ESCUELA POLITÉCNICA NACIONAL

FACULTAD DE INGENIERÍA MECÁNICA

ANÁLISIS DE SINGULARIDADES Y ESPACIO DE TRABAJO DEL ROBOT PARALELO DE TIPOLOGÍA 3UPE-RPU

TRABAJO DE TITULACIÓN PREVIO A LA OBTENCIÓN DEL TÍTULO DE MAGÍSTER EN MECATRÓNICA Y ROBÓTICA

DIANA CAROLINA TUMBACO MENDOZA
diana.tumbaco01@epn.edu.ec

DIRECTOR: ING. IVÁN ZAMBRANO, M.Sc.
ivan.zambrano@epn.edu.ec

CODIRECTOR: ING. LUIS RICARDO SOTO A, M.Sc.
ricardo.soto@epn.edu.ec

Quito, abril, 2019

CERTIFICACIÓN

Certifico que el presente trabajo fue desarrollado por **Diana Carolina Tumbaco Mendoza**, bajo mi supervisión.

Ing. Iván Zambrano, M.Sc.	Ing. Luis Ricardo Soto, M.Sc.
DIRECTOR DE PROYECTO	**CODIRECTOR DE PROYECTO**

DECLARACIÓN

Yo, **Diana Carolina Tumbaco Mendoza**, declaro bajo juramento que el trabajo aquí descrito es de mi autoría; que no ha sido previamente presentado para ningún grado o calificación profesional; y, que he consultado las referencias bibliográficas que se incluyen en este documento.

A través de la presente declaración cedo mis derechos de propiedad intelectual correspondiente a este trabajo, a la Escuela Politécnica Nacional, según lo establecido por la Ley de Propiedad Intelectual, por su Reglamento y por la normativa institucional vigente.

_____

Diana Carolina Tumbaco Mendoza

DEDICATORIA

Primeramente, a Dios, por todas las bendiciones que me ha dado. A mis padres, Luis y Rosa, por brindarme su esfuerzo y dedicación, sobre todo por la motivación para ser mejor persona. A mis hermanos, Javier y Paul, por su apoyo y respaldo incondicional.

Diana Carolina Tumbaco Mendoza

ÍNDICE

Certificación i
Declaración ii
Dedicatoria iii
Índice iv
Índice de figuras vii
Índice de tablas x
Resumen xi
Abstract xii
INTRODUCCIÓN 1
Pregunta de Investigación 3
Objetivo general 3
Objetivos específicos 3
Alcance 3
1. MARCO TEÓRICO 4
1.1. Mecanismo paralelo 4
1.1.1. Robot paralelo para diagnosis y rehabilitación de rodilla 5
1.1.2. Descripción del 3UPE-RPU 7
1.2. Análisis cinemático 10
1.2.1. Coordenadas Generalizadas 10
1.2.2. Determinación de las ecuaciones de restricción 11
1.2.3. Resolución del problema de posición 11
1.2.4. Resolución del problema de velocidad 12
1.2.5. Resolución del problema de aceleración 12
1.3. Cinemática del robot 3UPE-RPU 13
1.4. Matriz Jacobiana 14
1.5. Singularidades 15
1.5.1. Singularidad de cinemática inversa (tipo I) 16
1.5.2. Singularidad de cinemática directa (tipo II) 17
1.5.3. Singularidad combinada (tipo III) 17
1.5.4. Casos especiales de singularidades 17
1.6. Teoría de tornillos 18
1.6.1. Conceptos básicos de la teoría de tornillos 19
1.6.2. Velocidad de un cuerpo rígido expresado en tornillos 20
1.6.3. Representación de los pares cinemáticos 21
1.7. Tornillos recíprocos 22

1.7.1. Condición de reciprocidad de dos tornillos 22
1.8. Matriz jacobiana basada en la teoría de tornillos 22
1.9. Homogenización de la matriz jacobiana 23
1.10. Determinación de singularidades por matrices jacobianas de traslación y rotación 24
1.10.1. Matriz jacobiana de traslación y rotación 24
1.10.2. Matriz jacobiana general de un robot paralelo de 4 GDL 24
1.11. Espacio de trabajo y su representación. 28
1.11.1. Algoritmo para el cálculo del volumen de trabajo 30
1.11.2. Restricciones de las articulaciones y colisiones 30
1.11.3. Implementación computacional 31
2. METODOLOGÍA 33
2.1. Análisis cinemático 34
2.1.1. Cinemática inversa, posición 34
2.1.2. Cinemática directa, posición 36
2.1.3. Determinación de la matriz jacobiana basado en la teoría de tornillos 41
2.2. Homogenización de la matriz jacobiana 46
2.2.1. Singularidades 50
2.2.2. Singularidades: casos especiales 51
2.3. Análisis del espacio de trabajo 52
2.3.1. Restricciones 52
2.3.2.Técnica de discretización para el cálculo del espacio de trabajo a orientación constante 53
2.3.3. Función “Posicion_prueba” 54
2.3.4. Algoritmo de búsqueda del espacio de trabajo 54
2.3.5. Algoritmo de límite del espacio de trabajo 55
2.3.6. Evaluación del volumen 58
3. RESULTADOS Y DISCUSIÓN 59
3.1. Representación de la cinemática inversa del robot 3UPE-RPU 59
3.2. Cálculo de la matriz jacobiana analítica 64
3.3. Cálculo de la matriz jacobiana del robot 65
3.3.1. Análisis de singularidades 66
3.3.2. Singularidades de Fichter y Hunt 72
3.3.3. Determinación de la posición del espacio de trabajo 76
4. CONCLUSIONES Y RECOMENDACIONES 83
4.1. Conclusiones 83
4.2. Recomendaciones 84
Referencias bibliográficas 85

Anexos 88
A1. Algoritmo para la obtención simbólica del jacobiano analítico dependiente e independiente del modelo del robot 3UPR/RPU 88
A2. Algoritmo para la obtención del jacobiano geométrico y homogenización de la matriz del modelo del robot 3UPR/RPU en forma simbólica. 92
A3. Algoritmo para localizar singularidades en el espacio de trabajo con sus restricciones. 93

ÍNDICE DE FIGURAS

Figura 1. Configuración del manipulador paralelo 3UPE-RPU. 1
Figura 1.1. Mecanismos paralelos. 5
Figura 1.2. Movimientos requeridos para el trabajo de rehabilitación. 6
Figura 1.3. Robot paralelo 3UPE+RPU. 7
Figura 1.4. Nomenclatura 3UPE-RPU 8
Figura 1.5. Movimiento de la plataforma móvil. 8
Figura 1.6. Configuración de robot 3UPE-RPU. 9
Figura 1.7. Configuración singular de Hunt. 18
Figura 1.8. Configuración singular de Fichter 18
Figura 1.9 a. Representación gráfica del teorema de Chasles y **b**. Representación gráfica del teorema de Poinsot. 19
Figura 1.10.Representacion de un tornillo. 19
Figura 1.11. Pares cinemáticos más usuales. (a) Junta de revoluta; (b) par prismático; (c)par helicoidal; (d) par cilíndrico; (e) par esférico. 21
Figura 1.12. Tornillos recíprocos típicos. 22
Figura 1.13. Robot paralelo de 4 GDL 25
Figura 1.14. Ángulo de rotación de la junta esférica. 30
Figura 1.15. Coordenadas esféricas. 31
Figura 1.16. Algoritmo de búsqueda del vector 32
Figura 2.1. Cadenada cinemática para la pata 3. 34
Figura 2.2. Estructura de la matriz jacobiana dependiente. 39
Figura 2.3. Estructura de la matriz jacobiana independiente. 40
Figura 2.4. a) Ubicación de los tornillos en una extremidad, b) Tornillos en la extremidad central. 41
Figura 2.5. Ubicación de los tornillos en una extremidad del 3UPE-RPU. 42
Figura 2.6. Estructura de la matriz jacobiana directa. 45
Figura 2.7. Estructura de la matriz jacobiana inversa. 46
Figura 2.8 Estructura de la matriz jacobiana homogenizada. 50
Figura 2.9. Configuracion singular de Fichter: El movimiento de la plataforma rotado $\pm 90°$ alrededor del eje vertical. 51

Figura 2.10. Configuracion singular de Hunt: dos miembros y la plataforma móvil son coplanares. 51
Figura 2.11. Angulo de rotación de las juntas y vector unitario de sus direcciones. 53
Figura 2.12. Interferencia entre extremidades 53
Figura 2.13. Diagrama de flujo de vector de búsqueda. 55
Figura 2.14. diagrama de flujo algoritmo de búsqueda frontera. 57
Figura 2.15. Calculo de volumen del sector del espacio de trabajo. 58
Figura 3.1. Representación cinemática inversa 3UPE-RPU. 60
Figura 3.2. Movimiento en la posición $Xm = 10\ mm\ , Zm = 600\ mm,$ 65
Figura 3.3. Diagrama de flujo en el cálculo de singularidades. 67
Figura 3.4. Configuración singular inversa: a) extremidades extendidas, b) extremidades retraídas. 68
Figura 3.5. configuración singular del manipulador. 70
Figura 3.6. Configuración singular del robot. 71
Figura 3.7. Configuración singular del 3UPE-RPU: a) vista isométrica 72
Figura 3.8. Configuración singular de Fichter ($\theta = 90°$): a) vista isométrica b) Vista superior. 73
Figura 3.9 Configuración singular de Fichter ($\theta = -90°$): a) vista isométrica, b) Vista superior. 74
Figura 3.10. Configuración singular de Hunt: a) vista isométrica 75
Figura 3.11. Configuración singular de Hunt: dos miembros y la plataforma móvil son coplanares. 76
Figura 3.12. Espacio de trabajo del manipulador plano XZ, orientación constante ($\varphi = 0; \theta = 0$). 77
Figura 3.13. Singularidades dentro del espacio de trabajo cuando ($\varphi = 0; \theta = 0$) 77
Figura 3.14. *Espacio de trabajo del manipulador en el plano XZ, orientación fija ($\varphi = 0; \theta = \pi 18$).* 78
Figura 3.15. Singularidades dentro del espacio de trabajo cuando ($\varphi = 0; \theta = \pi 18$) 78
Figura 3.16.Espacio de trabajo del manipulador a orientación constante ($\varphi = \pi 4;\ \theta = \pi 4$). 79
Figura 3.17. Espacio de trabajo del manipulador en el plano XZ, orientación constante 79
Figura 3.18. Espacio de trabajo del manipulador en el plano XZ a orientación constante 80
Figura 3.19. Posición singular en: X= [466.4265 0 300.0000 0 1.0472 0] 80

Figura 3.20. Puntos de singulares del manipulador a una orientación constante ($\varphi = \pi;\ \theta = 0$).....81
Figura 3.21. Espacio de trabajo del manipulador a una orientación constante ($\varphi = 0;\ \theta = 0$).....81
Figura 3.22. Puntos de singulares del manipulador a una orientación constante ($\varphi = 0;\ \theta = 0$).....81

p

ÍNDICE DE TABLAS

Tabla 1.1. Nomenclatura del robot 3UPE-RPU 9
Tabla 1.2. Pares cinemáticos representados mediante tornillo. 21
Tabla 1.3. Condición de reciprocidad de dos tronillos. 22
Tabla 1.4. Tipos de espacio de trabajo. 29
Tabla 3.1. Parámetros geométricos del robot 59
Tabla 3.2. Valores iniciales para el cálculo cinemático 59
Tabla 3.3. Parámetros geométricos del robot. 76
Tabla 3.4. Restricciones 76

RESUMEN

Este trabajo es la continuación de la primera etapa del proyecto de investigación PIMI 15-04 que se refiere a un robot paralelo de tipología 3UPE-RPU y es llevado a cabo por la Facultad de Ingeniería Mecánica de la Escuela Politécnica Nacional (EPN). Uno de los objetivos planteados en este trabajo es hacer una revisión preliminar de la cinemática del robot 3UPE-RPU, donde se encuentra la posición, velocidad y aceleración, establecidas con el principio de coordenadas generalizadas, además se hace una comparación entre el jacobiano analítico y el jacobiano geométrico, con el fin de cumplir con nuestro objetivo principal. Ya cumplido con el primer objetivo, el propósito de este trabajo es representar el espacio de trabajo a orientación constante del robot de 4GDL usando un algoritmo que permita discretizar y evaluar su espacio de trabajo, además se realiza un análisis matemático en base a la teoría de tornillo y se clasifica las singularidades de acuerdo con el trabajo realizado por Gosselin y Angeles, cuya herramienta principal para dicho análisis es el determinante de la matriz jacobiana, además se identifican algunos casos especiales de singularidades estudiados por Hunt y Fichter. Por la configuración del robot se obtiene una matriz jacobiana adimensional, en consecuencia, se usa un método de homogenización, que facilita el análisis del rango de la matriz jacobiana. Para validar los resultados se utiliza MATLAB, donde se resuelve las ecuaciones y se desarrolla los cálculos matemáticos, además se realiza una representación del espacio de trabajo, los resultados obtenidos permiten visualizar el espacio de trabajo del robot y analizar sus singularidades.

Palabras clave: Singularidades, 3UPE-RPU, espacio de trabajo, matriz jacobiana, homogenización.

ABSTRACT

This work is the continuation of the first stage of the PIMI research project 15-04 which refers to a parallel robot of typology 3UPE-RPU and is carried out by the Faculty of Mechanical Engineering of the National Polytechnic School (EPN). One of the objectives proposed in this paper is to make a preliminary review of the kinematics of the 3UPE-RPU robot, where the position, velocity and acceleration are found, established with the principle of generalized coordinates, in addition a comparison is made between the analytical Jacobian and the geometric Jacobian, in order to fulfill our main objective. Already fulfilled with our first objective, the purpose of this work is to represent the working space with constant orientation of the 4GDL robot using an algorithm that allows us to discretize and evaluate their workspace, in addition a mathematical analysis is performed based on the theory of screw and classifies the singularities according to the work done by Gosselin and Angeles, whose main tool for this analysis is the determinant of the Jacobian matrix, in addition some special cases of singularities studied by Hunt and Fichter are identified. By the configuration of the robot a dimensionless Jacobian matrix is obtained, consequently, a homogenization method is used, which facilitates the analysis of the Jacobian matrix range. In order to validate the results, MATLAB is used, where the equations are solved and the mathematical calculations are developed, in addition a representation of the workspace is made, the obtained results allow to visualize the working space of the robot and analyze its singularities.

Keywords: Singularities, 3UPE-RPU, workspace, Jacobian matrix, homogenization.

ANÁLISIS DE SINGULARIDADES Y ESPACIO DE TRABAJO DEL ROBOT PARALELO DE TIPOLOGÍA 3UPE-RPU

INTRODUCCIÓN

El proyecto PIMI 15-04 realizado en la Facultad de Ingeniería Mecánica de la EPN, tiene por objetivo el desarrollo de un sistema que permita rehabilitar el miembro inferior del cuerpo humano, a través del empleo de un robot paralelo de tipología 3UPE-RPU. (Morales, 2017).

Al investigar los requisitos del sistema, el mecanismo paralelo propuesto es un robot de 4GDL que consta de 3 brazos: una junta universal, un actuador lineal y una junta esférica (UPE), los cuales forman los vértices de la plataforma móvil y una pata central con una junta de rotación, un actuador lineal y una junta universal (RPU), obteniendo un robot con dos movimientos rotacionales (2R) y dos traslacionales (2T).

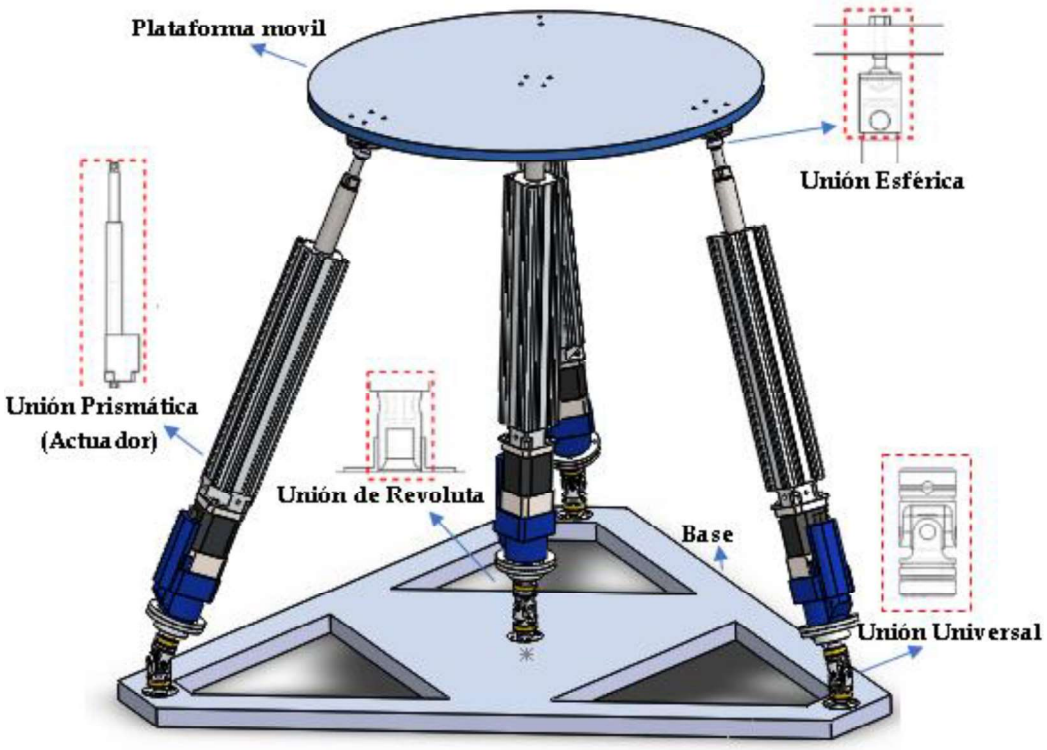


Figura .1. Configuración del manipulador paralelo 3UPE-RPU.
(Fuente: Autor)

El estudio de la cinemática de un robot trata inevitablemente el problema de configuraciones singulares, sobre todo los mecanismos con configuración paralela que alcanzan ciertas posiciones que ganan o pierden GDL(Grados de libertad). Dichas posiciones deben ser analizadas, ya que cerca de ellas, el mecanismo va a mostrar un

movimiento descontrolado que puede provocar daños permanentes en el manipulador, debido a lo cual, el análisis de singularidades y el espacio de trabajo, es uno de los pasos más importantes y uno de los primeros procedimientos en el diseño del robot. (Sevillano, 2014)

Enfocándose en esta premisa y con el propósito de avanzar en el proyecto se toma como referencia el estudio de la cinemática del robot paralelo 3UPE-RPU propuesto por (Fernández & Sotomayor, Análisis cinématico inverso y directo del robot paralelo, 2016) y (Morales, 2017), para estudiar las singularidades determinadas por la matriz jacobiana, que normalmente se deriva de un análisis de velocidades, que de acuerdo al estudio que se realice se puede considerar dos tipos de jacobiano, el analítico y el geométrico. Independientemente del tipo de jacobiano esta matriz proporciona información útil sobre el funcionamiento del manipulador, en este trabajo la teoría de tornillos recíprocos se utiliza para formular la matriz jacobiana y el estudio de singularidades se basa en la clasificación de Gosselin & Angeles (1990) y el análisis de Hunt y Fichter (Fichter, 1986) (Hunt, 1978) .

De acuerdo con la clasificación propuesta por Gosselin & Angeles (1990) las singularidades de un mecanismo paralelo se presentan de 3 tipos. El primer tipo de singularidad se da cuando convergen distintas ramas del problema cinemático inverso, lo que resulta en una pérdida de movilidad. Este tipo de singularidades ocurren en los límites del espacio de trabajo. El segundo tipo de singularidad se da cuando convergen distintas ramas del problema cinemático directo, conduciendo a la aparición de grados de libertad adicionales en el efector final del robot. Esto implica fuerzas o pares en el efector final que los actuadores no podrían soportar. El tercer tipo de singularidad se da cuando el manipulador está en una posición tal que se producen simultáneamente singularidades de los dos primeros tipos (Silva, 2005).

Estudios previos condujeron a la identificación de algunos casos especiales de singularidades. Hunt (1978) señaló una singularidad, que ocurre cuando todas las líneas asociadas con los actuadores prismáticos se intersecan en una línea común. Fichter (1986) mostró que se logra una configuración singular cuando la plataforma gira alrededor de un eje ortogonal al plano de la base en un ángulo de ± 90 grados (St-Onge & Gosselin, 2000).

Además, para localizar las singularidades de manera consistente, la matriz jacobiana debe ser dimensionalmente homogénea y se propone un método para homogenizar o normalizar la matriz jacobiana adimensional.

Por otra parte, el espacio de trabajo de un robot es uno de los aspectos primordiales para tener en cuenta en su diseño, para este estudio se emplea un algoritmo de discretización

para visualizar el área de trabajo a orientación constante (Oliva, Obtención de espacio de trabajo de la plataforma de Gough-Stewart mediante técnicas CAD, 2014).

La finalidad de este trabajo es proporcionar las bases necesarias para poder efectuar un control optimo en algún proyecto posterior.

Pregunta de Investigación

¿Se puede conseguir representar el espacio de trabajo total del robot 3UPE-RPU con sus respectivas limitaciones en un software matemático y además aplicar alguna metodología para obtener la matriz jacobiana del robot y poder analizar sus singularidades?

Objetivo general

Representar el espacio de trabajo y analizar las singularidades de un robot paralelo de tipología 3UPE-RPU.

Objetivos específicos

- Revisar matemáticamente la cinemática del robot.
- Determinar y calcular la matriz jacobiana analítica en base a derivadas parciales.
- Determinar y calcular la matriz jacobiana de robot en base a la teoría de tornillos.
- Analizar las condiciones suficientes de la existencia de singularidades.
- Validar el algoritmo para la representación del espacio de trabajo a orientación constante en base a una herramienta matemática.

Alcance

Este trabajo cubre el conocimiento preliminar de la modelización geométrica y cinemática del robot 3UPE-RPU. También comprende el estudio del área de trabajo del manipulador, es decir, las posiciones que puede alcanzar a una orientación fija de la plataforma móvil, así como la clasificación de singularidades a través de la determinación de la matriz jacobiana.

Para facilitar el estudio, los algoritmos se implementan en Matlab, automatizando todos los cálculos numéricos y proporcionando representaciones gráficas que ayudan a la compresión de los resultados.

1. MARCO TEÓRICO

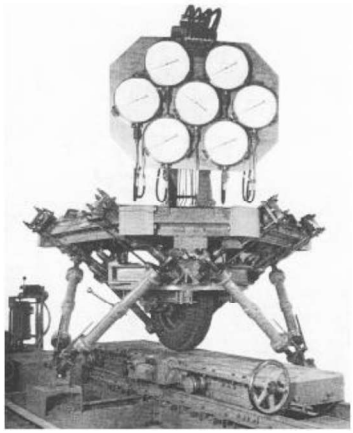
1.1. Mecanismo paralelo

Los manipuladores o mecanismos paralelos (MP) está compuesto de una plataforma fija y una plataforma móvil, conectadas por al menos dos cadenas cinemáticas seriadas en paralelo, estas cadenas cinemáticas en serie se denominan piernas o extremidades (Jiang, 2008).

Los mecanismos paralelos poseen ventajas significativas sobre los mecanismos de serie en términos de propiedades dinámicas, capacidad de carga, alta precisión y rigidez. Esto se debe a que los mecanismos paralelos se caracterizan por varias cadenas cinemáticas que conectan la base con el efector final, lo que permite ubicar los actuadores en la base del mecanismo o cerca de ella. Por lo tanto, se pueden utilizar mecanismos paralelos en muchas aplicaciones en las que estas propiedades son de importancia primaria, aunque tenga un espacio de trabajo limitado (Jiang, 2008).

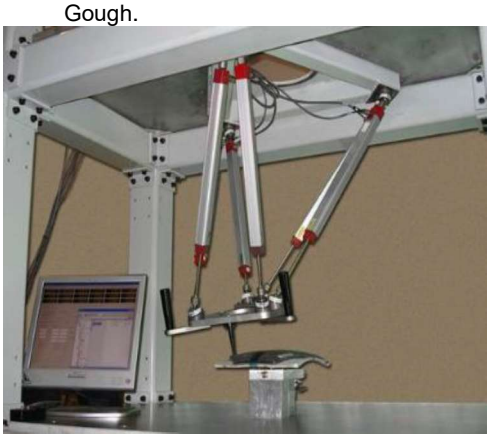
La cantidad de aplicaciones en MP ha ido aumentando constantemente (Figura 1.1) y varios prototipos se han construido, por ejemplo, los mecanismos paralelos también pueden usarse como máquinas herramientas (Figura. 1.1 c) o incluso con fines médicos (Figura. 1.1 d).

Desafortunadamente, hay algunos factores que limitan la aplicación de mecanismos paralelos. Un factor principal es que el espacio de trabajo es bastante limitado, el otro es que las configuraciones singulares pueden existir dentro del espacio de trabajo. Cuando un mecanismo paralelo está en una configuración singular, el número de grados de libertad del mecanismo cambia instantáneamente, volviéndolo incontrolable e incluso puede ocasionar un desglose del mecanismo, por lo tanto, es de importancia buscar el espacio de trabajo máximo, libre de singularidades (Jiang, 2008).



a) Máquina para probar neumáticos de Gough.

b) Simulador de vuelo (CAE Electronics Ltd, Canada)



c) Máquina cinemática paralela.

d) Robot paralelo Delta (Humboldt-University at Berlin)

Figura 1.1. Mecanismos paralelos.
(Fuente: (Jiang, 2008))

1.1.1. Robot paralelo para diagnosis y rehabilitación de rodilla

El uso de robots paralelos cada vez va en aumento sobre todo en aplicaciones de rehabilitación, el objetivo es diseñar un robot para la rehabilitación de miembros inferiores que tome en cuenta la movilidad de la pierna humana e imiten el trabajo realizado por los fisioterapeutas.

En base a estudios realizados y ejemplificados la mejor propuesta es un robot paralelo 3UPE-RPU capaz de realizar movimientos que se aplican en el proceso de rehabilitación. Para un mejor entendimiento los lectores interesados pueden revisar el trabajo “Diseño de dispositivo robótico para la rehabilitación y diagnosis de extremidades inferiores”, (Araujo, Diaz, & Mata, 2017), la tesis de (Zamora, 2016) y en cuanto a los movimientos básicos necesarios se ha establecido de acuerdo a la figura 1.2 (Aginaga, Iriarte, Plaza, & Mata, 2018).

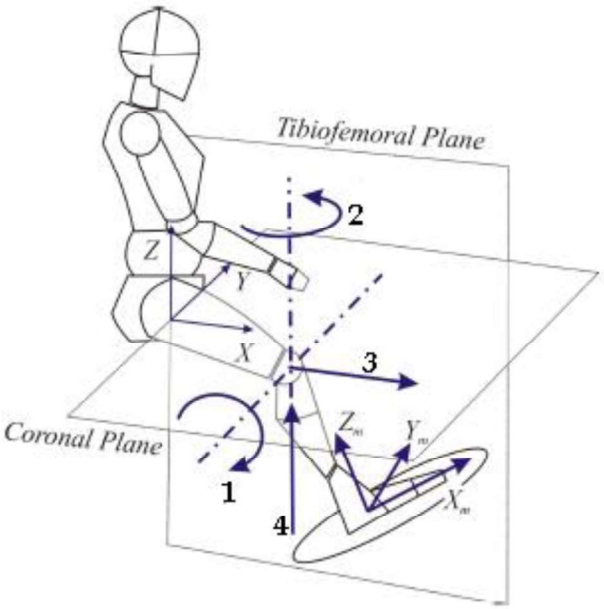
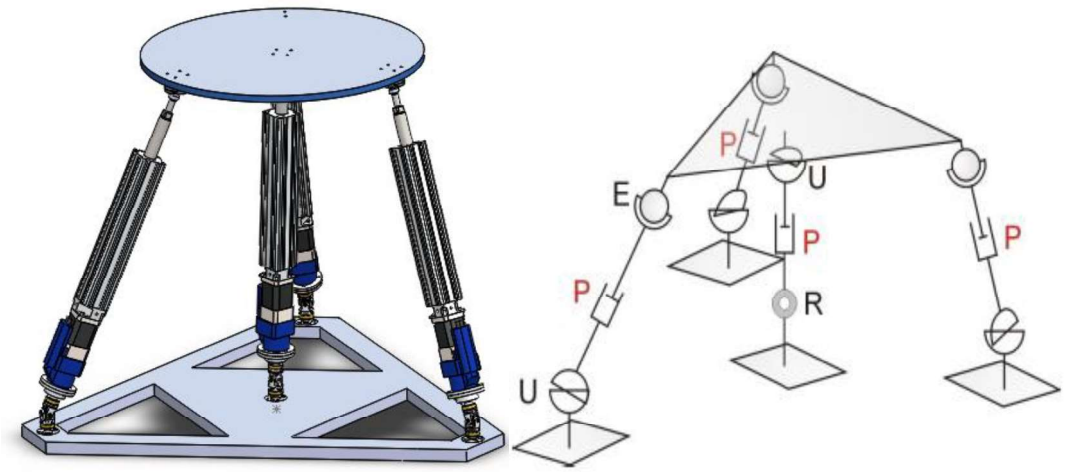


Figura 1.2. Movimientos requeridos para el trabajo de rehabilitación.
(Fuente: (Aginaga, Iriarte, Plaza, & Mata, 2018))

1. Flexión de la extremidad en dirección perpendicular al plano tibiofemoral. El rango total es de al menos 60 grados para fines de rehabilitación.
2. Rotación de la extremidad a lo largo de un eje perpendicular a la plataforma móvil. El rango de movimiento debe ser de al menos $\pm$10 grados.
3. Traslación en la dirección de un eje horizontal contenido en el plano tibiofemoral. El desplazamiento es pequeño, 15–20 mm, para reproducir las pruebas de Pivot o Lachman (Zamora, 2016). Sin embargo, para realizar algunos de los movimientos planeados para la rehabilitación, este desplazamiento horizontal debe incrementarse a un total de al menos 400 mm.
4. Traslación en la dirección de un eje vertical contenido en el plano tibiofemoral. Es posible que se requieran cambios mayores, al menos 200 mm, en coordinación con el movimiento horizontal descrito anteriormente para reproducir los movimientos de rehabilitación. Además, desde un punto de vista práctico, se requería un mayor desplazamiento para ubicar correctamente la plataforma móvil donde descansan los pies de los pacientes.

1.1.2. Descripción del 3UPE-RPU

Manipulador paralelo con 3 brazos, que se compone de una unión universal, una unión prismática (actuador lineal) y una unión esférica (UPE) los cuales forman los vértices de la plataforma móvil y fija; y una pata central con una unión de rotación, unión prismática (actuador lineal) y una unión universal (RPU), ver en la figura 1.3 como se distribuye:



a) Representación CAD b) Esquema

Figura 1.3. Robot paralelo 3UPE+RPU.
(Fuente: Autor & (Zamora, 2016))

De acuerdo con fórmula de Grübler, los grados de libertad del robot paralelo 3UPE+RPU se obtiene aplicando la ecuación 1.1:

$$F = 6(n - j - 1) + \sum fi \qquad \text{Ecuación 1.1}$$

F Grados de libertad

n Número de eslabones

j Número de juntas

fi Número de grados de libertad de la junta *i*

Por lo tanto, se tiene:

$$n = 10$$

$$j = 12$$

$$\sum fi = 22$$

$$F = 6(10 - 12 - 1) + 22 = 4\ GDL$$

Entonces, de forma matemática se obtiene $4GDL$, y constructivamente se restringe el desplazamiento en el eje “Y” y la rotación en el eje “X”, obteniendo dos movimientos de traslación y dos de Rotación (2T2R) (Zamora, 2016).

En la figura 1.4, figura 1.5 y figura 1.6, se tiene la nomenclatura, movimientos y configuración del robot.

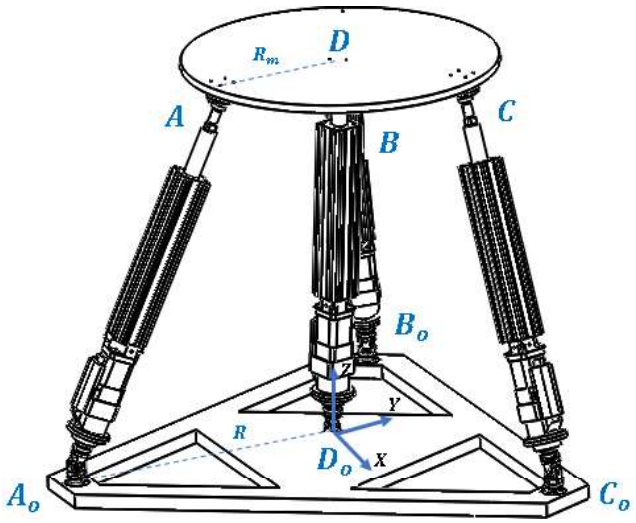
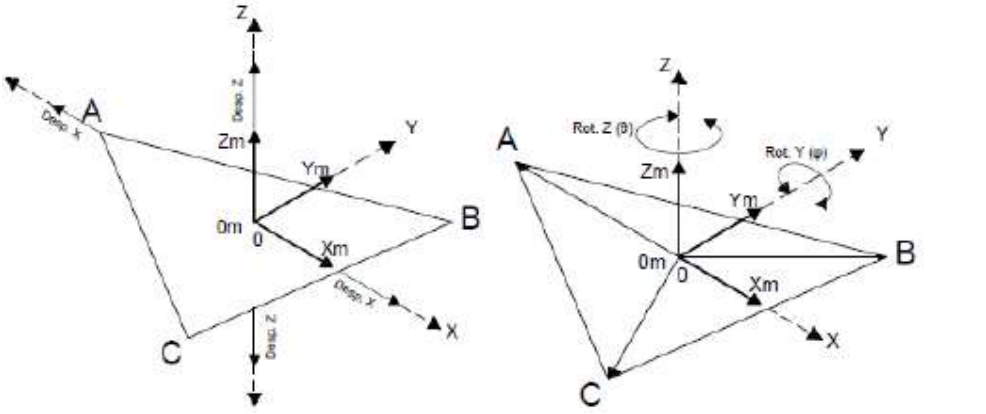
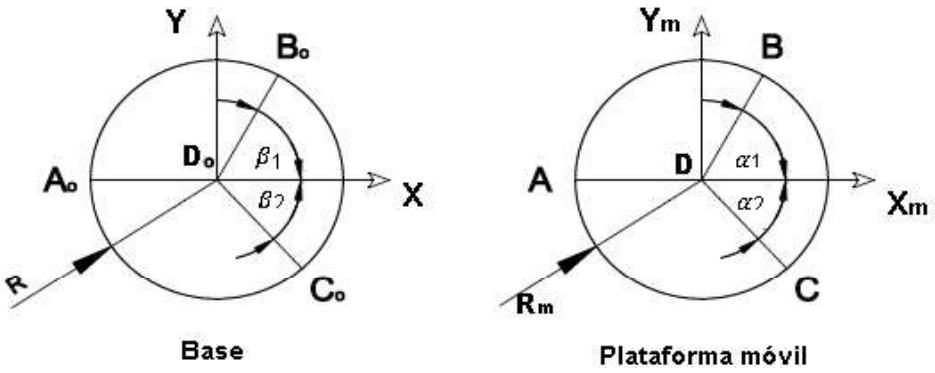


Figura 1.4. Nomenclatura 3UPE-RPU
(Fuente: ((Morales, 2017), (Zamora, 2016) y Autor)).



Movimientos de rotación de la plataforma. Movimiento de desplazamiento de laplataforma.

Figura 1.5. Movimiento de la plataforma móvil.
(Fuente: ((Morales, 2017), (Zamora, 2016) y Autor)).



Configuración del robot

Figura 1.6. Configuración de robot 3UPE-RPU.
(Fuente: ((Morales, 2017), (Zamora, 2016) y Autor))

En la tabla 1,1 se tiene una descripción de los puntos de unión y las variables de la figura 1.4, figura 1.5 y figura 1.6.

Tabla 1.1. Nomenclatura del robot 3UPE-RPU

Símbolo	Descripción	Símbolo	Descripción
$\boldsymbol{R}$	Radio de la plataforma fija	R_m	Radio de la plataforma móvil
$\boldsymbol{A_o}$	Punto de unión del brazo1 con la plataforma fija	A	Punto de unión del brazo1 con la plataforma móvil
$\boldsymbol{B_o}$	Punto de unión del brazo2 con la plataforma fija	B	Punto de unión del brazo2 con la plataforma móvil
$\boldsymbol{C_o}$	Punto de unión del brazo3 con la plataforma fija	C	Punto de unión del brazo3 con la plataforma móvil
$\boldsymbol{D_o}$	Centro del eje de coordenadas de la plataforma fija	D	Centro del eje de coordenadas de la plataforma móvil
$\boldsymbol{\beta_1}$	Ángulo de posición del punto B_o en la plataforma fija	α_1	Ángulo de posición del punto B en la plataforma móvil
$\boldsymbol{\beta_2}$	Ángulo de posición del punto C_o en la plataforma fija	α_2	Ángulo de posición del punto C en la plataforma móvil

(Fuente: (Zamora, 2016))

- **Ubicación de los vértices de la plataforma fija** (Morales, 2017)**.**

$$A_o = [-R \quad 0 \quad 0]^T \qquad \text{Ecuación 1.2}$$

$$B_o = [Rcos(\beta_1) \quad Rsin(\beta_1) \quad 0]^T \qquad \text{Ecuación 1.3}$$

$$C_o = [Rcos(\beta_2) \quad -Rsin(\beta_2) \quad 0]^T \qquad \text{Ecuación 1.4}$$

$$D_o = [0 \quad 0 \quad 0]^T \qquad \text{Ecuación 1.5}$$

- **Ubicación de los vértices de la Plataforma móvil** (Morales, 2017).

$$A = [-R_m \quad 0 \quad 0]^T \qquad \text{Ecuación 1.6}$$

$$B = [R_m cos(\alpha_1) \quad R_m sin(\alpha_1) \quad 0]^T \qquad \text{Ecuación 1.7}$$

$$C = [R_m cos(\alpha_2) \quad -R_m sin(\alpha_2) \quad 0]^T \qquad \text{Ecuación 1.8}$$

$$D_m = [0 \quad 0 \quad 0]^T \qquad \text{Ecuación 1.9}$$

1.2. Análisis cinemático

El análisis cinemático tiene como finalidad determinar el movimiento (posición, velocidad, y aceleración) de cada uno de los elementos que conforman el sistema mecánico, existen dos tipos de cinemática, directa e inversa. No se detalla cómo solucionar este problema, pues ya han sido expuestas en trabajos anteriores (Fernández & Sotomayor, Análisis cinématico inverso y directo del robot paralelo, 2016), (Morales, 2017) y (Zamora, 2016).

La obtención del análisis cinemático se utiliza para analizar configuraciones singulares y representar el área de trabajo del robot, uno de los métodos que se usa es el de las coordenadas generalizadas siendo ventajoso para realizar un posterior modelado dinámico del sistema. Los sólidos rígidos que constituye un sistema mecánico en el espacio tienen un movimiento general de 6 *GDL*, por lo mismo, su posición y orientación se pueden fijar mediante 6 coordenadas generalizadas, pero la presencia de pares cinemáticos entre los sólidos rígidos reduce la movilidad del sistema mecánico, siendo necesario el uso de las ecuaciones de enlace cinemático que describen las relaciones entre coordenadas generalizadas (Bacaicoa, 2017)

1.2.1. Coordenadas Generalizadas

El grupo de variables que pueden describir completamente el comportamiento de un sistema, se llaman coordenadas generalizadas y se distinguen dos tipos. (Pareja & Pazmiño, 2018). Y se escribe como:

$$q = \{q_1 \quad q_2 \quad \ldots . q_m\} \qquad \text{Ecuación 1.10}$$

m Número de coordenadas generalizadas

- **Coordenadas Independientes:** El número de coordenadas independientes concuerda con el número de grados de libertad del mecanismo, por lo que se requiere el conjunto de variables independientes, así como la información geométrica del mecanismo (Chang, 2016)

- **Coordenadas Dependientes:** El número de coordenadas dependientes es mayor al número de grado de libertad del sistema mecánico, estas coordenadas definen el movimiento de cada elemento del mecanismo y están relacionadas mediante ecuaciones de restricción (Chang, 2016).
- **Ecuaciones de restricción:** Es una modelación del sistema mecánico, limitado por sus pares cinemáticos y eslabones. Este tipo de modelación utiliza el método de partición de coordenadas generalizadas, obteniendo vectores dependientes e independientes (Bacaicoa, 2017), esto es:

$$q = \begin{Bmatrix} q_{ind} \\ q_{dep} \end{Bmatrix}$$ Ecuación 1.11

ind Coordenadas independientes

dep Coordenadas dependientes.

1.2.2. Determinación de las ecuaciones de restricción

Una ecuación de restricción indica las características geométricas del mecanismo y su configuración. Además, se utilizan para determinar velocidades y aceleraciones; en el modelado de las ecuaciones es necesario conocer las restricciones del sistema que dependen del tipo de unión del par cinemático y las condiciones impuestas al movimiento del sistema mecánico.

Introduciendo la partición de coordenadas expuestas en la ecuación 1.11 en el vector de restricciones, $\boldsymbol{\Phi}(\boldsymbol{q})$ (Bacaicoa, 2017), se pueden expresar como:

$$\Phi\left(x, q_{ind}, q_{dep}\right) = 0$$ Ecuación 1.12

$$\Phi(\mathrm{x}, \mathrm{q}) = 0$$ Ecuación 1.13

1.2.3. Resolución del problema de posición

Para la resolución del problema de posición de un sistema mecánico se determina localizando las coordenadas dependientes, q_{dep}, en función de las coordenadas independientes, q_{ind}.

El vector de coordenadas generalizadas de posición es: $\mathrm{q} = \left({q_{ind}}^T, {q_{dep}}^T\right)^T$

La Ecuación 1.13 no es lineal respecto a las coordenadas generalizas, y puede resolverse utilizando el método de Newton-Raphson que se expresa como: (Bacaicoa, 2017).

$$J_{dep}^{(i)}\left(q_{dep}^{(i)} - q_{dep}^{(i+1)}\right) = \Phi(x^{(i)}, q^{(i)})$$ Ecuación 1.14

Donde:

$$J_{dep} = \frac{\partial \Phi(\mathrm{x}, \mathrm{q})}{\partial q_{dep}} \qquad \text{Ecuación 1.15}$$

J_{dep} Matriz jacobiana obtenida de las ecuaciones de restricción respecto a las coordenadas generalizadas dependientes, y el subíndice ***i*** hace referencia a la evaluación de la interación ***i***- ésima (Bacaicoa, 2017).

1.2.4. Resolución del problema de velocidad

Con la solución del problema de posición y resuelta las coordenadas generalizas de la ecuación 1.13 a continuación se resuelve el problema de velocidad para obtener las velocidades generalizadas dependientes, $\dot{q_{dep}}$, a partir del valor de las velocidades generalizadas independientes, $\dot{q}_{ind}$. A comparación del anterior este problema es lineal y consiste en resolver la derivada temporal de las ecuaciones de posición ver ecuación 1.14. Por tanto, la ecuación de la velocidad es: (Bacaicoa, 2017)

$$J_{dep}\dot{q}_{dep} = -J_{ind}\dot{q}_{ind} \qquad \text{Ecuación 1.16}$$

donde:

$$J_{ind} = \frac{\partial \Phi(\mathrm{x}, \mathrm{q})}{\partial q_{ind}} \qquad \text{Ecuación 1.17}$$

J_{ind} Matriz jacobiana obtenida de las ecuaciones de restricción respecto de las coordenadas generalizadas independientes.

1.2.5. Resolución del problema de aceleración

Para obtener las aceleraciones de las coordenadas generalizadas dependientes, $\ddot{q_{dep}}$, en función de las aceleraciones generalizadas independientes, $\ddot{q_{ind}}$l se realiza la derivada temporal de las ecuaciones de velocidad ver ecuación 1.16, por tanto, el problema de aceleraciones se resuelve como: (Bacaicoa, 2017).

$$J_{dep}\ddot{q}_{dep} = -\dot{J}_{dep}\dot{q}_{dep} - \dot{J}_{ind}\dot{q}_{ind} - J_{ind}\ddot{q}_{ind} \qquad \text{Ecuación 1.18}$$

1.3. Cinemática del robot 3UPE-RPU

A continuación, se presenta la resolución cinemática del robot 3UPE-RPU, ver figura 1.4, figura 1.5 y figura 1.6, obtenidas por Morales (2017).

$$X = [X_m;\ Y_m;\ Z_m;\ \psi;\ \phi;\ \theta] \qquad \text{Ecuación 1.19}$$

Donde:

X Vector posición del centro de la plataforma móvil.

X_m Coordenada en el eje "X" del centro de la plataforma móvil.

Y_m Coordenada en el eje "Y" del centro de la plataforma móvil.

Z_m Coordenada en el eje "Z" del centro de la plataforma móvil.

ψ Ángulo de rotación de la plataforma móvil alrededor del eje X.

ϕ Ángulo de rotación de la plataforma móvil alrededor del eje Y.

θ Ángulo de rotación de la plataforma móvil alrededor del eje Z.

Para el propósito del mecanismo $Y_m = 0$ y $\psi = 0$.

A partir de la ecuación 1.19 y el método propuesto por Morales (2017) el cual utilizo los parámetros Denavit-Hartenberg (DH) para calcular la cinemática inversa se obtuvo las coordenadas generalizadas de las 4 extremidades ver ecuación 1.20 y las 11 ecuaciones de restricción del sistema (Fernández & Sotomayor, 2016), ver ecuaciones 1.21, ecuación 1.22, ecuación 1.23:

$$q = (A_1, A_2, A_3, B_1, B_2, B_3, C_1, C_2, C_3, D_1, D_2), \qquad \text{Ecuación 1.20}$$

q Vector de coordenadas generalizadas de posición

$$\Phi_i(x, q_{ind}, q_{dep}) = 0 \qquad \text{Ecuación 1.21}$$

$$\Phi_i(x, q) = 0 \qquad \text{Ecuación 1.22}$$

$$\Phi_i(X_m, Z_m, \phi, \theta, A_3, B_3, C_3, D_1, A_1, A_2, B_2, B_1, C_1, C_2, D_2,) = 0 \qquad i = 1,2,3 \ldots\ldots 11$$

Donde:

Φ_i Ecuación de restricción

Las ecuaciones de restricción son (Fernández & Sotomayor,2016):

Ecuación 1.23

$$\Phi(1) = R_m \cos(\phi)\cos(\theta) - X_m - R - A_1\cos(A_3)\sin(A_2)$$

$$\Phi(2) = R_m\cos(\phi)\sin(\theta) - A_1\sin(A_3)$$

$$\Phi(3) = A_1\cos(A_2)\cos(A_3) - R_m\sin(\phi) - Z_m$$

$$\Phi(4) = B_1\sin(B_3)\sin(bB) - Rcos(bB) - X_m + R_m\sin(\alpha_1)\sin(\theta) - R_m\cos(\alpha_1)\cos(\phi)\cos(\theta) - B_1\cos(B_3)\sin(B_2)\cos(bB)$$

$$\Phi(5) = -Rsin(bB) - B_1\sin(B_3)\cos(bB) - R_m\sin(\alpha_1)\cos(\theta) - R_m\cos(\alpha_1)\cos(\phi)\sin(\theta) - B_1\cos(B_3)\sin(B_2)\sin(bB)$$

$$\Phi(6) = R_m\cos(\alpha_1)\sin(\phi) - Z_m + B_1\cos(B_2)\cos(B_3)$$

$$\Phi(7) = C_1\sin(C_3)\sin(bC) - Rcos(bC) - X_m + R_m\sin(\alpha_2)\sin(\theta) - R_m\cos(\alpha_2)\cos(\phi)\cos(\theta) - C_1\cos(C_3)\sin(C_2)\cos(bC)$$

$$\Phi(8) = -Rsin(bC) - C_1\sin(C_3)\cos(bC) - R_m\sin(\alpha_2)\cos(\theta) - R_m\cos(\alpha_2)\cos(\phi)\sin(\theta) - C_1\cos(C_3)\sin(C_2)\sin(bC)$$

$$\Phi(9) = R_m\cos(\alpha_2)\sin(\phi) - Z_m + C_1\cos(C_2)\cos(C_3)$$

$$\Phi(10) = D_1\sin(D_2) - X_m$$

$$\Phi(11) = D_1\cos(D_2) - Z_m$$

Donde:

bB, bC:	Ángulos de rotación para cada brazo.
($A_1, A_2, A_3, B_1, B_2, B_3, C_1, C_2, C_3, D_1, D_2$):	Corresponde a las variables de las articulaciones hallados por el método D&H (Morales, 2017).

Durante el cálculo de la solución de la cinemática directa es necesario utilizar la matriz jacobiana analítica y se obtiene derivando el vector de restricciones respecto de las coordenadas generalizadas, de la ecuación 1.23.

1.4. Matriz Jacobiana

La matriz Jacobiana está formada de derivadas parciales de primer orden de una función $\Phi_i(x, q)$. Entre sus aplicaciones más importante es la posibilidad de aproximarse linealmente a la función en un punto y para interés de este trabajo se utiliza en el modelado cinemático de un robot y sobre todo para encontrar configuraciones singulares del mismo (Hernández, 2018).

El jacobiano es una herramienta que relaciona la velocidad de las coordenadas articulares, posición y orientación del efector final del robot.

Las aplicaciones anteriormente mencionadas atrajeron la atención de diferentes investigadores, entre ellos Gosselin y Ángeles que en 1990 estudiaron las singularidades de los mecanismos de cadena cinemática cerrada, en este trabajo se divide la matriz jacobiana en dos matrices, una asociada a la cinemática directa y la otra a la cinemática inversa (Gosselin & Angeles, 1990).

La matriz jacobiana se puede obtener de diferentes maneras y se presentan de dos tipos:

Jacobiano geométrico: También conocido como jacobiano del robot J_v, relaciona el mundo articular (velocidades articulares) con el mundo cartesiano (velocidad lineal y angular) del extremo del robot. Es el más importante de la cinemática diferencial y depende de la configuración del robot (Villegas, 2016).

Relaciona a velocidades geométricas: (Bona, 2016).

$$v_p = \begin{bmatrix} \dot{x} \\ \omega \end{bmatrix} = J_v \dot{q} \qquad \text{Ecuación 1.24}$$

Jacobiano analítico: Por el procedimiento con el que se obtiene, el nombre es jacobiano analítico, aquí relaciona las velocidades articulares con las velocidades del extremo del robot y el cambio de los ángulos de euler. Consiste en la diferenciación de la función directa de la cinemática, pero con este procedimiento se pierde el significado físico de la velocidad angular.

El jacobiano analítico, en general, difiere del geométrico, ya que este es adoptado cuando cantidades del espacio operacional son de interés y se utiliza habitualmente en métodos numéricos iterativos para resolver la cinemática directa, el jacobiano geométrico se usa cuando cantidades físicas son de interés (Villegas, 2016).

Relaciona a velocidades analíticas: (Bona, 2016)

$$\dot{p} = \begin{bmatrix} \dot{x} \\ \dot{\alpha} \end{bmatrix} = J_a \dot{q} \qquad \text{Ecuación 1.25}$$

1.5. Singularidades

Aquellas configuraciones en las cuales la matriz jacobiana es de rango deficiente se definen como singularidades. En consecuencia, las configuraciones que conducen a las singularidades necesitan ser encontradas, sabiendo que cada vez que una matriz pierde su rango, su determinante se convierte en cero (Khrev, 2016).

Muchos investigadores han prestado atención al análisis de las singularidades de mecanismos paralelos en las últimas dos décadas, este estudio es inevitable para el diseño, la planificación de trayectoria y el control. Una de las primeras obras para abordar las singularidades es el análisis basado en la teoría del tornillo de Hunt (Hunt, 1978).

En general, el análisis de singularidades se basa en la relación de la cinemática instantánea entre la velocidad $\dot{X}$ de la plataforma móvil y la velocidad de las articulaciones activas $\dot{q}$, que se describe por:

$$J_X \dot{X} = J_q \dot{q} \qquad \text{Ecuación 1.26}$$

donde J_X y J_q son matrices jacobianas.

J_X Matriz Jacobiana directa permite obtener la velocidad del efector final del robot a partir de las velocidades articulares

J_q Matriz Jacobiana inversa permite obtener la velocidad que tienen que adquirir cada una de las articulaciones para obtener una velocidad determinada en el efector final.

La ecuación 1.26 es fundamental para el análisis de singularidades de robots paralelos.

Gosselin y Angeles, han demostrado que las singularidades encontradas en las cadenas cinemáticas de ciclo cerrado se pueden dividir en tres grupos principales basados en la ecuación 1.26 (Gosselin & Angeles, 1990).

La primera clase, la cual se designa como tipo I, se relaciona con el análisis del jacobiano inverso, mientras que la segunda clase se le relaciona con el análisis del jacobiano directo y se designan como tipo II. La última clase es una combinación de las dos primeras y se conoce como singularidad tipo III (Salgado, 2008).

1.5.1. Singularidad de cinemática inversa (tipo I)

Este tipo de singularidades ocurre cuando:

$$\det(J_q) = 0 \qquad \text{Ecuación 1.27}$$

$\det(J_q)$ Determinante de la matriz jacobiana inversa.

En esta configuración, ver ecuación 1.27 se pierde uno o más grados de libertad, en este tipo de singularidades un robot paralelo puede resistir fuerzas o momentos en algunas direcciones.

La singularidad tipo I se obtiene computarizando el borde del espacio de trabajo del manipulador (Gosselin C. , 1988).

1.5.2. Singularidad de cinemática directa (tipo II)

Este tipo de singularidades ocurre cuando:

$$\det(J_x) = 0 \qquad \text{Ecuación 1.28}$$

$\det(J_x)$ Determinante de la matriz jacobiana directa.

En este tipo de singularidades, ver ecuación 1.28 el movimiento del robot se encuentra limitado mientras todos sus actuadores están completamente bloqueados. En este caso el prototipo gana 1 o más grados de libertad, en esta configuración el robot no puede resistir fuerzas o momentos en alguna dirección.
El segundo tipo de singularidad se halla en las expresiones del determinante de la matriz jacobiana del mecanismo (Gosselin C. , 1988).

1.5.3. Singularidad combinada (tipo III)

Una singularidad combinada ocurre cuando los determinantes de J_x y J_q son cero. Este tipo de singularidades solo puede ocurrir en manipuladores con una arquitectura cinemática especial.

A demás, el tercer tipo de singularidad se evita mediante una elección adecuada de parámetros cinemáticos.

1.5.4. Casos especiales de singularidades

Las expresiones obtenidas en la sección anterior son completamente generales y pueden incluir todas las configuraciones singulares de los mecanismos, pero existen casos especiales que necesitan ser revisados.

a) Configuración Singular de Hunt

Hunt es un investigador que identificó un caso especial de configuración singular cuando todas las líneas asociadas con los actuadores prismáticos intersecan en una línea común, ver figura 1.7 (Hunt, 1978).

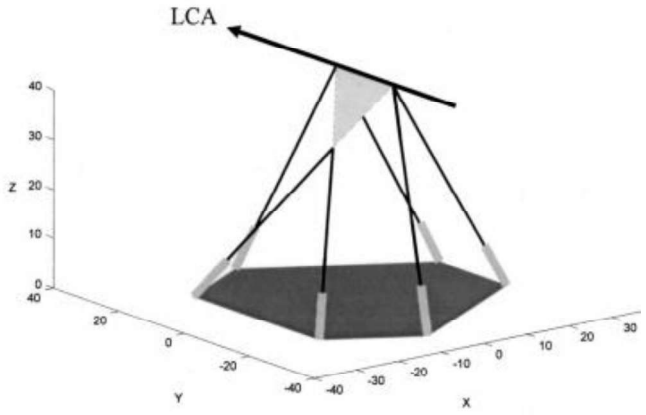


Figura 1.7. Configuración singular de Hunt.
(Fuente:(St-Onge & Gosselin, 2000))

b) Configuración Singular de Fichter

Fichter identificó un segundo caso especial de singularidad (1986) y se logra cuando la plataforma móvil es rotada por un ángulo de $\mp 90°$ alrededor del eje z y gana un grado de libertad extra, ver figura 1.8 (St-Onge & Gosselin, 2000).

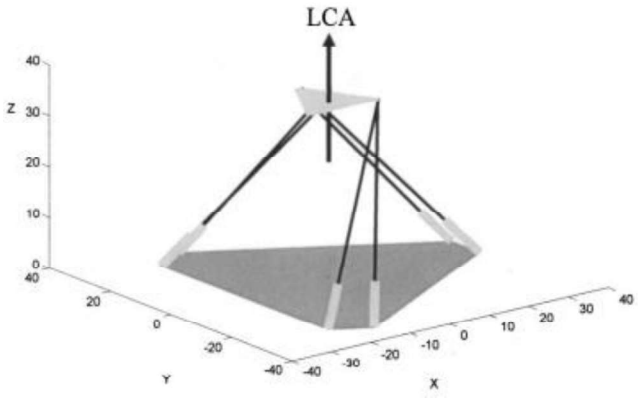


Figura 1.8. Configuración singular de Fichter
(Fuente:(St-Onge & Gosselin, 2000))

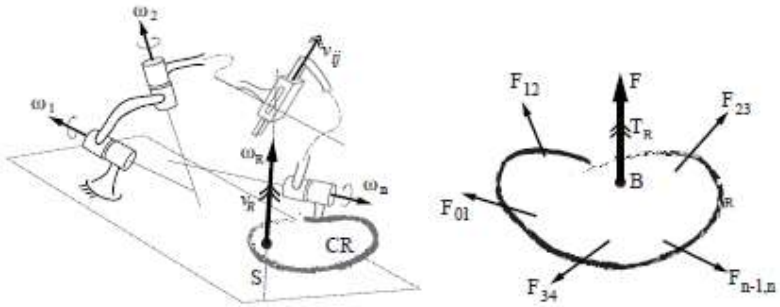
Para determinar la matriz jacobiana directa e inversa se usa el método de la teoría de tornillos y su propiedad de tornillo reciproco (Hunt, 1978).

1.6. Teoría de tornillos

La teoría de tornillos ha sido usada en la cinemática por más de 2 siglos. Robert Ball fue el que enuncio esta teoría (Ball, 1990) y desarrollada posteriormente por (Hunt, 1978) y quien hizo importantes avances en este campo. Desde entonces, la teoría ha sido usada extensivamente en cinemática: desde análisis de velocidad y singularidades en

manipuladores serial hasta análisis de fuerza, velocidad y aceleración en manipuladores paralelos (Abbasi, 2001).

La teoría de tornillos se basa en dos teoremas muy conocidos. El primero se relaciona con el desplazamiento del cuerpo rígido, conocido como el teorema de Chasles, y el otro tiene que ver con las fuerzas que actúan sobre el cuerpo rígido, llamado teorema de Poinsot (Cárdenas, 2016).



a. **b.**

Figura 1.9 a. Representación gráfica del teorema de Chasles y **b**. Representación gráfica del teorema de Poinsot.

(Fuente: (Cárdenas, 2016))

1.6.1. Conceptos básicos de la teoría de tornillos

Un tornillo es un vector de 6 componentes que permite representar campos de velocidades o fuerzas de manera sencilla y compacta. Por medio de las coordenadas de tornillo puede definirse el eje del movimiento asociado a un determinado par cinemático (Salgado, 2008).

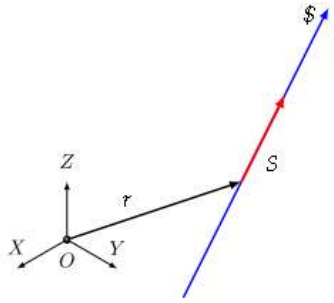


Figura 1.10.Representacion de un tornillo.
(Fuente: (Salgado, 2008))

De acuerdo con la figura 1.10 un tornillo puede ser representado matemáticamente a través de la siguiente expresión: (Cárdenas, 2016), (Salgado, 2008)

$$\$ = \begin{bmatrix} S \\ S \times r + hS \end{bmatrix} \quad \text{Si } h \text{ es finito,} \qquad \text{Ecuación 1.29}$$

$$\$ = \begin{bmatrix} 0 \\ S \end{bmatrix} \quad \text{Si } h \to \infty$$

Donde:

S Vector unitario en la dirección del eje del tornillo $

r Vector que va desde cualquier punto sobre el eje del tornillo al origen del sistema coordenado de referencia O_{XYZ}

h Es un valor escalar, representa el paso del eje al que hace referencia y es el cociente entre la traslación y la rotación permitida por el par cinemático asociado al tornillo.

1.6.2. Velocidad de un cuerpo rígido expresado en tornillos

Sean *A* y *B* un par de cuerpos rígidos, y sea *O* un punto arbitrario fijo al cuerpo *B*; más aún, sea ${}^{A}\dot{S}^{B}$ un vector arbitrario a lo largo de la dirección de la velocidad angular ${}^{A}\omega^{B}$, entonces:

$$ {}^{A}\omega^{B} = {}^{A}\omega^{B}\ {}^{A}S^{B} \qquad \text{Ecuación 1.30} $$

Siendo ${}_{A}h_{B}$ el paso del tornillo y $r_{O/P}$ el vector de posición del punto *O* respecto a un punto *P* a lo largo del eje instantáneo de rotación del movimiento del cuerpo *B;* visto desde el sistema referencia *A.* Entonces, el par-momento, ${}^{A}S_{O}^{B}$, se define como (García, 2014):

$$ {}^{A}S_{O}^{B} = {}_{A}h_{B}.\ {}^{A}S^{B} + {}^{A}S^{B} \times r_{O/P} \qquad \text{Ecuación 1.31} $$

Un tornillo normalizado se representa como:

$$ {}^{A}\dot{\$}^{B} = \begin{bmatrix} {}^{A}\dot{S}^{B} \\ {}^{A}\dot{S}_{O}^{B} \end{bmatrix};\ |{}^{A}\dot{S}^{B}| = 1 \qquad \text{Ecuación 1.32} $$

El estado de velocidad de un cuerpo rígido *B*, visto desde el cuerpo A, es el giro infinitesimal de un tornillo y se representa como (García, 2014):

$$ {}^{A}V^{B} = {}^{A}\omega^{B}\ {}^{A}\dot{\$}^{B} = {}^{A}\omega^{B} \begin{bmatrix} {}^{A}\dot{S}^{B} \\ {}^{A}\dot{S}_{O}^{B} \end{bmatrix} = \begin{bmatrix} {}^{A}\omega^{B}\ {}^{A}\dot{S}^{B} \\ {}^{A}\omega^{B}\ {}^{A}\dot{S}_{O}^{B} \end{bmatrix} \qquad \text{Ecuación 1.33} $$

donde ${}^{A}\omega^{B}$ es la velocidad angular del cuerpo rígido en movimiento mientras que ${}^{A}v_{O}^{B}$ es la velocidad del punto O, fijo al mismo cuerpo, y que en el instante considerado coincide con un punto fijo al sistema de referencia.

De donde (García, 2014),

$$ {}^{A}\omega^{B}\ {}^{A}\$^{B} = \begin{bmatrix} {}^{A}\omega^{B} & {}^{A}S^{B} \\ {}^{A}\omega^{B} & {}^{A}S_{O}^{B} \end{bmatrix} = \begin{bmatrix} {}^{A}\omega^{B} \\ {}^{A}v_{O}^{B} \end{bmatrix} = {}^{A}V^{B} $$ Ecuación 1.34

1.6.3. Representación de los pares cinemáticos

A continuación, las representaciones más usuales, mediante tornillos de los pares cinemáticos, son: ver tabla 1.2 y figura 1.11.

Tabla 1.2. Pares cinemáticos representados mediante tornillo.

Par cinemático	Definición	Representación
Par de revoluta, R	El paso *h* del tornillo es cero.	$\$_R = \begin{bmatrix} S \\ S \times r + hS \end{bmatrix}$
Par prismático, P	Permite una traslación de un eslabón respecto a otro a lo largo de un eje.	$\$_P = \begin{bmatrix} 0 \\ S \end{bmatrix}$
Par cilíndrico, C	Esta clase de par se puede simular mediante la acción simultanea de un par de revoluta, $\$_R$, y un par prismático, $\$_P$.	$\$_C$ Combinación $\$_R$ y $\$_P$
Par helicoidal, H	Unión de dos elementos que rotan y se trasladan uno con respecto al otro a lo largo de un eje definido por la geometría de la unión. Sin embargo, la traslación se relaciona con la rotación por medio del paso, *h* del tornillo.	
Par esférico, S	Se modela mediante tres tornillos cuyas direcciones están representadas por vectores linealmente independientes, usualmente perpendiculares y que pasan por el centro geométrico del par esférico.	

(Fuente: (García, 2014))

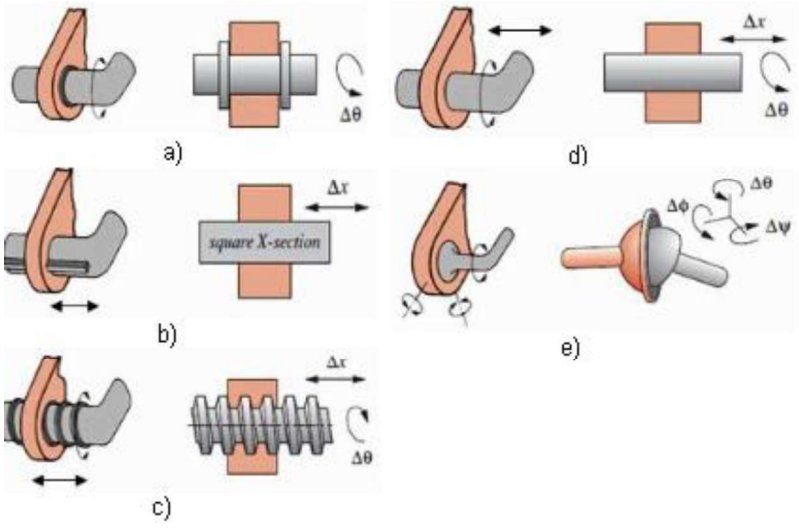


Figura 1.11. Pares cinemáticos más usuales. (a) Junta de revoluta; (b) par prismático; (c)par helicoidal; (d) par cilíndrico; (e) par esférico.

(Fuente: (García, 2014))

1.7. Tornillos recíprocos

Un concepto importante para simplificar este análisis es la reciprocidad (García, 2014). Dos tornillos $\$$ y $\$_r$, se dice que son reciprocos si satisfacen la siguiente condición:

$$\$^T{}_r\$ = 0 \qquad \text{Ecuación 1.35}$$

Donde la transpuesta de un tornillo es definida como:

$$\$^T = [S_4 \;\; S_5 \;\; S_6 \;\; S_1 \;\; S_2 \;\; S_3] \qquad \text{Ecuación 1.36}$$

Tal que el producto ortogonal de dos tornillos esta dado por:

$$\$^T{}_r\$ = S_{r4}S_1 + S_{r5}S_2 + S_{r6}S_3 + S_{r1}S_4 + S_{r2}S_5 + S_{r3}S_6 \qquad \text{Ecuación 1.37}$$

1.7.1. Condición de reciprocidad de dos tornillos

La condicion de reciprocidad de dos tornillos se describe en la siguiente tabla.

Tabla 1.3. Condición de reciprocidad de dos tronillos.

No.	Condición
1	Dos líneas de vector deben ser coplanar.
2	Dos pares siempre deben ser recíprocos.
3	La línea vector y par son recíprocos solo si ellos son perpendiculares.

(Fuente: Autor)

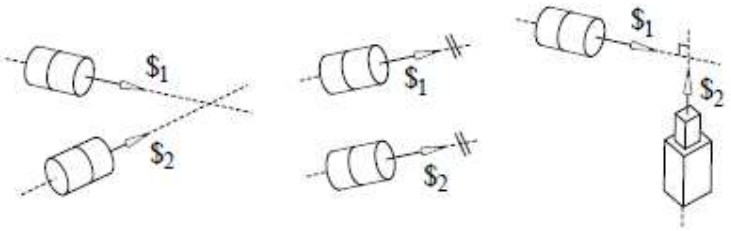


Figura 1.12. Tornillos recíprocos típicos.
(Fuente: (García, 2014))

1.8. Matriz jacobiana basada en la teoría de tornillos

Esta sección muestra cómo generar la matriz jacobiana utilizando el principio de la teoría de tornillos. Los movimientos instantáneos de los cuerpos rígidos pueden ser "llevados a cabo" por un tornillo, se conoce que el movimiento instantáneo de un cuerpo rígido de un elemento puede describirse por un par de vectores ω y v_O por lo que un tornillo $\$_i$ representa el movimiento del elemento i, entonces el par es: $\$_i = (\omega_i, v_{O_i})$.

Esta expresión, $\$_p = \grave{\$}_i \dot{q}$, permite determinar la matriz jacobiana casi directamente, algunas veces por simple inspección geométrica del robot.

El movimiento instantáneo del efector final $\$_n$ es la suma de los movimientos instantáneos, $\$_i$, de los elementos que componen una cadena serial.(Martins, 2002),(Ribeiro, Guenther, & Martins, 2008)

$$\dot{x} = \begin{bmatrix} \omega \\ v \end{bmatrix} = \$_n = \sum_{i=1}^{n} \$_i = \sum_{i=1}^{n} \begin{bmatrix} J_v \\ J_\omega \end{bmatrix} \dot{q}_i = \sum_{i=1}^{n} \hat{\$}_i \dot{q}_i \qquad \text{Ecuación 1.38}$$

La ecuación puede ser reescrita como:

$$\dot{x} = J\dot{q} \qquad \text{Ecuación 1.39}$$

Donde J es la matriz jacobiana dada por:

$$J = \left[\hat{\$}_1 \dots \hat{\$}_n\right] \qquad \text{Ecuación 1.40}$$

1.9. Homogenización de la matriz jacobiana

Cuando el robot es de más de tres grados de libertad, además controlan posición y orientación o sus articulaciones no son del mismo tipo, las unidades de los elementos de la matriz jacobiana no son homogéneas, y esta situación dificulta la localización de singularidades. (Moreno, Saltearen, Carrera, Puglisi, & Aracil)

Diferentes estrategias se han propuesto para resolver el problema:

Aplicación de la **longitud característica** L_c, (selección de esta longitud es arbitraria) para homogeneizar la matriz Jacobiana se dividen los elementos de la matriz relacionados a las componentes de la velocidad angular por la longitud característica. (Moreno, Saltearen, Carrera, Puglisi, & Aracil)

La matriz normalizada está dada por:

$$J = \begin{bmatrix} {s^T}_{i,1} & \left(b_1 \times s_{i,1}\right)^T \Big/ L_c \\ {s^T}_{i,2} & \left(b_2 \times s_{i,2}\right)^T \Big/ L_c \\ {s^T}_{i,3} & \left(b_3 \times s_{i,3}\right)^T \Big/ L_c \\ {s^T}_{i,4} & \left(b_4 \times s_{i,4}\right)^T \Big/ L_c \end{bmatrix} \qquad \text{Ecuación 1.41}$$

Donde:

$s_{i,j}$ Vectores unitarios a lo largo de las extremidades del manipulador.

b_i Vectores posición de las articulaciones sobre la plataforma móvil.

Sin embargo, esta estrategia no es confiable debido a que la selección de la longitud característica es en forma arbitraria y su significado físico es difícil de determinar.

La siguiente estrategia consiste en la partición de la matriz jacobiana en dos sub-matrices, ver ecuación 1. 48 y ecuación 1.49, correspondientes a la traslación y rotación del efector final. (Moreno, Saltearen, Carrera, Puglisi, & Aracil)

$$\begin{bmatrix} \boldsymbol{v} \\ \boldsymbol{\omega} \end{bmatrix} = \begin{bmatrix} \boldsymbol{J_v} \\ \boldsymbol{J_\omega} \end{bmatrix} \boldsymbol{q} \qquad \text{Ecuación 1.42}$$

donde v y ω son la velocidad lineal y angular del efector final.

1.10. Determinación de singularidades por matrices jacobianas de traslación y rotación

Dado el complicado proceso para determinar singularidades de un robot paralelo de 4 GDL que provee una matriz jacobiana de 4 X 6, se transforma en un simple proceso para determinar singularidades desde un jacobiano general **J**.

1.10.1. Matriz jacobiana de traslación y rotación

Dada la matriz jacobiana de 4X6, se utiliza una matriz jacobiana traslacional $\boldsymbol{J_v}$ de 3x4 y una matriz jacobiana rotacional $\boldsymbol{J_\omega}$ de 3x4, a partir de estas matrices de traslación/rotación genera un jacobiano general $\boldsymbol{J}$ de dimensiones 4x4. (Lu, Shi, & Yu, 2009)

1.10.2. Matriz jacobiana general de un robot paralelo de 4 GDL

Los Manipuladores Paralelos de 4 GDL se clasifican como: *1T3R, 2T2R y 3T1R* según sus características de movimiento, donde *T* es traslación y *R* es la rotación. (Lu, Shi, & Yu, 2009). Ejemplo: como se muestra en la figura 1.13.

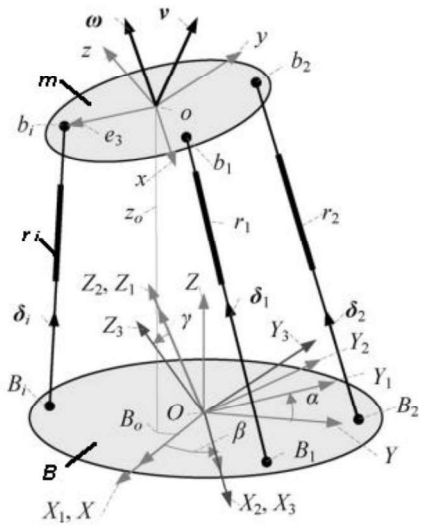


Figura 1.13. Robot paralelo de 4 GDL
(Fuente: (Lu, Shi, & Yu, 2009))

Donde:

B Plataforma fija $(O - XYZ)$

m Plataforma móvil $(o - xyz)$

$(r_i = 1,2 \ldots .3\ o\ 4)$ 3 o 4 extremidades activas con actuadores lineales.

b_i 3 o 4 vértices ubicados en la plataforma móvil.

B_i 3 o 4 vértices ubicados en la plataforma fija.

o Punto central en la plataforma móvil.

O Punto central en la plataforma fija.

Antes de determinar las singularidades es necesario obtener la cinemática inversa y las velocidades del robot. (Lu, Shi, & Yu, 2009)

Ecuación 1.43

$$B_i = \begin{bmatrix} X_{Bi} \\ Y_{Bi} \\ Z_{Bi} \end{bmatrix}, \qquad {}^{m}b_i = \begin{bmatrix} x_{bi} \\ y_{bi} \\ z_{bi} \end{bmatrix}, \quad o = \begin{bmatrix} X_o \\ Y_o \\ Z_o \end{bmatrix},$$

$$ {}^{B}_{m}R = \begin{bmatrix} x_l & y_l & z_l \\ x_m & y_m & z_m \\ x_n & y_n & z_n \end{bmatrix}, \quad b_i^m = {}^{B}_{m}R\, b_i + o$$

Ecuación 1.44

$$r_i = |b_i - B_i|, \ \ \delta_i = {}^{(b_i - B_i)}/_{r_i} = [\delta_{ix} \quad \delta_{iy} \quad \delta_{iz}]^T,$$

$$e_i = b_i - o_i = [e_{ix} \quad e_{iy} \quad e_{iz}]^T$$

Donde:

B_i Vector posición del punto B_i en B.

${}^{m}b_i$ Vector posición del punto b_i en m.

o Vector posicion del punto o sobre m en B.

${}^{B}_{m}R$ Matriz de rotacion de m con respecto a B.

δ_i Vector unitario de la extremidad activa.

e_i Vector desde el centro o de m a la junta b_i.

$r_i (i = 1,2,3\ o\ 4)$, longitud de la extremidad.

Ademas $s_\varphi = \sin\varphi$, $c_\varphi = \cos\varphi$, $t_\varphi = \tan\varphi$.

φ Representa a uno de los ángulos de euler (α, β, γ) de m.

C_1, C_2 y C_3 Vectores unitarios de la matriz rotacional en los ejes (α, β, γ), respectivamente.

$x_i (i = 1,2,3,4)$ Parámetros independientes de posicion de m donde, $x_i \epsilon (X_o,\ Y_o,\ Z,\ \alpha,\ \beta,\ \gamma)$ y se puede representar asi:

$$X_o = X_o(x_1 \quad x_2 \quad x_3 \quad x_4),$$
$$Y_o = X_o(x_1 \quad x_2 \quad x_3 \quad x_4),$$
$$Z_o = X_o(x_1 \quad x_2 \quad x_3 \quad x_4),$$
$$\alpha = \alpha(x_1 \quad x_2 \quad x_3 \quad x_4),$$
$$\beta = \beta(x_1 \quad x_2 \quad x_3 \quad x_4),$$
$$\gamma = \gamma(x_1 \quad x_2 \quad x_3 \quad x_4)$$

Ecuación 1.45

Acontinuacion, las ecuaciones de la velocidad y matriz jacobiana, diferenciando cada uno de los componentes de la posicion $(X_o,\ Y_o,\ Z_o,\ \alpha,\ \beta,\ \gamma)$ con respecto al tiempo se obtiene:

Ecuación 1.46

$$\dot{X}_o = \sum_{i=1}^{4} \frac{\partial X_o}{\partial x_i} \dot{x}_\iota \qquad \dot{Y}_o = \sum_{i=1}^{4} \frac{\partial Y_o}{\partial x_i} \dot{x}_\iota \qquad \dot{Z}_o = \sum_{i=1}^{4} \frac{\partial Z_o}{\partial x_i} \dot{x}_\iota$$

$$\dot{\alpha}_o = \sum_{i=1}^{4} \frac{\partial \alpha}{\partial x_i} \dot{x}_\iota \qquad \dot{\beta} = \sum_{i=1}^{4} \frac{\partial \beta}{\partial x_i} \dot{x}_\iota \qquad \dot{\gamma}_o = \sum_{i=1}^{4} \frac{\partial \gamma}{\partial x_i} \dot{x}_\iota$$

Ecuación 1.47

$$V = \begin{bmatrix} v \\ \omega \end{bmatrix}, \quad v = \begin{bmatrix} \dot{X}_o \\ \dot{Y}_o \\ \dot{Z}_o \end{bmatrix}, \quad \omega = \begin{bmatrix} \omega_x \\ \omega_y \\ \omega_z \end{bmatrix}, \quad v_e = \begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \\ \dot{x}_3 \\ \dot{x}_4 \end{bmatrix}$$

V representa la velocidad directa de m en el centro o que consta de v y ω que son velocidad lineal y angular respectivamente, v_e representa a la velocidad equivalente de $x_i (i = 1,2,3,4)$.

Con la ecuación 1.46 y la ecuación 1.47, v queda expresada como: (Lu, Shi, & Yu, 2009):

$$v = J_v v_e$$

$$\boldsymbol{J}_v = [J_{v1} \quad J_{v2} \quad J_{v3} \quad J_{v4}]$$

Ecuación 1.48

$$\boldsymbol{J}_v = \begin{bmatrix} \frac{\partial X_o}{\partial x_1} & \frac{\partial X_o}{\partial x_2} & \frac{\partial X_o}{\partial x_3} & \frac{\partial X_o}{\partial x_4} \\ \frac{\partial Y_o}{\partial x_1} & \frac{\partial Y_o}{\partial x_2} & \frac{\partial Y_o}{\partial x_3} & \frac{\partial Y_o}{\partial x_4} \\ \frac{\partial Z_o}{\partial x_1} & \frac{\partial Z_o}{\partial x_2} & \frac{\partial Z_o}{\partial x_3} & \frac{\partial Z_o}{\partial x_4} \end{bmatrix}$$

$\boldsymbol{J}_v$ Matriz jacobiana traslacional de 3x4.

ω es expresada desde la ecuación 1.49 de siguiente manera: (Lu, Shi, & Yu, 2009)

Ecuación 1.49

$$\omega = C \begin{bmatrix} \dot{\alpha} \\ \dot{\beta} \\ \dot{\gamma} \end{bmatrix} = J_\omega v_e,$$

$$C = [C_1 \quad C_2 \quad C_3] = \begin{bmatrix} C_{1x} & C_{2x} & C_{3x} \\ C_{1y} & C_{2y} & C_{3y} \\ C_{1z} & C_{2z} & C_{3z} \end{bmatrix},$$

$$\boldsymbol{J}_\omega = [J_{\omega 1} \quad J_{\omega 2} \quad J_{\omega 3} \quad J_{\omega 4}]$$

$$\boldsymbol{J}_\omega = C \begin{bmatrix} \frac{\partial \alpha}{\partial x_1} & \frac{\partial \alpha}{\partial x_2} & \frac{\partial \alpha}{\partial x_3} & \frac{\partial \alpha}{\partial x_4} \\ \frac{\partial \beta}{\partial x_1} & \frac{\partial \beta}{\partial x_2} & \frac{\partial \beta}{\partial x_3} & \frac{\partial \beta}{\partial x_4} \\ \frac{\partial \gamma}{\partial x_1} & \frac{\partial \gamma}{\partial x_2} & \frac{\partial \gamma}{\partial x_3} & \frac{\partial \gamma}{\partial x_4} \end{bmatrix}$$

$\boldsymbol{J}_\omega$ Matriz jacobiana rotacional de 3x4.

Combinando la ecuación 1.48 y la ecuación 1.49 da como resultado: (Lu, Shi, & Yu, 2009)

$$J_e = \begin{bmatrix} J_v \\ J_\omega \end{bmatrix} = \begin{bmatrix} J_{v1} & J_{v2} & J_{v3} & J_{v4} \\ J_{\omega 1} & J_{\omega 2} & J_{\omega 3} & J_{\omega 4} \end{bmatrix}_{6\times 4}$$

Ecuación 1.50

Donde.

$\boldsymbol{J_e}$ Matriz jacobiana equivalente de 6 x 4

La velocidad escalar v_{ri} de r_i es la siguiente: (Lu, Shi, & Yu, 2009)

$$v_{ri} = [\delta_i^T \quad (e_i \times \delta_i)^T]V. \qquad \text{Ecuación 1.51}$$

Cuando nos dan 4 entradas de desplazamiento traslacionales $r_i(1,2,3,4)$ a lo largo de las extremidades activas, el vector desplazamiento X_{in} y su vector velocidad V_{in} es expresado como: (Lu, Shi, & Yu, 2009)

$$\boldsymbol{X}_{in} = [r_1 \quad r_2 \quad r_3 \quad r_4\ ]^T \qquad \text{Ecuación 1.52}$$
$$\boldsymbol{v}_{in} = [v_{r1} \quad v_{r2} \quad v_{r3} \quad v_{r4}\ ]^T$$

A partir de la ecuación 1.50 a la ecuación 1.51 se expresa:

$$\boldsymbol{v}_{in} = J_e V = J_r J_e v_e = J v_e \qquad \text{Ecuación 1.53}$$

Donde:

$$v_e = J^{-1} v_{in}, \quad J = [J_r]_{4\times6}\,[J_e]_{6\times4}$$

$$v_{in} = \begin{bmatrix} v_{r1} \\ v_{r2} \\ v_{r3} \\ v_{r4} \end{bmatrix}, \quad J_r = \begin{bmatrix} \delta_1^T & (e_1 \times \delta_1)^T \\ \delta_2^T & (e_2 \times \delta_2)^T \\ \delta_3^T & (e_3 \times \delta_3)^T \\ \delta_4^T & (e_4 \times 4)^T \end{bmatrix}_{4\times6}$$

Entonces $\boldsymbol{J}$ es la matriz jacobiana general 4 x 4, y $\boldsymbol{J^{-1}}$ existe.

$|\boldsymbol{J}|$ Determinante de la matriz jacobiana general.

Donde:

1. Cuando $|\boldsymbol{J}| = \boldsymbol{0}$, las singularidades se presentan en los límites del manipulador.
2. Cuando $|\boldsymbol{J}| \rightarrow \infty$, singularidades locales en el manipulador.
3. Cuando $|\boldsymbol{J}| \rightarrow \boldsymbol{0/0}$, las singularidades ocurren en la estructura del manipulador.

1.11. Espacio de trabajo y su representación.

El espacio de trabajo es la región que puede alcanzar el efector final, la representación del espacio de trabajo en manipuladores paralelos con más de tres grados de libertad es un problema bastante complejo, debido a que la posición y la orientación del efector final están fuertemente acoplados y un espacio de n>3 dimensiones no es representable ni

interpretable por lo tanto aparecen diferentes tipos de espacios de trabajo, ver tabla 1.4, tomando en cuenta 4 factores y son:(Oliva, 2014).

- Geometría del manipulador.
- Limitaciones de los actuadores.
- Restricciones de las juntas pasivas.
- Auto-colisiones.

Tabla 1.4. Tipos de espacio de trabajo.

Tipos de espacio de trabajo	
Espacio de trabajo a orientación constante	El volumen que puede cubrir el punto de referencia del órgano terminal manteniendo este una orientación determinada
Espacio de trabajo en orientación	El conjunto de todas las orientaciones que pueden ser adoptadas por el efector final manteniendo fijo su origen en un punto.
Espacio de trabajo alcanzable	El conjunto de todas las posiciones que pueden ser alcanzadas por el origen de efector final con al menos una orientación de la plataforma.
Espacio de trabajo en orientación total	El conjunto de posiciones en las que el punto de referencia del efector final puede alcanzar con todas las orientaciones de un conjunto dado mediante rangos de los ángulos de orientación.
Espacio de trabajo diestro	El conjunto de posiciones en las que el punto de referencia del efector final puede alcanzar todas las orientaciones posibles. Es un caso particular del espacio de trabajo en orientación total donde los rangos de orientación son [0°, 360º].
Espacio de trabajo en orientación total reducido	El conjunto de posiciones en las que el punto de referencia del efector final puede alcanzar con todas las orientaciones de un conjunto dado mediante rangos de 1 ó 2 de los ángulos de orientación dejando el resto de los ángulos en un valor fijo.

(Fuente: (Oliva,2014))

Para representación del espacio de trabajo de robots paralelos se tiene dos formas:

Algoritmos de discretización: Para localizar el espacio se divide según una malla cartesiana o polar y en cada punto se resuelve la cinemática inversa del robot considerando las limitaciones de los actuadores, las articulaciones pasivas y las auto-colisiones entre extremidades, estableciéndose así los puntos que pertenecen al espacio de trabajo y la frontera de este.

La calidad de esta depende de la finura del mallado y, con la calidad, se incrementa notablemente el tiempo de cálculo de los computadores (Oliva,2014).

Métodos geométricos: Se determina el espacio de trabajo en base a las restricciones geométricas impuestas por las dimensiones del robot, los límites de los actuadores y las articulaciones pasivas e, incluso, las intersecciones entre eslabones (Oliva,2014).

Este método proporciona soluciones exactas del espacio de trabajo a un bajo coste computacional pero no es viable para robot paralelos con arquitectura compleja.

1.11.1. Algoritmo para el cálculo del volumen de trabajo

El método de discretización es el más fácil para el cálculo de los volúmenes de trabajo en robots complejos, obviamente este método demanda un enorme costo computacional dependiendo del paso deseado para la malla.

Las restricciones cinemáticas básicas que limitan el espacio de trabajo se ilustran a continuación (Conti, Clinton, & Zhang, 1998) (Li, Cao, Zhang, & Wang, 2013):

1.11.2. Restricciones de las articulaciones y colisiones

Un elemento de interés en el análisis del espacio de trabajo de un robot paralelo y que tiene relación directa en su realización física son las restricciones de movimiento de los actuadores, las articulaciones pasivas y las colisiones entre eslabones.

a) Restricción de desplazamiento de actuadores prismáticos

Si el manipulador paralelo tiene actuadores prismáticos se define una restricción de distancia $L_{min} < L_i < L_{max}$.

b) Restricción de Articulaciones universales y esféricas

Es un movimiento relativo que no debe exceder la región definida por un cono circular. El eje del cono es normal a la articulación y la apertura de este, es dada por el valor δ_{max} y el ápice en el centro de la articulación.

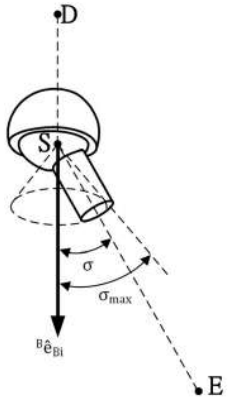


Figura 1.14. Ángulo de rotación de la junta esférica.
(Fuente: (Huang & Tsai, 2014))

c) Restricción de articulaciones rotacionales

Está definido entre un máximo y un mínimo ángulo $\theta_{min} < \theta_i < \theta_{max}$, que corresponde con un valor de giro.

d) Colisiones entre eslabones

Se consideran los eslabones como cuerpos cilíndricos; para determinar la colisión, se determina la distancia entre dos cilindros, dada como la mínima distancia a lo largo de la perpendicular común.

La carrera de los actuadores es uno de los principales factores que limitan el espacio de trabajo del manipulador. Generalmente, hay un mínimo y máximo valor para las posibilidades físicas de la longitud de los actuadores prismáticos asociados, el mismo que se toma en cuenta en el desarrollo del algoritmo (Conti, Clinton, & Zhang, 1998) (Li, Cao, Zhang, & Wang, 2013).

1.11.3. Implementación computacional

Para determinar el espacio de trabajo a orientación constante del robot, existe un método de búsqueda numérica, que fue inicialmente aplicado a un robot Hexápodo (Conti, Clinton, & Zhang, 1998). En este algoritmo, se emplean coordenadas esféricas para encontrar los límites del espacio de trabajo.

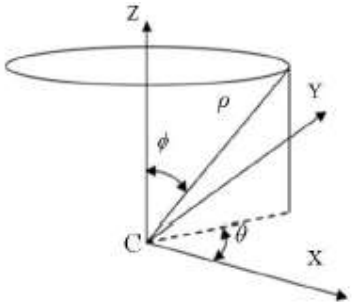


Figura 1.15. Coordenadas esféricas.
(Fuente: (Gharahsofloo & Rahmani, 2015)

En primer lugar, se necesita localizar un punto central *C* aproximado que se encuentre en el espacio de trabajo ver figura 1.16, es un requerimiento general para inicializar la búsqueda del espacio de trabajo, entonces para cualquier conjunto dado de φ y θ ver figura 1,15, se encuentra el radio del punto $P = (X_p,\ Y_p,\ Z_p)$ en espacio de trabajo, es decir, ρ ver figura 1.16.

- Si *P<0* el punto está dentro del espacio de trabajo.
- Si *P=0* el punto está en el borde del espacio de trabajo.

- Si *P>0* el punto está afuera del espacio de trabajo.

Variando los valores de φ, θ, se encuentra todo el volumen del espacio de trabajo (Gharahsofloo & Rahmani, 2015).

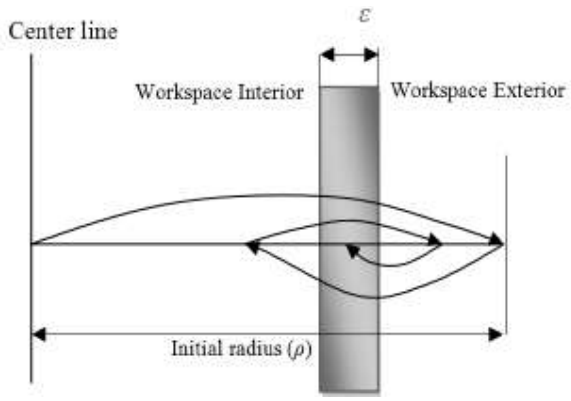


Figura 1.16. Algoritmo de búsqueda del vector
(Fuente: (Gharahsofloo & Rahmani, 2015))

2. METODOLOGÍA

Para el análisis de singularidades y representación del espacio de trabajo del robot paralelo 3UPE-RPU con 4 GDL se toma como referencia a los trabajos realizados por la FIM.
Los pasos que seguir para realizar este trabajo son los siguientes:

- Revisar la modelación cinemática del robot paralelo **3UPE-RPU**, (Fernández & Sotomayor, 2016), (Morales, 2017).
- Establecer la matriz característica a partir de las ecuaciones de restricción, ver ecuación 1.23, utilizada en la modelación dinámica del robot paralelo conocida también como la matriz jacobiana analítica.
- Obtener la matriz jacobiana geométrica a partir de la teoría de tornillos, ver ecuación 1.40.
- Homogenizar la matriz jacobiana obtenida en el paso anterior y analizar las singularidades de acuerdo con el marco teórico expuesto en la sección anterior.
- Obtener el espacio de trabajo a orientación constante utilizando las ecuaciones de la cinemática inversa y un algoritmo de discretización para representar el mismo.

2.1. Análisis cinemático

2.1.1. Cinemática inversa, posición

La cinemática inversa equivale a dada la posición $(X_m;\ Y_m;\ Z_m;\ \psi;\ \phi;\ \theta)$ de la plataforma móvil, hallar la posición de las variables ($A_1, A_2\ , A_3, B_1, B_2\ , B_3, C_1, C_2\ , C_3, D_1\ , D_2$) que están en las articulaciones del robot 3UPE-RPU.

Para el siguiente análisis es necesario las ecuaciones que describe la cinemática del robot, las cuales son desarrollados como lazo cerrado.

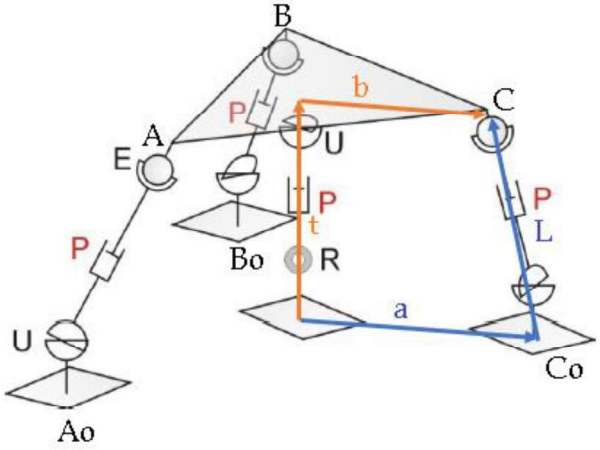


Figura 2.1. Cadenada cinemática para la pata 3.
(Fuente: Autor)

A partir de la geometría del 3UPE-RPU ilustrada en la figura 2.1 la relación cinemática en lazo cerrado para cada pata, i=1,2,3,4, esta expresada como:

$$L_i S_i = t + Rp * b_i - a_i \qquad \text{Ecuación 2.1}$$

$$t + Rp * b_i = T_R = \begin{bmatrix} \cos(\phi)\cos(\theta) & -\sin(\theta) & \cos(\theta)\sin(\phi) & X_m \\ \cos(\phi)sin(\theta) & \cos(\theta) & \sin(\phi)sin(\theta) & 0 \\ -\sin(\phi) & 0 & \cos(\phi) & Z_m \\ 0 & 0 & 0 & 1 \end{bmatrix} \qquad \text{Ecuación 2.2}$$

$$a_i = \begin{bmatrix} Rcos\beta_i \\ Rsin\beta_i \\ 0 \end{bmatrix} \qquad \text{Ecuación 2.3}$$

Donde:

a_i Vector de posición de la plataforma fija

Rp Matriz de rotación.

S_i Vector unitario i-ésima de la extremidad.

L_i La longitud i-ésima de la extremidad es determinada a partir de la norma euclidiana.

T_R Matriz de transformación de rotación y traslación

Con las ecuaciones de ubicación de la plataforma móvil y plataforma fija conocidas en la sección anterior, ver ecuaciones 1.2 hasta 1.9 y la matriz de transformación de rotación y traslación, se obtiene las siguientes ecuaciones:

- **Vectores posición de la plataforma móvil trasladada y rotada**

$$A^p = T_R A = \begin{bmatrix} X_m - R_m \cos(\phi)\cos(\theta) \\ -R_m \cos(\phi)\sin(\theta) \\ Z_m + R_m \sin(\phi) \\ 1 \end{bmatrix} \quad \text{Ecuación 2.4}$$

$$B^p = T_R B = \begin{bmatrix} X_m - R_m \sin(\alpha_1)\sin(\theta) + R_m \cos(\alpha_1)\cos(\phi)\cos(\theta) \\ R_m \, s\,in(\alpha_1)\cos(\theta) + R_m \cos(\alpha_1)\cos(\phi)\sin(\theta) \\ Z_m - R_m \cos(\alpha_1)\sin(\phi) \\ 1 \end{bmatrix} \quad \text{Ecuación 2.5}$$

$$C^p = T_R C = \begin{bmatrix} X_m - R_m \sin(\alpha_2)\sin(\theta) + R_m \cos(\alpha_2)\cos(\phi)\cos(\theta) \\ R_m \sin(\alpha_2)\cos(\theta) + R_m \cos(\alpha_2)\cos(\phi)\sin(\theta) \\ Z_m - R_m \cos(\alpha_2)\sin(\phi) \\ 1 \end{bmatrix} \quad \text{Ecuación 2.6}$$

$$O_m^p = T_R O_m = [X_m \quad 0 \quad Z_m \quad 1]^T \quad \text{Ecuación 2.7}$$

Con la posición de la plataforma móvil $X = [X_m;\ Y_m;\ Z_m;\ \psi;\ \phi;\ \theta]$ se obtiene las coordenadas generalizadas de las 4 extremidades, ver la ecuación 1.20, los valores de interés son $A_1,\ B_1,\ C_1$ para los brazos UPE y D_1 para el brazo RPU, que equivalen a la longitud de los 4 actuadores lineales.

Para el caso de estudio es necesario conocer las longitudes de las extremidades y se puede calcular mediante las siguientes fórmulas:

$$A_1 = L_1 = \|A^p - A_o\|$$

$$L_1 = \left\| \begin{bmatrix} R + X_m - R_m \cos(\phi)\cos(\theta) \\ -R_m \sin(\theta)\cos(\phi) \\ Z_m + R_m \sin(\phi) \end{bmatrix} \right\| \quad \text{Ecuación 2.8}$$

$$B_1 = L_2 = \|B^p - B_o\|$$

$$L_2 = \left\| \begin{bmatrix} X_m - Rcos(\beta_1) - R_m \sin(\alpha_1)\sin(\theta) + R_m \cos(\alpha_1)\cos(\theta)\cos(\phi) \\ R_m \sin(\alpha_1)\cos(\theta) - Rsin(\beta_1) + R_m \cos(\alpha_1)\cos(\phi)\sin(\theta) \\ Z_m - R_m \cos(\alpha_1)\sin(\phi) \end{bmatrix} \right\| \quad \text{Ecuación 2.9}$$

$$C_1 = L_3 = \|C^p - C_o\| \qquad \text{Ecuación 2.10}$$

$$L_3 = \left\| \begin{bmatrix} X_m - Rcos(\beta_2) - R_m \sin(\alpha_2)\sin(\theta) + R_m \cos(\alpha_2)\cos(\theta)\cos(\phi) \\ -Rsin(\beta_2) + R_m \cos(\alpha_2)\cos(\phi)\sin(\theta) + R_m \cos(\theta)\sin(\alpha_2) \\ Z_m - R_m \cos(\alpha_2)\sin(\phi) \end{bmatrix} \right\|$$

$$D_1 = L_4 = \left\|O_m^p - O_m\right\| \qquad \text{Ecuación 2.11}$$

$$L_4 = \| \ [X_m \quad 0 \quad Z_m \quad 1]^T \ \|$$

- **Vectores unitarios a lo largo de la junta prismática**

Ecuación 2.12

$$S_1 = \frac{A^p - A_o}{\|A^p - A_o\|}, \qquad S_2 = \frac{B^p - B_o}{\|B^p - B_o\|}, \qquad S_3 = \frac{C^p - C_o}{\|C^p - C_o\|}, \qquad S_4 = \frac{O_m^p - O_f}{\left\|O_m^p - O_f\right\|}$$

2.1.2. Cinemática directa, posición

En el problema de la cinemática inversa, se obtuvo una serie de ecuaciones no lineales, que tienen una gran ventaja: están desacopladas. El problema de la cinemática directa en los robots de arquitectura paralela ya no es tan simple. La manera tradicional para llegar a una solución resulta en un conjunto de ecuaciones no lineales acopladas de múltiples soluciones en la cual se debe utilizar herramientas computacionales, este apartado no enfatiza en la resolución de la cinemática directa, sin embargo, si, se determina la matriz jacobiana analítica.

Se inicia con las ecuaciones de restricción, ver ecuación 1.23. Luego particionando las coordenadas generalizadas en independientes $q_{ind} = (A_1, B_1, C_1, D_1)$ y dependientes $q_{dep} = (X_m, Z_m, \theta, \phi, A_2, A_3, B_2, B_3, C_2, C_3, D_2)$, se aplica la ecuación 1.15.

A continuación, se indica el programa en Matlab que calcula la matriz Jacobiana dependiente e independiente del robot 3UPE-RPU a partir de la ecuación 2.13. y ecuación 2.14.

```
% ECUACIONES DE RESTRICCIÓN

 f(1)=Rm*cos(phi)*cos(th) - Xm - R - A1*cos(A3)*sin(A2);
 f(2)=Rm*cos(phi)*sin(th) - A1*sin(A3);
 f(3)=A1*cos(A2)*cos(A3) - Rm*sin(phi) - Zm;
 f(4)=B1*sin(B3)*sin(bB) - R*cos(bB) - Xm + Rm*sin(al1)*sin(th) -
Rm*cos(al1)*cos(phi)*cos(th) - B1*cos(B3)*sin(B2)*cos(bB);
```

```
 f(5)=- R*sin(bB) - B1*sin(B3)*cos(bB) - Rm*sin(al1)*cos(th) -
Rm*cos(al1)*cos(phi)*sin(th) - B1*cos(B3)*sin(B2)*sin(bB);
 f(6)=Rm*cos(al1)*sin(phi) - Zm + B1*cos(B2)*cos(B3);
 f(7)=C1*sin(C3)*sin(bC) - R*cos(bC) - Xm + Rm*sin(al2)*sin(th) -
Rm*cos(al2)*cos(phi)*cos(th) - C1*cos(C3)*sin(C2)*cos(bC);
 f(8)=- R*sin(bC) - C1*sin(C3)*cos(bC) - Rm*sin(al2)*cos(th) -
Rm*cos(al2)*cos(phi)*sin(th) - C1*cos(C3)*sin(C2)*sin(bC);
 f(9)=Rm*cos(al2)*sin(phi) - Zm + C1*cos(C2)*cos(C3);
 f(10)=D1*sin(D2) - Xm;
 f(11)=D1*cos(D2) - Zm;

 % VARIABLES INDEPENDIENTES
 %-------------------------------------------------------------------

 A1 = sym('A1','real');B1 = sym('B1','real');C1 = sym('C1','real');
 D1 = sym('D1','real');
 VAR_IND=[A1,B1,C1,D1];

 % VARIABLES SECUNDARIAS(DEPENDIENTES)
 %-------------------------------------------------------------------

 A2 = sym('A2','real');B2 = sym('B2','real');C2 = sym('C2','real');
 D2 = sym('D2','real');
 A3 = sym('A3','real');B3 = sym('B3','real');C3 = sym('C3','real');
 Xm = sym('Xm','real');Zm = sym('Zm','real');th = sym('th','real');
 phi = sym('phi','real');
 VAR_S=[Xm,Zm,th,phi,A2,A3,B2,B3,C2,C3,D2];

 for i=1:11
     for j=1:11
         Js(i,j)=diff(f(i),VAR_S(j));
     end
 end
 Jdep=Js;


for i=1:11
     for j=1:4
         Jid(i,j)=diff(f(i),VAR_IND(j));
     end
 end
 Jind=Jid;
 %spy(Jind)
```

Luego la matriz Jacobiana obtenida en el código anterior es:

$$J_{dep} = \left[\frac{\partial \Phi(\mathrm{x},\mathrm{q})}{\partial q_{dep}}\right]_{11\times 11}$$ Ecuación 2.13

La jacobiana esta presentada en columnas $(11 \times n)$debido al tamaño de sus dimensiones.

$$J_{dep} = \begin{bmatrix} -1 & 0 & -R_m\cos(\phi)\sin(\theta) \\ 0 & 0 & R_m\cos(\theta) \\ 0 & -1 & R_m\sin(\phi)\sin(\theta) \\ -1 & 0 & R_m\cos(\alpha)\cos(\phi)\sin(\theta) + R_m\cos(\phi)\sin(\alpha)\cos(\theta) \\ 0 & 0 & R_m\sin(\alpha)\sin(\theta) - R_m\cos(\alpha)\cos(\theta) \\ 0 & -1 & -R_m\cos(\alpha)\sin(\phi)\sin(\theta) - R_m\sin(\alpha)\cos(\theta)\sin(\phi) \\ -1 & 0 & R_m\cos(\alpha)\cos(\phi)\sin(\theta) - R_m\cos(\phi)\sin(\alpha)\cos(\theta) \\ 0 & 0 & -R_m\cos(\alpha)\cos(\theta) - R_m\sin(\alpha)\sin(\theta) \\ 0 & -1 & R_m\sin(\alpha)\cos(\theta)\sin(\phi) - R_m\cos(\alpha)\sin(\phi)\sin(\theta) \\ -1 & 0 & 0 \\ 0 & -1 & 0 \end{bmatrix}_{11\times 3}$$

$$J_{dep} = \begin{bmatrix} -R_m\cos(\theta)\sin(\phi) \\ 0 \\ -R_m\cos(\phi)\cos(\theta) \\ R_m\cos(\alpha)\cos(\theta)\sin(\phi) - R_m\sin(\alpha)\sin(\phi)\sin(\theta) \\ 0 \\ R_m\cos(\alpha)\cos(\phi)\cos(\theta) - R_m\cos(\phi)\sin(\alpha)\sin(\theta) \\ R_m\cos(\alpha)\cos(\theta)\sin(\phi) + R_m\sin(\alpha)\sin(\theta)\sin(\phi) \\ 0 \\ R_m\cos(\alpha)\cos(\phi)\cos(\theta) + R_m\cos(\phi)\sin(\alpha)\sin(\theta) \\ 0 \\ 0 \end{bmatrix}_{11\times 4}$$

$$J_{dep}$$
$$= \begin{bmatrix} -A_1\cos(A_2)\cos(A_3) & A_1\sin(A_2)\sin(A_3) & 0 \\ 0 & -A_1\cos(A_3) & 0 \\ -A_1\cos(A_3)\sin(A_2) & -A_1\cos(A_2)\sin(A_3) & 0 \\ 0 & 0 & -B_1\cos(B_2)\cos(B_3)\cos(bB) \\ 0 & 0 & -B_1\cos(B_2)\cos(B_3)\sin(bB) \\ 0 & 0 & -B_1\cos(B_3)\sin(B_2) \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}_{11\times 7}$$

$$J_{dep}$$
$$= \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ B_1\cos(B_3)\sin(bB) + B_1\sin(B_2)\sin(B_3)\cos(bB) & 0 \\ B_1\sin(B_2)\sin(B_3)\sin(bB) - B_1\cos(B_3)\cos(bB) & 0 \\ -B_1\cos(B_2)\sin(B_3) & 0 \\ 0 & -C_1\cos(C_2)\cos(C_3)\cos(bC) \\ 0 & -C_1\cos(C_2)\cos(C_3)\sin bC \\ 0 & -C_1\cos(C_3)\sin(C_2) \\ 0 & 0 \\ 0 & 0 \end{bmatrix}_{11\times 9}$$

$$J_{dep} = \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ C_1\cos(C_3)\sin(bC) + C_1\sin(C_2)\sin(C_3)\cos(bC) & 0 \\ C_1\sin(C_2)\sin(C_3)\sin(bC) - C_1\cos(C_3)\cos(bC) & 0 \\ -C_1\cos(C_2)\sin(C_3) & 0 \\ 0 & D_1\cos(D_2) \\ 0 & -D_1\sin(D_2) \end{bmatrix}_{11\times 11}$$

La estructura de la matriz jacobiana J_{dep}, construida a partir de las primitivas anteriores, no se puede observar su dispersión por la dimensionalidad de 11x11, entonces se usa la función **spy** de Matlab donde los ceros en la matriz se representan como espacios en blanco y el número de elementos distintos de cero con (nz). Observar figura 2.2. dispersión de la matriz dependiente.

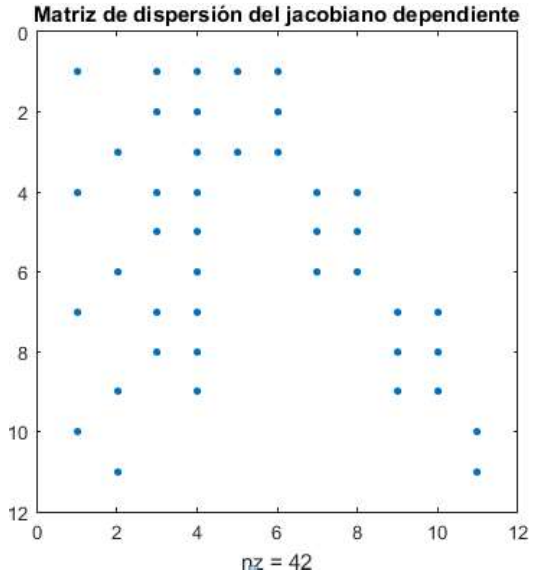


Figura 2.2. Estructura de la matriz jacobiana dependiente.
(Fuente: Autor)

Para resolución del problema de velocidades se necesita determinar el jacobiano independiente ver ecuación 1.17, y se procede igual para:

$$J_{ind} = \left[\frac{\partial \Phi(\mathrm{x},\mathrm{q})}{\partial q_{ind}}\right]_{11\times 4} \qquad \text{Ecuación 2.14}$$

$$J_{ind} = \begin{bmatrix} -\cos(A_3)\sin(A_2) & 0 \\ -\sin(A_3) & 0 \\ \cos(A_2)\cos(A_3) & 0 \\ 0 & \sin(B_3)\sin(bB) - \cos(B_3)\sin(B_2)\cos(bB) \\ 0 & -\sin(B_3)\cos(bB) - \cos(B_3)\sin(B_2)\sin(bB) \\ 0 & \cos(B_2)\cos(B_3) \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \end{bmatrix}_{11\times 2}$$

$$J_{ind} = \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ \sin(C_3)\sin(bC) - \cos(C_3)\sin(C_2)\cos(bC) & 0 \\ -\sin(C_3)\cos(bC) - \cos(C_3)\sin(C_2)\sin(bC) & 0 \\ \cos(C_2)\cos(C_3) & 0 \\ 0 & \sin(D_2) \\ 0 & \cos(D_2) \end{bmatrix}_{11\times 4}$$

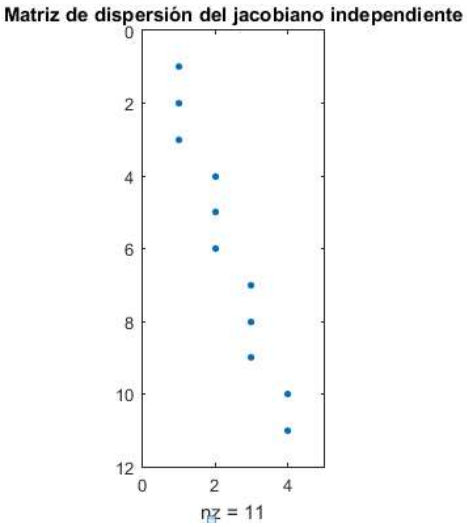


Figura 2.3. Estructura de la matriz jacobiana independiente.
(Fuente: Autor)

La manera como se obtuvieron las matrices anteriores le dan el nombre de jacobiano analítico.

Para el análisis de singularidades, esta matriz es muy compleja de interpretar, por lo que se usa la teoría de tornillos para determinar la matriz jacobiana geométrica.

2.1.3. Determinación de la matriz jacobiana basado en la teoría de tornillos

Para el cálculo de la matriz jacobiana geométrica se usa un método que facilita su hallazgo conocido como teoría de tornillos, este procedimiento simplifica la identificación de configuraciones singulares, una tarea que a través de los procedimientos derivativos convencionales tiende a ser complicado.

Para determinar la matriz jacobiana es necesario desarrollar el modelo de velocidad del robot, el cual es presentado en este trabajo a través de la teoría de tornillos, la cual se basa en que las articulaciones de un mecanismo pueden ser asociadas a tornillos infinitesimales, cuya representación depende del tipo de articulación a tratar, considérese la geometría de la figura 2.4.

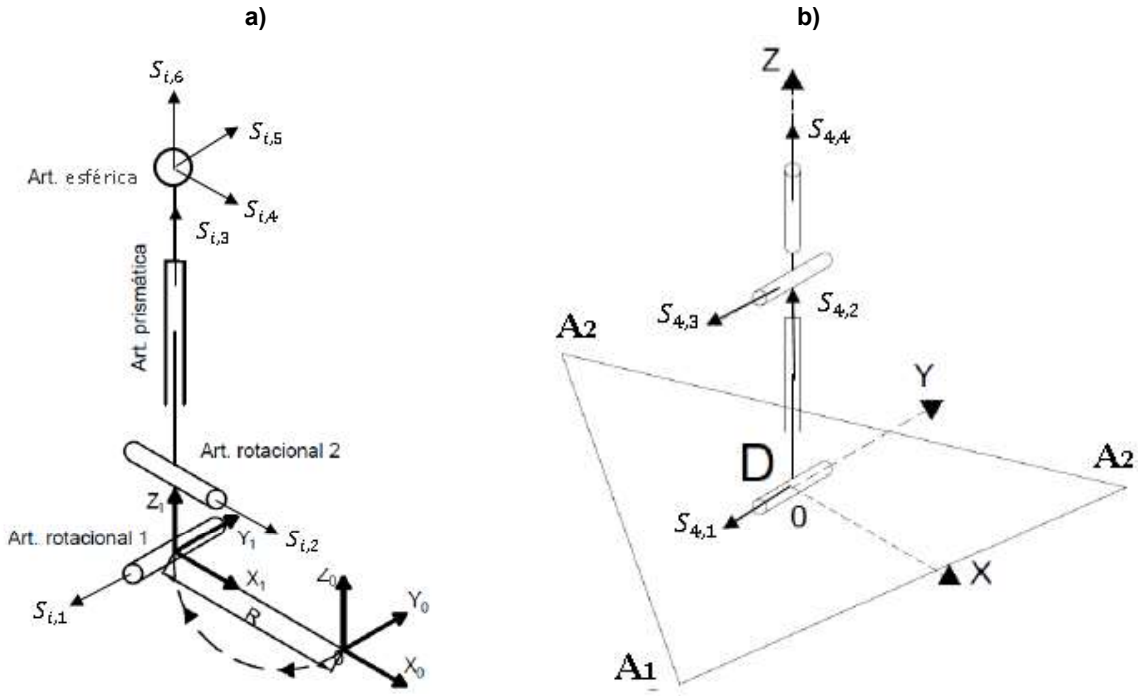


Figura 2.4. a) Ubicación de los tornillos en una extremidad, b) Tornillos en la extremidad central. (Fuente: Autor).

Cada brazo se modela como un manipulador serie de tipo Universal, Prismático, Esférico, tal y como se muestra en la Figura 2.4.a).

Nótese que la junta universal, se descompone en dos pares de revoluta, $\$_{i,1}$ y $\$_{i,2}$, cuyos ejes son ortogonales y se intersecan. Por su parte el par prismático se encuentra asociado al tornillo $\$_{i,3}$, mientras que el par esférico se representa mediante un conjunto de tres tornillos, $\{\$_{i,4}, \$_{i,5}, \$_{i,6}\}$,cuyos ejes se intersecan y son mutuamente ortogonales.

El brazo central se modela como un manipulador serie tipo Revoluta, Prismático, Universal, tal y como se muestra en la Figura 2.4.b).

La junta revoluta, tiene un tornillo, $\$_{4,1}$. El par prismático se encuentra asociado al tornillo $\$_{4,2}$, mientras que el par universal se representa mediante dos tornillos, $\$_{4,3}$ y $\$_{4,4}$, cuyos ejes se intersecan y son mutuamente ortogonales.

En forma general, el vector $S_{i,j}$ es un vector unitario en la dirección del j-ésimo eje de la articulación de la i-ésima del brazo, ver figura 2.5. De esta forma, los tornillos para cada uno de los ejes de las articulaciones son:

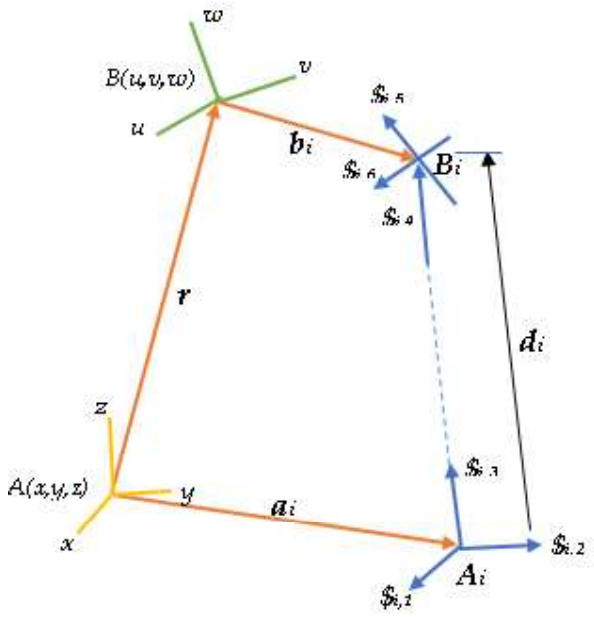


Figura 2.5. Ubicación de los tornillos en una extremidad del 3UPE-RPU.
(Fuente: Autor)

El tornillo $\$_{i,1}$ está asociado a la primera revoluta de su correspondiente cadena cinemática i. Su eje instantáneo es perpendicular al plano de la plataforma fija y puede ser calculado como:

$$\$_{i,1} = \begin{pmatrix} S_{i,1} \\ (b_i - d_i) \times S_{i,1} \end{pmatrix} \qquad \text{Ecuación 2.15}$$

Donde:

- Para encontrar un vector perpendicular al plano que contiene $\overline{A_i B_i}$ y a_i se realiza el producto cruz entre ellos:

$$\$_{i,1} = \frac{\$_{i,3} \times a_1}{\|\$_{i,3} \times a_1\|}$$

- a_i: vector posición de la i^{th}-ésima cadena en la plataforma fija. *i:1,2,3*
- b_i: vector posición de la i^{th}-ésima cadena en la plataforma móvil. *i:1,2,3*
- d_i: vector posición de la i^{th}-ésima de la articulación prismática. *i:1,2,3*

La segunda revoluta de la i^{th}-ésima cadena está representada por el tornillo $\$_{i,2}$.

$$\hat{\$}_{i,2} = \begin{pmatrix} S_{i,2} \\ a_i \times S_{i,2} \end{pmatrix}$$ Ecuación 2.16

Donde:

- El correspondiente vector unitario se encuentra dividiendo $\overline{A_i B_i}$ entre su magnitud:

$$\$_{i,3} = \frac{\overline{A_i B_i}}{\|\overline{A_i B_i}\|}$$

$$S_{i,2} = S_{i,3} \times S_{i,1}$$

Por otro lado, el tornillo que representa el par actuado se calcula como sigue:

$$\$_{i,3} = \begin{pmatrix} 0 \\ S_{i,3} \end{pmatrix}$$ Ecuación 2.17

El par esférico compuesto de tres pares de revoluta representados por los tornillos $\{\$_{i,4}\$_{i,5}$ $\$_{i,6}\}$ cuyos ejes son normales entre sí. Note que la dirección del primero de ellos es paralela a la del $S_{i,2}$, tal que:

$$\$_{i,4} = \begin{pmatrix} S_{i,4} \\ b_i \times S_{i,4} \end{pmatrix}$$ Ecuación 2.18

$\$_{i,4} = S_{i,2}$, además, el eje del tornillo, $\$_{i,3}$ es colineal con el del $\$_{i,6}$, calculando se da:

$$\$_{i,5} = \begin{pmatrix} S_{i,5} \\ b_i \times S_{i,5} \end{pmatrix}$$ Ecuación 2.19

$$\$_{i,6} = \begin{pmatrix} S_{i,6} \\ b_i \times S_{i,6} \end{pmatrix}$$ Ecuación 2.20

Donde:

$$S_{i,6} = \frac{1}{\|b_i - a_i\|}(b_i - a_i)$$

$$S_{i,5} = S_{i,4} \times S_{i,6}$$

Dado que a cada brazo se la analizó como una cadena cinemática abierta, se describir el movimiento de la plataforma móvil sumando los tornillos de cada una de las articulaciones aplicando la expresión de la ecuación 1.38.

$$V = \begin{bmatrix} \omega \\ v \end{bmatrix} = \$_{UPE} = \dot{\theta}_1\$_{i,1} + \dot{\theta}_2\$_{i,2} + \dot{L}_3\$_{i,3} + \dot{\theta}_4\$_{i,4} + \dot{\theta}_5\$_{i,5} + \dot{\theta}_6\$_{i,6}$$

Ecuación 2.21

Para i=1,2,3

$$\$_{RPU} = \dot{\theta}_1\$_{4,1} + \dot{L}_2\$_{4,2} + \dot{\theta}_3\$_{4,3} + \dot{\theta}_4\$_{4,4}$$ Ecuación 2.22

Como los ejes de los tornillos de cada una de las articulaciones que no están actuadas se intersecan en los puntos a_i y b_i el único tornillo reciproco a todas ellos es:

$$\$_{ri,3} = \begin{pmatrix} S_{i,3} \\ b_i \times S_{i,3} \end{pmatrix}$$ Ecuación 2.23

$$\$_{r4,2} = \begin{pmatrix} S_{4,2} \\ b_i \times S_{4,2} \end{pmatrix}$$ Ecuación 2.24

Tomando el producto de la ecuación (2.21) y (2.22) con la ecuación (2.23) y (2.24) se tiene:

$$\$_{r3,i}^{\boldsymbol{T}}\$_{\boldsymbol{UPE}} = \dot{L}_{\boldsymbol{i}}$$ Ecuación 2.25

$$\$_{r4,2}^{\boldsymbol{T}}\$_{RPU} = \dot{L}_{\boldsymbol{4}}$$ Ecuación 2.26

Agrupando en forma matricial la expresión anterior, el estado de velocidad puede ser calculado de la expresión:

$$J_x\dot{X} = J_q\dot{q}$$ Ecuación 2.27

$$J_x\dot{X} = J_q\dot{q}$$

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

$$J = J_x * J_q^{-1}$$

Donde:

$J = [\$_{1,3}\ \$_{2,3}\ \$_{3,3}\ \$_{4,2}\ ]$; y

$\dot{q} = [\dot{L}_1\ \ \dot{L}_2\ \ \dot{L}_3\ \ \dot{L}_4]^T$.

Obviamente, J debe ser invertible, de otro modo el manipulador se encontraría en una singularidad.

De este modo puede ser calculado el estado de velocidad $\dot{X} = [\ \dot{X}_m \quad \dot{Z}_m \quad \dot{\phi} \quad \dot{\theta}]^T$ sin necesidad de conocer las razones de velocidades pasivas, lo que implica una gran ventaja y la matriz jacobiana directa es la siguiente:

$$J_x = \begin{bmatrix} S^T{}_{1,3} & \left(b_1 \times S_{1,3}\right)^T \\ S^T{}_{2,3} & \left(b_2 \times S_{2,3}\right)^T \\ S^T{}_{3,3} & \left(b_3 \times S_{3,3}\right)^T \\ S^T{}_{4,2} & \left(O_m \times S_{4,2}\right)^T \end{bmatrix}_{4\times 6}$$

Ecuación 2.28

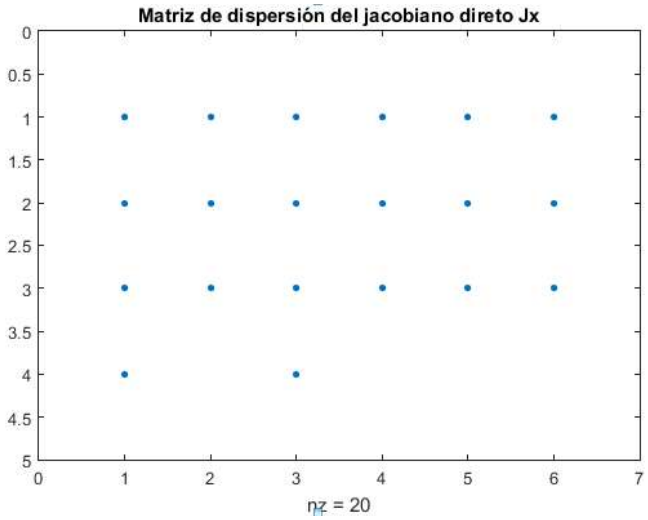


Figura 2.6. Estructura de la matriz jacobiana directa.
(Fuente: Autor)

Y la Jacobiana inversa es:

$$J_q = \begin{bmatrix} (S^T{}_{1,3})L_1 & 0 & 0 & 0 \\ 0 & (S^T{}_{2,3})L_2 & 0 & 0 \\ 0 & 0 & (S^T{}_{3,3})L_3 & 0 \\ 0 & 0 & 0 & (S^T{}_{4,2})L_4 \end{bmatrix}_{4\times 4}$$

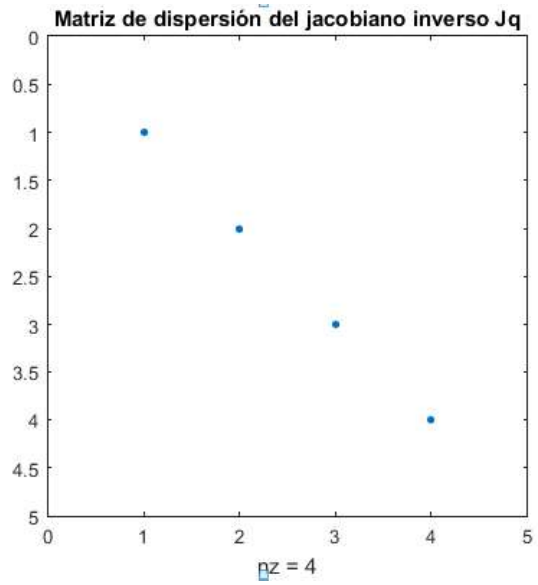


Figura 2.7. Estructura de la matriz jacobiana inversa.
(Fuente: Autor)

2.2. Homogenización de la matriz jacobiana

El manipulador 3UPE-RPU de 4GDL tiene 2 traslaciones (X_m, Z_m) y 2 rotaciones ϕ alrededor de ***y***, θ alrededor de ***z*** y presenta una matriz jacobiana directa de dimensión 4x6, por lo que se la homogeniza.

Donde:

El vector velocidad $v_e$es expresada como sigue:

$$x_1 = X_m, \qquad x_2 = Z_m\ , \qquad x_3 = \phi, \qquad x_4 = \theta$$

$$v_e = \begin{bmatrix} \dot{X_m} & \dot{Z_m} & \dot{\phi} & \dot{\theta} \end{bmatrix}^T \qquad \text{Ecuación 2.29}$$

La matriz jacobiana traslacional $\boldsymbol{J_v}$ se obtiene de la ecuación 1.48, entonces:

$$\boldsymbol{J_v} = [J_{v1} \quad J_{v2} \quad J_{v3} \quad J_{v4}\ ] \qquad \text{Ecuación 2.30}$$

Donde:

$$\boldsymbol{J_v} = \begin{bmatrix} \frac{\partial X_m}{\partial X_m} & \frac{\partial X_m}{\partial Z_m} & \frac{\partial X_m}{\partial \phi} & \frac{\partial X_m}{\partial \theta} \\ \frac{\partial Y_m}{\partial X_m} & \frac{\partial Y_m}{\partial Z_m} & \frac{\partial Y_m}{\partial \phi} & \frac{\partial Y_m}{\partial \theta} \\ \frac{\partial Z_m}{\partial X_m} & \frac{\partial Z_m}{\partial Z_m} & \frac{\partial Z_m}{\partial \phi} & \frac{\partial Z_m}{\partial \phi} \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix}$$

La plataforma móvil rota en orden YZ, la matriz jacobiana de rotacional $\boldsymbol{J_\omega}$ (3x4) se tiene de la expresión (1.45)

$$\boldsymbol{J}_\omega = [J_{\omega 1} \quad J_{\omega 2} \quad J_{\omega 3} \quad J_{\omega 4}\,] \qquad \text{Ecuación 2.31}$$

$$\boldsymbol{J}_\omega = R_{yz} * \begin{bmatrix} \frac{\partial \delta}{\partial X_m} & \frac{\partial \delta}{\partial Z_m} & \frac{\partial \delta}{\partial \varphi} & \frac{\partial \delta}{\partial \theta} \\ \frac{\partial \phi}{\partial X_m} & \frac{\partial \phi}{\partial Z_m} & \frac{\partial \phi}{\partial \phi} & \frac{\partial \phi}{\partial \theta} \\ \frac{\partial \theta}{\partial X_m} & \frac{\partial \theta}{\partial Z_m} & \frac{\partial \theta}{\partial \phi} & \frac{\partial \theta}{\partial \theta} \end{bmatrix} = R_{yz} * \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$J_\omega = \begin{bmatrix} 0 & 0 & -\cos\phi \sin\theta & \sin\phi \\ 0 & 0 & \cos\theta & 0 \\ 0 & 0 & \sin\theta \sin\phi & \cos\phi \end{bmatrix}$$

$$\boldsymbol{J}_e = \begin{bmatrix} \boldsymbol{J}_v \\ J_\omega \end{bmatrix}, \qquad J_x = \begin{bmatrix} {S^T}_{1,3} & (b_1 \times S_{1,3})^T \\ {S^T}_{2,3} & (b_2 \times S_{2,3})^T \\ {S^T}_{3,3} & (b_3 \times S_{3,3})^T \\ {S^T}_{4,2} & (b_4 \times S_{4,2})^T \end{bmatrix}_{4x6}$$

$$\boldsymbol{J} = \boldsymbol{J_x J_e}$$

Aquí $\boldsymbol{J}$ es una matriz general de 4x4, y J^{-1} existe y en base a sus componentes se analiza sus singularidades.

El código de este algoritmo se observa en el anexo **A.2** y se obtiene los elementos de la matriz general.

$$\boldsymbol{J} = J_x J_e = (J_r)_{4\times 6} \begin{bmatrix} J_{v1} & J_{v2} & J_{v3} & J_{v4} \\ J_{\omega 1} & J_{\omega 2} & J_{\omega 3} & J_{\omega 4} \end{bmatrix}_{4\times 4} \qquad \text{Ecuación 2.32}$$

$$\boldsymbol{J} = \begin{bmatrix} J_{11} & J_{12} & J_{13} & J_{14} \\ J_{21} & J_{22} & J_{23} & J_{24} \\ J_{31} & J_{32} & J_{33} & J_{34} \\ J_{41} & J_{42} & J_{43} & J_{44} \end{bmatrix}_{4\times 4} \quad \text{Donde:}$$

$$J_{11} = \frac{R + X_m - R_m \cos(\theta)\cos(\varphi)}{\sqrt{{R_m}^2 \cos\varphi^2 \sin\theta^2 + (Z_m + R_m \sin\varphi)^2 + (R + X_m - R_m \cos\theta \cos\varphi)^2}}$$

$$J_{12} = \frac{Z_m + R_m sin(\varphi)}{\sqrt{{R_m}^2 \sin\theta^2 \cos\varphi^2 + (Z_m + R_m \sin\varphi)^2 + (R + X_m - R_m \cos\theta \cos\varphi)^2}}$$

$$J_{13} = \frac{R_m \cos\theta \, (Z_m \cos\varphi \cos\theta + R \sin\varphi + X_m \sin\varphi)}{\sqrt{{R_m}^2 \sin\theta^2 \cos\varphi^2 + (Z_m + R_m \sin\varphi)^2 + (R + X_m - R_m \cos\theta \cos\varphi)^2}} + \frac{R_m Z_m \sin\theta^2 \cos\varphi}{\sqrt{{R_m}^2 \sin\theta^2 \cos\varphi^2 + (Z_m + R_m \sin\varphi)^2 + (R + X_m - R_m \cos\theta \cos\varphi)^2}}$$

$$J_{14} = \frac{R_m \cos\varphi \sin\theta \, (R + X_m)}{\sqrt{{R_m}^2 \sin\theta^2 \cos\varphi^2 + (Z_m + R_m \sin\varphi)^2 + (R + X_m - R_m \cos\theta \cos\varphi)^2}} + \frac{R_m \cos\varphi^2 \sin\theta \, (R + X_m)}{\sqrt{{R_m}^2 \sin\theta^2 \cos\varphi^2 + (Z_m + R_m \sin\varphi)^2 + (R + X_m - R_m \cos\theta \cos\varphi)^2}} - \frac{R_m Z_m \cos\varphi \sin\theta \cos\theta \sin\varphi}{\sqrt{{R_m}^2 \sin\theta^2 \cos\varphi^2 + (Z_m + R_m \sin\varphi)^2 + (R + X_m - R_m \cos\theta \cos\varphi)^2}}$$

$$J_{21} = \frac{a}{\sqrt{a^2 + b^2 + c^2}}$$

$$J_{22} = \frac{b}{\sqrt{a^2 + b^2 + c^2}}$$

Donde:

$a = X_m - R\cos\beta_1 + R_m \cos\alpha_1 \cos\theta \cos\varphi - R_m \sin\alpha_1 \sin\theta$

$b = Z_m - R_m \cos\alpha_1 \sin\varphi$

$c = R_m \sin(\alpha_1)\cos(\theta) - R\sin\beta_1 + R_m\cos(\alpha_1)\cos(\varphi)\sin\theta$

$$J_{23} = -\frac{d * e}{\sqrt{h^2 + i^2 + j^2}} - \frac{f * g}{\sqrt{h^2 + i^2 + j^2}}$$

Donde:

$d = R_m \cos\theta$

$e = X_m \cos(\alpha_1)\sin(\varphi) - Z_m \sin(\alpha_1)\sin(\theta) - R\cos(\alpha_1)\cos(\beta_1)\sin(\varphi) + Z_m\cos(\alpha_1)\cos(\varphi)\cos(\theta)$

$f = R_m \sin\theta$

$g = Z_m \sin(\alpha_1)\cos(\theta) - R\cos(\alpha_1)\sin(B_1)\sin(\varphi) + Z_m\cos(\alpha_1)\cos(\varphi)\sin(\theta)$

$h = X_m - \mathrm{R}\cos(\beta_1) - R_m \sin(\alpha_1)\sin(\theta) + R_m \cos(\alpha_1)\cos(\varphi)\cos(\theta)$

$i = Z_m - R_m \cos(\alpha_1)\sin(\varphi)$

$j = R_m \sin(\alpha_1)\cos(\theta) - R sin(B_1) + R_m \cos(\alpha_1)\cos(\varphi) sin(\theta)$

$$J_{24} = \frac{k * g}{\sqrt{h^2 + i^2 + j^2}} - \frac{l * m}{\sqrt{h^2 + i^2 + j^2}} - \frac{n * o}{\sqrt{h^2 + i^2 + j^2}}$$

Donde:

$k = R_m \cos(\theta)\sin(\varphi)$

$l = R_m sin(\theta) sin(\varphi)$

$m = X_m \cos(\alpha_1)\sin(\varphi) - Z_m \sin(\alpha_1)\sin(\theta) - R cos(\alpha_1)\cos(\beta_1)\sin(\varphi) + Z_m \cos(\alpha_1)\cos(\varphi)\cos(\theta)$

$n = R_m \cos\varphi$

$o = si\, n(\alpha_1)\cos(\theta) - R\cos(\beta_1)\sin(\alpha_1)\cos(\theta) + X_m \cos(\alpha_{1)}\cos(\varphi)\sin(\theta) - R\sin(\alpha_1) sin(B_1)\sin(\theta) - R\cos(\alpha_1)\cos(\beta_1)\cos(\varphi)\sin(\theta) + R\cos(\alpha_{1)}\cos(\varphi) sin(B_1)\cos(\theta)$

$$J_{31} = \frac{p}{\sqrt{p^2 + q^2 + r^2}}$$

Donde:

$p = X_m - \mathrm{R}\cos(\beta_2) - R_m \sin(\alpha_2)\sin(\theta) + R_m \cos(\alpha_2)\cos(\varphi)\cos(\theta)$

$\mathrm{q} = Z_m - R_m \cos(\alpha_2)\sin(\varphi)$

$r = R_m \sin(\alpha_2)\cos(\theta) - R sin(B_2) + R_m \cos(\alpha_2)\cos(\varphi) sin(\theta)$

$$J_{32} = \frac{s}{\sqrt{p^2 + s^2 + r^2}}$$

$s = Z_m - R_m \cos(\alpha_2)\sin(\varphi)$

$$J_{33} = -\frac{d * t}{\sqrt{p^2 + q^2 + r^2}} - \frac{f * u}{\sqrt{p^2 + q^2 + r^2}}$$

Donde:

$t = X_m \cos(\alpha_2)\sin(\varphi) - Z_m \sin(\alpha_2)\sin(\theta) - R cos(\alpha_2)\cos(\beta_2)\sin(\varphi) + Z_m \cos(\alpha_2)\cos(\varphi)\cos(\theta)$

$u = Z_m \sin(\alpha_2)\cos(\theta) - R\cos(\alpha_2)\sin(B_2)\sin(\varphi) + Z_m \cos(\alpha_2)\cos(\varphi)\sin(\theta)$

$$J_{34} = \frac{k * u}{\sqrt{p^2 + q^2 + r^2}} - \frac{l * t}{\sqrt{p^2 + q^2 + r^2}} - \frac{n * v}{\sqrt{p^2 + q^2 + r^2}}$$

$v = si\,n(\alpha_2)\cos(\theta) - R\cos(\beta_2)\sin(\alpha_2)\cos(\theta) + X_m\cos(\alpha_{2)}\cos(\varphi)\sin(\theta)$ $- R\sin(\alpha_2)sin(B_2)\sin(\theta) - R\cos(\alpha_2)\cos(\beta_2)\cos(\varphi)\sin(\theta)$ $+ R\cos(\alpha_{2)}\cos(\varphi)sin(B_2)\cos(\theta)$
$J_{41} = \dfrac{X_m}{\sqrt{{X_m}^2 + {Z_m}^2}}$
$J_{42} = \dfrac{Z_m}{\sqrt{{X_m}^2 + {Z_m}^2}}$
$J_{43} = 0$
$J_{44} = 0$

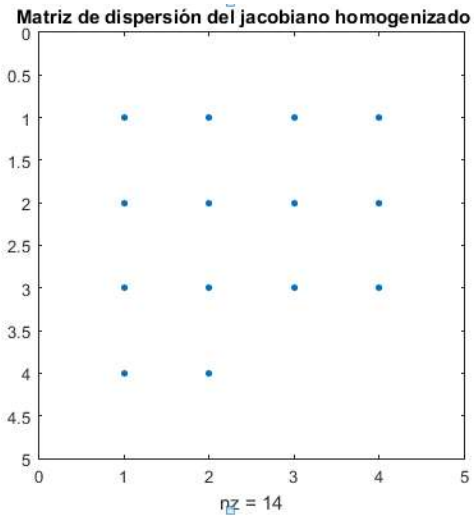


Figura 2.8 Estructura de la matriz jacobiana homogenizada.
(Fuente: Autor)

2.2.1. Singularidades

La existencia de singularidades en los mecanismos de cadena cinemática cerrada, tienden a perder rigidez mientras gana grados de libertad, y representa el punto en el que la estructura puede colapsar.

Las singularidades se pueden presentar cuando J_x o J_q son singulares o cuando ambas lo son, es decir cuando su determinante es igual a cero.

Por las dimensiones de la matriz jacobiana se normaliza dicha matriz para obtener su determinante y facilitar la localización de puntos singulares en el espacio de trabajo.

- Cuando $|J| = 0$, las singularidades se presentan en los límites del manipulador.
- Cuando $|J| \to \infty$, singularidades locales en el manipulador paralelo.
- Cuando $|J| \to 0/0$, singularidades en la estructura del manipulador paralelo.

2.2.2. Singularidades: casos especiales

Configuracion singular de Fichter

Fichter analiza el movimiento de mecanismos paralelos cuando rotan $\pm 90°$ alrededor del eje vertical.

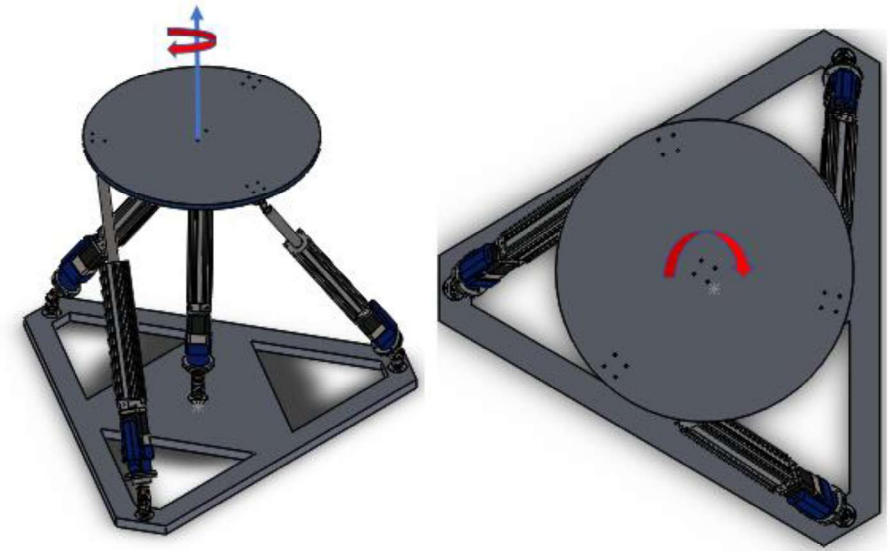


Figura 2.9. Configuracion singular de Fichter: El movimiento de la plataforma rotado $\pm 90°$ alrededor del eje vertical.
(Fuente:Autor)

Configuracion singular de Hunt

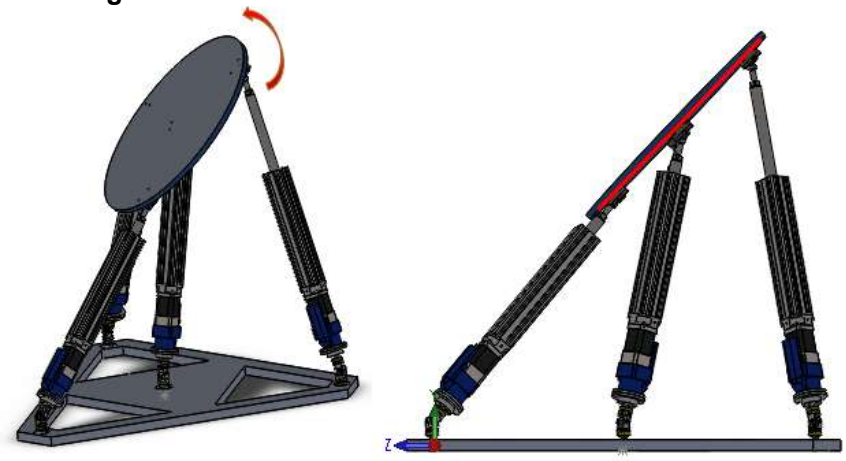


Figura 2.10. Configuracion singular de Hunt: dos miembros y la plataforma móvil son coplanares.
(Fuente:Autor)

2.3. Análisis del espacio de trabajo

Para la obtención del espacio de trabajo se utiliza un algoritmo de discretización discutida en la base teórica, a continuación, se presentan todas las restricciones e implementación del algoritmo.

2.3.1. Restricciones

Las restricciones utilizadas en el cálculo del espacio de trabajo son las siguientes:

Límites físicos de los actuadores prismáticos

Los actuadores prismáticos tienen un recorrido limitado; Por lo tanto, el estado de cada actuador debe encontrarse dentro de los límites físicos del actuador. Es decir que cada actuador debe satisfacer:

$$\begin{matrix} \|L_i\| = L_i \\ L_{min} < L_i < L_{max} \qquad para\ i = 1,2,3,4 \end{matrix} \qquad \text{Ecuación 2.33}$$

Límites físicos de las articulaciones esféricas

Una articulación esférica se puede considerar como una junta universal que permite una rotación adicional alrededor de un eje normal a los ejes de junta universales.

$$\theta_{pi} = \cos^{-1}\frac{L_i . R_{ot} . n_{pi}}{\|L_i\|} \leq \theta_{p\,max} \qquad \text{Ecuación 2.34}$$

Límites físicos de las juntas universales

$$\theta_{bi} = \cos^{-1}\frac{L_i . n_{bi}}{\|L_i\|} \leq \theta_{b\,max} \qquad \text{Ecuación 2.35}$$

Aquí θ_{pi} y θ_{bi} son los ángulos de rotación de las juntas como se muestra en la figura 2.9. L_{min} y L_{max} denota el minimo y máximo longitud de la extremidad, respectivamente. $\theta_{p\,max}$ es el máximo ángulo montada en la plataforma móvil, mientras $\theta_{b\,max}$ es el máximo ángulo montado en la plataforma fija, R_{ot} representa a la matriz de rotación y n_{bi}, n_{pi} son vectores unitarios de las juntas en la base y la plataforma móvil, respectivamente.

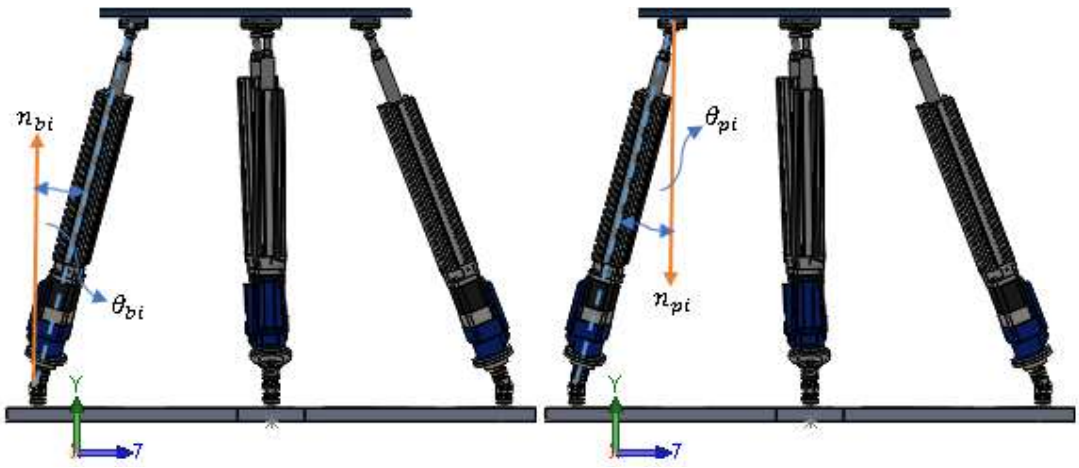


Figura 2.11. Angulo de rotación de las juntas y vector unitario de sus direcciones.
(Fuente: Autor)

Colisiones entre extremidades

La tercera condición que se necesita para que una configuración pertenezca al espacio de trabajo es que las patas no choquen entre ellas es decir colisiones entre extremidades.

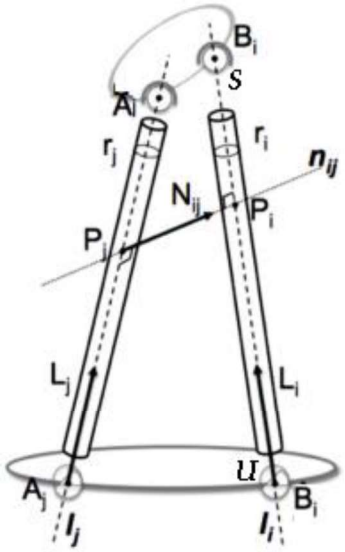


Figura 2.12. Interferencia entre extremidades
(Fuente: Autor)

2.3.2. Técnica de discretización para el cálculo del espacio de trabajo a orientación constante

Los pasos principales para la implementación son los siguientes:

1. Definir las restricciones y la función llamada “posicion_prueba”, la cual devuelve el resultado de la evaluación, verdadero o falso.
2. Implementar un algoritmo de búsqueda para determinar el centro aproximado del espacio de trabajo.

3. Implementar un criterio de error para determinar el límite del espacio de trabajo.
4. Determinar el volumen del espacio de trabajo.
5. Visualización gráfica del espacio de trabajo.

2.3.3. Función "Posicion_prueba"

Esta función primero calcula la cinemática inversa del robot y luego verifica si las restricciones establecidas son satisfechas. La función está definida como una función Booleana, que da como respuesta verdadero o falso si todas las restricciones son satisfechas.

$$Respuesta = Posicion_prueba(X_m, Y_m, Z_m, ; \psi; \phi; \theta)$$

```
function [L S Resp] =Posicion_prueba(x,y,z,psi_x,theta_y,phi_z)

%Parámetros geómetricos de la plataforma fija
R=400;   be1=40*(pi/180);   be2=50*(pi/180);
%Parámetros geómetricos de la plataforma móvil
Rm=200;   al1=30*(pi/180);   al2=40*(pi/180);
%Movimientos Ym=0 y angX=0
y=0;psi_x=0;
%Vértices de la plataforma fija
 Ao=[-R;0;0];    Bo=[R*cos(be1);R*sin(be1);0];  Co=[R*cos(-be2);R*sin(-be2);0];   Do=[0;0;0];
%Vértices de la plataforma móvil
 A1=[-Rm; 0; 0];  B1=[Rm*cos(al1);Rm*sin(al1);0]; C1=[Rm*cos(-al2);Rm*sin(-al2);0]; D1=[x;0;z];
```

2.3.4. Algoritmo de búsqueda del espacio de trabajo

El algoritmo del espacio de trabajo realiza una búsqueda para determinar las posiciones que puede alcanzar el robot con una orientación constante es decir O ($\psi; \phi; \theta$). El primer paso, suponer la ubicación de un centro C aproximado del espacio de trabajo que es conocida para una orientación dada de la plataforma, luego este algoritmo conduce a una búsqueda para determinar el límite del espacio de trabajo en un sistema de coordenadas esféricas (ρ, φ, θ) con centro en C, y para buscar todo el espacio de trabajo se logra mediante la discretización del rango de los ángulos φ ∈ [0,180], θ ∈ [0, 360]. Para cada par de ellos, el radio ρ se incrementa hasta detectar una restricción. Cuando se encuentra que un valor para ρ es ubicado fuera del espacio de trabajo, que constituye una primera aproximación para la ubicación del límite del espacio de trabajo a lo largo del rayo esférico, un segundo procedimiento, denominado algoritmo de límite de espacio de trabajo, se utiliza para refinar el resultado.

Como se ilustra en la figura 1.15 a, el vector de búsqueda se expresa en coordenadas esféricas, $V = \{\rho, \Theta, \Phi\}$. El vector V se considera que esta cerca del borde del espacio de trabajo y tiene que satisfacer la siguiente condicion:

$$Respuesta = posicion_prueba(\rho, \Theta, \Phi, \text{Orientación}) = True$$

$$Respuesta = posicion_prueba(\rho, \Theta, \Phi, \text{Orientación}) = false$$

El proceso de búsqueda del espacio completo se realiza por la rotación del vector discretizando $\Delta\Theta, \Delta\Phi$.

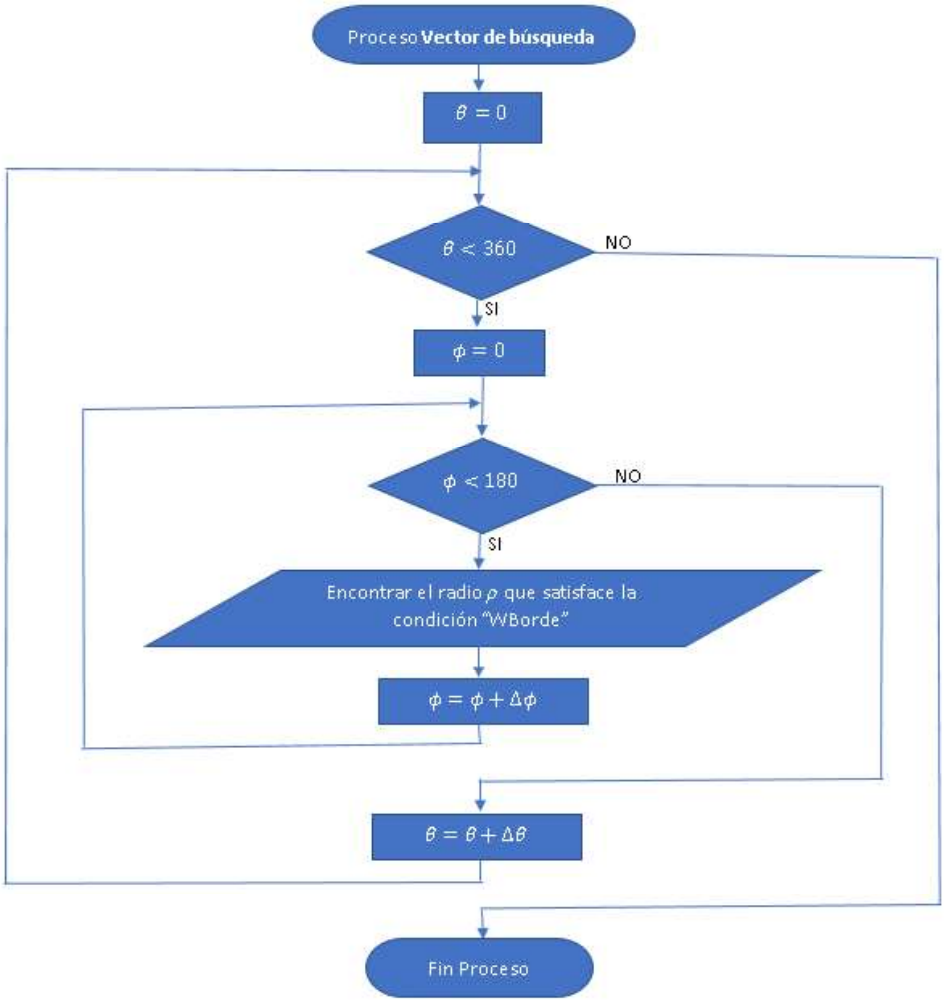


Figura 2.13. Diagrama de flujo de vector de búsqueda.
(Fuente: Autor)

2.3.5. Algoritmo de límite del espacio de trabajo

Se basa en el uso de una técnica de búsqueda a la mitad del intervalo para garantizar que ρ está dentro de $\pm\varepsilon$, del límite del espacio de trabajo, donde ε es una tolerancia de error dada.

a) Implementar un criterio de error para determinar el límite del espacio de trabajo. Asumir un error aceptable, ε y el algoritmo busca el límite del espacio de trabajo dentro de una tolerancia ε / 2.
b) Defina un valor inicial para ρ para iniciar el algoritmo. El algoritmo calcula el límite del área de trabajo obteniendo valor de radio ρ con la tolerancia ε / 2, para cualquier $\varphi \in [0,180]$, $\theta \in [0, 360]$.
c) Al principio de cada paso, ρ se divide por 2. Si la función "posicion_prueba" para el nuevo valor de ρ devuelve true, este ρ se agrega al anterior, de lo contrario se resta.
d) El algoritmo continúa reduciendo el incremento (Δ) a ρ / 2, ρ / 4, ρ / 8 ... hasta que es menor o igual a ε / 2 y la función Test-Pose devuelve true.
e) Determinar el volumen del espacio de trabajo.
f) Visualización gráfica del espacio de trabajo.

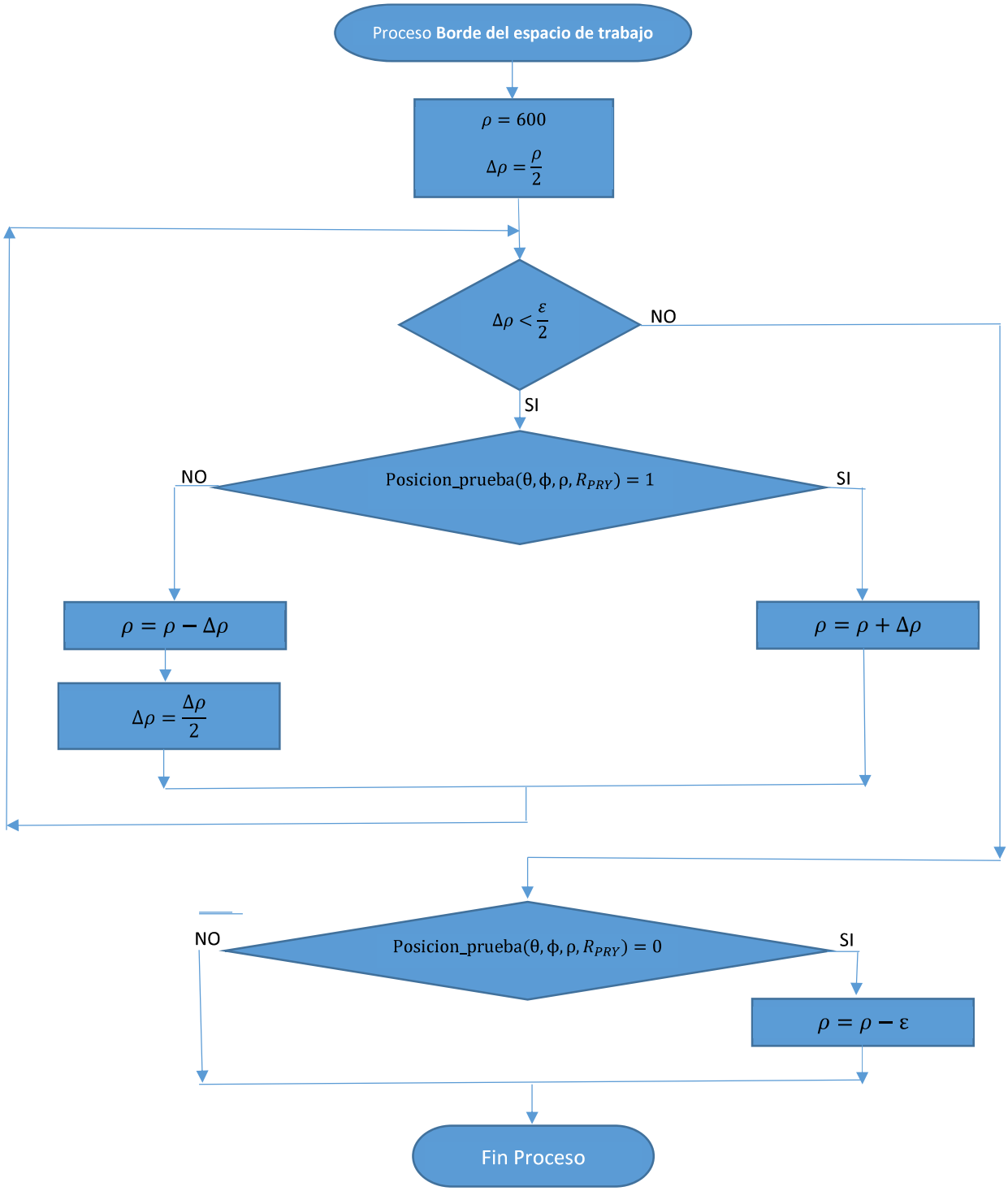


Figura 2.14. diagrama de flujo algoritmo de búsqueda frontera.
(Fuente: Autor)

2.3.6. Evaluación del volumen

El espacio de trabajo obtenido eventualmente se define por una gran cantidad de puntos fronterizos del origen de búsqueda C. Para evaluar su volumen, el método de cálculo se basa principalmente en acumulación. Definir un sector por cuatro puntos de datos de área de trabajo adyacentes V_1, V_2, V_3, V_4 y la búsqueda del origen, mostrada en la figura 2.15

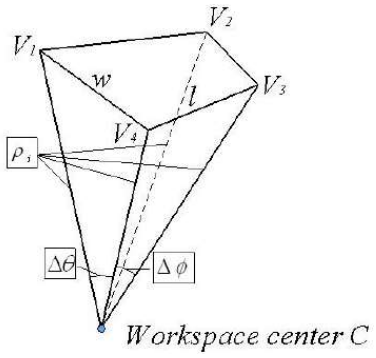


Figura 2.15. Calculo de volumen del sector del espacio de trabajo.
(Fuente:(Conti et al., 1998))

Luego, sumando el volumen de los todos sectores, se puede determinar el volumen total del espacio de trabajo.

Este apartado no aplica en el cálculo de volumen del espacio de trabajo, el robot 3UPE-RPU no trabaja en las 3 coordenadas, entonces solo se presenta el espacio de trabajo en el plano XZ.

3. RESULTADOS Y DISCUSIÓN

3.1. Representación de la cinemática inversa del robot 3UPE-RPU

Para representar al robot paralelo se usa los archivos `CinInv_3UPE_RPU.m` y `Repres_3UPE_RPU.m` donde se calcula la cinemática inversa y se presenta su arquitectura. Los datos para la simulación se detallan a continuación en la tabla 3.1 y tabla 3.2, (Zamora, 2016)

Tabla 3.1. Parámetros geométricos del robot

Radio Plataforma móvil	Radio Base	Ángulo Pata 2 base β_1	Ángulo Pata 2 móvil α_1	Ángulo Pata 3 base β_2	Ángulo pata 3 móvil α_2
200mm	400mm	40°	30°	50°	40°

(Fuente: Autor)

Tabla 3.2. Valores iniciales para el cálculo cinemático

N°	Coordenada en X_m	Coordenada en Y_m	Coordenada en Z_m	Ángulo de rotación en X Ψ	Ángulo de rotación en Y ϕ	Ángulo de rotación en Z θ
1	134.2mm	0mm	400mm	0°	0°	0°
2	-134.2mm	0mm	400mm	0°	0°	0°
3	134.2mm	0mm	400mm	0°	20°	0°
4	134.2mm	0mm	400mm	0°	-20°	0°
5	90mm	0mm	400mm	0°	0°	30°
6	90mm	0mm	400mm	0°	0°	-30°

(Fuente: Autor)

En el espacio cartesiano, la condición X esta determinado por las variables de posición (X_m Y_m Z_m) y variables de orientación (Ψ ϕ θ), la cual se expresa con la siguiente ecuación:

$$X = [X_m \;\; Y_m \;\; Z_m \;\; \Psi \;\; \phi \;\; \theta\,] \qquad \text{Ecuación 3.1}$$

Considerando que los movimientos están restringidos para Y_m y Ψ.

La ecuación 3.1 se ingresa en el archivo del cálculo cinemático con los siguientes resultados, ver figura 3.1

El código de programación para la resolución cinemática es:

```
clc;clear all
Rm=200;
R=400;
alfa=pi/6;
```

```
beta=5*pi/18;

X=[120 0 400 0 0 -pi/6];

%Función Calcular Cinemática Inversa
[Joints L vL S] = CinInv_3UPE_RPU(Xp,Rm,R,alfa,beta)

%Función presenta al robot 3UPE-RPU
Repres_3UPE_RPU(L,Xp,Rm,R,alfa,beta)
```

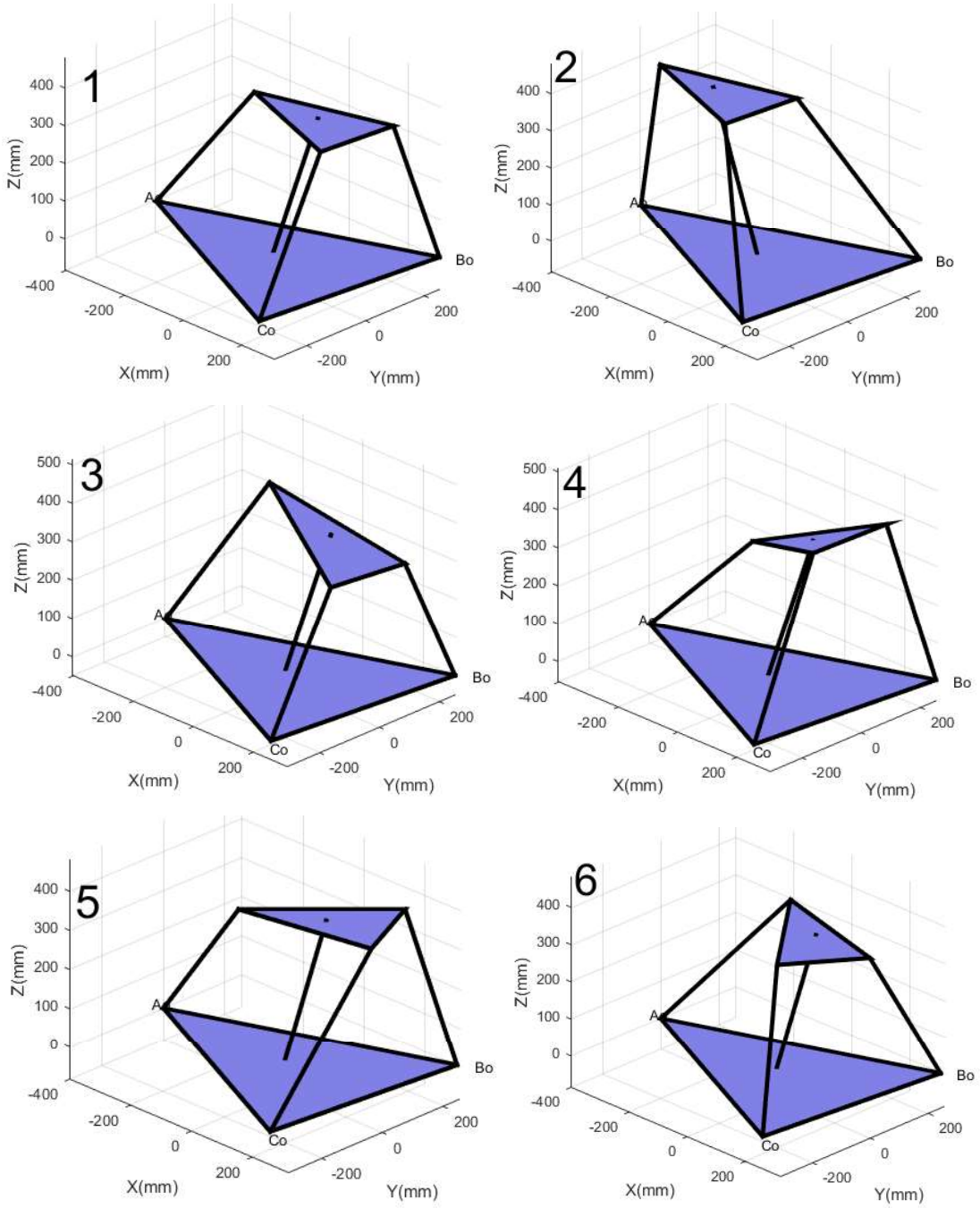


Figura 3.1. Representación cinemática inversa 3UPE-RPU.
(Fuente: Autor)

El propósito de modelar el sistema es determinar las longitudes de las extremidades y los ángulos de las juntas universales (UPE) y longitud, junta rotacional (RPU) requeridas para llegar a una cierta posición. A continuación, se detalla los valores de las juntas pasivas, longitud de los actuadores y sus vectores unitarios obtenidos de las posiciones de acuerdo con la tabla 3.2.

Ejemplo 3.1.1. `X = [134.2 0 400 0 0 0];`

- **Valor de las juntas**

```
Joints =
   Brazo A     Brazo B     Brazo C     Brazo D
  521.2386    429.7512    438.8075    427.2397
   -0.6960     -0.2455     -0.2840      0.3590
    0.0000     -0.2854      0.3188           0
```

- **Longitud de los actuadores**

```
L =

  521.2386  429.7512  429.7512  427.2397
```

- **Vector unitario a lo largo de cada extremidad**

```
S =
       L1          L2          L3          L4
   0.6412      0.0023      0.0023      0.3514
  -0.0000     -0.3656     -0.3656           0
   0.7674      0.9308      0.9308      0.9362
```

Ejemplo 3.1.2. `X=[-134.2 0 400 0 0 0];`

- **Valor de las juntas**

```
Joints =
   Brazo A     Brazo B     Brazo C     Brazo D
  405.3759    506.1568    498.3260    427.2397
   -0.1630     87.3118     99.9048     -0.3590
    0.0000     18.9515     56.4116           0
```

- **Longitud de los actuadores**

```
L =

  405.3759  506.1568  506.1568  427.2397
```

- **Vector unitario a lo largo de cada extremidad.**

```
S =
       L1          L2          L3          L4
   0.1623     -0.5283     -0.5283     -0.3514
   0.0000     -0.3104     -0.3104           0
   0.9867      0.7903      0.7903      0.9362
```

Ejemplo 3.1.3 `X=[134.2 0 400 0 pi/9 0];`

- **Valor de las juntas**

```
Joints =
   Brazo A     Brazo B     Brazo C     Brazo D
  582.4941   375.3561   391.0281   427.2397
   -0.6366    -0.3076    -0.3394     0.3590
    0.0000    -0.3094     0.3401          0
```

- **Longitud de los actuadores**

```
L =

  582.4941  375.3561  375.3561  427.2397
```

Vector unitario a lo largo de cada extremidad.

```
S =
       L1         L2         L3         L4
   0.5944    -0.0252    -0.0252     0.3514
   0.0000    -0.4186    -0.4186          0
   0.8041     0.9078     0.9078     0.9362
```

Ejemplo 3.1.4. `X=[134.2 0 400 0 -pi/9 0];`

- **Valor de las juntas**

```
Joints =
   Brazo A     Brazo B     Brazo C     Brazo D
  479.4298   485.4644   486.5632   427.2397
   -0.8070    -0.2315    -0.2649     0.3590
    0.0000    -0.2376     0.2714          0
```

- **Longitud de los actuadores**

```
L =

  479.4298  485.4644  485.4644  427.2397
```

Vector unitario a lo largo de cada extremidad.

```
S =
       L1         L2         L3         L4
   0.7222    -0.0195    -0.0195     0.3514
   0.0000    -0.3236    -0.3236          0
   0.6916     0.9460     0.9460     0.9362
```

Ejemplo 3.1.5. `X = [90 0 400 0 0 pi/6];`

- **Valor de las juntas**

```
Joints =
    Brazo A     Brazo B     Brazo C     Brazo D
  519.9606   424.9635   484.4639   412.4739
   -0.6698   -56.8923    -0.4413     0.2466
    0.1935   113.1222     0.4199          0
```

- **Longitud de los actuadores**

```
L =

  519.9606  424.9635  424.9635  412.4739
```

Vector unitario a lo largo de cada extremidad.

```
S =
       L1        L2        L3        L4
   0.6093   -0.2739   -0.2739    0.2441
  -0.1923   -0.1975   -0.1975         0
   0.7693    0.9413    0.9413    0.9698
```

Ejemplo 3.1.6. `X=[90 0 400 0 0 -pi/6];`

- **Valor de las juntas**

```
Joints =
   Brazo A     Brazo B     Brazo C     Brazo D
  519.9606   475.7917  -428.6971   412.4739
   -0.6698    -0.4184   188.1276     0.2466
   -0.1935    -0.4026    72.2579          0
```

- **Longitud de los actuadores**

```
L =

  519.9606  475.7917  475.7917  412.4739
```

Vector unitario a lo largo de cada extremidad.

```
S =
       L1        L2        L3        L4
   0.6093   -0.0345   -0.0345    0.2441
   0.1923   -0.5404   -0.5404         0
   0.7693    0.8407    0.8407    0.9698
```

3.2. Cálculo de la matriz jacobiana analítica

Para ello se usa la matriz jacobiana constituida por las derivadas parciales de las ecuaciones de restricción de la ecuación 1.23 que se generan manualmente utilizando el archivo `Calculo_Jacobiana_Analitica.m` que resuelve la jacobiana analítica independiente y dependiente. Ver código en el anexo A.1

Ejemplo 3.2.1. Obtener la jacobiana analítica para la posición:

$X_m = 45\ mm\ , Z_m = 647\ mm,\ \ \phi = 0\ rad,\ \theta = 0.$

Matriz jacobiana de variables dependientes

```
Jdep =

   -1.0000         0         0 -100.0000 -647.0000  -47.2420         0         0         0         0         0
         0         0  200.0000         0         0 -734.0815         0         0         0         0         0
         0   -1.0000         0 -173.2051  346.7949  -88.1373         0         0         0         0         0
   -1.0000         0   86.6025   86.6025         0         0  322.6024  563.7977         0         0         0
         0         0 -173.2051         0         0         0 -558.7638  338.0890         0         0         0
         0   -1.0000  -50.0000  150.0000         0         0  125.5546   55.9868         0         0         0
   -1.0000         0  -86.6025   86.6025         0         0         0         0  296.9993 -504.7495         0
         0         0 -173.2051         0         0         0         0         0  514.4179  366.1635         0
         0   -1.0000   50.0000  150.0000         0         0         0         0  178.3905 -215.5427         0
   -1.0000         0         0         0         0         0         0         0         0         0  647.0000
         0   -1.0000         0         0         0         0         0         0         0         0 -134.2287
```

Matriz jacobiana de variables independiente

```
Jind =

    0.4681         0         0         0
   -0.1350         0         0         0
    0.8733         0         0         0
         0   -0.1700         0         0
         0    0.1216         0         0
         0    0.9779         0         0
         0         0   -0.4306         0
         0         0   -0.0636         0
         0         0    0.9003         0
         0         0         0    0.2031
         0         0         0    0.9792
```

Ejemplo 3.2.2. Obtener la jacobiana analítica para la posición:

Para: $X_m = 120\ mm\ , Z_m = 747\ mm,\ \phi = 0\ rad,\ \theta = pi/6.$

Matriz jacobiana de variables dependientes

```
Jdep =

   -1.0000         0         0         0 -747.0000   -0.0000         0         0         0         0         0
         0         0  200.0000         0         0 -812.6555         0         0         0         0         0
         0   -1.0000         0 -200.0000  320.0000   -0.0000         0         0         0         0         0
   -1.0000         0  173.2051         0         0         0  373.5000  648.6127         0         0         0
         0         0 -100.0000         0         0         0 -646.9210  396.5818         0         0         0
         0   -1.0000         0  100.0000         0         0  140.0000  102.1446         0         0         0
   -1.0000         0 -173.2051         0         0         0         0         0  373.5000 -648.6127         0
         0         0 -100.0000         0         0         0         0         0  646.9210  396.5818         0
         0   -1.0000         0  100.0000         0         0         0         0  140.0000 -102.1446         0
   -1.0000         0         0         0         0         0         0         0         0         0  747.0000
         0   -1.0000         0         0         0         0         0         0         0         0 -134.2287
```

Matriz jacobiana de variables independientes

```
Jind =

    0.3938         0         0         0
   -0.0000         0         0         0
    0.9192         0         0         0
         0   -0.2086         0         0
         0    0.0903         0         0
         0    0.9738         0         0
         0         0   -0.2086         0
         0         0   -0.0903         0
         0         0    0.9738         0
         0         0         0    0.1769
         0         0         0    0.9842
```

3.3. Cálculo de la matriz jacobiana del robot

Para el cálculo del jacobiano geométrico del robot se ejecuta el siguiente archivo considerando los parámetros geométricos de la tabla 3.1, **`Calcul_Jacobiano_3UPE_RPU.M`**. Ver anexo A.2

Ejemplo 3.3.1. Calcular la matriz jacobiana del robot en la posición:

Para: $X_m = 10\ mm\ , Z_m = 600\ mm,\ \phi = 0,$ y $\theta = -\frac{\pi}{6}\ rad,$

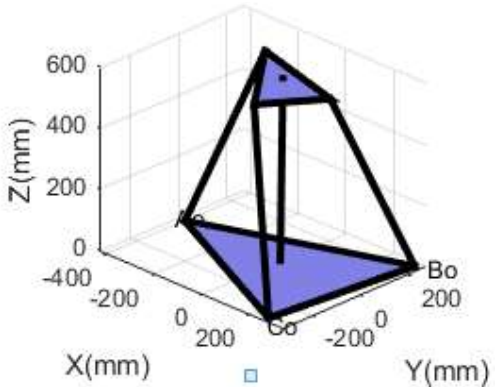


Figura 3.2. Movimiento en la posición $X_m = 10\ mm\ , Z_m = 600\ mm,$ $\phi = 0$ y $\theta = -\frac{\pi}{6}\ rad$
(Fuente: Autor)

Jacobiano directo

```
Jx =

    0.3628    0.1532    0.9192   91.9200  159.2101  -62.8120
   -0.0259   -0.3798    0.9247   92.4720 -160.1661  -63.1888
   -0.0259   -0.3798    0.9247 -184.9439   -0.0000   -5.1777
    0.0186         0    0.9998         0         0         0
```

Jacobiano inverso

```
Jq =

  652.7418         0         0         0
         0  659.8519         0         0
         0         0  659.8519         0
         0         0         0  598.0996
```

Jacobiano General

```
Jc =

    0.3628    0.9192  183.8399  -62.8120
   -0.1461    0.9093 -157.4948  -77.9306
   -0.1461    0.9093 -139.3123  -54.1151
    0.0186    0.9998         0         0
```

3.3.1. Análisis de singularidades

En base a la ecuación 2.32, se verifica si se presenta una deficiencia de rango de la matriz jacobiana J es decir su determinante igual a cero det(J)=0, si es así entonces el robot 3UPE/RPU se encuentra en una posición singular.

A continuación, el programa en MATLAB se construye utilizando el diagrama de flujo que se muestra en la figura 3.3, donde conociendo los parámetros geométricos y dando una posición se calcula la cinemática inversa obteniendo las longitudes de las extremidades con sus respectivos vectores unitarios para formar la matriz jacobiana y determinar las posiciones singulares del manipulador paralelo.

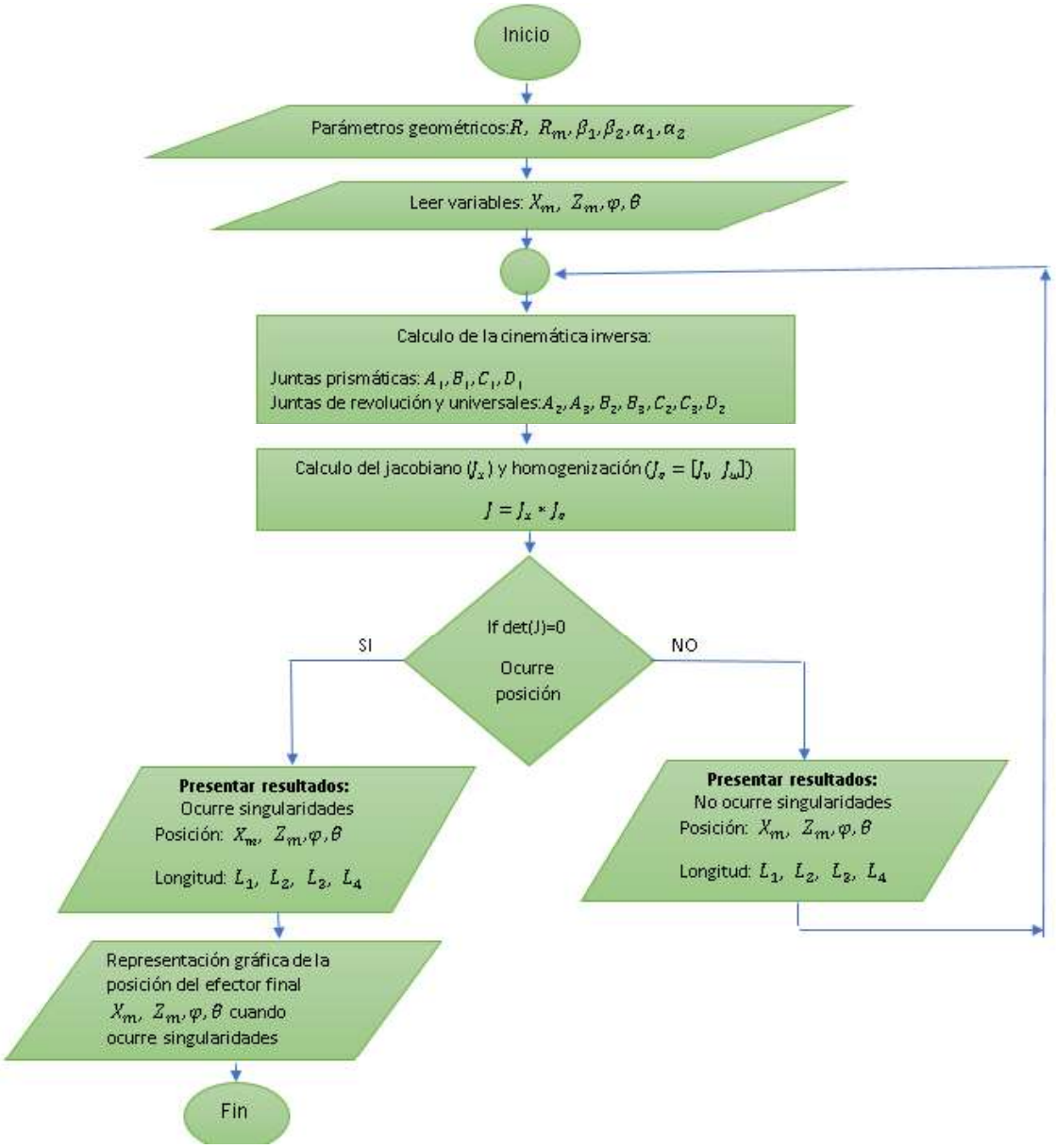


Figura 3.3. Diagrama de flujo en el cálculo de singularidades.

(Fuente: Autor)

El programa MATLAB se puede aplicar a cualquier posición de la plataforma móvil para verificar la singularidad dentro del espacio de trabajo del robot paralelo.

En la sección anterior se aborda la importancia y como obtener la matriz jacobiana utilizando la teoría de tornillos recíprocos, a continuación, el estudio a distintos mecanismos paralelos en base al determinante de esta matriz investigada por (Gosselin & Angeles, 1990) se establece tres tipos de singularidades.

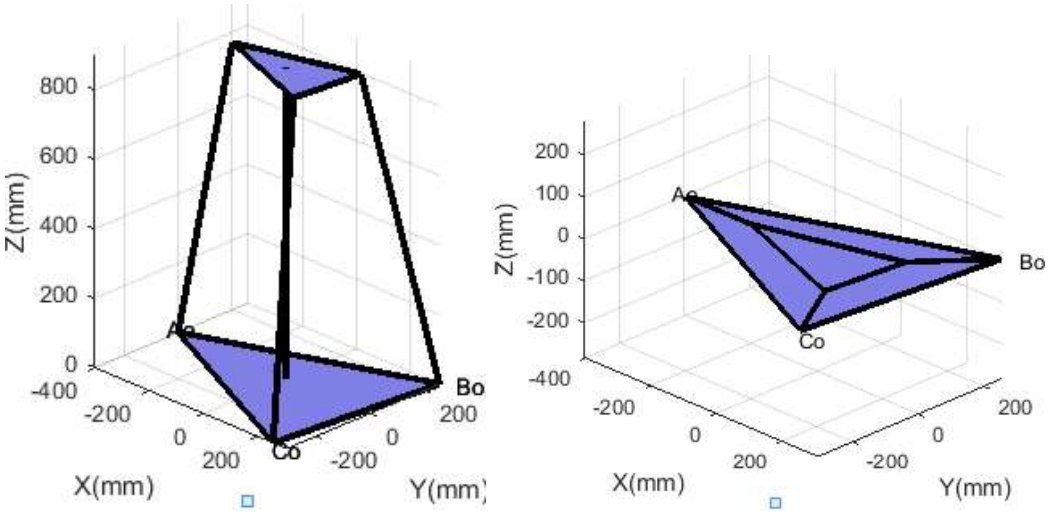
Singularidad de cinemática inversa (tipo I)

$$Det(J_q) = 0$$ Ecuación 3.2

Donde:

$$J_q = \begin{bmatrix} (S^T{}_{1,3})L_1 & 0 & 0 & 0 \\ 0 & (S^T{}_{2,3})L_2 & 0 & 0 \\ 0 & 0 & (S^T{}_{3,3})L_3 & 0 \\ 0 & 0 & 0 & (S^T{}_{4,2})L_4 \end{bmatrix}_{4\times4}$$

En este tipo de singularidad el robot pierde un grado de libertad en la dirección de la extremidad, se presenta en los límites del espacio de trabajo, es decir cuando la extremidad está totalmente extendida o retraída.



a) b)

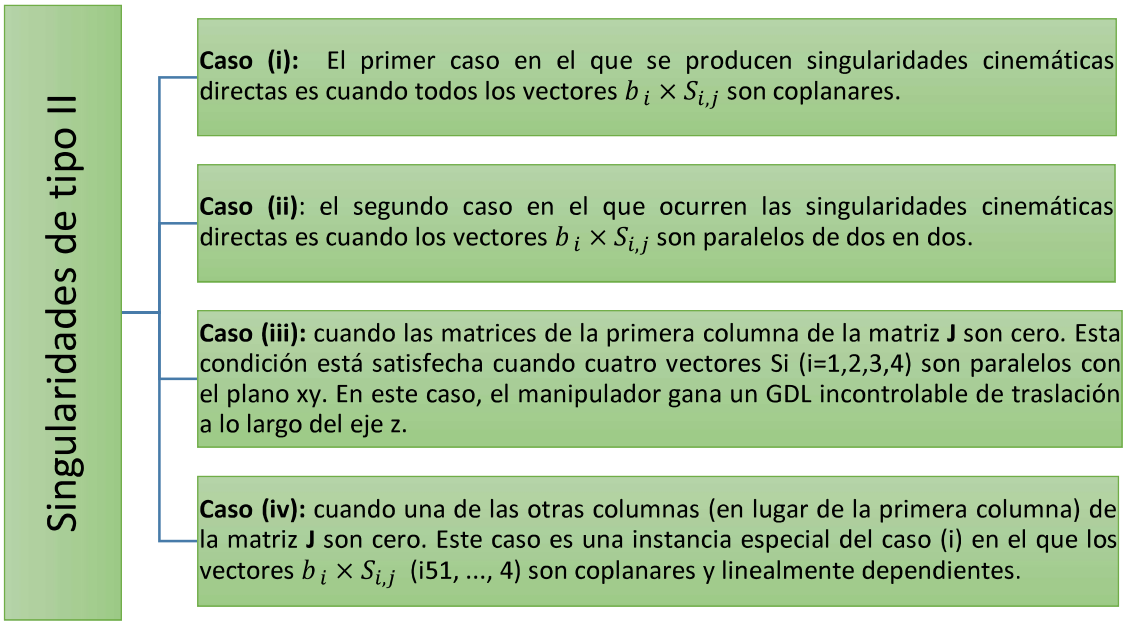
Figura 3.4. Configuración singular inversa: a) extremidades extendidas, b) extremidades retraídas. (Fuente: Autor)

Aquí el robot 3UPE- RPU no puede rotar alrededor del eje z, perdiendo uno de los 4 GDL. De acuerdo con las restricciones y parámetros geométricos impuestos en la tabla 3.1, se puede notar que J_q por su estructura da como resultado una matriz de identidad, lo que significa que el espacio de trabajo del mecanismo no presenta singularidades cinemáticas inversas.

Singularidad de cinemática directa (tipo II)

$$Det(J_x) = 0$$ Ecuación 3.3

Se pueden identificar algunos casos para las singularidades cinemáticas directas.



Singularidad combinada (tipo III)

Este tipo de singularidades se presentan cuando:

$$Det(J_x) = 0 \text{ y } Det(J_q) = 0. \qquad \text{Ecuación 3.4}$$

Para excluir las singularidades directas o inversas en el espacio de configuraciones, se necesita imponer

$$Det(J_x) \neq 0 \qquad \text{y/o} \qquad Det(J_q) \neq 0.$$

Según convenga. Por lo general se dice que son las más perjudiciales para el mecanismo. Es necesario excluir las singularidades directas debido a que estas comprometen el control y puede producir rotura del mecanismo. También se puede excluir las singularidades inversas, aunque no es deseable porque reduce el espacio de trabajo del robot.

A continuación, se presenta algunos ejemplos donde se calculan el jacobiano directo, jacobiano inverso y su homogenización obteniendo el determinante de la matriz y se halla las configuraciones singulares en el robot 3UPE- RPU.

Ejemplo 3.3.1. Calcular las matrices jacobinas de la siguiente posición:

```
X=[0 0 850 0 pi 0];
```

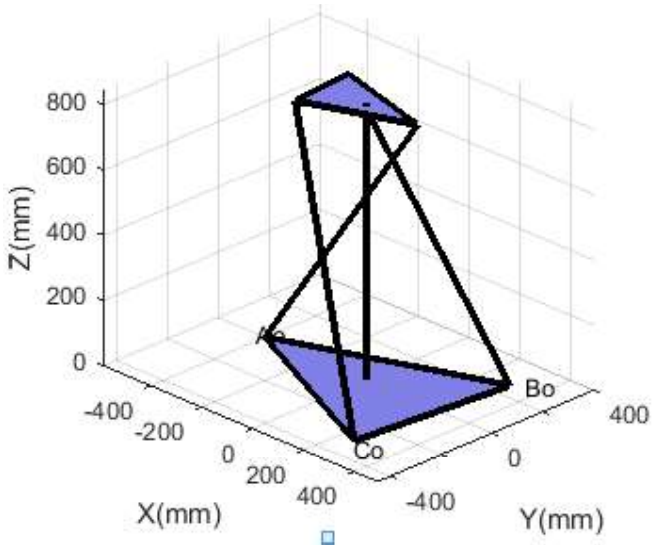


Figura 3.5. configuración singular del manipulador.
(Fuente: Autor)

- **Jacobiano directo**

```
Jx =

    0.5767    0.0000    0.8170   -0.0000 -163.3936    0.0000
   -0.4852   -0.1589    0.8598   85.9849  148.9302   76.0464
   -0.4852   -0.1589    0.8598 -110.5400  131.7365  -38.0232
         0         0    1.0000         0         0         0
```

- **Jacobiano inverso**

```
Jq =

   1.0e+03 *

    1.0404         0         0         0
         0    0.9885         0         0
         0         0    0.9885         0
         0         0         0    0.8500
```

- **Jacobiano General**

```
Jc =

    0.5767    0.8170 -163.3936   -0.0000
   -0.4852    0.8598  148.9302  -76.0464
   -0.4852    0.8598  131.7365   38.0232
         0    1.0000         0         0
```

- **Determinante**

```
detJC =

   -0.0011
```

- **Rango**

Rank (JC)=3

Ejemplo 3.3.2. Calcular las matrices jacobinas de la siguiente posición:

```
X=[0 0 414.3514  0 pi 0];
```

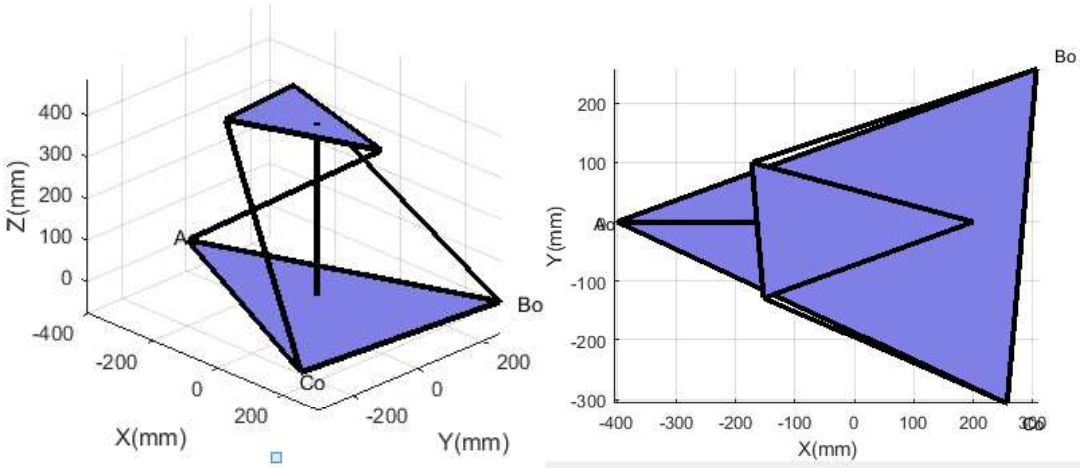


Figura 3.6. Configuración singular del robot.
(Fuente: Autor)

- **Jacobiano directo**

```
Jx =

    0.8229    0.0000    0.5683   -0.0000 -113.6503    0.0000
   -0.7345   -0.2406    0.6345   63.4534  109.9045  115.1230
   -0.7345   -0.2406    0.6345  -81.5741   97.2163  -57.5615
         0         0    1.0000         0         0         0
```

- **Jacobiano inverso**

```
Jq =

  729.1688         0         0         0
         0  653.0010         0         0
         0         0  653.0010         0
         0         0         0  414.3514
```

- **Jacobiano general**

```
Jc =

    0.8229    0.5683 -113.6503   -0.0000
   -0.7345    0.6345  109.9045 -115.1230
   -0.7345    0.6345   97.2163   57.5615
         0    1.0000         0         0
```

- **Determinante**

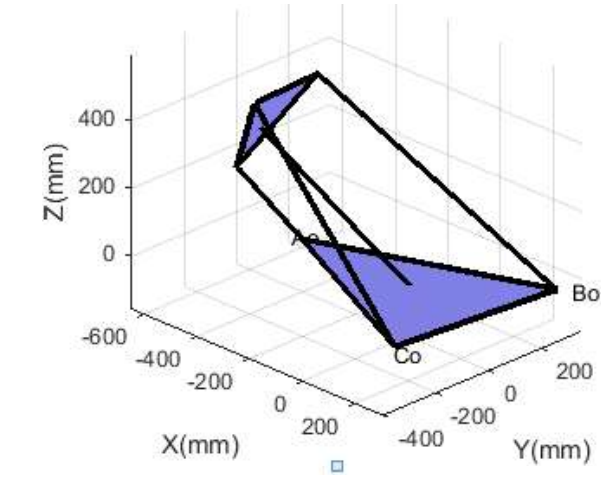
```
detJC =

    0.0017
```

- **Rango**

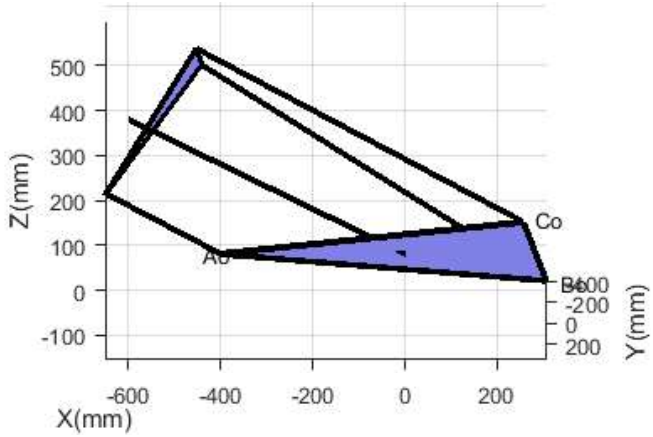
Rank (JC)=3

Ejemplo 3.3.3. Calcular las matrices jacobinas de la siguiente posición:

```
X=[-537.9812   0  300.0000   0   -0.9817   0 ];
```



a)



b)

Figura 3.7. Configuración singular del 3UPE-RPU: a) vista isométrica b) Vista lateral.

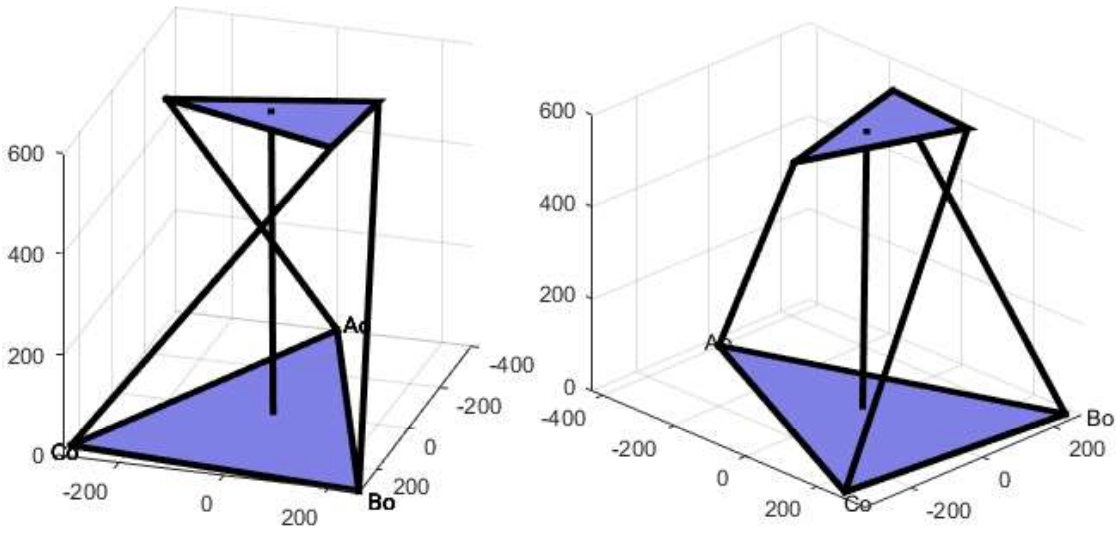
(Fuente: Autor)

3.3.2. Singularidades de Fichter y Hunt

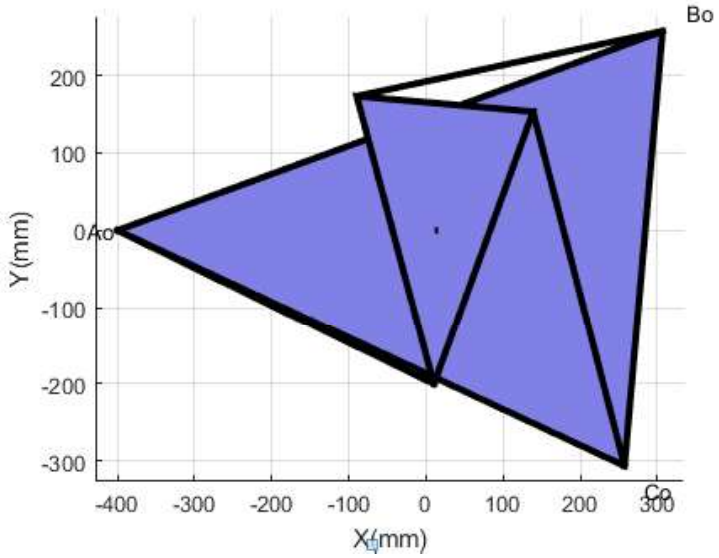
En el capítulo 1 abarca dos tipos de singularidades especiales de dos investigadores como lo es Fichter y Hunt, para aplicar su trabajo se realiza un ejemplo en el 3UPE/RPU.

Ejemplo 3.3.4: Configuración Singular de Fichter

Rotación de la plataforma móvil en $\pm 90°$ alrededor del eje z, ver figura 3.8 y 3.9.

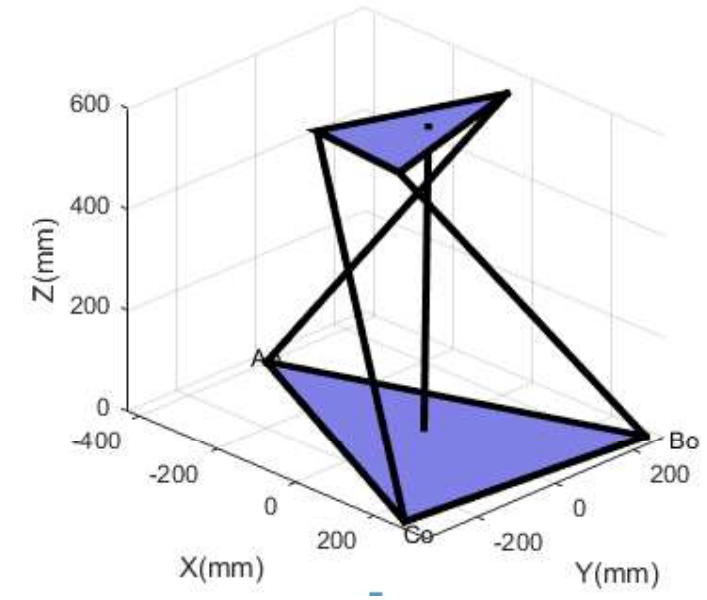


a)

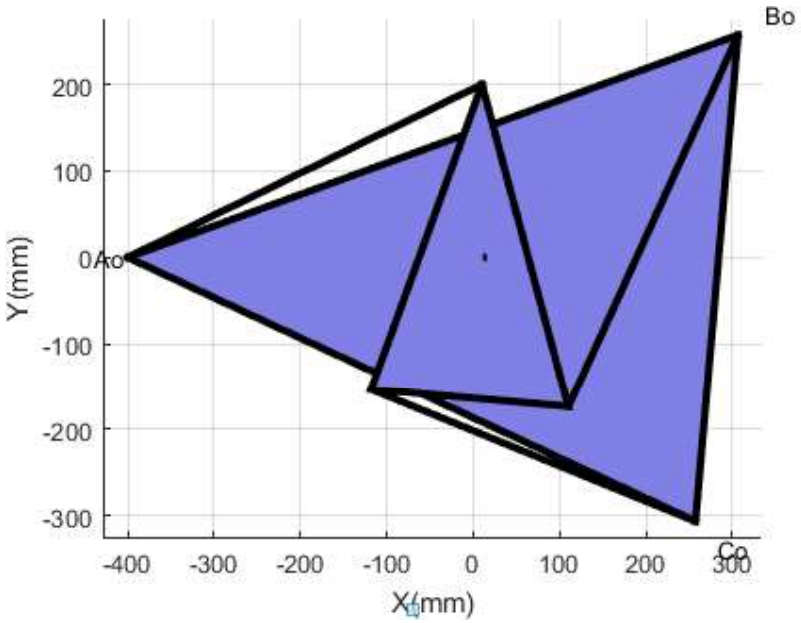


b)

Figura 3.8. Configuración singular de Fichter ($\theta = 90°$): a) vista isométrica b) Vista superior.
(Fuente: Autor)



a)

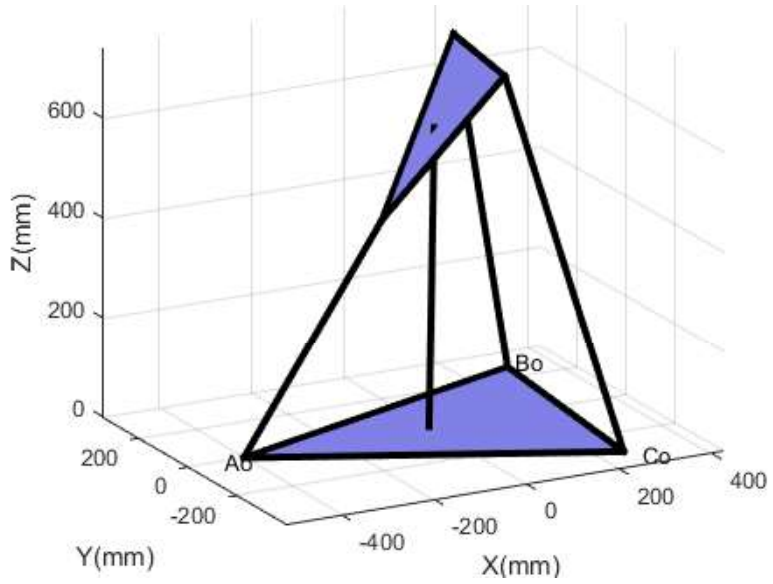


b)

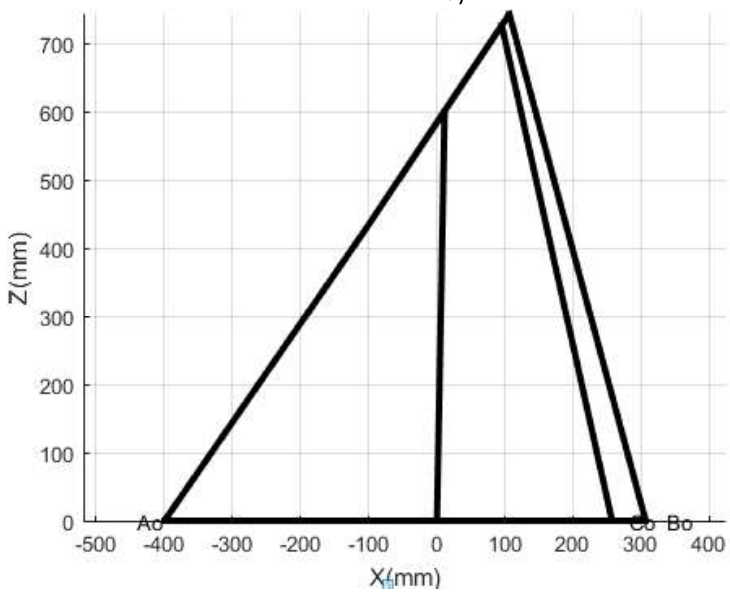
Figura 3.9 Configuración singular de Fichter ($\theta = -90°$): a) vista isométrica, b) Vista superior.
(Fuente: Autor)

Configuración Singular de Hunt

Coordenada donde existe singularidad: `Xp=[10 0 600 0 -pi/3.2 0];`



a)



b)

Figura 3.10. Configuración singular de Hunt: a) vista isométrica b) Vista lateral.

(Fuente: Autor)

Coordenada donde existe singularidad: `Xp =[10 0 600 0 pi/3.05 0]`

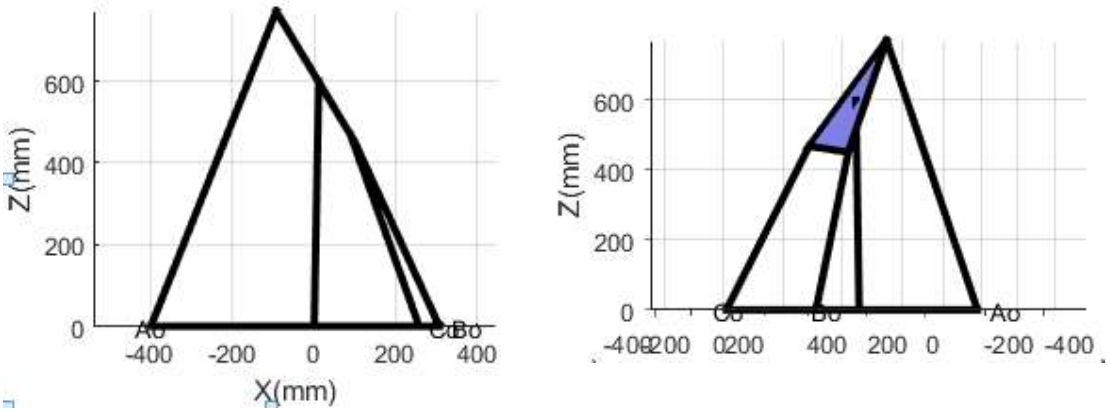


Figura 3.11. Configuración singular de Hunt: dos miembros y la plataforma móvil son coplanares. (Fuente: Autor)

3.3.3. Determinación de la posición del espacio de trabajo

Se introduce un algoritmo para representar el límite del espacio de trabajo del robot a orientación constante, este algoritmo trata de evitar singularidades en base a restricciones impuestas que están detalladas en la parte teórica, aunque las singularidades también dependen de la arquitectura simétrica o asimétrica que presente el robot.

Para evitar el problema de singularidades, se propone una disposición asimétrica de las extremidades. Los parámetros físicos se detallan en la tabla 3.3 y sus restricciones en la tabla 3.4.

El análisis del espacio de trabajo del 3UPE-RPU se realiza en el entorno Matlab, a través de la búsqueda de coordenadas esférica y el cálculo cinemático inverso,

Tabla 3.3. Parámetros geométricos del robot.

Radio Plataforma móvil	Radio Base	Ángulo Pata 2 base β_1	Ángulo Para 2 móvil α_1	Ángulo Pata 3 base β_2	Ángulo pata 3 móvil α_2
200mm	400mm	40°	30°	50°	40°

(Fuente: Autor)

Tabla 3.4. Restricciones.

Desplazamiento mínimo del actuador prismático l_{min}	Desplazamiento máximo del actuador prismático l_{max}	Ángulos articulaciones Universal pasivas θ_{bi}	Ángulos articulaciones Esféricas pasivas θ_{pi}	Colisiones entre diámetros de eslabones D
500mm	800mm	35°	85°	30mm

(Fuente: Autor)

La figura 3.12 ilustra el espacio de trabajo cartesiano para el 3UPE-RPU con la plataforma orientada en $\varphi = 0, \theta = 0$

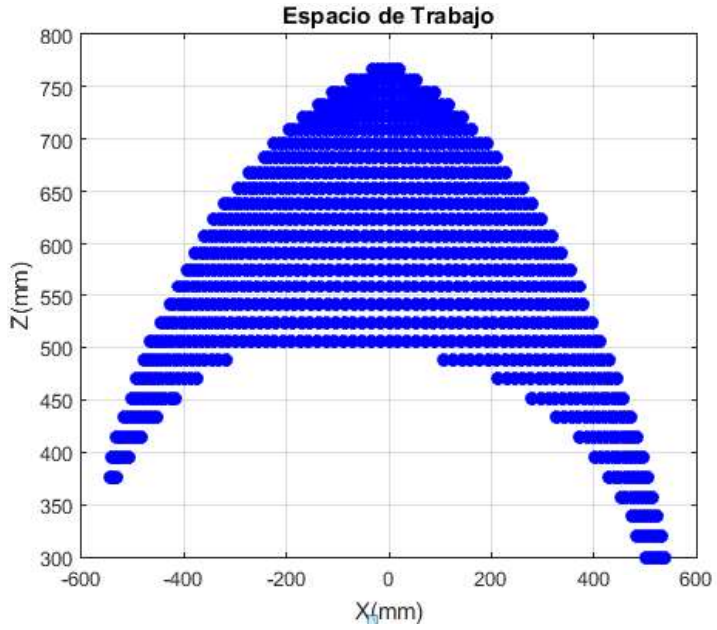


Figura 3.12. Espacio de trabajo del manipulador plano XZ, orientación constante ($\varphi = 0; \theta = 0$).
(Fuente: Autor)

Con los parámetros geométricos de la tabla 3.3 y como muestra la figura 3.12, se afirma que puede realizar todos los movimientos deseados (Aginaga, Iriarte, Plaza, & Mata, 2018).

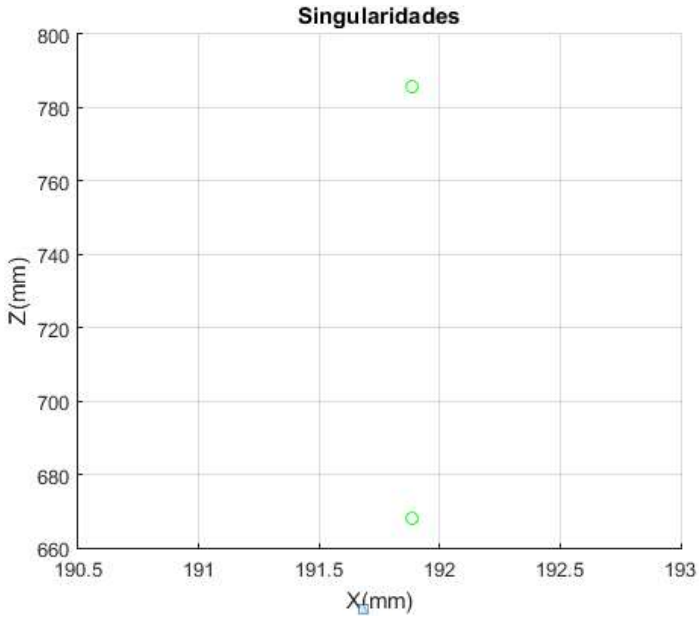


Figura 3.13. Singularidades dentro del espacio de trabajo cuando ($\varphi = 0; \theta = 0$)
(Fuente: Autor)

En la figura 3.13, se observa en el plano XZ dos posiciones donde el robot 3UPE-RPU presenta singularidades cuando la orientación es $\varphi = 0; \theta = 0$, aquí la presencia de singularidades es mínima y por la ubicación de los puntos singulares se puede decir que es una singularidad tipo II.

En la siguiente figura 3.15 se observa que variando el ángulo de rotación del eje z a θ =10°, hay un incremento de posiciones singulares.

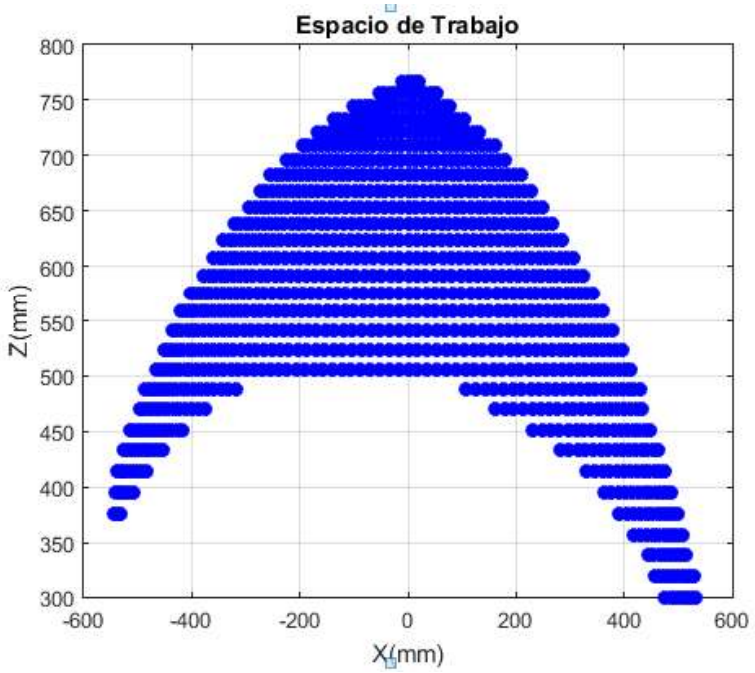


Figura 3.14. Espacio de trabajo del manipulador en el plano XZ, orientación fija ($\varphi = 0; \theta = \frac{\pi}{18}$).
(Fuente: Autor)

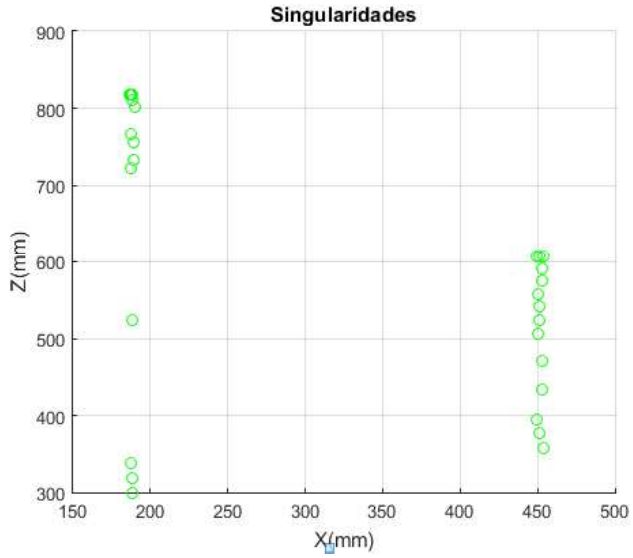


Figura 3.15. Singularidades dentro del espacio de trabajo cuando ($\varphi = 0; \theta = \frac{\pi}{18}$)
(Fuente: Autor)

En las siguientes graficas se presentan espacios de trabajo del manipulador a diferentes orientaciones con sus respectivas posiciones singulares y se observa que las restricciones se vuelven prominentes para las orientaciones de plataforma con grandes ángulos de rotación.

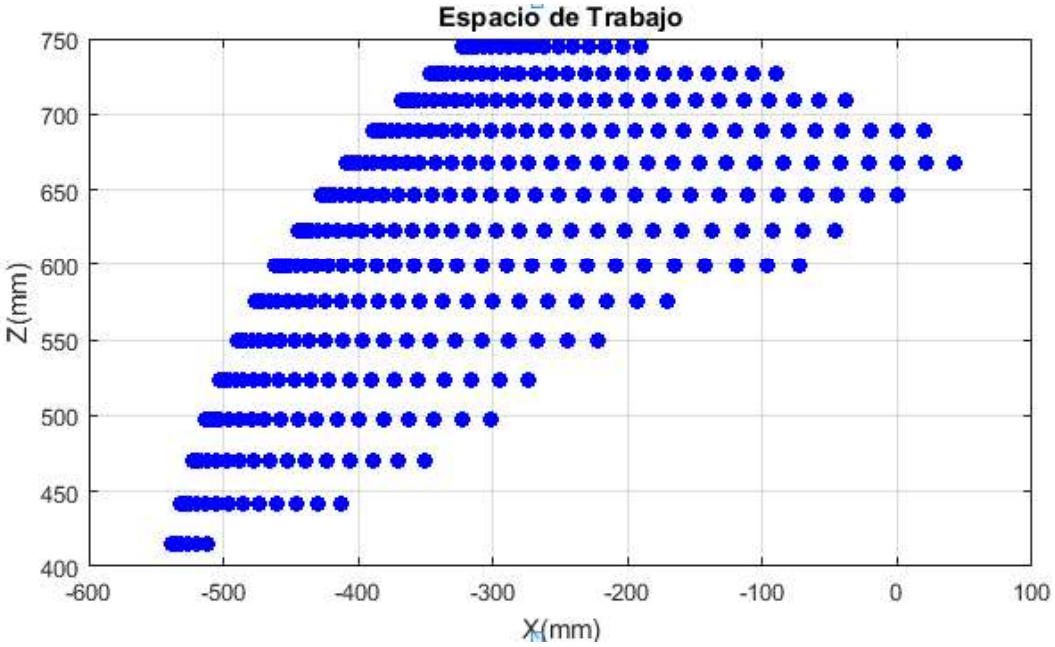


Figura 3.16.Espacio de trabajo del manipulador a orientación constante *($\varphi = \frac{\pi}{4}$; $\theta = \frac{\pi}{4}$).*
(Fuente: Autor)

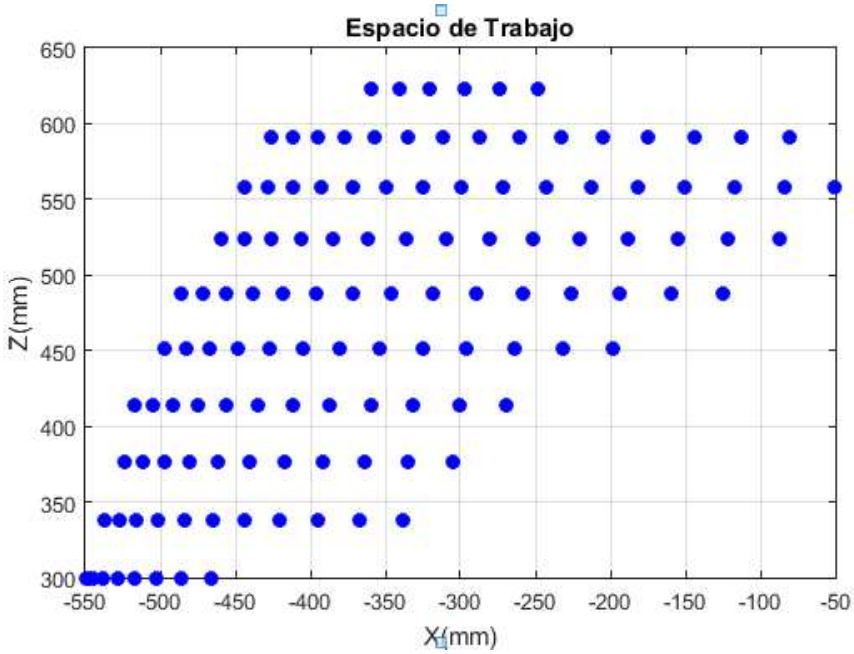


Figura 3.17. Espacio de trabajo del manipulador en el plano XZ, orientación constante *($\varphi = \frac{\pi}{3}$; $\theta = 0$).*
(Fuente: Autor)

Figura 3.18. Espacio de trabajo del manipulador en el plano XZ a orientación constante $(\varphi = \frac{\pi}{8};\ \theta = 0)$.
(Fuente: Autor)

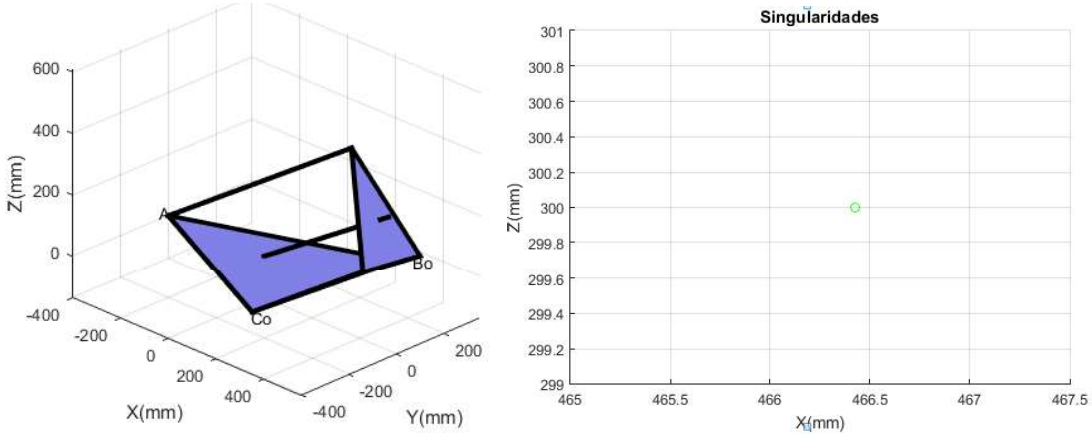


Figura 3.19. Posición singular en: X= [466.4265 0 300.0000 0 1.0472 0]
(Fuente: Autor)

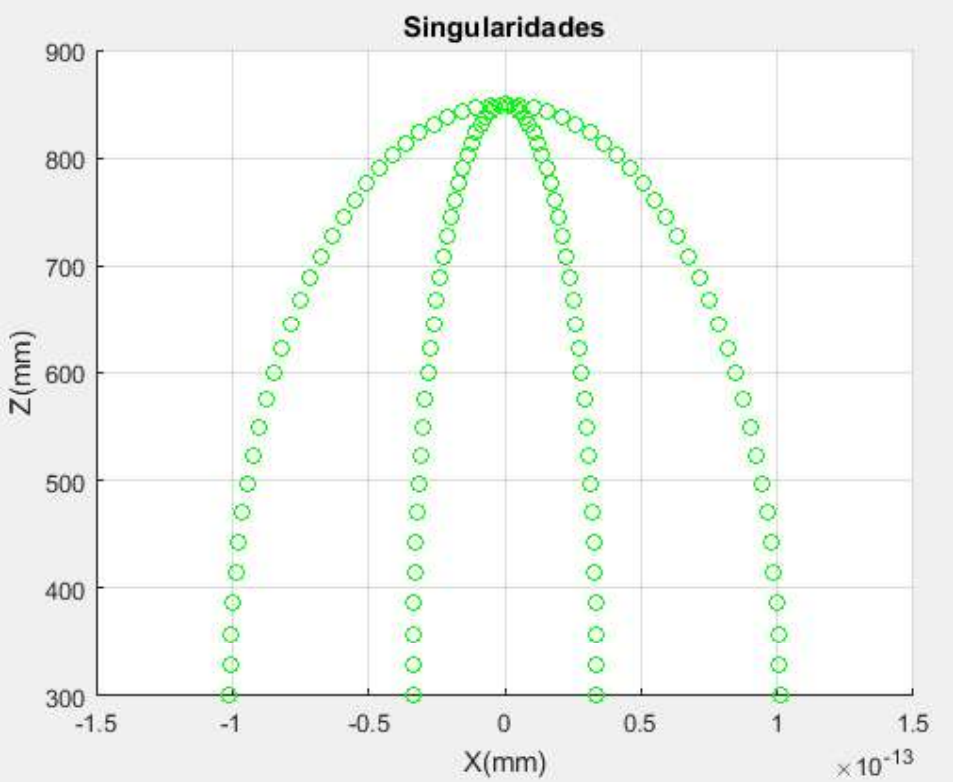


Figura 3.20. Puntos de singulares del manipulador a una orientación constante ($\varphi = \pi;\ \theta = 0$).
(Fuente: Autor)

En la figura 3.20 muestra la ubicación de posiciones singulares cuando la orientación está en $\varphi = \pi$, se observa que en el eje horizontal entre $[-1; 1]$mm el rango de la matriz jacobiana es deficiente.

Tabla 3.5. Parámetros geométricos del robot, disposición simétrica.

Radio Plataforma móvil	Radio Base	Ángulo Pata 2 base β_1	Ángulo Para 2 móvil α_1	Ángulo Pata 3 base β_2	Ángulo pata 3 móvil α_2
200mm	**400mm**	**60°**	**60°**	**60°**	**60°**

(Fuente: Autor)

Tabla 3.6. Restricciones.

Desplazamiento mínimo del actuador prismático l_{min}	Desplazamiento máximo del actuador prismático l_{max}	Ángulos mínimos de articulaciones pasivas φ_{min}	Ángulos máximos de articulaciones pasivas φ_{max}	Colisiones entre diámetros de eslabones D
500mm	**800mm**	**10°**	**25°**	**30mm**

(Fuente: Autor)

La figura 3.21 ilustra el espacio de trabajo cartesiano para el 3UPE-RPU con la plataforma orientada en $\varphi = 0, \theta = 0$ y la figura 3.22 muestra los puntos singulares que deben ser evitados.

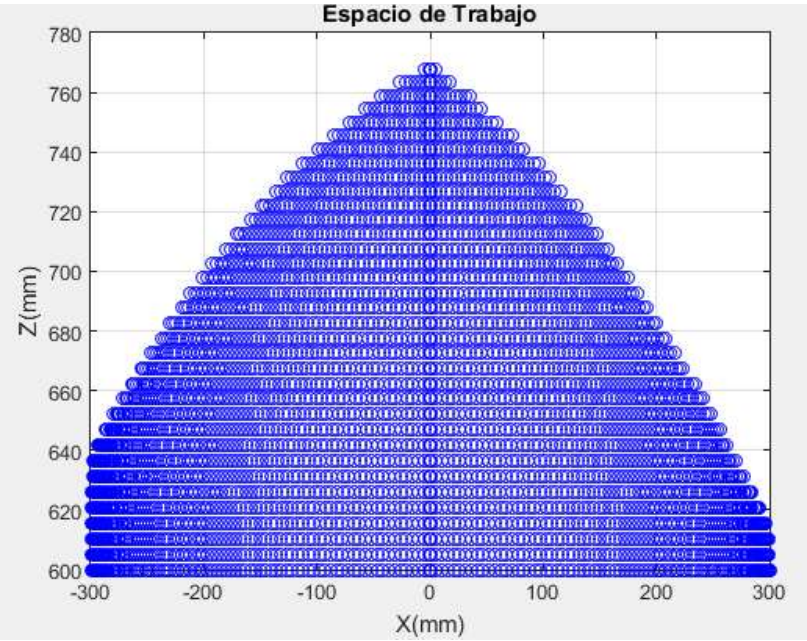


Figura 3.21. Espacio de trabajo del manipulador plano XZ, orientación constante ($\varphi = 0; \theta = 0$).

Fuente: Propia.

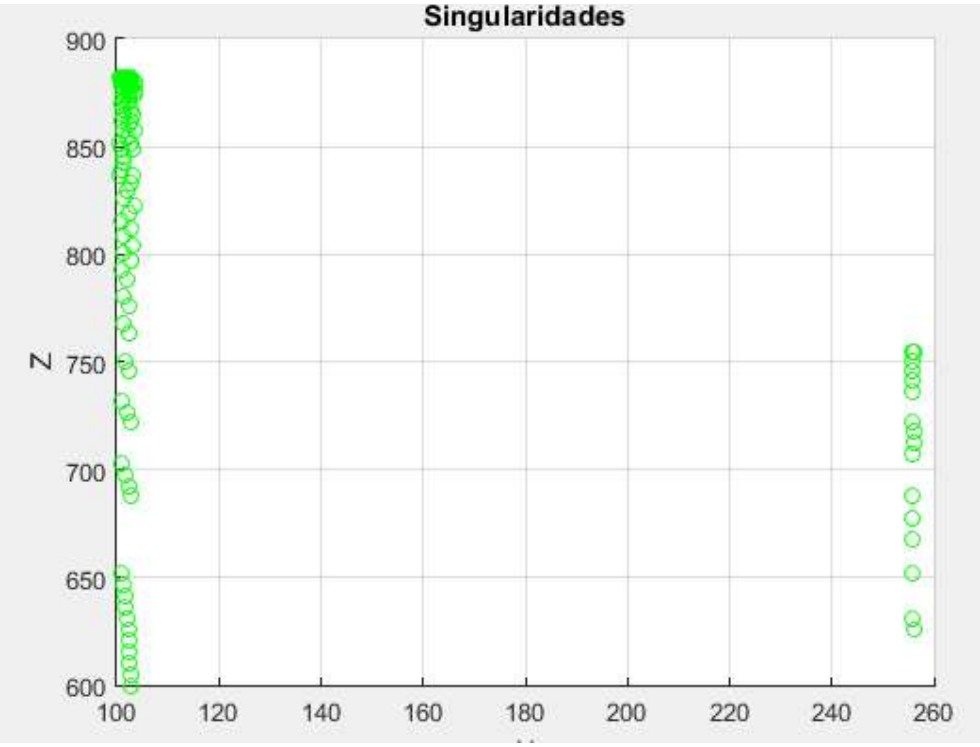


Figura 3.22. Puntos singulares en plano XZ, orientación constante ($\varphi = 0; \theta = 0$).

(Fuente: Autor)

4. CONCLUSIONES Y RECOMENDACIONES

4.1. Conclusiones

- Una vez establecida la cinemática inversa del robot 3UPE-RPU se define el espacio de trabajo, por el cual se determina la geometría del robot, las restricciones propias de las articulaciones rotacionales y sobre todo que las longitudes e inferencias de los actuadores estén dentro del rango establecido para definir las posiciones específicas que puede alcanzar el robot en una orientación fija.
- La presentación del espacio de trabajo del 3UPE-RPU dificulta su visualización en 3D, al ser un robot de 4 GDL cuyas ecuaciones de traslación de la plataforma móvil depende de la orientación, se decidió escoger entre los tipos de espacio de trabajo al de orientación fija obteniendo como resultado puntos alcanzables en el plano XZ, desplazamiento requerido para propósitos del robot.
- El espacio de trabajo de la figura 3.12 representa el desplazamiento en el plano XZ cuando la rotación alrededor del eje "y_m" y "z_m" son nulas donde se observa movimientos con mínimas singularidades, concluyendo que el del robot 3UPE-RPU es apropiado para llevar a cabo las tareas de rehabilitación debido a que alcanza los movimientos deseados y necesarios para la prueba de Lachman y el desplazamiento de pivote, pruebas de diagnóstico reconocidas en la actualidad.
- Las matrices jacobianas obtenidas en las ecuaciones 2.13 y ecuación 2.14 están en función de las sumas y productos de los senos y cosenos de los ángulos formados por las articulaciones activas y pasivas del robot 3UPE-RPU, donde encontrar las configuraciones singulares es complicado, a este tipo de matriz se la conoce como jacobiana analítica y se le dará uso en el análisis dinámico. Sin embargo, para presentar el análisis de singularidades se determina la jacobiana geométrica usando la teoría del tornillo,
- En el cálculo de la matriz jacobiana del robot, es de interés analizar las singularidades, donde se obtuvo una matriz jacobiana que no es dimensionalmente homogénea, esta característica dificulta localizar las configuraciones singulares, Sin embargo, se optó por homogenizar la matriz mediante la partición de dos matrices: traslación y rotación las cuales proporcionan resultados que han mostrado su eficacia al localizar dichas configuraciones.
- En el presente trabajo son pocas las singularidades que se pueden alcanzar sobre todo al realizar movimientos de rotación en el eje Z que deben estar en rango de $\pm 10°$,entonces los parámetros geométricos son apropiadamente seleccionados para evitar este tipo de dificultades dentro del espacio de trabajo.

4.2. Recomendaciones

- Actualmente, se han llevado a cabo muchas investigaciones sobre el análisis de la configuración singular de los manipuladores paralelos usando varias metodologías (teoría de tornillos, líneas geométricas, algebra de Grassmann Cayley), debido a que dicho análisis facilita también el futuro control de la plataforma. Una mejora consistiría en realizar dicho estudio más profundamente, en vez de restringir los movimientos del robot a un área en la que se ha podido comprobar que estaba alejada de las singularidades.

Referencias bibliográficas

Abbasi, W. (2001). Screw theory based path tracking techniques for autonomous robots and manipulators. Florida.

Aginaga, J., Iriarte, X., Plaza, A., & Mata, V. (2018, Septiembre). Kinematic Design of a New Four Degree-of-Freedom Parallel Robot for Knee Rehabilitation. *Journal of Mechanical Design*.

Araujo, P., Diaz, M., & Mata, V. (2017, Octubre). Diseño de dispositivo robótico para la rehabilitación y diagnosis de extremidades inferiores.

Bacaicoa, J. (2017, Junio). Identificación de parámetros dinámicos de un robot 3-PRS por máxima verosimilitud. *Tesis de Pregrado*. Pamplona: E.T.S de Ingeniería Industrial, Informatica y de Telecomunicación.

Ball, R. (1990). *A treatise on the theory of screws.* Cambridge. Retrieved from https://archive.org/details/atreatiseontheo02ballgoog

Bona, B. (2016). Kinematic Functions. Politecnico di Torino.

Cárdenas, P. (2016, Enero). Contribución al análisis cinemático y dinámico de manipuladores paralelos. *Tesis Doctoral*. Universidad Politécnica de Madrid.

Chang, A. (2016, Junio). Diseño, modelación y simulación de un sistema de impresión 3D basado en sistemas de cadena cinemática cerrada con 4 grados de libertad tipo delta. *Tesis de maestría*. Quito: Escuela Politécnica Nacional.

Conti, J., Clinton, C., & Zhang. (1998). Workspace Variation of Hexapod Machine Tool. University of Maryland.

Fernández, L., & Sotomayor, L. (2016, Noviembre). Análisis cinemático inverso y directo del robot paralelo. *Tesis de Maestria* . Quito: Escuela Politécnica Nacional.

Fernández, L., & Sotomayor, L. (2016, Noviembre). Análisis cinématico inverso y directo del robot paralelo. *Tesís de Maestría*. Quito: Escuela Politécnica Nacional.

Fichter, E. (1986). A Stewart Platform- Based Manipulator: General Theory and Practical Construction. *The International Journal of Robotics Research*. doi:10.1177/027836498600500216

García, M. (2014, Noviembre). Dinámica de un manipulador paralelo para tareas de maquinado considerando fuerzas de corte. *Tesis Doctoral*. Santiago de Querétaro: Instituto Politécnico Nacional.

Gharahsofloo, A., & Rahmani, A. (2015). An Efficient Algorithm for Workspace Generation of Delta Robot. *International Journal Of Robotics*, 48-53.

Gosselin, C. (1988, June 15). Kinematic Analysis, Optimization and Programming of Parallel Robotic Manipulator. Canada: McGill University.

Gosselin, C., & Angeles, J. (1990, June). Singularity analysis of closed-loop kinematic chains. *Paper*. IEEE. doi:10.1109/70.56660

Hernández, V. (2018, Enero 11). Matrices.

Huang, C., & Tsai, K. (2014, Noviembre). A general method to determine compatible orientation workspaces for different types of 6-DOF parallel manipulators. Elsevier.

Hunt, K. (1978). Kinematic Geometry of Mechanisms. Oxford : Clarendon, 1978.

Jiang, Q. (2008). Singularity-free workspace analysis and geometric optimization of parallel mechanisms. *Tesis Doctoral*.

Li, B., Cao, Y., Zhang, Q., & Wang, C. (2013, June). Singularity Representation and Workspace Determination of a Special Class of the Gough-Stewart Plataforms. *Intech*. doi:10.5772/56792

Lu, Y., Shi, Y., & Yu, J. (2009). Determination of singularities of some 4-DOF parallel manipulators by traslational/rotational Jacobian matrices. China: Robotics Research Center, College of Mechanical Engineering, Yanshan University, Quihuangdao. doi:10.1017/S0263574709990518

Morales, W. (2017, Marzo). Análisis cinemático y simulación del robot paralelo de tipología 3UPE-RPU. *Tesís de Pregrado*. Quito: Escuela PolitécnicaNacional.

Moreno, H., Saltearen, R., Carrera, I., Puglisi, L., & Aracil, R. (n.d.). Índices de desempeño de robots manipuladores: unarevision del estado del arte.

Oliva, M. (2014, Marzo 14). Obtención del espacio de trabajo de la plataforma de Gough-Stewart mediantes técnicas CAD. *Tesis Doctoral*. Elche: Universidad Miguel Hernández de Elche.

Oliva, M. (2014, Mayo). Obtención de espacio de trabajo de la plataforma de Gough-Stewart mediante técnicas CAD. *Tesis Doctoral*. Elche: Universidad Miguel Hernández de Elche.

Pareja, D., & Pazmiño, M. (2018, Febrero). Estudio de la aplicación de las ecuaciones de gibbs appell en el análisis dinámico de un robot paralelo eUPE-1RPU. *Tesis de Pregrado*. Quito: Escuela Politécnica Nacional.

Salgado, O. (2008, Enero). Síntesis, Análisis y Diseño de Manipuladores Paralelos de Baja Movilidad. *Tesis Doctoral*. Bilbao: Universidad del País Vasco Euskal Herriko Unibertsitatea.

Sevillano, G. (2014, Febrero). Diseño mecánico de un simulador de marcha normal basado en la plataforma stewart-gough. *Tesís de Pregrado*. Lima: Pontificia Universidad Católica del Perú.

Silva, L. (2005). Control visual de robots paralelos. Análisis, desarrollo y aplicación a la plataforma robotenis . *Tesis Doctoral*. Escuela Técnica Superior de Ingenieros Industriales.

St-Onge, B., & Gosselin, C. (2000, Marzo). Singularity analysis and representation of the general gough-stewart platform. (Sage, Ed.) *The International Journal of Robotics Research*. doi:10.1177/02783640022066860

Villegas, M. (2016, Marzo). Control cartesiano de una mano mecánica de dos dedos. *Maestria en Ingenieria Eléctrica*. México: Universidad Nacional Autonoma de Mexico.

Zamora, P. (2016, Septiembre). Diseño de un robot paralelo para aplicaciones de diagnosis y rehabilitación de rodilla. *Tesis de Maestría*. Valencia: Universidad Politécnica de Valencia.

Khrev, A. (2016). Kinhmatikh anaΛγΣh enoΣ KUKA youBot. ΕΘΝΙΚΟ ΜΕΤΣΟΒΙΟ ΠΟΛΥΤΕΧΝΕΙΟ.

Anexos

A1. Algoritmo para la obtención simbólica del jacobiano analítico dependiente e independiente del modelo del robot 3UPR/RPU

```
%=========================================================================
==============================
% Escuela Politécnica Nacional
% Maestría en Mecatrónica y Robótica
% Proyecto de desarrollo
% Ing. Diana Tumbaco
%ANÁLISIS DE SINGULARIDADES Y ESPACIO DE TRABAJO DEL ROBOT PARALELO DE
TIPOLOGÍA 3UPS+1RPU
%=========================================================================
==============================

clear all; clc;clearvars;
%PARAMÉTROS GEOMÉTRICOS
 Xm = sym('Xm','real'); Zm = sym('Zm','real');phi = sym('phi','real');
th = sym('th','real'); Rm = sym('Rm','real'); R = sym('R','real');
 al1 = sym('al1','real');be1 = sym('be1','real');
 al2 = sym('al2','real');be2 = sym('be2','real');
 f = sym('f','real');

%MATRIZ DE MOV DE TRASLACIÓN Y ROTACIÓN DE PLATAFORMA MÓVIL
Txz=[1 0 0 Xm;0 1 0 0;0 0 1 Zm;0 0 0 1];
Ry=[cos(phi) 0 sin(phi) 0;0 1 0 0;-sin(phi) 0 cos(phi) 0;0 0 0 1];
Rz=[cos(th) -sin(th) 0 0 ;sin(th) cos(th) 0 0;0 0 1 0;0 0 0 1];
Rzy=Rz*Ry;
TR=Txz*Rzy;

%POSICIÓN VÉRTICES DE LA PLATAFORMA MÓVIL (Pi=[A,B,C]) Coordenadas
iniciales
Pi=sym(zeros(4,3));
Pi(:,1)=[-Rm; 0; 0; 1];                %A
Pi(:,2)=[Rm*cos(al1);Rm*sin(al1);0;1];%B
Pi(:,3)=[Rm*cos(al2);Rm*sin(al2);0;1];%C

%POSICIÓN VÉRTICES DE LA PLATAFORMA MÓVIL (Pi=[Ai,Bi,Ci]) Coordenadas
finales
Pf=sym(zeros(4,3));
for i=1:3
    Pf(:,i)=TR*Pi(:,i);
end
Pfv=Pf;

%Resolución de parámetros de articulaciones para el brazo A
%-------------------------------------------------------------
 syms A1 A2 A3 bA
% Articulación 1
 a1=-R; alp1=90; d1=0; th1=0;
 T01A=[cos(th1) -(cosd(alp1))*(sin(th1)) (sind(alp1))*(sin(th1))
a1*(cos(th1));sin(th1) (cosd(alp1))*(cos(th1)) -
(sind(alp1))*(cos(th1))...
     a1*sin(th1);0 sind(alp1) cosd(alp1) d1; 0 0 0 1];%Matriz de
transformación homogénea
% Articulación 2
 T12A=[-sin(A2) 0 cos(A2) 0;cos(A2) 0 sin(A2) 0;0 1 0 0;0 0 0 1];
```

```
% Articulacion 3
 a3=A1;alp3=0;d3=0;th2=A3;
 T23A=[cos(th2) -(cosd(alp3))*(sin(th2)) (sind(alp3))*(sin(th2))
a3*(cos(th2));sin(th2) (cosd(alp3))*(cos(th2)) -
(sind(alp3))*(cos(th2))...
     a3*sin(th2);0 sind(alp3) cosd(alp3) d3; 0 0 0 1];%Matriz de
transformación homogénea

% Matrices de transformación homogénea entre sistemas de coordenadas
consecutivos
 PA=T01A*T12A*T23A;
 PAx=PA(1,4)-Pfv(1,1)==0;
 PAy=PA(2,4)-Pfv(2,1)==0;
 PAz=PA(3,4)-Pfv(3,1)==0;

%Resolución de parámetros de articulaciones para el brazo B
%-------------------------------------------------------------
 syms B1 B2 B3 bB
% Articulación 1
 %bB=-180+beta;
a1=-R; alp1=90; d1=0; th1=bB;
T01B=[cos(th1) -(cosd(alp1))*(sin(th1)) (sind(alp1))*(sin(th1))
a1*(cos(th1));sin(th1) (cosd(alp1))*(cos(th1)) -
(sind(alp1))*(cos(th1))...
     a1*sin(th1);0 sind(alp1) cosd(alp1) d1; 0 0 0 1];%Matriz de
transformación homogénea
 % Articulación 2
 T12B=[-sin(B2) 0 cos(B2) 0;cos(B2) 0 sin(B2) 0;0 1 0 0;0 0 0 1];
 %Articulacion 3
 a3=B1;alp3=0;d3=0;th2=B3;
 T23B=[cos(th2) -(cosd(alp3))*(sin(th2)) (sind(alp3))*(sin(th2))
a3*(cos(th2));sin(th2) (cosd(alp3))*(cos(th2)) -
(sind(alp3))*(cos(th2))...
     a3*sin(th2);0 sind(alp3) cosd(alp3) d3; 0 0 0 1];%Matriz de
transformación homogénea

% Matrices de transformación homogénea entre sistemas de coordenadas
consecutivos
 PB=T01B*T12B*T23B;
 PBp=PB(1:3,4);
 PBx=PB(1,4)-Pf(1,2)==0;
 PBy=PB(2,4)-Pf(2,2)==0;
 PBz=PB(3,4)-Pf(3,2)==0;

% Resolución de parámetros de articulaciones para el brazo C
%-------------------------------------------------------------
 syms C1 C2 C3 bC
% Articulación 1
%bC=180-beta;
 a1=-R; alp1=90; d1=0; th1=bC;
 T01C=[cos(th1) -(cosd(alp1))*(sin(th1)) (sind(alp1))*(sin(th1))
a1*(cos(th1));sin(th1) (cosd(alp1))*(cos(th1)) -
(sind(alp1))*(cos(th1))...
     a1*sin(th1);0 sind(alp1) cosd(alp1) d1; 0 0 0 1];%Matriz de
transformación homogénea
 % Articulación 2
 T12C=[-sin(C2) 0 cos(C2) 0;cos(C2) 0 sin(C2) 0;0 1 0 0;0 0 0 1];
 %Articulacion 3
 a3=C1;alp3=0;d3=0;th2=C3;
```

```
 T23C=[cos(th2) -(cosd(alp3))*(sin(th2)) (sind(alp3))*(sin(th2))
a3*(cos(th2));sin(th2) (cosd(alp3))*(cos(th2)) -
(sind(alp3))*(cos(th2))...
     a3*sin(th2);0 sind(alp3) cosd(alp3) d3; 0 0 0 1];%Matriz de
transformación homogénea

 % Matrices de transformación homogénea entre sistemas de coordenadas
consecutivos
 PC=T01C*T12C*T23C;
 %PCp=PC(1:3,4);
 PCx=PC(1,4)-Pf(1,3)==0;
 PCy=PC(2,4)-Pf(2,3)==0;
 PCz=PC(3,4)-Pf(3,3)==0;

 %Resolución de parámetros de articulaciones para el brazo D
 %-------------------------------------------------------------
 syms D1 D2
 % Articulación 1
 a1=0; alp1=D2; d1=0; th1=90;
 T01D=[cosd(th1) -(cos(alp1))*(sind(th1)) (sin(alp1))*(sind(th1))
a1*(cosd(th1));sind(th1) (cos(alp1))*(cosd(th1)) -
(sin(alp1))*(cosd(th1))...
     a1*sind(th1);0 sin(alp1) cos(alp1) d1; 0 0 0 1];%Matriz de
transformación homogénea
 % Articulación 2
 a2=0;alp2=0;d2=D1;t2=0;
 T12D=[cos(t2) -(cos(alp2))*(sin(t2)) (sin(alp2))*(sin(t2))
a2*(cos(t2));sin(t2) (cos(alp2))*(cos(t2)) -(sin(alp2))*(cos(t2))...
     a2*sin(t2);0 sin(alp2) cos(alp2) d2; 0 0 0 1];
  %Articulacion 3
 a3=0;alp3=0;d3=0;th2=0;
 T23D=[cos(th2) -(cosd(alp3))*(sin(th2)) (sind(alp3))*(sin(th2))
a3*(cos(th2));sin(th2) (cosd(alp3))*(cos(th2)) -
(sind(alp3))*(cos(th2))...
     a3*sin(th2);0 sind(alp3) cosd(alp3) d3; 0 0 0 1];%Matriz de
transformación homogénea

 % Matrices de transformación homogénea entre sistemas de coordenadas
consecutivos
 PD=T01D*T12D*T23D;
  PDx=PD(1,4)-Xm==0;
  PDz=PD(3,4)-Zm==0;

 % ECUACIONES DE RESTRICCIÓN
 f(1)=Rm*cos(phi)*cos(th) - Xm - R - A1*cos(A3)*sin(A2);
 f(2)=Rm*cos(phi)*sin(th) - A1*sin(A3);
 f(3)=A1*cos(A2)*cos(A3) - Rm*sin(phi) - Zm;
 f(4)=B1*sin(B3)*sin(bB) - R*cos(bB) - Xm + Rm*sin(al1)*sin(th) -
Rm*cos(al1)*cos(phi)*cos(th) - B1*cos(B3)*sin(B2)*cos(bB);
 f(5)=- R*sin(bB) - B1*sin(B3)*cos(bB) - Rm*sin(al1)*cos(th) -
Rm*cos(al1)*cos(phi)*sin(th) - B1*cos(B3)*sin(B2)*sin(bB);
 f(6)=Rm*cos(al1)*sin(phi) - Zm + B1*cos(B2)*cos(B3);
 f(7)=C1*sin(C3)*sin(bC) - R*cos(bC) - Xm + Rm*sin(al2)*sin(th) -
Rm*cos(al2)*cos(phi)*cos(th) - C1*cos(C3)*sin(C2)*cos(bC);
 f(8)=- R*sin(bC) - C1*sin(C3)*cos(bC) - Rm*sin(al2)*cos(th) -
Rm*cos(al2)*cos(phi)*sin(th) - C1*cos(C3)*sin(C2)*sin(bC);
 f(9)=Rm*cos(al2)*sin(phi) - Zm + C1*cos(C2)*cos(C3);
 f(10)=D1*sin(D2) - Xm;
 f(11)=D1*cos(D2) - Zm;
```

```
 % VARIABLES INDEPENDIENTES
 %-------------------------------------------------------------------------
---------
 A1 = sym('A1','real');B1 = sym('B1','real');C1 = sym('C1','real');D1 =
sym('D1','real');
 VAR_IND=[A1,B1,C1,D1];

 % VARIABLES SECUNDARIAS(DEPENDIENTES)
 %-------------------------------------------------------------------------
----------
 A2 = sym('A2','real');B2 = sym('B2','real');C2 = sym('C2','real');D2 =
sym('D2','real');
 A3 = sym('A3','real');B3 = sym('B3','real');C3 = sym('C3','real');
 Xm = sym('Xm','real');Zm = sym('Zm','real');th = sym('th','real');phi =
sym('phi','real');
 VAR_S=[Xm,Zm,th,phi,A2,A3,B2,B3,C2,C3,D2];

 for i=1:11
     for j=1:11
         Js(i,j)=diff(f(i),VAR_S(j));
     end
 end
 Jdep=Js;
%spy(Jdep); %patrón de dispersión.(Representa en una figura los elementos
distintos de cero de la matriz );

 for i=1:11
     for j=1:4
         Jid(i,j)=diff(f(i),VAR_IND(j));
     end
 end
 Jind=Jid;
 %spy(Jind)
```

A2. Algoritmo para la obtención del jacobiano geométrico y homogenización de la matriz del modelo del robot 3UPR/RPU en forma simbólica.

```
clc;clear all
Rm=200;  R=400;
alfa=pi/6;beta=5*pi/18;

A1 = [-Rm;0;0];  B1= [Rm*cos(alfa);Rm*sin(alfa);0];
C1 = [Rm*cos(-2*pi/9);Rm*sin(-2*pi/9);0];
Om = [0;0;0];

% Xp=[-69.6070 0 623.2819  0 0 pi/2];
%Xp = [-69.6070         0  843.2286         0         0    1.5708]
Xp =[537.2439         0  328.7848         0    1.2000         0];

%Función Calcular Cinematica Inversa
[Joints L vL S] = CinInv_3UPE_RPU(Xp,Rm,R,alfa,beta)
%Función presenta al robot 3UPE-RPU
Repres_3UPE_RPU(L,Xp,Rm,R,alfa,beta)

Ry=[cos(Xp(5)) 0 sin(Xp(5)) 0;0 1 0 0;-sin(Xp(5)) 0 cos(Xp(5)) 0;0 0 0
1];
Rz=[cos(Xp(6)) -sin(Xp(6)) 0 0 ;sin(Xp(6)) cos(Xp(6)) 0 0;0 0 1 0;0 0 0
1];
Ryz=Rz*Ry;   Ryz1=Ryz(1:3,1:3);

Jx=single([S(:,1)',(cross(Ryz1*A1,S(:,1)))';S(:,2)',(cross(Ryz1*B1,S(:,2)
))';S(:,3)',...
    (cross(Ryz1*C1,S(:,3)))';S(:,4)',(cross(Ryz1*Om,S(:,4)))'])

Jq=[dot(S(:,1),vL(:,1)) 0 0 0;0 dot(S(:,2),vL(:,2)) 0 0;0 0
dot(S(:,3),vL(:,3)) 0;...
    0 0 0 dot(S(:,4),vL(:,3))]
detJq=det(Jq);

Jv=[1 0 0 0;0 0 0 0;0 1 0 0];
Jw=Ryz1*[0 0 0 0;0 0 1 0;0 0 0 1];
Je=[Jv;Jw];
Jc=single(Jx*Je)
detJC=det(Jc)
rank(Jc)
```

A3. Algoritmo para localizar singularidades en el espacio de trabajo con sus restricciones.

```
if rank(Jc)<4
     Jc=single(Jx*Je)
    Determ=det(Jc)
    ra=rank(Jc)
    pos_singular=[x,z,theta_y,phi_z]
    figure (1)
    hold on
    title('Singularidades')
    plot(x,z,'go')
    grid on
     drawnow;
      pause(0.3)
            xlabel('X(mm)'); %ylabel('Y');
            ylabel('Z(mm)');
            zlabel('\theta (rad)')
            figure (2)
            Xp=[x y z  psi_x theta_y phi_z]
            alfa=pi/6;beta=5*pi/18;
 end
lmin=500;lmax=lmin+300;
ang_u=35;
ang_s=85;
D=30;

pos=longitud_Act(L,lmin,lmax);   %Función longitud_Act.m
if pos==1
   pos=angulo_joint(Rzy1,S,ang_u,ang_s); %Función angulo_joint.m
      if pos==1
       pos=colision(Rzy1,Ai,Bi,D); %Función colision.m
       if pos ==1
         Resp=1
              obs=[x,z];
            figure (2)
            title('Espacio de Trabajo')
           plot(x,z,'bo', 'MarkerFaceColor','[0 0.00 1.00]')
           grid on
          drawnow;
            pause(0.3)
%
             hold on
             xlabel('X(mm)'); %ylabel('Y');
             ylabel('Z(mm)')
             %zlabel('\theta (rad)')


        else
            Resp=0;
      end
  end
```