ & (9 * m3 * sin(3 * q2 + 2 * q3) * diff(q1(t), t) * diff(q2(t), t))/2 - (27 * m3 * sin(2 * q2 + 3 * \\ & q3) * diff(q1(t), t) * diff(q3(t), t))/2 - (9 * m3 * sin(2 * q2 + 4 * q3) * diff(q1(t), t) * diff \\ & (q2(t), t))/2 - (9 * m3 * sin(2 * q2 + 4 * q3) * diff(q1(t), t) * diff(q3(t), t))/2 + (9 * m3 * \\ & sin(q2 + q3) * diff(q1(t), t) * diff(q2(t), t))/4 \end{aligned}$$

$$\begin{aligned}
\tau(3) = & (45 * m3 * diff(q2(t), t, t))/4 + (45 * m3 * diff(q3(t), t, t))/4 + (9 * m3 * cos(q2 - q3) * \\
& diff(q2(t), t, t))/2 + 9 * m3 * cos(q2 - 2 * q3) * diff(q2(t), t, t) + 3 * g * m3 * cos(q2 + 2 * \\
& q3) + (9 * m3 * sin(2 * q2 + 2 * q3) * diff(q1(t), t)^2)/8 + (9 * m3 * sin(2 * q2 + 3 * q3) * \\
& diff(q1(t), t)^2)/2 + (9 * m3 * sin(3 * q2 + 2 * q3) * diff(q1(t), t)^2)/2 + (9 * m3 * sin(2 * \\
& q2 + 4 * q3) * diff(q1(t), t)^2)/2 + 9 * m3 * cos(q3) * diff(q2(t), t, t) + 9 * m3 * cos(q3) * \\
& diff(q3(t), t, t) + (3 * g * m3 * cos(q2 + q3))/2 - (9 * m3 * sin(q2 - q3) * diff(q1(t), t)^2)/4 - \\
& (9 * m3 * sin(q2 - q3) * diff(q2(t), t)^2)/2 - (9 * m3 * sin(q2 - 2 * q3) * diff(q1(t), t)^2)/2 - \\
& 9 * m3 * sin(q2 - 2 * q3) * diff(q2(t), t)^2 + (9 * m3 * sin(3 * q2 + q3) * diff(q1(t), t)^2)/4
\end{aligned}$$

A.2 Manipulador SCARA de 4 grados de libertad

Ecuaciones dinámicas del robot:

$$\begin{aligned}
\tau(1) = & (45 * m1 * \text{diff}(q1(t), t, t))/4 + (63 * m2 * \text{diff}(q1(t), t, t))/4 + (63 * m2 * \text{diff}(q2(t), t, t))/4 \\
& + (31 * m3 * \text{diff}(q1(t), t, t))/100 + (31 * m3 * \text{diff}(q2(t), t, t))/100 - (31 * m3 * \text{diff}(q3(t), t, \\
& t))/100 + (27 * m2 * \cos(q1 - q2) * \text{diff}(q1(t), t, t))/2 + 9 * m3 * \cos(q1 - q2) * \text{diff}(q1(t), t, t) + \\
& \text{diff}(q1(t), t)^2)/10 + (93 * m3 * \sin(q2 + q3) * \text{diff}(q2(t), t)^2)/10 + 9 * m4 * \sin(q2 + q3) * \\
& \text{diff}(q1(t), t)^2 + 9 * m4 * \sin(q2 + q3) * \text{diff}(q2(t), t)^2 + (93 * m3 * \cos(q2 + q3) * \\
& \text{diff}(q1(t), t, t))/10 + (93 * m3 * \cos(q2 + q3) * \text{diff}(q2(t), t, t))/10 + 9 * m4 * \cos(q2 + q3) * \\
& \text{diff}(q1(t), t, t) + 9 * m4 * \cos(q2 + q3) * \text{diff}(q2(t), t, t) - 9 * m2 * \sin(q1) * \text{diff}(q1(t), t)^2 + \\
& 9 * m2 * \sin(q2) * \text{diff}(q1(t), t)^2 + 9 * m2 * \sin(q2) * \text{diff}(q2(t), t)^2 + 9 * m3 * \sin(q2) * \\
& \text{diff}(q1(t), t)^2 + 9 * m3 * \sin(q2) * \text{diff}(q2(t), t)^2 - (3 * m3 * \sin(q3) * \text{diff}(q1(t), t)^2)/10 + \\
& 9 * m4 * \sin(q2) * \text{diff}(q1(t), t)^2 - (3 * m3 * \sin(q3) * \text{diff}(q2(t), t)^2)/10 + 9 * m4 * \sin(q2) * \\
& \text{diff}(q2(t), t)^2 - (3 * m3 * \sin(q3) * \text{diff}(q3(t), t)^2)/10 - (93 * m3 * \sin(q1 - q2 - q3) * \\
& \text{diff}(q1(t), t)^2)/10 - 9 * m4 * \sin(q1 - q2 - q3) * \text{diff}(q1(t), t)^2 + 9 * m1 * \cos(q1) * \\
& \text{diff}(q1(t), t, t) + 9 * m2 * \cos(q1) * \text{diff}(q1(t), t, t) + 18 * m2 * \cos(q2) * \text{diff}(q1(t), t, t) + \\
& 18 * m2 * \cos(q2) * \text{diff}(q2(t), t, t) + 9 * m3 * \cos(q2) * \text{diff}(q1(t), t, t) + 9 * m3 * \cos(q2) * \\
& \text{diff}(q2(t), t, t) + (3 * m3 * \cos(q3) * \text{diff}(q1(t), t, t))/10 + 9 * m4 * \cos(q2) * \text{diff}(q1(t), t, t) + \\
& (3 * m3 * \cos(q3) * \text{diff}(q2(t), t, t))/10 + 9 * m4 * \cos(q2) * \text{diff}(q2(t), t, t) - (3 * m3 * \cos(q3) * \\
& \text{diff}(q3(t), t, t))/10 + (93 * m3 * \cos(q1 - q2 - q3) * \text{diff}(q1(t), t, t))/10 + 9 * m4 * \cos(q1 - q2 \\
& - q3) * \text{diff}(q1(t), t, t) - (27 * m2 * \sin(q1 - q2) * \text{diff}(q1(t), t)^2)/2 - 9 * m3 * \sin(q1 - q2) * \\
& \text{diff}(q1(t), t)^2 - 9 * m4 * \sin(q1 - q2) * \text{diff}(q1(t), t)^2 + 18 * m2 * \sin(q2) * \text{diff}(q1(t), t) * \\
& \text{diff}(q2(t), t) + 18 * m3 * \sin(q2) * \text{diff}(q1(t), t) * \text{diff}(q2(t), t) - (3 * m3 * \sin(q3) * \\
& \text{diff}(q1(t), t) * \text{diff}(q2(t), t))/5 + 18 * m4 * \sin(q2) * \text{diff}(q1(t), t) * \text{diff}(q2(t), t) + (3 * m3 * \\
& \sin(q3) * \text{diff}(q1(t), t) * \text{diff}(q3(t), t))/5 + (3 * m3 * \sin(q3) * \text{diff}(q2(t), t) * \text{diff}(q3(t), t))/5 \\
& + (93 * m3 * \sin(q2 + q3) * \text{diff}(q1(t), t) * \text{diff}(q2(t), t))/5 + 18 * m4 * \sin(q2 + q3) * \\
& \text{diff}(q1(t), t) * \text{diff}(q2(t), t)
\end{aligned}$$

$$\begin{aligned}
\tau(2) = & (m3 * (\cos(q3) * (\sin(q2) * (3 * \cos(q1) * \text{diff}(q1(t), t)^2 + 3 * \sin(q1) * \text{diff}(q1(t), t, t)) - \\
& \cos(q2) * (3 * \sin(q1) * \text{diff}(q1(t), t)^2 - 3 * \cos(q1) * \text{diff}(q1(t), t, t)) + 3 * \sin(q2) * \\
& (\text{diff}(q1(t), t) + \text{diff}(q2(t), t))^2 + 3 * \cos(q2) * (\text{diff}(q1(t), t, t) + \text{diff}(q2(t), t, t))) + \sin(q3) * \\
& (\cos(q2) * (3 * \cos(q1) * \text{diff}(q1(t), t)^2 + 3 * \sin(q1) * \text{diff}(q1(t), t, t)) - 3 * \sin(q2) * \\
& (\text{diff}(q1(t), t, t) + \text{diff}(q2(t), t, t)) + \sin(q2) * (3 * \sin(q1) * \text{diff}(q1(t), t)^2 - 3 * \cos(q1) * \\
& \text{diff}(q1(t), t, t)) + 3 * \cos(q2) * (\text{diff}(q1(t), t) + \text{diff}(q2(t), t))^2) + \text{diff}(q1(t), t, t)/10 + \\
& \text{diff}(q2(t), t, t)/10 - \text{diff}(q3(t), t, t)/10))/10 + (3 * \cos(q2)^2 + 3 * \sin(q2)^2) * (m3 * (\cos(q3) * \\
& (\sin(q2) * (3 * \cos(q1) * \text{diff}(q1(t), t)^2 + 3 * \sin(q1) * \text{diff}(q1(t), t, t)) - \cos(q2) * (3 * \sin(q1) * \\
& \text{diff}(q1(t), t)^2 - 3 * \cos(q1) * \text{diff}(q1(t), t, t)) + 3 * \sin(q2) * (\text{diff}(q1(t), t) + \text{diff}(q2(t), t))^2 + \\
& 3 * \cos(q2) * (\text{diff}(q1(t), t, t) + \text{diff}(q2(t), t, t))) + \sin(q3) * (\cos(q2) * (3 * \cos(q1) * + \\
& \text{diff}(q1(t), t)^2 3 * \sin(q1) * \text{diff}(q1(t), t, t)) - 3 * \sin(q2) * (\text{diff}(q1(t), t, t) + \text{diff}(q2(t), t, t)) + \\
& \sin(q2) * (3 * \sin(q1) * \text{diff}(q1(t), t)^2 - 3 * \cos(q1) * \text{diff}(q1(t), t, t)) + 3 * \cos(q2) * \\
& (\text{diff}(q1(t), t) + \text{diff}(q2(t), t))^2) + \text{diff}(q1(t), t, t)/10 + \text{diff}(q2(t), t, t)/10 - \\
& \text{diff}(q3(t), t, t)/10) + m4 * (\cos(q3) * (\sin(q2) * (3 * \cos(q1) * \text{diff}(q1(t), t)^2 + 3 * \sin(q1) * \\
& \text{diff}(q1(t), t, t)) - \cos(q2) * (3 * \sin(q1) * \text{diff}(q1(t), t)^2 - 3 * \cos(q1) * \text{diff}(q1(t), t, t)) + 3 * \\
& \sin(q2) * (\text{diff}(q1(t), t) + \text{diff}(q2(t), t))^2 + 3 * \cos(q2) * (\text{diff}(q1(t), t, t) + \text{diff}(q2(t), t, t))) + \\
& \sin(q3) * (\cos(q2) * (3 * \cos(q1) * \text{diff}(q1(t), t)^2 + 3 * \sin(q1) * \text{diff}(q1(t), t, t)) - 3 * \sin(q2) * \\
& (\text{diff}(q1(t), t, t) + \text{diff}(q2(t), t, t)) + \sin(q2) * (3 * \sin(q1) * \text{diff}(q1(t), t)^2 - 3 * \cos(q1) * \\
& \text{diff}(q1(t), t, t)) + 3 * \cos(q2) * (\text{diff}(q1(t), t) + \text{diff}(q2(t), t))^2))) + m2 * (3 * \cos(q2) + 3/2) * \\
& (\sin(q2) * (3 * \cos(q1) * \text{diff}(q1(t), t)^2 + 3 * \sin(q1) * \text{diff}(q1(t), t, t)) - \cos(q2) * (3 * \sin(q1) * \\
& \text{diff}(q1(t), t)^2 - 3 * \cos(q1) * \text{diff}(q1(t), t, t)) + 3 * \sin(q2) * (\text{diff}(q1(t), t) + \text{diff}(q2(t), t))^2 + \\
& 3 * \cos(q2) * (\text{diff}(q1(t), t, t) + \text{diff}(q2(t), t, t)) + (3 * \text{diff}(q1(t), t, t))/2 + (3 * \\
& \text{diff}(q2(t), t, t))/2) - 3 * m2 * \sin(q2) * (\cos(q2) * (3 * \cos(q1) * \text{diff}(q1(t), t)^2 + 3 * \sin(q1) * \\
& \text{diff}(q1(t), t, t)) - 3 * \sin(q2) * (\text{diff}(q1(t), t, t) + \text{diff}(q2(t), t, t)) + \sin(q2) * (3 * \sin(q1) * \\
& \text{diff}(q1(t), t)^2 - 3 * \cos(q1) * \text{diff}(q1(t), t, t)) + 3 * \cos(q2) * (\text{diff}(q1(t), t) + \text{diff}(q2(t), t))^2 + \\
& ((3 * \text{diff}(q1(t), t))/2 + (3 * \text{diff}(q2(t), t))/2) * (\text{diff}(q1(t), t) + \text{diff}(q2(t), t))))
\end{aligned}$$

$$\begin{aligned}
\tau(3) = & -(m3 * (\cos(q3) * (\sin(q2) * (3 * \cos(q1) * \text{diff}(q1(t), t)^2 + 3 * \sin(q1) * \text{diff}(q1(t), t, t)) - \\
& \cos(q2) * (3 * \sin(q1) * \text{diff}(q1(t), t)^2 - 3 * \cos(q1) * \text{diff}(q1(t), t, t)) + 3 * \sin(q2) * \\
& (\text{diff}(q1(t), t) + \text{diff}(q2(t), t))^2 + 3 * \cos(q2) * (\text{diff}(q1(t), t, t) + \text{diff}(q2(t), t, t))) + \sin(q3) * \\
& (\cos(q2) * (3 * \cos(q1) * \text{diff}(q1(t), t)^2 + 3 * \sin(q1) * \text{diff}(q1(t), t, t)) - 3 * \sin(q2) * \\
& (\text{diff}(q1(t), t, t) + \text{diff}(q2(t), t, t)) + \sin(q2) * (3 * \sin(q1) * \text{diff}(q1(t), t)^2 - 3 * \cos(q1) * \\
& \text{diff}(q1(t), t, t)) + 3 * \cos(q2) * (\text{diff}(q1(t), t) + \text{diff}(q2(t), t))^2) + \text{diff}(q1(t), t, t)/10 + \\
& \text{diff}(q2(t), t, t)/10 - \text{diff}(q3(t), t, t)/10))/10 \\
\tau(4) = & -m4 * (g - \text{diff}(q4(t), t, t))
\end{aligned}$$

A.3 Manipulador angular de 6 grados de libertad

Elementos de la Matriz Jacobiana:

$$\begin{aligned}
J(1, 1) = & 3 * \cos(q1) * \sin(q3) - (\cos(q1) * \cos(q3))/10 - 3 * \cos(q1) + 3 * \cos(q2) * \cos(q3) * \sin(q1) \\
& + (\cos(q1) * \cos(q5) * \sin(q3))/5 + \cos(q2) * \sin(q1) * \sin(q3))/10 + (\cos(q2) * \cos(q3) * \\
& (\cos(q5) * \sin(q1))/5 + (\cos(q1) * \cos(q3) * \cos(q4) * \sin(q5))/5 - (\sin(q1) * \sin(q2) * \\
& \sin(q4) * \sin(q5))/5 - (\cos(q2) * \cos(q4) * \sin(q1) * \sin(q3) * \sin(q5))/5 \\
J(1, 2) = & (\cos(q1) * (30 * \cos(q3) * \sin(q2) + \sin(q2) * \sin(q3) + 2 * \cos(q3) * \cos(q5) * \sin(q2) + \\
& 2 * \cos(q2) * \sin(q4) * \sin(q5) - 2 * \cos(q4) * \sin(q2) * \sin(q3) * \sin(q5)))/10 \\
J(1, 3) = & 3 * \cos(q3) * \sin(q1) + (\sin(q1) * \sin(q3))/10 - (\cos(q1) * \cos(q2) * \cos(q3))/10 + 3 * \\
& \cos(q1) * \cos(q2) * \sin(q3) + (\cos(q3) * \cos(q5) * \sin(q1))/5 + (\cos(q1) * \cos(q2) * \cos(q5) * \\
& \sin(q3))/5 - (\cos(q4) * \sin(q1) * \sin(q3) * \sin(q5))/5 + (\cos(q1) * \cos(q2) * \cos(q3) * \\
& \cos(q4) * \sin(q5))/5 \\
J(1, 4) = & - (\sin(q5) * (\cos(q3) * \sin(q1) * \sin(q4) - \cos(q1) * \cos(q4) * \sin(q2) + \cos(q1) * \cos(q2) * \\
& \sin(q3) * \sin(q4)))/5 \\
J(1, 5) = & - (\cos(q5) * (\cos(q4) * (\sin(q1) * \sin(q3 - \pi/2) - \cos(q1) * \cos(q2) * \cos(q3 - \pi/2)) - \\
& \cos(q1) * \sin(q2) * \sin(q4)))/5 - (\sin(q5) * (\cos(q3 - \pi/2) * \sin(q1) + \cos(q1) * \cos(q2) * \\
& \sin(q3 - \pi/2)))/5 \\
J(2, 1) = & 3 * \sin(q1) * \sin(q3) - (\cos(q3) * \sin(q1))/10 - 3 * \sin(q1) - 3 * \cos(q1) * \cos(q2) * \\
& \cos(q3) - (\cos(q1) * \cos(q2) * \sin(q3))/10 + (\cos(q5) * \sin(q1) * \sin(q3))/5 - (\cos(q1) * \\
& \cos(q2) * \cos(q3) * \cos(q5))/5 + (\cos(q3) * \cos(q4) * \sin(q1) * \sin(q5))/5 + (\cos(q1) * \\
& \sin(q2) * \sin(q4) * \sin(q5))/5 + (\cos(q1) * \cos(q2) * \cos(q4) * \sin(q3) * \sin(q5))/5
\end{aligned}$$

$$\begin{aligned}
J(2, 2) &= (\sin(q1) * (30 * \cos(q3) * \sin(q2) + \sin(q2) * \sin(q3) + 2 * \cos(q3) * \cos(q5) * \sin(q2) + \\
&\quad 2 * \cos(q2) * \sin(q4) * \sin(q5) - 2 * \cos(q4) * \sin(q2) * \sin(q3) * \sin(q5))) / 10 \\
J(2, 3) &= 3 * \cos(q2) * \sin(q1) * \sin(q3) - (\cos(q1) * \sin(q3)) / 10 - (\cos(q1) * \cos(q3) * \cos(q5)) / 5 - \\
&\quad (\cos(q2) * \cos(q3) * \sin(q1)) / 10 - 3 * \cos(q1) * \cos(q3) + (\cos(q2) * \cos(q5) * \sin(q1) * \\
&\quad \sin(q3)) / 5 + (\cos(q1) * \cos(q4) * \sin(q3) * \sin(q5)) / 5 + (\cos(q2) * \cos(q3) * \cos(q4) * \\
&\quad \sin(q1) * \sin(q5)) / 5 \\
J(2, 4) &= (\sin(q5) * (\cos(q1) * \cos(q3) * \sin(q4) + \cos(q4) * \sin(q1) * \sin(q2) - \cos(q2) * \sin(q1) * \\
&\quad \sin(q3) * \sin(q4))) / 5 \\
J(2, 5) &= (\cos(q5) * (\cos(q4) * (\cos(q1) * \sin(q3 - \pi/2) + \cos(q2) * \cos(q3 - \pi/2) * \sin(q1)) + \\
&\quad \sin(q1) * \sin(q2) * \sin(q4))) / 5 + (\sin(q5) * (\cos(q1) * \cos(q3 - \pi/2) - \cos(q2) * \sin(q1) * \\
&\quad \sin(q3 - \pi/2))) / 5 \\
J(3, 2) &= 3 * \cos(q2) * \cos(q3) + (\cos(q2) * \sin(q3)) / 10 - (\sin(q2) * \sin(q4) * \sin(q5)) / 5 + (\cos(q2) * \\
&\quad \cos(q3) * \cos(q5)) / 5 - (\cos(q2) * \cos(q4) * \sin(q3) * \sin(q5)) / 5 \\
J(3, 3) &= - (\sin(q2) * (30 * \sin(q3) - \cos(q3) + 2 * \cos(q5) * \sin(q3) + 2 * \cos(q3) * \cos(q4) * \\
&\quad \sin(q5))) / 10 \\
J(3, 4) &= (\sin(q5) * (\cos(q2) * \cos(q4) + \sin(q2) * \sin(q3) * \sin(q4))) / 5 \\
J(3, 5) &= (\cos(q2) * \cos(q5) * \sin(q4)) / 5 - (\cos(q3) * \sin(q2) * \sin(q5)) / 5 - (\cos(q4) * \cos(q5) * \\
&\quad \sin(q2) * \sin(q3)) / 5 \\
J(4, 2) &= ((\sin(q6) * (\cos(q5) * (\sin(q2) * \sin(q4) + \cos(q2) * \cos(q4) * \cos(q3 - \pi/2)) - \cos(q2) * \\
&\quad \sin(q5) * \sin(q3 - \pi/2)) - \cos(q6) * (\cos(q4) * \sin(q2) - \cos(q2) * \cos(q3 - \pi/2) * \\
&\quad \sin(q4))) / (\sin(q5) * (\cos(q2) * \sin(q4) - \cos(q4) * \cos(q3 - \pi/2) * \sin(q2)) - \cos(q5) * \\
&\quad \sin(q2) * \sin(q3 - \pi/2)) - ((\sin(q5) * (\sin(q2) * \sin(q4) + \cos(q2) * \cos(q4) * \\
&\quad \cos(q3 - \pi/2)) + \cos(q2) * \cos(q5) * \sin(q3 - \pi/2)) * (\sin(q6) * (\cos(q5) * (\cos(q2) \\
&\quad * \sin(q4) - \cos(q4) * \cos(q3 - \pi/2) * \sin(q2)) + \sin(q2) * \sin(q5) * \sin(q3 - \pi/2)) - \\
&\quad \cos(q6) * (\cos(q2) * \cos(q4) + \cos(q3 - \pi/2) * \sin(q2) * \sin(q4)))) / (\sin(q5) * \\
&\quad (\cos(q2) * \sin(q4) - \cos(q4) * \cos(q3 - \pi/2) * \sin(q2)) - \cos(q5) * \sin(q2) * \sin(q3 - \\
&\quad \pi/2))^2) / ((\sin(q6) * (\cos(q5) * (\cos(q2) * \sin(q4) - \cos(q4) * \cos(q3 - \pi/2) * \sin(q2)) + \\
&\quad \sin(q2) * \sin(q5) * \sin(q3 - \pi/2)) - \cos(q6) * (\cos(q2) * \cos(q4) + \cos(q3 - \pi/2) * \\
&\quad \sin(q2) * \sin(q4)))^2 / (\sin(q5) * (\cos(q2) * \sin(q4) - \cos(q4) * \cos(q3 - \pi/2) * \sin(q2)) - \\
&\quad \cos(q5) * \sin(q2) * \sin(q3 - \pi/2))^2 + 1)
\end{aligned}$$

$$\begin{aligned}
J(4, 3) = & - ((\sin(q6) * (\cos(q3 - \pi/2) * \sin(q2) * \sin(q5) + \cos(q4) * \cos(q5) * \sin(q2) \\
& * \sin(q3 - \pi/2)) + \cos(q6) * \sin(q2) * \sin(q4) * \sin(q3 - \pi/2)) / (\sin(q5) * (\cos(q2) * \\
& \sin(q4) - \cos(q4) * \cos(q3 - \pi/2) * \sin(q2)) - \cos(q5) * \sin(q2) * \sin(q3 - \pi/2)) + \\
& ((\cos(q5) * \cos(q3 - \pi/2) * \sin(q2) - \cos(q4) * \sin(q2) * \sin(q5) * \sin(q3 - \pi/2)) * \\
& (\sin(q6) * (\cos(q5) * (\cos(q2) * \sin(q4) - \cos(q4) * \cos(q3 - \pi/2) * \sin(q2)) + \\
& \sin(q2) * \sin(q5) * \sin(q3 - \pi/2)) - \cos(q6) * (\cos(q2) * \cos(q4) + \cos(q3 - \pi/2) * \\
& \sin(q2) * \sin(q4)))) / (\sin(q5) * (\cos(q2) * \sin(q4) - \cos(q4) * \cos(q3 - \pi/2) * \sin(q2)) - \\
& \cos(q5) * \sin(q2) * \sin(q3 - \pi/2))^2) / ((\sin(q6) * (\cos(q5) * (\cos(q2) * \sin(q4) - \cos(q4) * \\
& \cos(q3 - \pi/2) * \sin(q2)) + \sin(q2) * \sin(q5) * \sin(q3 - \pi/2)) - \cos(q6) * (\cos(q2) * \\
& \cos(q4) + \cos(q3 - \pi/2) * \sin(q2) * \sin(q4)))^2 / (\sin(q5) * (\cos(q2) * \sin(q4) - \cos(q4) * \\
& \cos(q3 - \pi/2) * \sin(q2)) - \cos(q5) * \sin(q2) * \sin(q3 - \pi/2))^2 + 1) \\
J(4, 4) = & - ((\cos(q6) * (\cos(q2) * \sin(q4) - \cos(q4) * \cos(q3 - \pi/2) * \sin(q2)) + \cos(q5) * \sin(q6) * \\
& (\cos(q2) * \cos(q4) + \cos(q3 - \pi/2) * \sin(q2) * \sin(q4))) / (\sin(q5) * (\cos(q2) * \sin(q4) - \\
& \cos(q4) * \cos(q3 - \pi/2) * \sin(q2)) - \cos(q5) * \sin(q2) * \sin(q3 - \pi/2)) - (\sin(q5) * \\
& (\sin(q6) * (\cos(q5) * (\cos(q2) * \sin(q4) - \cos(q4) * \cos(q3 - \pi/2) * \sin(q2)) + \sin(q2) * \\
& \sin(q5) * \sin(q3 - \pi/2)) - \cos(q6) * (\cos(q2) * \cos(q4) + \cos(q3 - \pi/2) * \sin(q2) * \\
& \sin(q4))) * (\cos(q2) * \cos(q4) + \cos(q3 - \pi/2) * \sin(q2) * \sin(q4))) / (\sin(q5) * (\cos(q2) * \\
& \sin(q4) - \cos(q4) * \cos(q3 - \pi/2) * \sin(q2)) - \cos(q5) * \sin(q2) * \sin(q3 - \pi/2))^2) / \\
& ((\sin(q6) * (\cos(q5) * (\cos(q2) * \sin(q4) - \cos(q4) * \cos(q3 - \pi/2) * \sin(q2)) + \sin(q2) * \\
& \sin(q5) * \sin(q3 - \pi/2)) - \cos(q6) * (\cos(q2) * \cos(q4) + \cos(q3 - \pi/2) * \sin(q2) * \\
& \sin(q4)))^2 / (\sin(q5) * (\cos(q2) * \sin(q4) - \cos(q4) * \cos(q3 - \pi/2) * \sin(q2)) - \cos(q5) * \\
& \sin(q2) * \sin(q3 - \pi/2))^2 + 1) \\
J(4, 5) = & (\sin(q6) + ((\cos(q5) * (\cos(q2) * \sin(q4) - \cos(q4) * \cos(q3 - \pi/2) * \sin(q2)) + \sin(q2) * \\
& \sin(q5) * \sin(q3 - \pi/2)) * (\sin(q6) * (\cos(q5) * (\cos(q2) * \sin(q4) - \cos(q4) * \\
& \cos(q3 - \pi/2) * \sin(q2)) + \sin(q2) * \sin(q5) * \sin(q3 - \pi/2)) - \cos(q6) * (\cos(q2) * \cos(q4) \\
& + \cos(q3 - \pi/2) * \sin(q2) * \sin(q4)))) / (\sin(q5) * (\cos(q2) * \sin(q4) - \cos(q4) * \cos(q3 \\
& - \pi/2) * \sin(q2)) - \cos(q5) * \sin(q2) * \sin(q3 - \pi/2))^2) / ((\sin(q6) * (\cos(q5) * (\cos(q2) * \\
& \sin(q4) - \cos(q4) * \cos(q3 - \pi/2) * \sin(q2)) + \sin(q2) * \sin(q5) * \sin(q3 - \pi/2)) - \\
& \cos(q6) * (\cos(q2) * \cos(q4) + \cos(q3 - \pi/2) * \sin(q2) * \sin(q4)))^2 / (\sin(q5) * (\cos(q2) * \\
& \sin(q4) - \cos(q4) * \cos(q3 - \pi/2) * \sin(q2)) - \cos(q5) * \sin(q2) * \sin(q3 - \pi/2))^2 + 1)
\end{aligned}$$

$$\begin{aligned}
J(4, 6) = & - (\cos(q6) * (\cos(q5) * (\cos(q2) * \sin(q4) - \cos(q4) * \cos(q3 - \pi/2) * \sin(q2)) + \sin(q2) * \\
& \sin(q5) * \sin(q3 - \pi/2)) + \sin(q6) * (\cos(q2) * \cos(q4) + \cos(q3 - \pi/2) * \sin(q2) * \\
& \sin(q4))) / (((\sin(q6) * (\cos(q5) * (\cos(q2) * \sin(q4) - \cos(q4) * \cos(q3 - \pi/2) * \sin(q2)) + \\
& \sin(q2) * \sin(q5) * \sin(q3 - \pi/2)) - \cos(q6) * (\cos(q2) * \cos(q4) + \cos(q3 - \pi/2) * \\
& \sin(q2) * \sin(q4)))^2 / (\sin(q5) * (\cos(q2) * \sin(q4) - \cos(q4) * \cos(q3 - \pi/2) * \sin(q2)) - \\
& \cos(q5) * \sin(q2) * \sin(q3 - \pi/2))^2 + 1) * (\sin(q5) * (\cos(q2) * \sin(q4) - \cos(q4) * \cos(q3 - \\
& \pi/2) * \sin(q2)) - \cos(q5) * \sin(q2) * \sin(q3 - \pi/2)))
\end{aligned}$$

$$J(6, 1) = 1$$

$$\begin{aligned}
J(6, 2) = & ((\cos(q6) * (\cos(q5) * (\cos(q2) * \sin(q1) * \sin(q4) - \cos(q4) * \cos(q3 - \pi/2) * \sin(q1) * \\
& \sin(q2)) + \sin(q1) * \sin(q2) * \sin(q5) * \sin(q3 - \pi/2)) + \sin(q6) * (\cos(q2) * \cos(q4) * \\
& \sin(q1) + \cos(q3 - \pi/2) * \sin(q1) * \sin(q2) * \sin(q4))) / (\sin(q6) * \sin(q4) * (\sin(q1) * \\
& (\sin(q3 - \pi/2) - \cos(q1) * \cos(q2) * \cos(q3 - \pi/2)) + \cos(q1) * \cos(q4) * \sin(q2)) \\
& - \cos(q6) * (\cos(q5) * (\cos(q4) * (\sin(q1) * \sin(q3 - \pi/2) - \cos(q1) * \cos(q2) * \cos(q3 - \\
& \pi/2)) - \cos(q1) * \sin(q2) * \sin(q4)) + \sin(q5) * (\cos(q3 - \pi/2) * \sin(q1) + \cos(q1) * \\
& \cos(q2) * \sin(q3 - \pi/2)))) + ((\sin(q6) * (\sin(q4) * (\cos(q1) * \sin(q3 - \pi/2) + \cos(q2) * \\
& \cos(q3 - \pi/2) * \sin(q1)) - \cos(q4) * \sin(q1) * \sin(q2)) - \cos(q6) * (\cos(q5) * (\cos(q4) * \\
& (\cos(q1) * \sin(q3 - \pi/2) + \cos(q2) * \cos(q3 - \pi/2) * \sin(q1)) + \sin(q1) * \sin(q2) * \\
& \sin(q4)) + \sin(q5) * (\cos(q1) * \cos(q3 - \pi/2) - \cos(q2) * \sin(q1) * \sin(q3 - \pi/2)))) * \\
& (\sin(q6) * (\cos(q1) * \cos(q2) * \cos(q4) + \cos(q1) * \cos(q3 - \pi/2) * \sin(q2) * \sin(q4)) + \\
& \cos(q6) * (\cos(q5) * (\cos(q1) * \cos(q2) * \sin(q4) - \cos(q1) * \cos(q4) * \cos(q3 - \pi/2) * \\
& \sin(q2)) + \cos(q1) * \sin(q2) * \sin(q5) * \sin(q3 - \pi/2)))) / (\sin(q6) * (\sin(q4) * \\
& (\sin(q1) * \sin(q3 - \pi/2) - \cos(q1) * \cos(q2) * \cos(q3 - \pi/2)) + \cos(q1) * \cos(q4) * \\
& \sin(q2)) - \cos(q6) * (\cos(q5) * (\cos(q4) * (\sin(q1) * \sin(q3 - \pi/2) - \cos(q1) * \cos(q2) * \\
& \cos(q3 - \pi/2)) - \cos(q1) * \sin(q2) * \sin(q4)) + \sin(q5) * (\cos(q3 - \pi/2) * \sin(q1) + \\
& \cos(q1) * \cos(q2) * \sin(q3 - \pi/2))))^2) / (((\sin(q6) * (\sin(q4) * (\cos(q1) * \sin(q3 - \\
& \pi/2) + \cos(q2) * \cos(q3 - \pi/2) * \sin(q1)) - \cos(q4) * \sin(q1) * \sin(q2)) - \cos(q6) * \\
& (\cos(q5) * (\cos(q4) * (\cos(q1) * \sin(q3 - \pi/2) + \cos(q2) * \cos(q3 - \pi/2) * \sin(q1)) + \\
& \sin(q1) * \sin(q2) * \sin(q4)) + \sin(q5) * (\cos(q1) * \cos(q3 - \pi/2) - \cos(q2) * \sin(q1) * \\
& \sin(q3 - \pi/2))))^2 / (\sin(q6) * (\sin(q4) * (\sin(q1) * \sin(q3 - \pi/2) - \cos(q1) * \cos(q2) * \\
& \cos(q3 - \pi/2)) + \cos(q1) * \cos(q4) * \sin(q2)) - \cos(q6) * (\cos(q5) * (\cos(q4) * (\sin(q1) * \\
& \sin(q3 - \pi/2) - \cos(q1) * \cos(q2) * \cos(q3 - \pi/2)) - \cos(q1) * \sin(q2) * \sin(q4)) + \\
& \sin(q5) * (\cos(q3 - \pi/2) * \sin(q1) + \cos(q1) * \cos(q2) * \sin(q3 - \pi/2))))^2 + 1)
\end{aligned}$$

$$J(1, 6) = J(2, 6) = J(3, 1) = J(3, 6) = J(4, 1) = J(5, 1) = 0$$