



ESPE
UNIVERSIDAD DE LAS FUERZAS ARMADAS
INNOVACIÓN PARA LA EXCELENCIA

**DEPARTAMENTO DE ELÉCTRICA, ELECTRÓNICA Y
TELECOMUNICACIONES**

**CARRERA DE INGENIERÍA EN ELECTRÓNICA,
AUTOMATIZACIÓN Y CONTROL**

**TRABAJO DE TITULACIÓN, PREVIO A LA OBTENCIÓN DEL TÍTULO
DE INGENIERO ELECTRÓNICO EN AUTOMATIZACIÓN Y CONTROL**

**TEMA: “MODELAMIENTO CINEMÁTICO DEL SIMULADOR DE
DESORIENTACIÓN ESPACIAL DEL CICTE PARA
ENTRENAMIENTO DE PILOTOS DE LAS FUERZAS ARMADAS”**

AUTOR:

GUERRÓN TAPIA, CRISTHIAN FELIPE

DIRECTOR: DR. AGUILAR CASTILLO, WILBERT GEOVANNY

SANGOLQUÍ

2018



**DEPARTAMENTO DE ELÉCTRICA, ELECTRÓNICA Y
TELECOMUNICACIONES**

**CARRERA DE INGENIERÍA EN ELECTRÓNICA, AUTOMATIZACIÓN Y
CONTROL**

CERTIFICACIÓN

Certifico que el trabajo de titulación, “*MODELAMIENTO CINEMÁTICO DEL SIMULADOR DE DESORIENTACIÓN ESPACIAL DEL CICTE PARA ENTRENAMIENTO DE PILOTOS DE LAS FUERZAS ARMADAS*” fue realizado por el señor *GUERRÓN TAPIA, CRISTHIAN FELIPE* el mismo que ha sido revisado en su totalidad y analizado por la herramienta de verificación de similitud de contenido; por lo tanto cumple con los requisitos teóricos, científicos, técnicos, metodológicos y legales establecidos por la Universidad de Fuerzas Armadas ESPE, razón por la cual me permito acreditar y autorizar para que lo sustenten públicamente.

Sangolqui, Agosto del 2018

.....
Ing. Wilbert Geovanny Aguilar Catillo PhD.

C.C: 0703844696



**DEPARTAMENTO DE ELÉCTRICA, ELECTRÓNICA Y
TELECOMUNICACIONES
CARRERA DE INGENIERÍA EN ELECTRÓNICA, AUTOMATIZACIÓN Y
CONTROL**

AUTORÍA DE RESPONSABILIDAD

Yo, **GUERRÓN TAPIA, CRISTHIAN FELIPE**, declaro que el contenido, ideas y criterios del trabajo de titulación: “**MODELAMIENTO CINEMÁTICO DEL SIMULADOR DE DESORIENTACIÓN ESPACIAL DEL CICTE PARA ENTRENAMIENTO DE PILOTOS DE LAS FUERZAS ARMADAS**” es de mi autoría y responsabilidad, cumpliendo con los requisitos teóricos, científicos, técnicos, metodológicos y legales establecidos por la Universidad de Fuerzas Armadas ESPE, respetando los derechos intelectuales de terceros y referenciando las citas bibliográficas.

Consecuentemente el contenido de la investigación mencionada es veraz.

Sangolquí, Agosto de 2018.

.....
Cristhian Felipe Guerrón Tapia

CC: 1726761586



**DEPARTAMENTO DE ELÉCTRICA, ELECTRÓNICA Y
TELECOMUNICACIONES**

**CARRERA DE INGENIERÍA EN ELECTRÓNICA, AUTOMATIZACIÓN Y
CONTROL**

AUTORIZACIÓN

Yo, **GUERRÓN TAPIA, CRISTHIAN FELIPE**, autorizo a la Universidad de las Fuerzas Armadas ESPE publicar el trabajo de titulación: “**MODELAMIENTO CINEMÁTICO DEL SIMULADOR DE DESORIENTACIÓN ESPACIAL DEL CICTE PARA ENTRENAMIENTO DE PILOTOS DE LAS FUERZAS ARMADAS**” en el Repositorio Institucional, cuyo contenido, ideas y criterios son de mi responsabilidad.

Sangolquí, Agosto del 2018

.....
Cristhian Felipe Guerrón Tapia

CC: 1726761586

DEDICATORIA

Este trabajo se lo dedico a mi madre Jeanneth que fue mi motor para seguir adelante inculcándome valores y consejos para seguir avanzando en cada etapa de mi vida.

A mi familia que siempre me apoyaron en todos los momentos para lograr finalizar esta etapa de mi vida.

Cristhian Felipe Guerrón Tapia

AGRADECIMIENTO

Agradezco a mi madre que es mi soporte en todas las situaciones que atravieso en cada etapa de mi vida, guiándome siempre con el ejemplo y el aliento para cumplir todas mis metas propuestas.

Agradezco a mi familia que me apoyaron en todo momento y me ayudaron en cada situación que tuve solucionar, para lograr mis objetivos y enseñándome a ser una persona madura y fuerte.

A mis amigos que me ayudaron para lograr finalizar esta etapa.

Al Dr Wilbert Aguilar, director de este trabajo, por guiarme con su conocimiento y colaboración durante el desarrollo de este proyecto.

Cristhian Felipe Guerrón Tapia

ÍNDICE DE CONTENIDO

CERTIFICACIÓN	i
AUTORÍA DE RESPONSABILIDAD	ii
AUTORIZACIÓN	iii
DEDICATORIA	iv
AGRADECIMIENTO	v
ÍNDICE DE CONTENIDO	vi
ÍNDICE DE TABLAS	x
ÍNDICE DE FIGURAS	xii
RESUMEN	xv
ABSTRACT	xvi
CAPITULO I	1
1. INTRODUCCIÓN	1
1.1 Antecedentes.....	1
1.2 Justificación.....	3
1.3 Alcance del Proyecto.....	4
1.4 Objetivos.....	6
1.4.1 Objetivo General	6
1.4.2 Objetivos Específicos	6
CAPITULO II	7
2. ESTADO DEL ARTE	7
2.1 Conceptos Previos	7
2.1.1 Desorientación espacial	7
2.1.2 Clasificación de la desorientación espacial	8
2.1.2.1 Tipo I: Desorientación espacial no reconocida	8
2.1.2.2 Tipo II: Desorientación espacial reconocida	8
2.1.2.3 Tipo III: Desorientación espacial incapacitante	8
2.1.3 Ilusiones de vuelo	9
2.1.3.1 Ilusiones de visuales	12
2.1.3.2 Ilusiones vestibulares.....	12
2.1.3.2.1 Ilusiones somatográficas.....	13
2.1.3.2.2 Ilusiones somatogiratorias	13

2.2 Simuladores de desorientación espacial	14
2.3 Métodos para modelar la cinemática de los simuladores de vuelo	19
2.3.1 Velocidad de referencia	19
2.3.2 Ángulos de Euler	20
2.3.3 Cuaterniones	23
2.3.4 Métodos Directos.....	24
2.4 Control cinemático a partir de modelos de movimiento	28
2.4.1 Control desacoplado y acoplado.....	29
2.4.2 Control cinemático por interpolación de trayectorias	31
2.4.3 Otros	34
2.5 Simulaciones de modelos cinemáticos	35
2.5.1 GUI MATLAB	35
2.5.2 Solidworks	35
2.5.3 Otros	36
CAPITULO III	39
3. MODELADO Y CONTROL CINEMÁTICO	39
3.1 Descripción del funcionamiento de la simulación en ANSYS	39
3.2 Especificaciones de funcionamiento	40
3.2.1 Estructura.....	40
3.2.2 Especificaciones de funcionamiento	43
3.2.3 Especificaciones de las ilusiones de vuelo	44
3.3 Modelado de Posición	45
3.3.1 Cinemática inversa	45
3.3.1.1 Cinemática del manipulador 3 RPS.....	48
3.3.1.1.1 Restricciones de movimiento del manipulador 3 RPS	52
3.3.1.2 Modelo de la plataforma giratoria	53
3.3.2 Cinemática directa	58
3.3.2.1 Método vectorial.....	58
3.3.2.2 Convención Denavit Hartenberg	62
3.4 Velocidad.....	65
3.4.1 Velocidad del modelo inverso	67
3.4.1.1 Singularidades de velocidad en el modelo inverso.....	68
3.4.2 Velocidad del modelo directo.....	68

3.4.2.1 Singularidades de velocidad en el modelo directo	70
3.4.3 Determinación de la velocidad de los actuadores.....	70
3.5 Aceleración.....	71
3.5.1 Aceleración del modelo inverso	72
3.5.1.1 Singularidades de aceleración en el modelo inverso	73
3.5.2 Aceleración del modelo directo.....	73
3.5.2.1 Singularidades de aceleración en el modelo directo.....	74
3.5.3 Determinación de la aceleración de los actuadores	74
3.6 Control cinemático por interpolación de trayectorias	75
3.6.1 Interpoladores	81
3.6.1.1 Interpolador Cuadrático.....	81
3.6.1.2 Interpolador cúbico.....	82
CAPITULO IV	87
4. INTERFAZ EN MATLAB DE LOS MODELOS CINEMÁTICOS OBTENIDOS	87
4.1 Interfaz del modelo cinemático inverso con datos estáticos	87
4.1.1 Parámetros de entrada y salida	88
4.1.2 Diseño de la pantalla de la interfaz.....	88
4.1.3 Algoritmo de funcionamiento de la interfaz inversa	89
4.2 Interfaz del modelo cinemático directo con datos estáticos	91
4.2.1 Datos de entrada y salida.....	91
4.2.2 Diseño de la pantalla de la interfaz.....	92
4.2.3 Algoritmo de funcionamiento de interfaz directa.....	93
4.3 Interfaz de simulación de movimiento a partir de las trayectorias de las ilusiones.....	94
4.3.1 Parámetro de entrada y salida de la interfaz de simulación de movimiento	95
4.3.2 Diseño de la pantalla de la interfaz simulación de movimiento	95
4.3.3 Algoritmo de funcionamiento de la interfaz.....	96
CAPITULO V	99
5. RESULTADOS	99
5.1 Comprobación de los resultados de la interfaz inversa con ANSYS.	99
5.2 Comprobación de resultados entre los modelos inverso y directo	104
5.3 Comprobación de las trayectorias de las ilusiones dinámicas de vuelo.	106
5.3.1 Comprobación de la Coriolis 1	107
5.3.2 Barrena 1	113

CAPÍTULO VI	118
6. CONCLUSIONES Y RECOMENDACIONES	118
6.1 Conclusiones.....	118
6.2 Recomendaciones	119
REFERENCIAS BIBLIOGRÁFICAS	121

ÍNDICE DE TABLAS

Tabla 1. Especificaciones de las articulaciones activas de los brazos.....	42
Tabla 2. Especificaciones de funcionamiento del manipulador R-3RPS	44
Tabla 3. Especificaciones de las ilusiones vestibulares de trabajo.....	44
Tabla 4. Descripción de las variables de la base móvil y plataforma.....	46
Tabla 5. Descripción de las variables del manipulador 3 RPS	50
Tabla 6. Descripción de las variables de la plataforma giratoria	53
Tabla 7. Variables dependientes e independientes del modelo cinemático inverso	55
Tabla 8. Descripción de las articulaciones activas en el manipulador R-3RPS	56
Tabla 9. Descripción de las variables de trabajo del método vectorial	59
Tabla 10. Parámetros Denavit Hartenberg	62
Tabla 11. Parámetros Denavit Hartenberg para el brazo 1.....	63
Tabla 12. Parámetros Denavit Hartenberg para el brazo 2.....	63
Tabla 13. Parámetros Denavit Hartenberg para el brazo 3.....	63
Tabla 14. Condiciones de posición de los tramos de la ilusión barrena.....	77
Tabla 15. Condiciones de posición de los tramos de la ilusión coriolis.....	77
Tabla 16. Condiciones de velocidad de los tramos de la ilusión barrena.....	78
Tabla 17. Condiciones de velocidad de los tramos de la ilusión coriolis.....	78
Tabla 18. Condiciones de aceleración y tiempo de los tramos de la ilusión barrena	79
Tabla 19. Condiciones de aceleración y tiempo de los tramos de la ilusión coriolis	79
Tabla 20. Condiciones de las combinaciones de la ilusión barrena	80
Tabla 21. Condiciones de las combinaciones de la ilusión coriolis	80
Tabla 22. Número de puntos de las combinaciones de la ilusión barrena.....	82
Tabla 23. Número de puntos de las combinaciones de la ilusión coriolis.....	82
Tabla 24. Valores de los coeficientes para el interpolador cúbico de barrena 1	83
Tabla 25. Valores de los coeficientes para el interpolador cúbico de barrena 2	83
Tabla 26. Valores de los coeficientes para el interpolador cúbico de barrena 3	83
Tabla 27. Valores de los coeficientes para el interpolador cúbico de coriolis 1	84
Tabla 28. Valores de los coeficientes para el interpolador cúbico de coriolis 2	84

Tabla 29. Parámetros de entrada y salida de la interfaz del modelo inverso.....	88
Tabla 30. Parámetros de entrada y salida de la interfaz del modelo directo	92
Tabla 31. Parámetros de entrada y salida de la interfaz de simulación	95
Tabla 32. Errores absolutos y porcentuales máximos entre el modelo inverso y ANSYS	100
Tabla 33. Errores porcentuales máximos ilusión coriolis 1 por interpolador cuadrático	109
Tabla 34. Errores porcentuales máximos ilusión coriolis 1 por interpolador cúbico.....	111
Tabla 35. Errores porcentuales máximos ilusión coriolis 1 por interpolador cuadrático	115

ÍNDICE DE FIGURAS

Figura 1. Sistema de ejes de la aeronave.....	10
Figura 2. Reacción de los canales semicirculares ante aceleraciones angulares.....	11
Figura 3. Reacción de los canales otolíticos ante aceleraciones lineales	11
Figura 4. Ilusión de pista.....	12
Figura 5. Ilusión de inversión.....	13
Figura 6. Ilusiones somatogiratorias	14
Figura 7. Estructura de la plataforma Stewart.....	16
Figura 8. Manipulador de 4 grados de libertad.	17
Figura 9. Estructura SIMPRO.	18
Figura 10. Manipulador 6 RUS.....	18
Figura 11. Modelado cinemático mediante velocidad de referencia.....	20
Figura 12. Movimientos sobre los ejes de la aeronave.	21
Figura 13. Representación de los ejes coordenados del simulador SIMPRO	25
Figura 14. Estructura de la máquina paralela Exechon.....	26
Figura 15. Representación estructural del manipulador 6 RUS.	27
Figura 16. Modelo cinemático del manipulador de dos grados de libertad	27
Figura 17. Esquema de control por espacio de estados.....	29
Figura 18. Control acoplado a la cinemática inversa	30
Figura 19. Control acoplado con lógica difusa	30
Figura 20. Esquema del filtro de washout.....	31
Figura 21. Interpolación de puntos en trayectoria.....	31
Figura 22. Esquema general del control cinemático	32
Figura 23. Representaciones típicas en el espacio cartesiano	33
Figura 24. Esquema cinemático con MS flight simulator.....	34
Figura 25. Simulación de vuelo en GUI MATLAB.....	35
Figura 26. Simulación de la plataforma Stewart en Solidworks	36
Figura 27. Simulación en Virtual Robot Simulator.....	36
Figura 28. Simulación MATLAB- ADAMS	37

Figura 29. Simulación en Unity 3D	37
Figura 30. Simulación con OpenGL	38
Figura 31. Geometría del manipulador R-3RPS	41
Figura 32. Planos de movimiento de los brazos.....	43
Figura 33. Coordenadas de (a) Plataforma giratoria (b) Base móvil para la cabina	46
Figura 34. (a) Manipulador 3RPS (b) Plataforma giratoria	48
Figura 35. Variables de trabajo del manipulador 3RPS.....	50
Figura 36. Vectores para posición de la plataforma giratoria	53
Figura 37. Variables de las articulaciones activas en manipulador R-3RPS	56
Figura 38. Variables para el manipulador R-3RPS por el método vectorial.....	59
Figura 39. Componentes de la longitud total de los brazos.	60
Figura 40. Manipulador R-3RPS graficado en Robotics de MATLAB.....	63
Figura 41. Esquema de funcionamiento de trayectorias con los modelos obtenidos.....	85
Figura 42. Variables de salida del actuador prismático con interpolador cuadrático	85
Figura 43. Variables de salida del actuador prismático con interpolador cúbico	86
Figura 44. Distribución de los parámetros en la pantalla de la interfaz inversa	88
Figura 45. Algoritmo de funcionamiento de la interfaz inversa	90
Figura 46. Interfaz inversa en MATLAB.....	91
Figura 47. Distribución de los parámetros en la pantalla de la interfaz directa	92
Figura 48. Algoritmo de funcionamiento de la interfaz directa	93
Figura 49. Interfaz directa en MATLAB	94
Figura 50. Distribución de los parámetros en la pantalla de la interfaz de simulación.....	95
Figura 51. Algoritmo de funcionamiento de la interfaz.....	97
Figura 52. Interfaz de simulación de movimiento en MATLAB	98
Figura 53. Gráfica del error absoluto en cada brazo hasta número de dato 90	100
Figura 54. Gráfica del error porcentual en cada brazo hasta número de dato 90.....	101
Figura 55. Gráfica del error absoluto en cada brazo a partir del número de dato 91	102
Figura 56. Gráfica del error porcentual en cada brazo a partir del número de dato 91.....	102
Figura 57. Espacio de trabajo del manipulador R-3RPS.....	103
Figura 58. Error porcentual método vectorial	104

Figura 59. Error porcentual Denavit Hartenberg	105
Figura 60. Gráfica con Robotics	106
Figura 61. Error porcentual en posición coriolis1 por interpolador cuadrático	107
Figura 62. Error porcentual en velocidad coriolis1 por interpolador cuadrático	108
Figura 63. Error porcentual en aceleración coriolis1 por interpolador cuadrático.....	108
Figura 64. Error porcentual en posición coriolis1 por interpolador cúbico	110
Figura 65. Error porcentual en velocidad coriolis1 por interpolador cúbico	110
Figura 66. Posición mediante los interpoladores y ANSYS	111
Figura 67. Velocidad mediante los interpoladores y ANSYS.....	112
Figura 68. Aceleración mediante los interpoladores y ANSYS.....	112
Figura 69. Error porcentual en posición barrena 1 por los dos interpoladores	113
Figura 70. Error porcentual en velocidad barrena 1 por los dos interpoladores	114
Figura 71. Error porcentual en posición barrena 1 por interpolador cuadrático	114
Figura 72. Posición barrena 1 mediante los interpoladores y ANSYS	116
Figura 73. Velocidad barrena 1 mediante los interpoladores y ANSYS.....	116
Figura 74. Aceleración barrena 1 mediante los interpoladores y ANSYS.....	117

RESUMEN

En el presente proyecto de titulación se realizó la obtención de forma analítica del modelo cinemático del simulador de desorientación espacial que se encuentra en desarrollo por el Centro de Investigación Científica y Tecnológica del Ejército (CICTE). El simulador mencionado se basa en la estructura de un manipulador paralelo R-3RPS, por lo que el modelo obtenido se fundamenta en la aplicación de métodos de resolución de cinemática inversa y directa para encontrar la posición, velocidad y aceleración en puntos de referencia del robot. El modelo cinemático inverso se encuentra mediante la metodología de los ángulos de Euler (Roll, Pitch, Yaw), mientras que el directo se lo calcula por dos métodos para posición, el primero es el cálculo geométrico de los puntos referenciales a partir de desplazamientos en los actuadores y el segundo es la convención de Denavit-Hartenberg, para velocidades y aceleraciones se trabaja con las derivadas de primer y segundo orden de la posición de ambos modelos. Mediante interfaces gráficas en MATLAB se realizó los cálculos de la posición, velocidad y aceleración de los puntos referenciales por cinemática inversa y directa, además se generó las trayectorias de las ilusiones a realizar por el simulador, se aplicó un control cinemático por interpolación de las mismas, con el uso de las ecuaciones de movimiento circular uniforme y variado, con la combinación de los modelos desarrollados se obtuvo los desplazamientos, velocidades, aceleraciones de los actuadores para las ilusiones de trabajo que comprueban los resultados de la simulación en ANSYS.

PALABRAS CLAVE:

- **CINEMÁTICA INVERSA**
- **INTERPOLACIÓN DE TRAYECTORIAS**
- **CINEMÁTICA DIRECTA**

ABSTRACT

In the present project, the analytical form of the kinematic model of the spatial disorientation simulator that is being developed by the Center for Scientific and Technological Research of the Army (CICTE). The aforementioned simulator is based on the structure of a R-3RPS parallel manipulator, so the model obtained is based on the application of direct and inverse kinematics resolution methods to find the position, speed and acceleration at robot reference points. The inverse kinematic model is found by the Euler angles methodology (Roll, Pitch, Yaw), while the direct one is calculated by two methods for position, the first is the geometric calculation of the referential points from displacements in the actuators and the second is the Denavit-Hartenberg convention, for speeds and accelerations we work with the derivatives of first and second order of the position of both models. Using graphical interfaces in MATLAB, the position, velocity and acceleration calculations of the referential points were made by inverse and direct kinematics, the trajectories of the illusions to be performed by the simulator were generated, the interpolation was applied, with the use of the equations of uniform and varied circular motion, with the combination of the models developed the displacements, speeds, accelerations of the actuators for the work illusions that verify the results of the simulation made in the software ANSYS were obtained.

KEYWORDS:

- **INVERSE KINEMATIC**
- **INTERPOLATION OF TRAJECTORIES**
- **DIRECT KINEMATIC**

CAPITULO I

1. INTRODUCCIÓN

1.1 Antecedentes

La desorientación espacial es un fenómeno que experimentan los pilotos de aeronaves, cuando no tienen referencias visuales del entorno, ni señales de los instrumentos de navegación, y la aceleración del sistema produce sensaciones por las cuales los pilotos pierden la noción sobre la posición e inclinación de la aeronave (OACI, 2012). Para tratar este fenómeno se desarrollan los simuladores de vuelo como una parte importante dentro del campo de la aeronáutica, para el entrenamiento de pilotos de aerolíneas comerciales y militares (Estevez, 2014), de tal forma de garantizar la seguridad y efectividad de los mismos antes de navegar en las aeronaves reales (Mauro, Gastaldi, Pastorelli, & Sorli, 2016). El funcionamiento de los simuladores de vuelo, consiste en reproducir determinadas condiciones de entorno para generar en el usuario una serie de sensaciones que resulten tan parecidas a la realidad como sea posible. Hay dos tipos de simuladores comúnmente usados que son los estáticos y dinámicos (Gastaldi, Pastorelli, & Sorli, 2009) (Limache, Rojas, & Murillo, 2010), en el primer caso el asiento del piloto es fijo y la generación de ilusiones se produce por medio de efectos visuales de entorno, mientras que los dinámicos utilizan un sistema de movimiento basados en estructuras robóticas, que permiten recrear una simulación de desorientación espacial lo más apegada a la realidad (Limache, Rojas, & Murillo, 2010).

En ámbito profesional se usan los simuladores dinámicos, que han evolucionado desde dispositivos basados en reacción a fuerzas aerodinámicas, como el de Sanders, creado en 1910 (Campos, de Faveri, Furtado, Reis, & Garcia, 2013) hasta entornos simulados en realidad virtual con el uso de manipuladores paralelos de 3 a 6 grados de libertad (López, 2008). Los robots paralelos de 6 grados de libertad dan la posibilidad generar movimientos con mayor rango angular en sus actuadores (González & Reinoso, 2011), el más representativo de estos manipuladores es la plataforma de Stewart, que se basa en 6 cadenas con juntas universales, prismáticas y esféricas, respectivamente (López, 2008), pero al tener mayor amplitud de movimiento angular, se producen posiciones donde pueden existir singularidades, que generan pérdida de movilidad en la articulaciones (Gastaldi, Pastorelli, & Sorli, 2009), por esta razón existen diseños basados en otras topologías de robots de igual o menor número de grados de libertad, y con menor rango angular de sus actuadores (Pastorelli & Battezzato, 2009), que permiten recrear las sensaciones de movimiento de un aeronave real, sin singularidades en su funcionamiento.

El modelado cinemático de estos manipuladores permite analizar singularidades y restricciones de movilidad del sistema robótico (Caetano, y otros, 2014), esto ayuda delimitar los rangos de funcionamiento del robot como factor fundamental para el manejo y control del movimiento a generar por los actuadores (Lyngé, Blanke, & Arne Johansen, 2015), en el caso de la desorientación espacial un requisito importante es utilizar métodos que ayuden a acercarse a las ecuaciones que gobiernan el movimiento de una aeronave real (Schopferer & Pfeifer, 2015), teniendo en cuenta a la fase cinemática como antesala al control dinámico que incluye el análisis con fuerzas y momentos inerciales (Arias, 2013). Los métodos cinemáticos que se emplean para modelar los movimientos de los manipuladores robóticos de simuladores de vuelo se basan en cinemática inversa como en la directa (Aristizabal & Giraldo, 2015).

Los modelos inversos se basan la localización de un punto referencia llamado centro de masa, mediante el análisis de traslaciones y rotaciones por los ángulos de Euler (roll, pitch, yaw) del punto en mención (Castañeda, 2013) (Oliveira & Encarnação, 2013), mientras la directa utiliza el planteamiento geométrico de puntos de referencia a partir de traslaciones y rotaciones de loa actuadores, otro método usado también es la convención de Denavit- Hartenberg (Asif, 2012) ,todos los métodos mencionados tanto inversos como directos, se ajustan a diferentes estructuras robóticas como la plataforma de Stewart (Arias, 2013) (Aristizabal & Giraldo, 2015), la 6 RUS (Rotacional-Universal-Esférica) (Campos, de Faveri, Furtado, Reis, & Garcia, 2013) y la 3 RPS (Rotacional-Prismática-Esférica) (Ospina, 2012), que son los manipuladores más utilizados para el desarrollo de este tipo de simulación.

El proyecto de titulación realizado presenta el modelo matemático y la simulación en MATLAB del mismo que representa el sistema cinemático en alto nivel del simulador de desorientación de vuelo que forma parte del proyecto que se desarrolla conjuntamente entre la Escuela de Aviación Naval y de la Universidad de las Fuerzas Armadas ESPE (CICTE, 2017).

1.2 Justificación

Actualmente el CICTE (Centro De Investigación Científica y Tecnológica Del Ejército) está a cargo del proyecto titulado “Construcción de un simulador de desorientación espacial para contribuir a la seguridad aérea y entrenamiento de los pilotos de las FF.AA” que involucra a investigadores de las carreras de Ingeniería Mecánica, Electrónica y Mecatrónica (CICTE, 2017). El proyecto en mención se encuentra dividido en áreas de desarrollo, donde una de estas ya tiene realizada la simulación del movimiento del manipulador robótico R-3RPS en el programa ANSYS que es la base del simulador en cuestión, el software de simulación permite el ingreso de desplazamientos, velocidades y aceleraciones de los actuadores activos del manipulador para

generar el movimiento a un alto nivel del mismo, y las salidas son las magnitudes cinemáticas mencionadas en las entradas pero en cualquier punto del manipulador, debido a que el simulador no se encuentra implementado, se requiere un sustento matemático que verifique la fiabilidad de los parámetros cinemáticos (posición, velocidad y aceleración en puntos referenciales del mecanismo) entregados por la simulación en ANSYS, la verificación mediante MATLAB produce la confiabilidad de los resultados de ANSYS, además de generar secuencias de puntos de control de las ilusiones dinámicas en el software como bases de pruebas del mecanismo real.

Es por esta razón que el desarrollo de este proyecto de titulación es de gran importancia ya que al finalizarlo se tendrá como producto final un algoritmo en MATLAB basado en el modelo matemático de la cinemática del robot R-3RPS, que permite encontrar los parámetros de posición (traslación y rotación), velocidades y aceleraciones finales ante una trayectoria definida, de cada actuador activo del simulador, la simulación en ANSYS será una guía para realizar pruebas al momento de desarrollar el modelo propuesto.

MATLAB es una herramienta que permite mediante sus paquetes como Simulink y GUI (Arias, 2013) (Aristizabal & Giraldo, 2015) (Alcocer, 2012) (Campos, de Faveri, Furtado, Reis, & Garcia, 2013) que facilitan la optimización de simulaciones que manejen sistemas de control cinemático de trayectorias y describan la cinemática de un robot, cada vez este software ofrece librerías más avanzadas para desarrollo de modelos simulados como es el caso de la simulación mecánica, por esta razón este software es útil y adecuado para el algoritmo del modelo planteado.

1.3 Alcance del Proyecto

Haciendo mención a la sección anterior, se necesita obtener el modelo matemático de la cinemática del simulador de desorientación espacial, teniendo en cuenta que el modelo en cuestión

se utilizará para la verificación de resultados obtenidos mediante la simulación realizada en el programa ANSYS.

Por lo que este proyecto de titulación se dividirá en las siguientes etapas:

- Estado del arte: Se recopilará los trabajos sobre métodos, modelos, simulaciones y software utilizados para representar la cinemática en simuladores dinámicos de vuelo, referenciado en los últimos 10 años.
- Modelado cinemático: Este proceso implica la deducción analítica del modelo, mediante métodos cinemáticos inverso y directo, adecuados para representar el movimiento del manipulador del simulador en cuestión.
- Simulación en MATLAB: Se plantea el desarrollo de un MATLAB que describa el modelo que permita la obtención de parámetros relevantes del simulador como: posición final (traslación eje X, Y, Z –rotación en ángulos pitch, yaw, roll), velocidad y aceleración de acuerdo a las trayectorias de movimiento especificadas por las ilusiones de trabajo del simulador.
- Comparación y análisis de resultados: Se verificará los resultados de ambas simulaciones tanto en ANSYS como en MATLAB, a partir de un control cinemático por interpolación de trayectorias de algunas ilusiones de trabajo del simulador, en caso de alguna diferencia notable entre estas se estudiará las opciones del error y se planteará los cambios necesarios, para rectificar los parámetros obtenidos.

La simulación en ANSYS fue facilitada en las instalaciones del CICTE en donde se trabajará de forma mutua para la realización del modelo cinemático.

1.4 Objetivos

1.4.1 Objetivo General

Modelar matemáticamente la cinemática del simulador de desorientación espacial, basado en un manipulador robótico paralelo R-3RPS, además de realizar un algoritmo en MATLAB que simule al modelo.

1.4.2 Objetivos Específicos

- Investigar sobre formas de obtención y simulación de modelos matemáticos que describan la cinemática de simuladores dinámicos de vuelo.
- Modelar la cinemática de simulador mediante el análisis matemático con el uso de métodos cinemáticos inversos y directos.
- Realizar un algoritmo del modelo matemático en MATLAB que permita la obtención y visualización de parámetros cinemáticos relevantes del simulador de vuelo.
- Comparar resultados obtenidos del modelo matemático con los datos de la simulación realizada en ANSYS, para hacer un análisis de parámetros en ambos casos.

CAPITULO II

2. ESTADO DEL ARTE

En este capítulo se revisa conceptos previos de desorientación espacial, investigaciones sobre métodos de modelado cinemático de los manipuladores utilizados para el desarrollo de simuladores dinámicos de desorientación espacial, además de formas de control cinemático a un alto nivel por interpolación de trayectorias para manipuladores destinados a este tipo de simulación, todo referenciado a los últimos 10 años.

2.1 Conceptos Previos

2.1.1 Desorientación espacial

La desorientación espacial se define como la pérdida de noción de una persona para ubicarse en un contorno tridimensional con respecto al horizonte terrestre, debido a una percepción errónea de la condición real (Cunliffe, 2009). En la aviación se define como la falta de percepción del piloto para detectar correctamente la posición, el movimiento y la altitud de su aeronave o de sí mismo dentro del sistema de coordenadas proporcionadas por la superficie de la tierra y la gravedad gravitacional, producida por una interpretación errónea de los parámetros de los instrumentos de control y navegación de la aeronave, ante una ilusión sensorial que lleva a diferencias entre las indicaciones del instrumento y lo que el piloto siente (Bozan, 2015).

En los seres humanos este fenómeno es percibido por el Sistema de Orientación y Equilibrio que es manejado por el Sistema Nervioso Central (SNC), que recepta y procesa la información percibida de una serie de aparatos sensoriales periféricos, entre los que sobresalen los aparatos visual, vestibular, auditivo y propioceptivo. Estos sistemas son los encargados de percibir las

condiciones no respetadas por las condiciones reales ni las fuerzas gravito-inerciales de la aviación (Escuela de Aviación Naval, 2016).

2.1.2 Clasificación de la desorientación espacial

Existen tres tipos de desorientación espacial que son: Tipo I (no reconocido), Tipo II (reconocido) y Tipo III (incapacitante).

2.1.2.1 Tipo I: Desorientación espacial no reconocida

El piloto no percibe ninguna sensación de desorientación espacial, sea natural o artificial, además piensa que la orientación y posición de la aeronave es correcta (Bozan, 2015). Esta sensación es la más temida, la más frecuente y la menos estudiada. La componente visual es la principal causa de esta desorientación, por esta razón es difícil distinguirla (Escuela de Aviación Naval, 2016).

2.1.2.2 Tipo II: Desorientación espacial reconocida

El piloto percibe sensación de desorientación espacial, puede tener un conflicto entre la orientación natural con la artificial, por diferencias sensoriales entre lo que está haciendo la aeronave con lo que le dicen los instrumentos (Escuela de Aviación Naval, 2016) (Bozan, 2015). Esta sensación es la más conocida y estudiada, en este caso este tipo de desorientación se produce por las ilusiones percibidas por el aparato vestibular o inconsistencia entre la información vestibular y visual (Cunliffe, 2009).

2.1.2.3 Tipo III: Desorientación espacial incapacitante

El piloto percibe la sensación de pérdida de orientación pero también pierde la capacidad de controlar la aeronave. Puede tener nistagmos (movimiento horizontal automático de los ojos) demasiado fuerte que le impide visualizar los indicadores de los instrumentos de navegación (Escuela de Aviación Naval, 2016) (Bozan, 2015), combinado con el mareo provocado por el

movimiento brusco de la aeronave, induce vértigo al piloto lo que impide una reacción ante estas condiciones. Este tipo de sensación es poco frecuente (Cunliffe, 2009).

La desorientación espacial tipo II es la más estudiada, debido a que permite recrear ilusiones sensoriales que estimulan al sistema vestibular y visual, con el uso de manipuladores robóticos y ambientes visuales, que reproducen las sensaciones de una aeronave real, que ayudan a los pilotos a entrenarse para solventar las sensaciones producidas por este fenómeno, en base a este tipo de desorientación se diseñan los simuladores dinámicos de vuelo (Escuela de Aviación Naval, 2016).

2.1.3 Ilusiones de vuelo

Las ilusiones de vuelo son los errores generados por los aparatos periféricos, entre la percepción y la realidad durante las fases de vuelo, los sistemas más sensibles a estos estímulos son el visual y el vestibular que se encuentra a la altura del oído interno (Escuela de Aviación Naval, 2016).

Dentro de la aviación, para entrar al campo de las ilusiones de vuelo, se requiere reconocer los ejes de rotación y traslación de la aeronave, para la rotación se define los términos de pitch o cabeceo (rotación alrededor del eje y), roll o alabeo (rotación alrededor del eje x) y yaw o guiñada (rotación alrededor del eje z), para la traslación los ejes cartesianos x, y, z, toman el nombre de eje longitudinal, lateral y vertical respectivamente (Garcia, Pereda, Giron-Sierra, & Espinosa, 2012) (Peddle, 2005) (Schopferer & Pfeifer, 2015), a partir de estos conceptos se basan las aeronaves para realizar los respectivos movimientos en el espacio, en la Figura 1 se muestra la ubicación de los ejes en la aeronave.

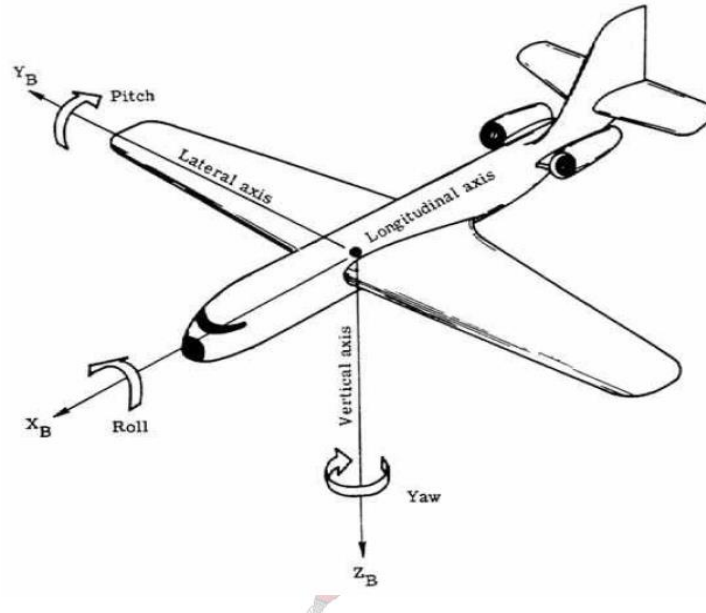


Figura 1. Sistema de ejes de la aeronave
Fuente: (Peddle, 2005)

La desorientación espacial es percibida por varios sistemas que actúan como sensores en el cuerpo humano, uno de estos aparatos es el vestibular que detecta el movimiento angular dependiendo de ciertos límites de estimulación la cual proporcional a la aceleración angular y la duración de esta magnitud, esta relación se la conoce como ley de Mulder, la expresión que relaciona esta ley es $\alpha\tau = k$, que indica que la magnitud de la aceleración angular, α , multiplicada por la duración de la aceleración aplicada, τ , debe ser igual a una constante especificada, k . Esta constante se llama constante de Mulder (Bozan, 2015).

Varias investigaciones sitúan a la constante de Mulder en un valor de 2.5 °/s, en los tres ejes de rotación de la aeronave, para que el sistema vestibular perciba una desorientación espacial, en la cual la multiplicación de aceleración con la duración de la misma debe aproximarse a dicho valor (Small, Wickens, Oster, Keller, & French, 2004) (Patterson & Chandler, 2010). Dentro del aparato vestibular se encuentran dos estructuras que perciben sensaciones una son los canales semicirculares que detectan la aceleración angular (Figura 2) y la otra son los canales otolíticos

que son sensibles la aceleración lineal, en la Figura 3, se muestra cómo reacciona el estos canales ante varias estimulaciones.

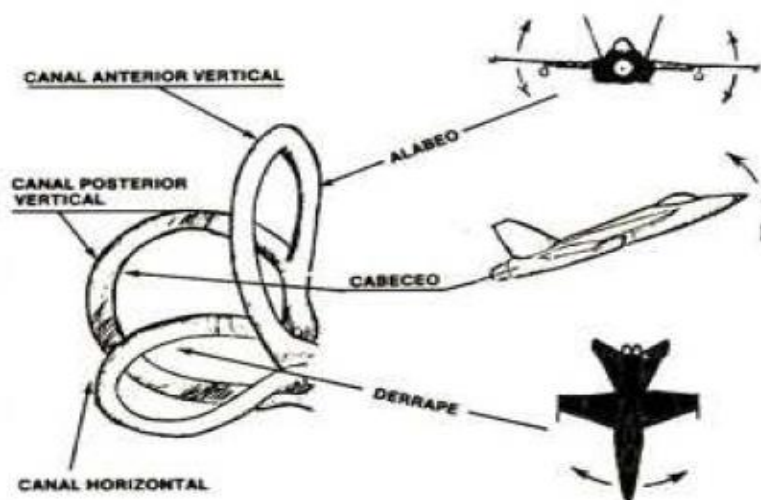


Figura 2. Reacción de los canales semicirculares ante aceleraciones angulares.

Fuente: (Arias, 2013)

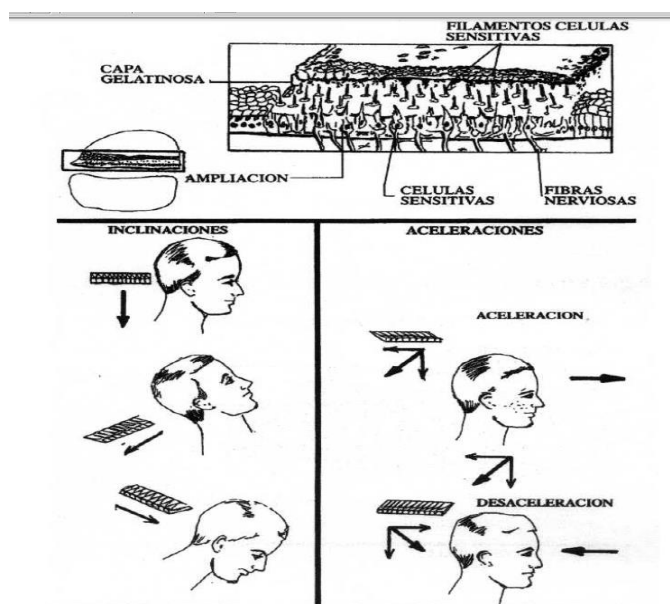


Figura 3. Reacción de los canales otolíticos ante aceleraciones lineales

Fuente: (Escuela de Aviación Naval, 2016)

De acuerdo a las condiciones fisiológicas mencionadas, se clasifican a las ilusiones en visuales y vestibulares.

2.1.3.1 Ilusiones de visuales

Las ilusiones visuales son generadas por el sistema visual central y periférico, dentro de las más importantes se encuentran, la ilusiones de pista, terreno, falso horizonte luminosidad, aproximación y de hoyo negro (Escuela de Aviación Naval, 2016). Las ilusiones de pista, terreno y aproximación se refieren a la perspectiva lineal de un plano inclinado, además de distinguir el ancho del mismo. En las ilusiones de luminosidad y de hoyo negro la visibilidad del piloto depende del nivel de intensidad de luz natural o artificial al que encuentra expuesto. En la Figura 4 se muestra un ejemplo de ilusión visual de pista.

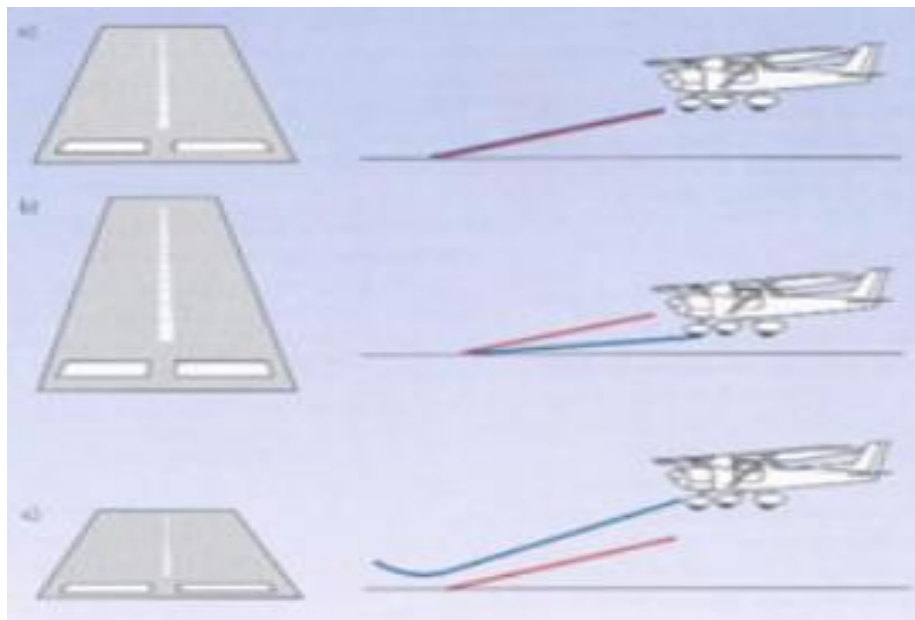


Figura 4. Ilusión de pista
Fuente: (Cantón Romero, 2016)

2.1.3.2 Ilusiones vestibulares

Las ilusiones vestibulares son originales por los canales semicirculares y órganos otolíticos dependiendo del tipo de aceleración a la que se encuentre expuesto el piloto, por esta razón se subdividen en ilusiones somatogiratorias (dependiente de los canales semicirculares) y somatográvicas (dependiente de los órganos otolíticos) (Cunliffe, 2009).

2.1.3.2.1 Ilusiones somatográficas

Este tipo de ilusión se genera por aceleraciones y desaceleraciones lineales que producen desplazamientos inerciales opuestos en la membrana otolítica, entre las más destacadas se encuentran: la ilusiones de inversión, oculográfica y de ascensor (Bozan, 2015).

La ilusión de inversión se produce al nivelar el vuelo de manera brusca a una altura deseada durante el ascenso formando una parábola soportando aceleraciones lineales verticales de -1 a +1 Gravedades (G) (Escuela de Aviación Naval, 2016), como se observa en la Figura 5. La ilusión oculográfica se produce cuando se aplica una aceleración lineal con la aeronave nariz arriba que crea sensación de elevación en el piloto, de igual manera con la aeronave nariz abajo se crea una sensación de depresión. La ilusión de ascensor se produce en un vuelo nivelado sometido a una aceleración vertical de forma brusca, genera el ascenso y descenso de la aeronave (Cunliffe, 2009).



Figura 5. Ilusión de inversión

Fuente: (Cantón Romero, 2016)

2.1.3.2.2 Ilusiones somatogiratorias

Se producen por la estimulación de los canales semicirculares mediante movimientos rotatorios constantes, que generan aceleraciones angulares en la ubicación del piloto, entre las más destacadas se encuentran la barrena, coriolis y ladeos (Bozan, 2015).

La ilusión barrena se produce por viraje e inclinación alar constante de forma descendente, por un tiempo en que se estimula los canales semicirculares del piloto mediante aceleraciones angulares en los tres ejes rotatorios de la aeronave (roll, pitch, yaw), generadas por el movimiento descrito (Escuela de Aviación Naval, 2016). La ilusión coriolis se genera por la estimulación de dos pares de canales semicirculares lo que crea aceleraciones angulares en roll y yaw, produciendo la sensación de un movimiento alar giratorio ascendente o en picada (Cunliffe, 2009). Los ladeos se producen por un movimiento alar hacia la derecha o izquierda de la aeronave, que genera una estimulación de un par de canales semicirculares, por medio de una aceleración angular en roll (Escuela de Aviación Naval, 2016), en la Figura 6 se observa las ilusiones somatogiratorias.

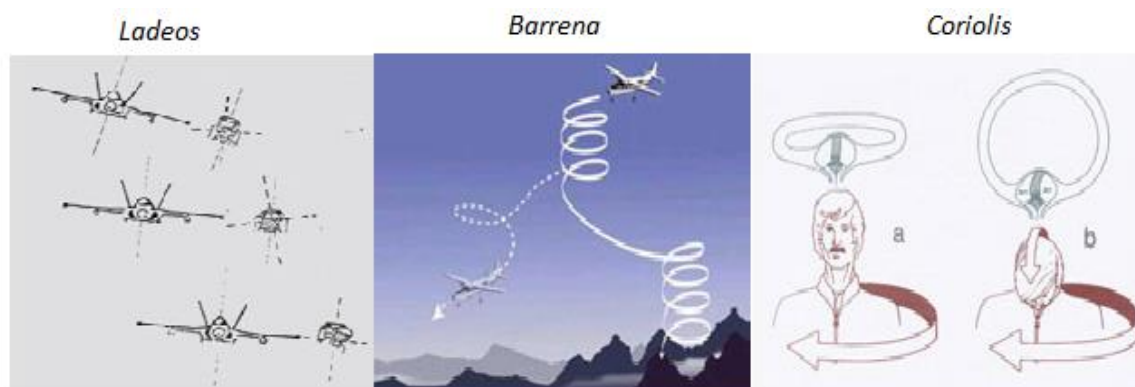


Figura 6. Ilusiones somatogiratorias
Fuente: (Cantón Romero, 2016)

2.2 Simuladores de desorientación espacial

En el ámbito de aviación los simuladores de desorientación espacial se los denomina como de vuelo, que varios trabajos los definen como una combinación de software y hardware, que reproducen una réplica fiel, precisa, semejante a la realidad del comportamiento de una aeronave en el espacio (Limache, Rojas, & Murillo, 2010), esto permite entrenar a los pilotos profesionales condiciones y parámetros de vuelo en tiempo real con el uso de computadoras de alto rendimiento (Mauro, Gastaldi, Pastorelli, & Sorli, 2016). Esta herramienta permite a los pilotos adquirir

conocimiento y experiencia necesaria para solucionar circunstancias producidas por desorientación espacial, condiciones climáticas y perturbaciones similares a las generadas en aeronaves reales (Limache, Rojas, & Murillo, 2010). Los simuladores de vuelo han evolucionado desde una aeronave sobre una articulación universal que reacciona con la presencia de viento, para la orientación e inclinación de la misma desarrollado por Sanders Teachers (Campos, de Faveri, Furtado, Reis, & Garcia, 2013), hasta softwares de simulación públicos que permiten visualizar las condiciones de vuelo de manera virtual. Se han clasificado a los simuladores de vuelo en dos grupos que son estáticos y dinámicos (Gastaldi, Pastorelli, & Sorli, 2009).

Los simuladores estáticos se basan en un asiento de piloto fijo, mientras que los dinámicos actúan mediante un sistema de movimiento generado por plataformas robóticas, en este caso se utiliza una réplica de la cabina del piloto, para recrear los movimientos de un avión real (Gastaldi, Pastorelli, & Sorli, 2009). Los simuladores dinámicos se complementan con simuladores virtuales que potencian la componente visual del movimiento, los cuales son de acceso público y profesional (Limache, Rojas, & Murillo, 2010).

Los simuladores de vuelo públicos son de código abierto y multiplataforma entre los destacados se encuentran X-Plane, Flight-Simulador y FlightGear (Limache, Rojas, & Murillo, 2010), que proporcionan una variedad de modelo de aviones, aeropuertos e interfaces de control, mientras los profesionales cumplen estándares y certificados para ser usados en el entrenamiento de pilotos, razón por la cual son pocas empresas las encargadas de desarrollar estos softwares, entre las más importantes se encuentran Rockwell-Collins, CAE y Flight-Safety-International (Limache, Rojas, & Murillo, 2010), la elección de cualquiera de estos programas depende de aplicación del simulador. En el ámbito profesional se utilizan los simuladores de vuelo dinámicos, por esta razón

es importante revisar los trabajos sobre los manipuladores robóticos que son la base de estos simuladores de vuelo.

En el ámbito profesional la simulación dinámica de vuelo se desarrolla en base a los robots espaciales, de 3 grados de libertad en adelante, debido a que permiten movimiento en espacios tridimensionales mediante la traslación y rotación en los tres ejes coordenados (López, 2008). En simuladores de vuelo profesionales el manipulador robótico mayormente usado es la plataforma Stewart (Pastorelli & Battezzato, 2009), el cual consiste en una base móvil unida mediante articulaciones esféricas a tres actuadores prismáticos, que se conectan a una base fija por articulaciones universales, este mecanismo es versátil, tiene un amplio rango de movilidad y presenta un posicionamiento de alta precisión (González & Reinoso, 2011). La estructura de esta plataforma se observa en la Figura 7.

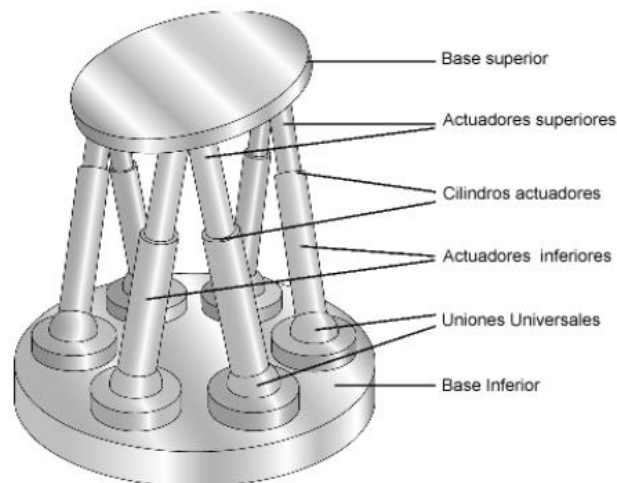


Figura 7. Estructura de la plataforma Stewart

Fuente: (González & Reinoso, 2011)

A parte de la plataforma Stewart, varios trabajos detallan otros manipuladores para el desarrollo de simuladores de vuelo, como lo plantearon Pouliot y Gosselin que demostraron satisfactoriamente que se pueden utilizar mecanismos de menor número de grados de libertad

(Pastorelli & Battezzato, 2009) (Gastaldi, Pastorelli, & Sorli, 2009), lo que implica diseños de simuladores a un costo bajo en comparación a los manipuladores de 6 grados de libertad manteniendo una alta precisión, rigidez y rendimiento (Gastaldi, Pastorelli, & Sorli, 2009), como se detalla a continuación.

En (Kvrgic, Visnjic, Cvijanovic, Divnic, & Mitrovic, 2015) y (Dančuo, Kvrgić, Rašuo, & Vidaković, 2013) se diseña un simulador de vuelo en base a un manipulador de cuatro grados de libertad, como se observa en la Figura 8, con ejes giratorios, para lograr simular diferentes orientaciones, fuerzas de aceleración que actúan sobre el piloto, propone un algoritmo de control que mejora la calidad, seguridad del movimiento, la precisión de posición y se reduce los errores en comparación a la eficacia de la plataforma Stewart.

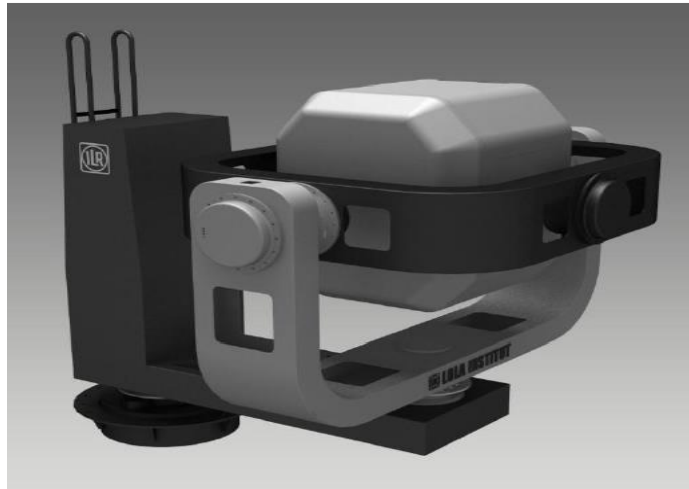


Figura 8. Manipulador de 4 grados de libertad.
Fuente: (Kvrgic, Visnjic, Cvijanovic, Divnic, & Mitrovic, 2015)

En (Hernández, Rubio, Prieto, Hernández, & Izaguirre, 2011) se realiza un simulador de desorientación espacial en base a una estructura SIMPRO la cual es un manipulador robótico paralelo de 3 grados de libertad, que se encuentra compuesto al igual que la plataforma de Stewart por dos bases móvil y fija unidas por una arquitectura RPSU-2PS, como se observa en la Figura 9,

esta configuración admite un rango angular bajo que bordea los 17° en roll y pitch, pero para la especificaciones de diseño de este simulador es suficiente.



Figura 9. Estructura SIMPRO.

Fuente: (Hernández, Rubio, Prieto, Hernández, & Izaguirre, 2011)

La plataforma Stewart es el manipulador de 6 grados de libertad más usado para simuladores dinámicos de vuelo, pero también hay otra topología de igual número de grados, que es utilizada, la cual es la estructura 6 RUS (Figura 10), que tiene articulaciones activa rotativas, lo que produce en cada cadena del mecanismo, la posibilidad de intercambio de las articulaciones universales por las esféricas sin ningún tipo de efecto en las características del manipulador (Campos, de Faveri, Furtado, Reis, & Garcia, 2013), esta es la segunda arquitectura que se utiliza para simuladores de movimiento.

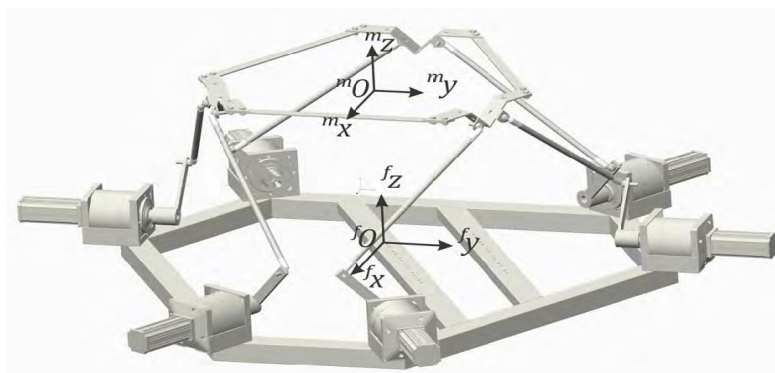


Figura 10. Manipulador 6 RUS

Fuente: (Campos, de Faveri, Furtado, Reis, & Garcia, 2013)

Las estructuras utilizadas dependen de las características de movimiento deseado, especificaciones del simulador y de las restricciones de movilidad planteadas en los requerimientos.

2.3 Métodos para modelar la cinemática de los simuladores de vuelo

Para el desarrollo de un algoritmo de simulación dinámica del vuelo, se comienza por el análisis de las restricciones cinemáticas del simulador de vuelo a partir del manipulador robótico base, tomando en cuenta las longitudes y rotaciones de los actuadores (Matlalcuatzi, y otros, 2014), razón por la cual se requiere conocer los métodos utilizados para representar la cinemática de una aeronave, como se detalla a continuación.

2.3.1 Velocidad de referencia

La cinemática de los simuladores de vuelo parte de los modelos cinemáticos que gobiernan las ecuaciones de movimiento de una aeronave, que salen de dos opciones, la primera es la ubicación del centro de masa del avión y la segunda es mediante una velocidad de referencia distribuida a los ejes de traslación del avión (Garcia, Pereda, Giron-Sierra, & Espinosa, 2012) (VanEykeren & Chu, 2014). La primera opción se basa en métodos matriciales que se detallan más adelante, mientras la segunda opción se enfoca en una velocidad de referencia como guía de la aeronave, esta magnitud se descompone alrededor de los ejes de traslación y de rotación en velocidades relativas, en diferentes puntos alrededor del horizonte terrestre en la Figura 11 se observa la distribución alrededor del eje longitudinal (Peddle, 2005) (Schopferer & Pfeifer, 2015).

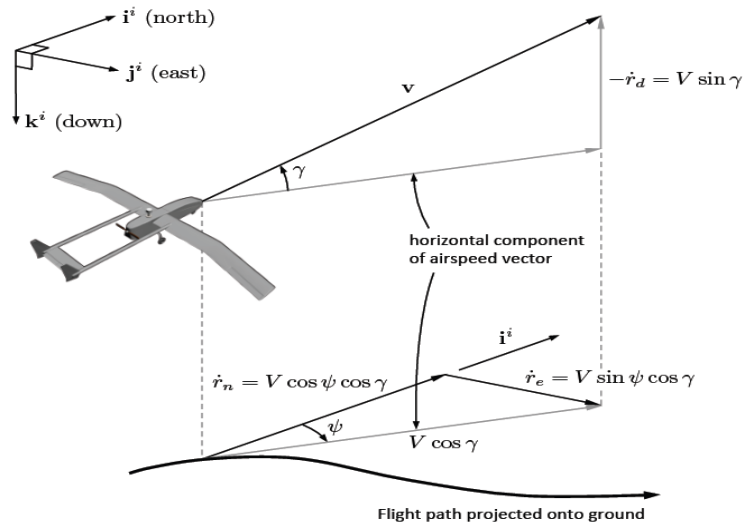


Figura 11. Modelado cinemático mediante velocidad de referencia.

Fuente: (Beard & McLain, 2013)

2.3.2 Ángulos de Euler

A partir de las componentes relativas de velocidad alrededor se realiza conversiones en velocidades lineales y angulares, con transformaciones de sistemas coordenados (Schopferer & Pfeifer, 2015), lo que lleva a explicar la primera opción de modelo cinemático basado en multiplicación matricial que se denomina ángulos de Euler, está metodología permite la orientación de un sistema de coordenadas tridimensional en relación a otro mediante secuencia de rotaciones alrededor de un eje a la vez determinados por los ángulos (roll(Φ), pitch(θ), yaw(ψ)) (Limache, Rojas, & Murillo, 2010). Las rotaciones de una aeronave se definen alrededor de los ejes longitudinal, lateral y vertical, respetando las siguientes condiciones:

La rotación alrededor del eje longitudinal o alabeo (Φ). El sentido positivo es cuando el ala derecha del avión se mueve hacia abajo, mientras el sentido negativo es con el ala izquierda (Limache, Rojas, & Murillo, 2010) (Pastorelli & Battezzato, 2009).

La rotación alrededor del eje lateral o cabeceo (θ). El sentido positivo es cuando la nariz del avión se mueve hacia arriba y el sentido negativo es cuando se mueve nariz abajo (Limache, Rojas, & Murillo, 2010) (Pastorelli & Battezzato, 2009).

La rotación alrededor del eje vertical o guiñada (ψ). El sentido positivo es cuando el avión gira hacia la derecha y negativo cuando gira a la izquierda (Limache, Rojas, & Murillo, 2010) (Pastorelli & Battezzato, 2009) (Salehinia, Najafi, Sadati, Shiee, & Salehinia, 2013). En la Figura 12 se muestra el sentido de las rotaciones en los tres ejes de manera visual.

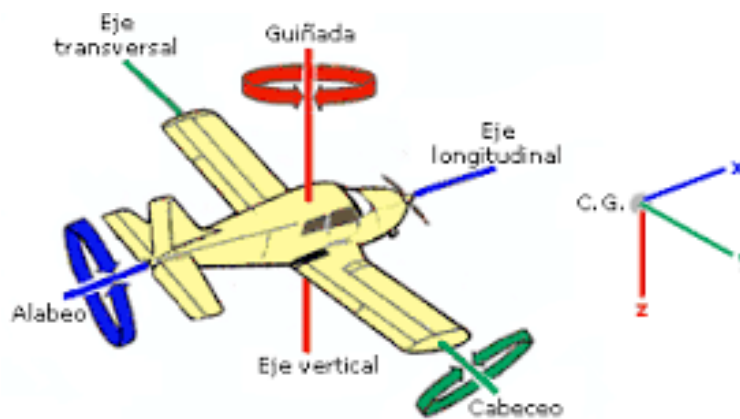


Figura 12. Movimientos sobre los ejes de la aeronave.

Fuente: (González, 2008)

Con las rotaciones descritas se formulan las siguientes expresiones matriciales denotadas con R (ángulo de Euler):

$$R(\phi) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\phi & -\text{sen}\phi \\ 0 & \text{sen}\phi & \cos\phi \end{bmatrix} \quad (1)$$

$$R(\theta) = \begin{bmatrix} \cos\theta & 0 & -\text{sen}\theta \\ 0 & 1 & 0 \\ \text{sen}\theta & 0 & \cos\theta \end{bmatrix} \quad (2)$$

$$R(\psi) = \begin{bmatrix} 0 & 0 & 0 \\ \cos\psi & -\text{sen}\psi & 0 \\ \text{sen}\psi & \cos\psi & 1 \end{bmatrix} \quad (3)$$

Mediante la multiplicación de las matrices (3), (2), (1) con esa secuencia se obtiene la matriz de rotación RPY (Roll, Pitch, Yaw) o ángulos de Euler, denotada por $R(\psi, \theta, \phi)$ como se observa a continuación:

$$R(\psi, \theta, \phi) = \begin{bmatrix} c\psi c\theta & c\psi s\theta s\phi - s\psi c\phi & c\psi s\theta c\phi + s\psi s\phi \\ s\psi c\theta & s\psi s\theta s\phi + c\psi c\phi & s\psi s\theta c\phi - c\psi s\phi \\ s\theta & -c\theta s\phi & c\theta c\phi \end{bmatrix} \quad (4)$$

En (4) c y s representan coseno y seno respectivamente, la matriz descrita es la base para el modelado cinemático de simuladores de vuelo a partir de un punto de referencia denotado como centro de masa o gravedad en términos de la aeronave real.

En (Caetano, y otros, 2013) (Oliveira & Encarnação, 2013) (Lynge, Blanke, & Arne Johansen, 2015) (Castañeda, 2013) realizan la aplicación de los ángulos de Euler en pasos de tiempo dentro de un campo inercial, con el objetivo de validar las ecuaciones generales de movimiento de un avión relacionando roll, pitch, yaw con alabeo, cabeceo y guiñada, que pueden ser ángulos mayores a 90° , además se obtienen la velocidades y aceleraciones mediante medidas de actitud discretas que después se diferencian usando las posiciones obtenidas por los ángulos de rotación y las traslaciones corporales con respecto al tiempo.

En varios trabajos realizados sobre simuladores de vuelo dinámicos este es el método que se aplica en los manipuladores robóticos, como cinemática inversa, comenzando con la ubicación del punto central en la base móvil del mecanismo, que tienen diferentes formas geométricas de contorno, el principio utilizado para como complemento a esta metodología es la suma vectorial desde el punto medio de la base fija hasta los vértices de la parte móvil pasando por el punto de referencia, a continuación se menciona algunas referencias en las cuales se utiliza a este método como cinemática inversa ya que parte del efector final como dato de entrada y los datos de las articulaciones del robot son salidas.

En (Arias, 2013) , (Bingul & Karahan, 2012) y (Aristizabal & Giraldo, 2015) se modela el movimiento con una estructura robótica, a pequeña escala basada en la plataforma Stewart, con bases triangulares, el análisis inverso se realiza mediante la metodología planteada de los ángulos de Euler y algebra vectorial para encontrar la posición angular y lineal de los actuadores a partir de movimientos del efector final o punto de referencia. En (López, 2008) y (Hernández, Rubio, Prieto, Hernández, & Izaguirre, 2011) se analiza la cinemática inversa con una variación de los ángulos de Euler, con la secuencia R (θ, Φ, ψ) y la distribución del movimiento del punto de referencia móvil de base pentagonal hacia los puntos de contacto de la articulaciones esféricas de la estructura robótica SIMPRO, en (Campos, de Faveri, Furtado, Reis, & Garcia, 2013) se modela la cinemática inversa del manipulador 6 RUS, a partir de la aplicación del método de la matriz RPY, en bases triangulares de la plataforma.

La desventaja de esta metodología radica en la presencia de singularidades que producen pérdida o ganancia de movilidad en los manipuladores robóticos debido a que este modelo trabaja con las funciones trigonométricas seno y coseno, que en ángulos críticos como 90° , 180° , 270° y 360° , principalmente en los ángulos roll y pitch, que generan valores indeterminados o infinitos, que afectan al movimiento del mecanismo, en el caso del ángulo yaw no hay restricciones para comportamiento dentro de este método, por lo cual puede tomar cualquier valor angular (Bingul & Karahan, 2012). Para solventar este problema se plantea el método de los cuaterniones.

2.3.3 Cuaterniones

La metodología de los cuaterniones se basa en sustituir los ángulos de Euler por magnitudes constituidas por senos y cosenos directores de ángulos medios, denotados por la letra q con su respectivo subíndice, gobernadas por las siguientes expresiones (Chablat, Jha, Rouillier, & Moroz, 2014) (Krause, Zapico, & Rodríguez, 2012):

$$q_1 = \cos \frac{\theta}{2} \cos \frac{(\phi + \psi)}{2} \quad (5)$$

$$q_2 = \sin \frac{\theta}{2} \cos \frac{(\phi - \psi)}{2} \quad (6)$$

$$q_3 = \sin \frac{\theta}{2} \sin \frac{(\phi - \psi)}{2} \quad (7)$$

$$q_4 = \cos \frac{\theta}{2} \sin \frac{(\phi + \psi)}{2} \quad (8)$$

Reemplazando (5), (6), (7) y (8) en la matriz RPY definida en (4) se obtiene la expresión R en función de los términos de q_1 , q_2 , q_3 , q_4 , como se muestra a continuación:

$$R(q) = \begin{bmatrix} 2q_1^2 + 2q_2^2 - 1 & -2q_1q_4 + 2q_2q_3 & 2q_1q_3 + 2q_2q_4 \\ 2q_1q_4 + 2q_2q_3 & 2q_1^2 + 2q_3^2 - 1 & -2q_1q_2 + 2q_3q_4 \\ -2q_1q_3 + 2q_2q_4 & 2q_1q_2 + 2q_3q_4 & 2q_1^2 + 2q_4^2 - 1 \end{bmatrix} \quad (9)$$

En (9) se obtiene la matriz de rotación RPY en función de los cuaterniones, este método permite eliminar las singularidades de movimiento, a partir del uso de 4 variables que significan dimensiones (Ran & Chuangchuang, 2015) (Lombaerts, Looye, Chu, & Mulder, 2012), debido que el método de los ángulos de Euler trabaja en un campo tridimensional, los cuaterniones eliminan las restricciones de movilidad ante puntos críticos por su espacio de 4 dimensiones, que indica que entra en el campo de los números complejos (Caetano, y otros, 2014). Esta metodología tiene una desventaja, la cual consiste en tener componentes redundantes, ya los cuaterniones no identifican el sentido de una magnitud, requieren de algoritmos predictores que complementen el método para solucionar este inconveniente (Chablat, Jha, Rouillier, & Moroz, 2014).

2.3.4 Métodos Directos

Los métodos cinemáticos directos permiten encontrar la posición y orientación del efector final, a partir de las variables articulares como entradas del modelo, en manipuladores paralelos presentan

soluciones complejas (Villarroel, 2011), en estructuras cerradas como las paralelas el análisis cinemático directo es un cálculo sumamente complicado (Arias, 2013) (Izaguirre, 2012), las metodologías mayormente utilizadas son las geométricas complementadas con algebra vectorial parecido a la cinemática inversa y la convención de Denavit Hartenberg. Varios trabajos utilizan este modelo como base de pruebas de los actuadores y articulaciones, a continuación se detalla las funciones de la cinemática directa dentro del modelado de movimiento de aeronaves.

En (López, 2008) se realiza el análisis directo como parte de verificación de la cinemática inversa, mediante la aplicación de la distancia entre dos puntos y algebra de segmentos denotada en (10), (11), (12) por R_i , para cada uno de los 3 brazos del simulador SIMPRO (Figura 13), el uso de esta metodología permite encontrar el desplazamiento de los actuadores prismáticos que son las articulaciones activas del manipulador, de las cuales depende la movilidad del mecanismo, para los cuales se utilizan válvulas proporcionales o electroválvulas para emular este movimiento.

$$R_1 = \overline{A_1O} + \overline{PB_1} - \overline{A_1C_1} \quad (10)$$

$$R_2 = \overline{A_2O} + \overline{PB_2} - \overline{A_2C_2} \quad (11)$$

$$R_3 = \overline{A_3O} + \overline{PB_3} - \overline{A_3C_3} \quad (12)$$

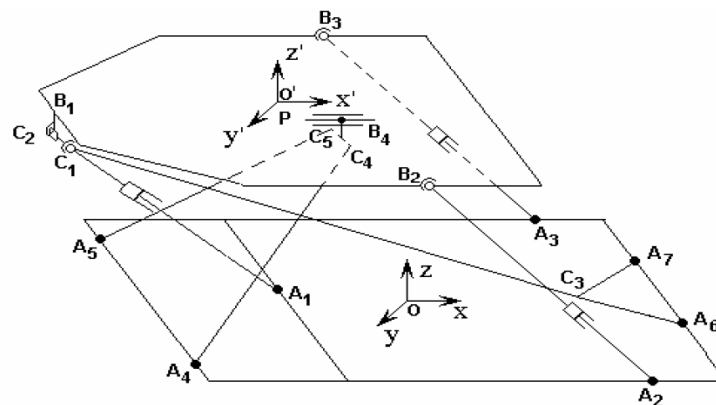


Figura 13. Representación de los ejes coordenados del simulador SIMPRO

Fuente: (López, 2008)

En (Bi & Jin, 2011), se realiza un análisis igual al del trabajo mencionado anteriormente pero se basa en una estructura de maquina paralela Exechon (Figura 14), que funciona como simulador de vuelo.

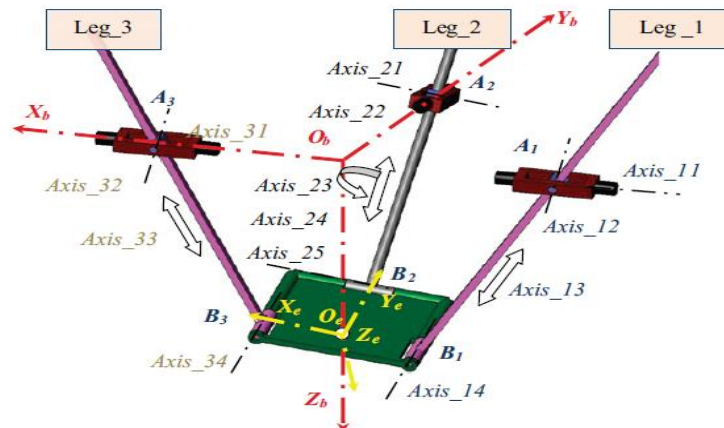


Figura 14. Estructura de la máquina paralela Exechon.

Fuente: (Bi & Jin, 2011)

En (Asif, 2012) se analiza la cinemática directa de una variante de la plataforma Stewart (Figura 15), mediante la convención de Denavit Harterberg, que maneja cuatro parámetros (θ , d , α , a) para representar el movimiento de las articulaciones con el objetivo de mover el efector final mediante el uso de la multiplicación de matrices homogéneas, descritas individualmente en (13) y como conjunto en (14), se utiliza este modelo para representar la cinemática del manipulador robótico y con una simulación virtual mover la plataforma real. En este caso utiliza tanto la inversa como la directa para cumplir el objetivo.

$$T = \begin{bmatrix} \cos\theta_i & -\cos\alpha_i \sin\theta_i & \sin\alpha_i \sin\theta_i & a \cos\alpha_i \\ \sin\theta_i & \cos\alpha_i \cos\theta_i & -\sin\alpha_i \cos\theta_i & a \sin\alpha_i \\ 0 & \sin\alpha_i & \cos\alpha_i & d \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (13)$$

$$T_{M1}^{G1} = T_{B1}^{G1} \chi T_{G1}^{B1} \quad (14)$$

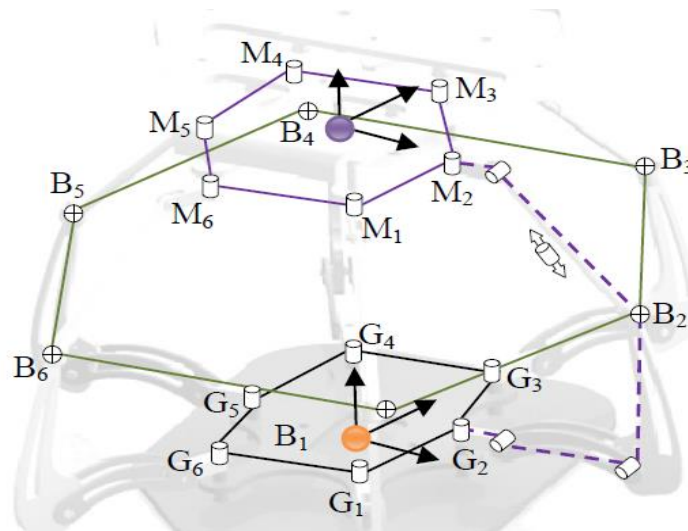


Figura 15. Representación estructural del manipulador 6 RUS.

Fuente: (Asif, 2012)

En (Lukács & Lantos, 2012) se obtiene el modelo no lineal de una aeronave mediante la adquisición de señales del movimiento de un manipulador de dos grados de libertad (Figura 16), que funciona como columna vertical móvil, que se controla mediante variables de estado y el modelado cinemático de la estructura robótica por la convención de Denavit Hartenberg.

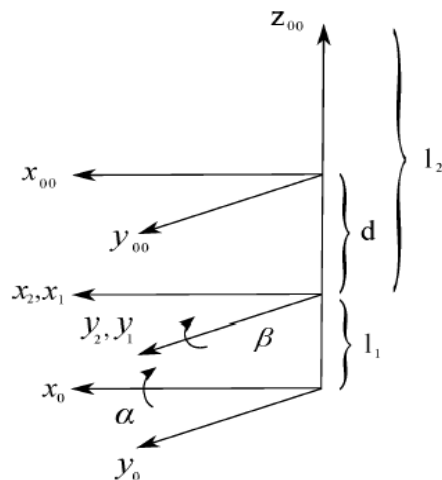


Figura 16. Modelo cinemático del manipulador de dos grados de libertad

Fuente: (Lukács & Lantos, 2012)

En (Dančuo, Kvirgić, Rašuo, & Vidaković, 2013) se combina la cinemática inversa con la directa, con el uso de la matriz RPY y la convención Denavit Hartenberg, descrito en (15) y (16)

respectivamente, para formar la matriz Jacobiana, a partir de las derivadas temporales de ambos modelos se encuentran las velocidades articulares en función de las cartesianas como se representa en (17) y (18), estas matrices J_i y J_d permiten la detección de singularidades en el mecanismo robótico mediante el cálculo de sus determinantes.

$$\begin{bmatrix} q_1 \\ q_2 \\ q_3 \end{bmatrix} = R(\psi, \theta, \phi) x \begin{bmatrix} x \\ y \\ z \end{bmatrix} \quad (15)$$

$$\begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix} = T_1^3 x \begin{bmatrix} q_1 \\ q_2 \\ q_3 \\ 1 \end{bmatrix} \quad (16)$$

$$\begin{bmatrix} \dot{q}_1 \\ \dot{q}_2 \\ \dot{q}_3 \end{bmatrix} = J_i x \begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{z} \end{bmatrix} \quad (17)$$

$$\begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{z} \\ 1 \end{bmatrix} = J_d x \begin{bmatrix} \dot{q}_1 \\ \dot{q}_2 \\ \dot{q}_3 \\ 1 \end{bmatrix} \quad (18)$$

En (Villaruel, 2011), (Gallardo, Rico, & Orozco, 2004) y (Fernández & Sotomayor, 2016) se realiza las mismas actividades que en el trabajo anterior con la diferencia que la cinemática directa se la calcula por medio de análisis geométrico y álgebra de segmentos, en este caso la matriz J_d se modifica y además se realiza el estudio en una plataforma Stewart por lo cual las matrices de transformación son de 6 x 6.

2.4 Control cinemático a partir de modelos de movimiento

Con las metodologías revisadas, para encontrar los modelos cinemáticos de los simuladores de vuelo, varios trabajos realizan la etapa de control ya sea a bajo nivel mediante el modelamiento de los actuadores que pueden ser motores neumáticos, hidráulicos para el caso de las articulaciones rotatorias y para las prismáticas se usan válvulas proporcionales o electroválvulas, el manejo del

modelo de las plantas obtenidas se realizan de una forma acoplada o desacoplada, añadiendo parte de la dinámica de los actuadores, este proceso se usa cuando tienen las plataformas robóticas reales a disposición. En el caso de no tener los mecanismos reales utilizan el control a alto nivel mediante la interpolación de trayectorias de trabajo. También hay casos especiales donde añaden dispositivos periféricos como palancas y pedales para probar el movimiento de los manipuladores entre otros métodos, a continuación se detallan los trabajos que utilizan la etapa de control para la comprobación de movimientos de los respectivos simuladores.

2.4.1 Control desacoplado y acoplado

El control desacoplado articular es usado en robots paralelos altamente acoplado y se realiza modelando servomotores y electroválvulas mediante su dinámica de funcionamiento, con sus respectivos controles PID cada uno, adecuando lo encontrado a un esquema de control por espacio de estados con acción integral (Aristizabal & Giraldo, 2015) (Figura 17) o también puede usarse el control a lazo cerrado (Hernández, Rubio, Prieto, Hernández, & Izaguirre, 2011) (Brusola Fernández-Portolés, 2016), y después se acopla este esquema al modelo cinemático inverso denominando a esta conexión control acoplado (Figura 18).

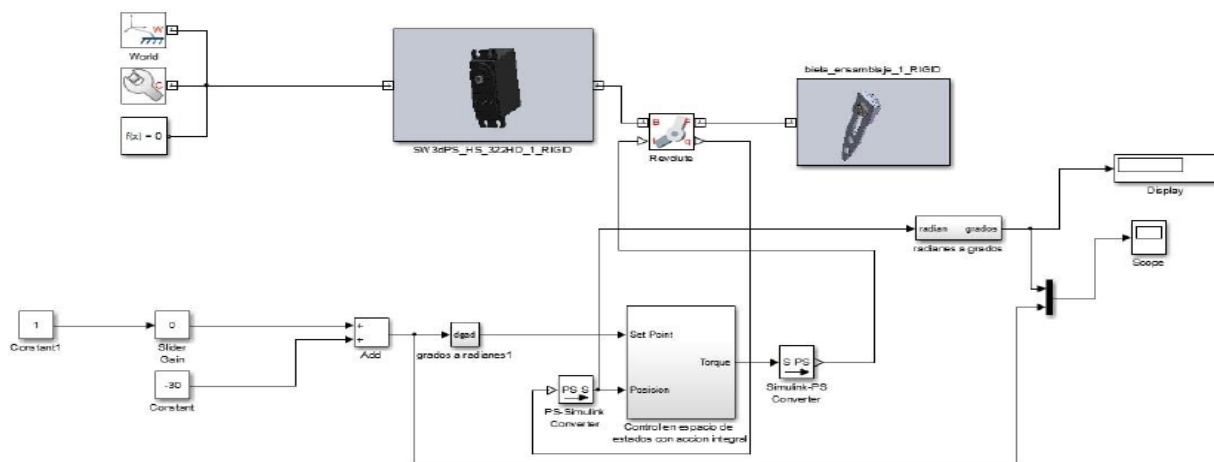


Figura 17. Esquema de control por espacio de estados

Fuente: (Aristizabal & Giraldo, 2015)

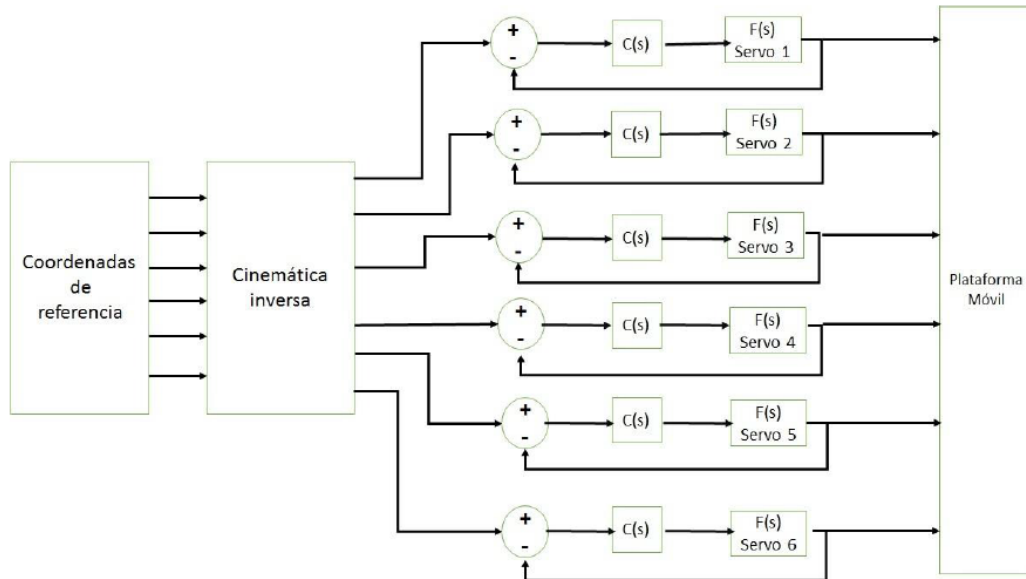


Figura 18. Control acoplado a la cinemática inversa
Fuente: (Hernández, Rubio, Prieto, Hernández, & Izaguirre, 2011)

Para pruebas de simulación de vuelo con el control acoplado se requiere de trayectorias que representen las ilusiones de una aeronave real, para ello se usa generadores de posiciones deseadas, donde se utiliza controladores difusos (González & Reinoso, 2011) o funciones determinadas por especificaciones de diseño, que permiten recrear las trayectorias como se muestra en la Figura 19.

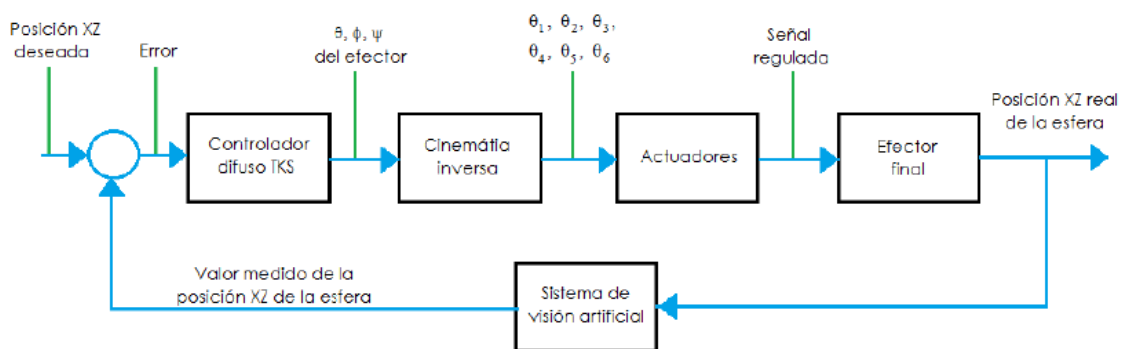


Figura 19. Control acoplado con lógica difusa
Fuente: (González & Reinoso, 2011)

Otra forma de control desacoplado es a partir de filtros pasa altos, donde la entrada son la aceleración cartesiana (x, y, z) y la velocidad angular, las cuales pasan por los filtros llamados

Washout (Figura 20), eliminando componentes de alta frecuencia y después se integra la señal para conseguir la posiciones y orientaciones del manipulador (Arias, 2013).

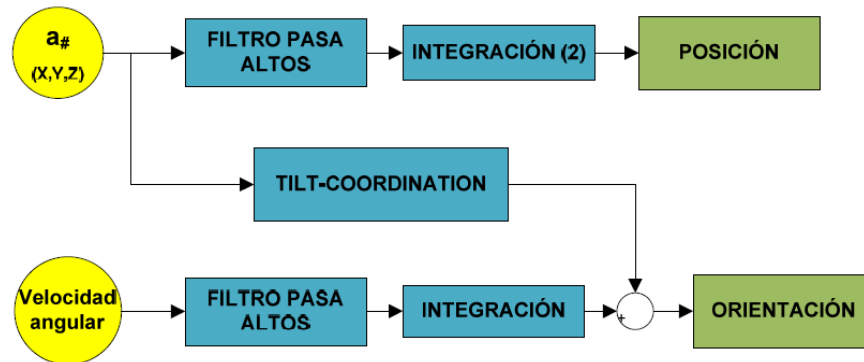


Figura 20. Esquema del filtro de washout.

Fuente: (Arias, 2013)

2.4.2 Control cinemático por interpolación de trayectorias

El control por interpolación de trayectorias se realiza para un robot cumpla un camino mediante trayectorias definidas por especificaciones del sistema, en las cuales los requerimientos de las mismas son puntos de inicio y final (Flórez Vergara & Castro Riveros, 2014). Este método interpola puntos intermedios dentro del camino requerido con funciones polinomiales, que permite describir el movimiento de un manipulador como una secuencia de puntos a lo largo del tiempo (Figura 21) (Iriarte & Noguera, 2012), cumpliendo con el esquema planteado en la Figura 22.

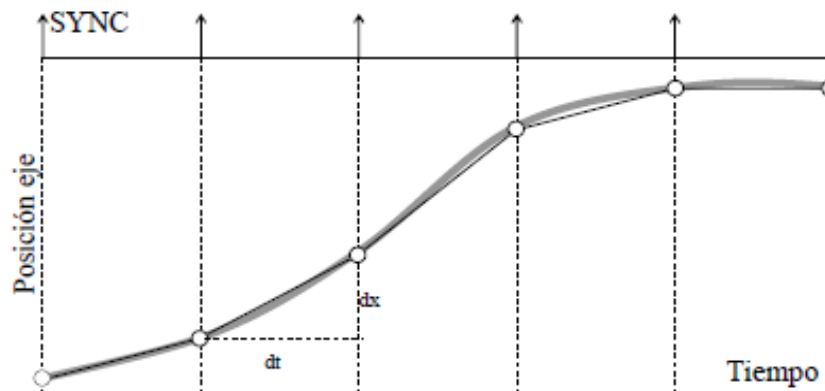


Figura 21. Interpolación de puntos en trayectoria.

Fuente: (Flórez Vergara & Castro Riveros, 2014)

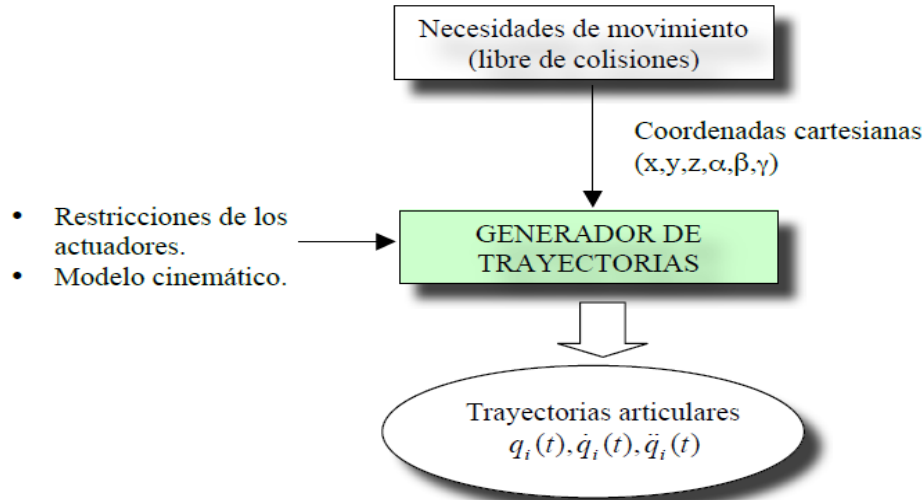


Figura 22. Esquema general del control cinemático
Fuente: (Iriarte & Noguera, 2012)

Los simuladores de vuelo se basan en estructuras paralelas por lo que ocupan trayectorias coordinadas, donde el movimiento de los actuadores es dependiente entre sí. Esta metodología puede trabajar en el espacio cartesiano y articular (Flórez Vergara & Castro Riveros, 2014), lo indica que usa trayectorias del efector final y de las articulaciones del robot, para ello se usa funciones en función del tiempo que permiten describir los caminos deseados para el mecanismo (Flórez-Vergara, E., Castro-Riveros, & Castillo-Esteba, 2016). En (19) se observa un ejemplo de interpolador cartesiano de grado 5 denotado por $x(t)$, donde t es el periodo entre cada punto, y a_i son los coeficientes que se calculan por sistemas de ecuaciones dependiendo de las especificaciones de movimiento (Gonzales-Villela, y otros, 2013), definiendo a_0 el punto inicial y a_5 es el punto final, mientras en (20), se muestra $q(t)$ es un interpolador articular.

$$x(t) = a_0 + a_1t + a_2t^2 + a_3t^3 + a_4t^4 + a_5t^5 \quad (19)$$

$$q(t) = a_0 + a_1t + a_2t^2 + a_3t^3 + a_4t^4 + a_5t^5 \quad (20)$$

Con esta metodología también se planifica trayectorias para velocidad y aceleración, coordinadas con los interpoladores de posición ya sea en el espacio cartesiano o articular (Chaparro

Altamirano, 2013), las magnitudes mencionadas anteriormente se puede obtener de dos maneras, la primera es derivar las expresiones de posición mostradas en (19) y (20) y la otra es mediante las ecuaciones de movimiento uniforme o variado (Hernández, Rubio, Prieto, Hernández, & Izaguirre, 2011), ya sea acelerado o desacelerado, definidas por $d(k)$, $v(k)$ y $a(k)$, donde k varía desde 0 hasta el punto final, como se muestra a continuación:

$$d(k) = d(k - 1) + v_o(k) * t + 0.5 * a(k) * t^2 \quad (21)$$

$$v(k) = v(k - 1) + v_o(k) + a(k) * t \quad (22)$$

$$t = \frac{T}{N} \quad \text{para } t > 0 \quad (23)$$

Donde T es el tiempo total de la trayectoria, mientras que N es el número de puntos interpolados y t es el periodo de cada punto, en otros trabajos plantean una tercera forma la cual es mediante el uso de los Jacobianos como se definen a continuación en (24) (Flórez-Vergara, E., Castro-Riveros, & Castillo-Esteba, 2016):

$$q(k + 1) = q(k) + (J_a(k))^{-1} * J_i * v(k) * t \quad (24)$$

A este tipo de control se lo denomina diferencial o discreto ya que se realiza a partir de una secuencia de puntos, en la Figura 23, se observa un ejemplo de trayectorias de posición, velocidad y aceleración.

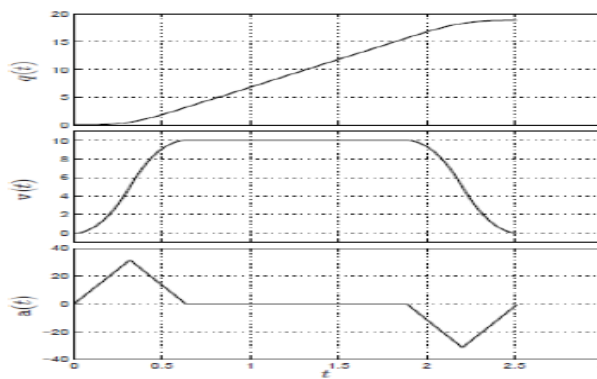


Figura 23. Representaciones típicas en el espacio cartesiano

Fuente: (Flórez Vergara & Castro Riveros, 2014)

2.4.3 Otros

En (Malamud & Scarpin, 2014) aplica un control por interpolación y extrapolación de coeficientes aerodinámicos, a partir de la ecuaciones de movimiento y modelamiento másico de una aeronave real, donde manejan 30 tablas de datos con un total de 14758 valores para interpolar los coeficientes de trayectorias dadas.

En (Mauro, Gastaldi, Pastorelli, & Sorli, 2016) se desarrolla una máquina paralela de tres grados de libertad, para pruebas de plantas que representan los canales semicirculares y otolíticos del cuerpo humano, para evaluar la respuesta dinámica y cinemática de la mismas ante aceleraciones lineales y angulares que se producen en simulaciones de vuelo, para reproducir esos escenarios utilizan pedales y palanca como entradas periféricas a la simulación conjunta entre Simulink de MATLAB y el software MS Flight Simulator que genera las trayectorias de vuelo, mediante el esquema planteado en la Figura 24.

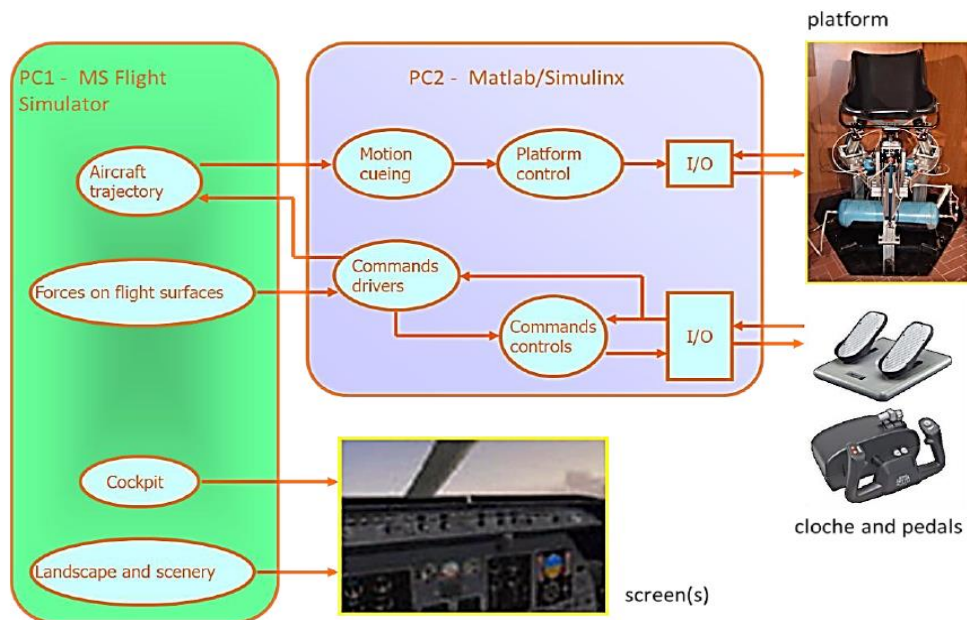


Figura 24. Esquema cinemático con MS flight simulator.

Fuente: (Mauro, Gastaldi, Pastorelli, & Sorli, 2016)

2.5 Simulaciones de modelos cinemáticos

2.5.1 GUI MATLAB

La herramienta GUI de MATLAB (Figura 25) se utiliza para representar los modelos cinemáticos obtenidos (Campos, de Faveri, Furtado, Reis, & Garcia, 2013), con el uso de combinación de conjuntos de funciones y archivos denominados como mex-files, que permiten la conexión con el lenguaje de C++, donde conecta con los actuadores (Arias, 2013).

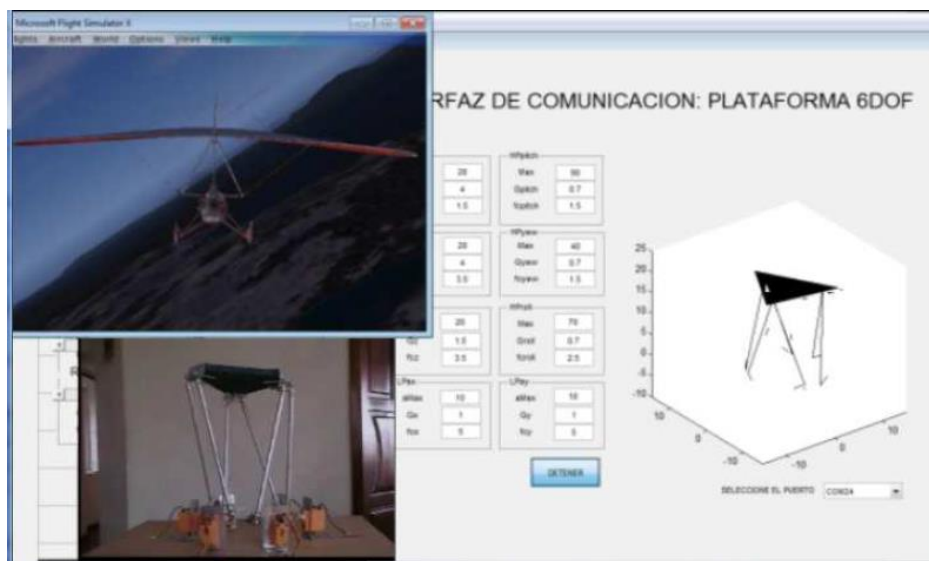


Figura 25. Simulación de vuelo en GUI MATLAB

Fuente: (Arias, 2013)

2.5.2 Solidworks

Hay trabajos usan un diseño CAD de las articulaciones mediante el programa Solidworks y se importa hacia MATLAB mediante el uso de la librería SimMechanics con Simulink (Alcocer, 2012) se realiza la simulación de la plataforma (Aristizabal & Giraldo, 2015) (Barraza, y otros, 2016) (Ramírez Rodríguez & Méndez Moreno, 2011) (Gouasmi, Ouali, Fernini, & Meghatria, 2012).

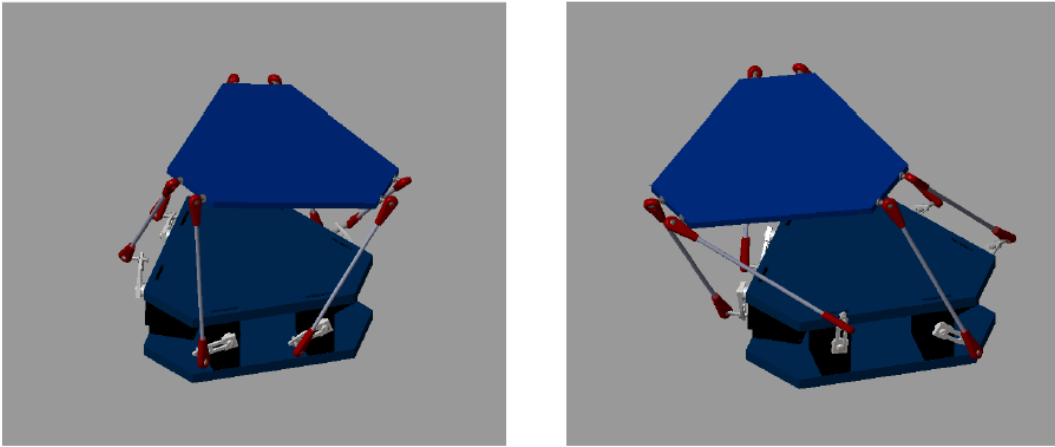


Figura 26. Simulación de la plataforma Stewart en Solidworks
Fuente: (Aristizabal & Giraldo, 2015)

2.5.3 Otros

En otros casos utilizan el Virtual Robot Simulator (VRS) que es una aplicación de modelado gráfico para la simulación de robots en conjunto en escenarios diseñados por el usuario, con la opción de conectarse a lenguaje C, además utiliza ficheros de Autodesk Inventor para diseño de manipuladores robóticos (López, 2008).

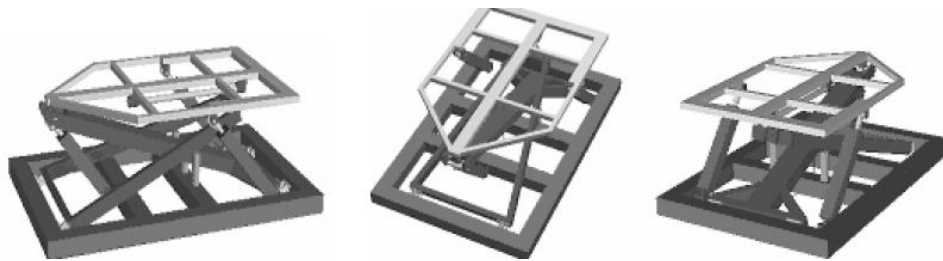


Figura 27. Simulación en Virtual Robot Simulator
Fuente: (López, 2008)

El software ADAMS es una opción de diseño gráfico mecánico que permite, la modelación, simulación, validación de ecuaciones cinemáticas y dinámica, mediante el diseño de la estructura robótica con el uso de fuerzas y momentos inerciales con esquema de conexión mostrado en la Figura 28 (Hernández, Rubio, Prieto, Hernández, & Izaguirre, 2011) (Huang, Wang, & Tham, 2017) (Izaguirre, 2012).

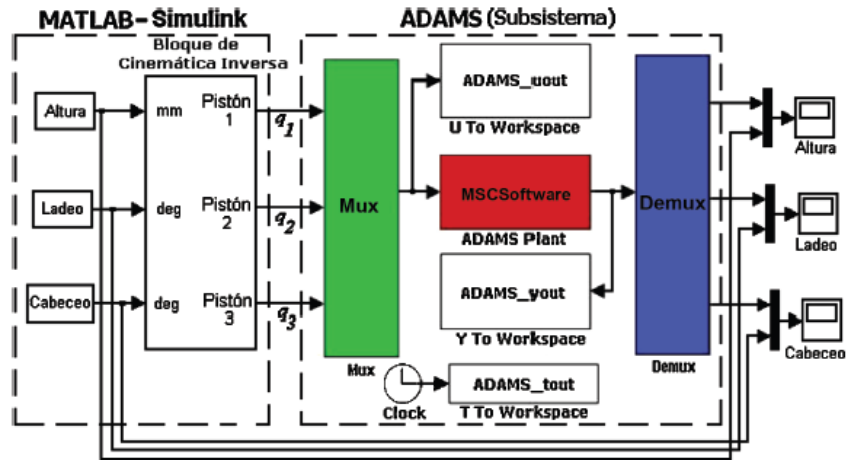


Figura 28. Simulación MATLAB- ADAMS

Fuente: (Huang, Wang, & Tham, 2017)

En (Castro & Zubieta, 2014) se desarrolla un ambiente virtual en base a la plataforma Stewart en el programa Unity 3D que permite crear entornos mediante librerías que reproducen escenarios de vuelo con elementos como terrenos, montañas, árboles, playa, también hay plantillas de motores, válvulas para diseñar mecanismos robóticos, el lenguaje de programación base es c#, pero la desventaja de este software es el precio de la licencia. Además de este software se complementa con la simulación de la plataforma en Solidworks para comprobar el movimiento de la misma.



Figura 29. Simulación en Unity 3D

Fuente: (Castro & Zubieta, 2014)

En (Malamud & Scarpin, 2014), (Rodríguez, 2012) implementan un simulador de vuelo virtual a partir del programa C++ ejecutable y con una interfaz gráfica en OpenGL, que permite la simulación en tiempo real con la interacción de varios usuarios.

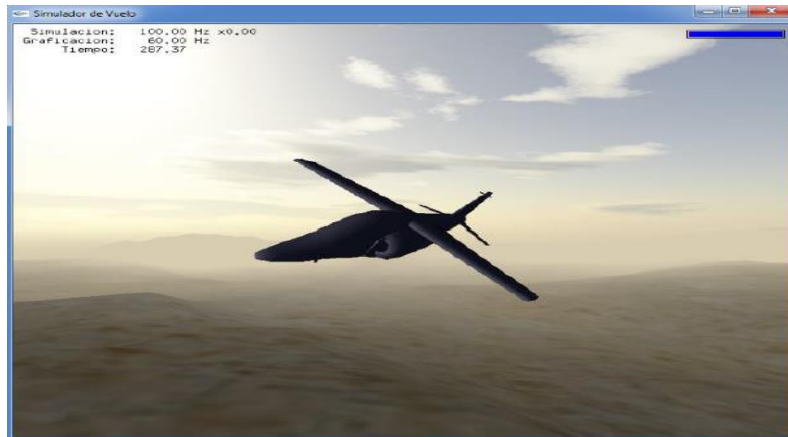


Figura 30. Simulación con OpenGL

Fuente: (Malamud & Scarpin, 2014)

CAPITULO III

3. MODELADO Y CONTROL CINEMÁTICO

En este capítulo se realiza la obtención de forma analítica del modelo que representa la cinemática del simulador de vuelo basado en una estructura R-3RPS, mediante la resolución la cinemática inversa y directa de manipulador en mención, se analiza las restricciones de movimiento, además se determina los jacobianos de velocidad mediante las derivadas parciales de la posición obtenida por métodos inversos y directos, para análisis de singularidades, se encuentra las aceleraciones tanto cartesianas como articulares en puntos referenciales. También se realiza el control cinemático por interpolación de trayectorias que permite encontrar los desplazamientos, velocidades y aceleraciones de los actuadores activos a partir de los modelos cinemáticos obtenidos que generan los datos de entrada a la simulación en ANSYS, la generación de las trayectorias usadas se basan las especificaciones de las ilusiones vestibulares proporcionadas por el manual de funcionamiento del simulador en cuestión.

3.1 Descripción del funcionamiento de la simulación en ANSYS

ANSYS es un software de diseño y simulación que permite el análisis estático y dinámico de mecanismos electromecánicos, electromagnéticos, termodinámicos, mecánicos (ANSYS, 2018), mediante el método de elemento finitos, que ayuda a dividir un sistema complejo en piezas más pequeñas (Vargas, Riaño, & Pineda, 2005), donde el software para cada una de las piezas calcula los parámetros de funcionamiento mediante ecuaciones diferenciales y métodos numéricos discretizados, por esta razón, para realizar pruebas en este programa se requiere una secuencia de puntos con información en cada uno de ellos, que permitan en el mecanismo realizar una acción. La información que puede estar en cada punto son magnitudes relacionadas entre sí como por

ejemplo tiempo, desplazamientos, velocidad, aceleración, temperatura, fuerzas, transferencia de calor entre otros (ANSYS, 2018). Para el caso del simulador de vuelo el software permite el ingreso de desplazamientos, velocidad y aceleración de los actuadores relacionados a periodos de tiempo, para cumplir una trayectoria definida, como salidas muestra las magnitudes de entrada pero en cualquier punto del mecanismo.

Por ese motivo mediante los modelos cinemáticos y la interpolación de las trayectorias de las ilusiones de trabajo, se busca encontrar los desplazamientos, velocidad y aceleración de las articulaciones activas para que estas sean las entradas a la simulación en ANSYS, y para verificar las salidas se van a comparar en puntos referenciales de la simulación realizada en este proyecto, con el objetivo de comprobar la cinemática del manipulador base del simulador de desorientación espacial.

3.2 Especificaciones de funcionamiento

3.2.1 Estructura

El simulador de desorientación espacial descrito en este proyecto se basa en una manipulador robótico paralelo simétrico R-3RPS de 4 grados de libertad, que está constituido dos bases móviles, ambas son de forma triangular equilátera unidas por tres brazos o cadenas separadas de manera equidistante entre sí, donde cada uno está formado por articulaciones rotatoria (1 grado de libertad), prismática (1 grado de libertad) y esférica (3 grados de libertad) (RPS), la geometría del manipulador se observa en la Figura 31.

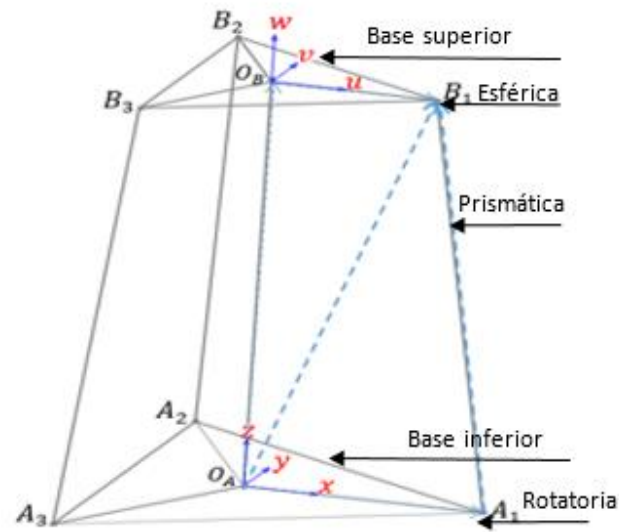


Figura 31. Geometría del manipulador R-3RPS

La base superior de lado igual a 900 mm, es denotada por los vértices B_1 , B_2 , B_3 , mientras que la inferior de lado igual a 1 m es denotada por los vértices A_1 , A_2 , A_3 , las bases triangulares representan dos sistemas coordenados referenciales, a la inferior se asigna el sistema A, con el punto de origen O_A , y los ejes coordenados asociados a este sistema son x , y , z , mientras que para la superior se asigna el sistema B, con el punto de origen O_B , y los ejes coordenados asociados a este sistema son u , v , w .

La base inferior solo permite la rotación alrededor del eje z , mientras que la superior permite rotaciones y traslaciones alrededor de los tres ejes coordenados u , v , w , el movimiento de esta base depende del movimiento de los 3 brazos unidos, que presentan ciertas características. Los brazos están ubicado sobre los vértices de las bases inferior y superior, la unión es denotada por A_iB_i donde i varía entre 1 y 3, referenciando el número de brazos.

Cada brazo es una estructura RPS, donde la articulación rotatoria que restringe el movimiento a solo el plano x - z , la parte prismática se comprende una válvula cuya longitud de su cilindro es fija y la del pistón es variable y se encuentra a media carrera inicialmente, la parte esférica es una

articulación que se ajusta de acuerdo al movimiento producido por las otras juntas activas, a continuación se muestra las especificaciones de las articulaciones activas de los brazos en la Tabla 1.

Tabla 1.

Especificaciones de las articulaciones activas de los brazos

Articulación rotatoria		Articulación prismática	
Elemento	Rango (°)	Elemento	Rango (mm)
Junta Rotatoria	60° - 120°	Cilindro	1191,2
		Pistón	0 – 914,4
		Actuador	1191,2 – 2105,6
		Actuador empieza	1648,4

Los brazos se encuentran separados de manera simétrica debido a que se encuentra ubicados en los vértices de las bases que son equiláteras que cumplen las características de lados y ángulos iguales, además que la forma de la base parte de la figura geométrica inscrita en una circunferencia por lo que también los brazos equidistan 120° entre sí, cumpliendo la simetría. Los brazos por movimiento propio de sus articulaciones solo se pueden mover en el plano x-z, como se observa en la Figura 32, denotados por pl_1 , pl_2 y pl_3 , pero también se puede tener un movimiento en el eje y, debido a que la base inferior tiene movilidad alrededor del eje z, lo que produce que todo el mecanismo encima de la misma gire. Se define el rango de funcionamiento de la plataforma giratoria de 0° a 360° en sentido positivo y de 0° a -360° para el negativo, que puede generar varias vueltas de la misma dependiendo las características y especificaciones de movimiento, los requerimientos de movilidad de la plataforma giratoria están especificados por los datos de las ilusiones dinámicas de desorientación espacial.

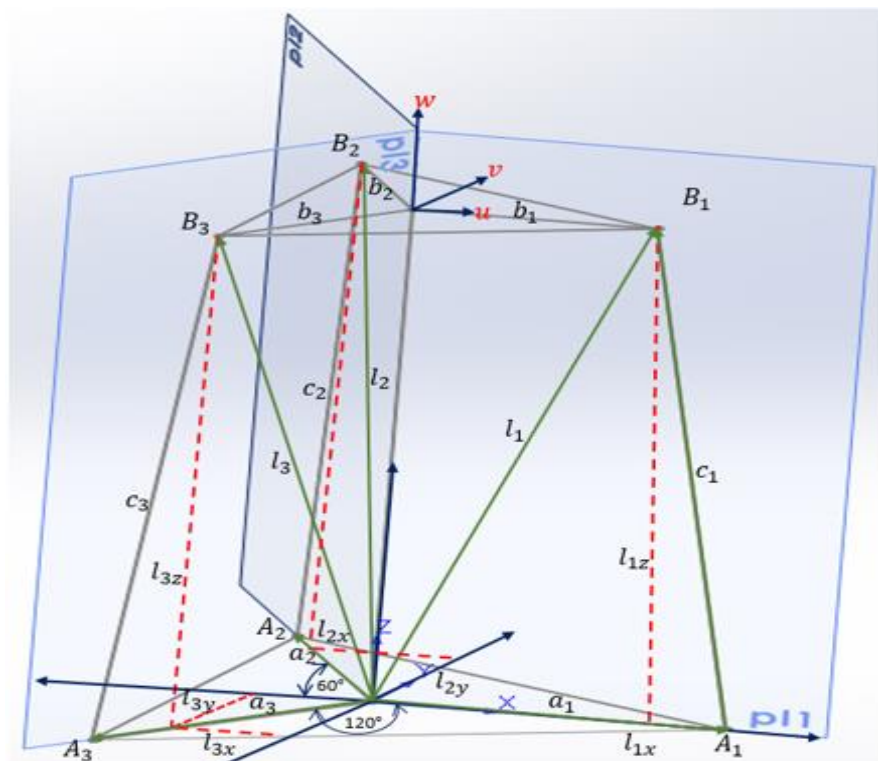


Figura 32. Planos de movimiento de los brazos

3.2.2 Especificaciones de funcionamiento

Descrito el funcionamiento del manipulador, se requiere explicar las especificaciones del mismo para limitar su uso para representar condiciones de vuelo. El mecanismo robótico en cuestión es la unión de una estructura 3RPS y una articulación rotatoria que gira todo el manipulador alrededor del eje z , las especificaciones se centran en las rotaciones basadas en los ángulos de orientación (roll (alabeo), pitch (cabeceo), yaw (guiñada)) con respecto al punto central de la base superior, también se especifica las velocidades rotacionales con respecto a los mismos ángulos, como se muestra en la Tabla 2, el manual de funcionamiento del simulador no presenta limitaciones en cuanto a traslación y aceleración del manipulador, pero más adelante se determina restricciones de movimiento que restringen las traslaciones del mecanismo.

Tabla 2.*Especificaciones de funcionamiento del manipulador R-3RPS*

Ángulos de orientación	Posición	Velocidad máxima
	Rango (°)	Rango(°/s)
Alabeo	-45 a 45	90
Cabeceo	-45 a 45	90
Guiñada	-360 a 360	4,5 a 80,21

3.2.3 Especificaciones de las ilusiones de vuelo

El simulador de vuelo va realizar varias ilusiones visuales y vestibulares que tienen sus propias especificaciones de funcionamiento, igualmente solo se centran en los ángulos de orientación, como se requiere probar el funcionamiento del manipulador se plantea el uso de las siguientes ilusiones vestibulares, ladeos, barrena y coriolis que engloban las rotaciones alrededor de los tres ejes x, y, z.

La Barrena especifica que el simulador debe tener una actitud de ascenso entre 20° a 30°, se produce la falta de sustentación que produce que el avión caiga nariz abajo de -20° a -30° con un alabeo entre 15° a 20°, dando 1 vuelta de manera constante (Escuela de Aviación Naval, 2016).

La coriolis especifica que el avión tendrá un alabeo constante de 30° a 45°, dando una vuelta con velocidad constante de 3°/s a 4.5°/s (Escuela de Aviación Naval, 2016).

En la Tabla 3 se muestran las especificaciones de las ilusiones mencionadas.

Tabla 3.*Especificaciones de las ilusiones vestibulares de trabajo*

Ilusión	Alabeo	Cabeceo	Guiñada
	Rango (°)	Rango (°)	Rango (vueltas)
Ladeos	1 - 2.5	0	0
Barrena	15 - 20	20 - 30	1
Coriolis	30 - 45	0	1

Con las especificaciones planteadas y el análisis de magnitudes de velocidad, aceleración se construye las trayectorias que representan las ilusiones mencionadas en la Tabla 3, esto permite la aplicación del control cinemático por interpolación, con las especificaciones de las Tabla 1 y Tabla 2, se prosigue a plantear los modelos cinemáticos inverso y directo, en las siguientes fases de este capítulo, para encontrar la posición, velocidad y aceleración en los puntos de la base superior.

3.3 Modelado de Posición

Mediante las especificaciones planteadas en la sección anterior en este punto se desarrolla los modelos cinemáticos inverso y directo, con métodos que permitan describir el movimiento del manipulador en función de la cinemática de vuelo.

3.3.1 Cinemática inversa

En esta sección se encuentra los desplazamientos de las articulaciones activas del manipulador, a partir de la posición y orientación del punto O_B , estableciendo a este punto como efector final. Las articulaciones activas son las encargadas de generar el movimiento del mecanismo, dependiendo del desplazamiento de los mismos, mientras que en las pasivas el desplazamiento de estas depende del movimiento de las activas. En lo que corresponde a desplazamiento de los actuadores pueden ser lineales o angulares, debido a que en el manipulador hay articulaciones prismáticas y rotatorias. Debido a que en la base superior se encuentra la cabina del piloto a este se le denomina base móvil para la cabina y a la inferior se le llama plataforma giratoria, el modelo se trabaja con las nuevas denominaciones mencionadas, para adecuación de términos.

El desarrollo del modelo se empieza, definiendo las coordenadas de la plataforma giratoria y la base móvil para la cabina, a partir de la Figura 33 y la descripción de las variables que las conforman en la Tabla 4.

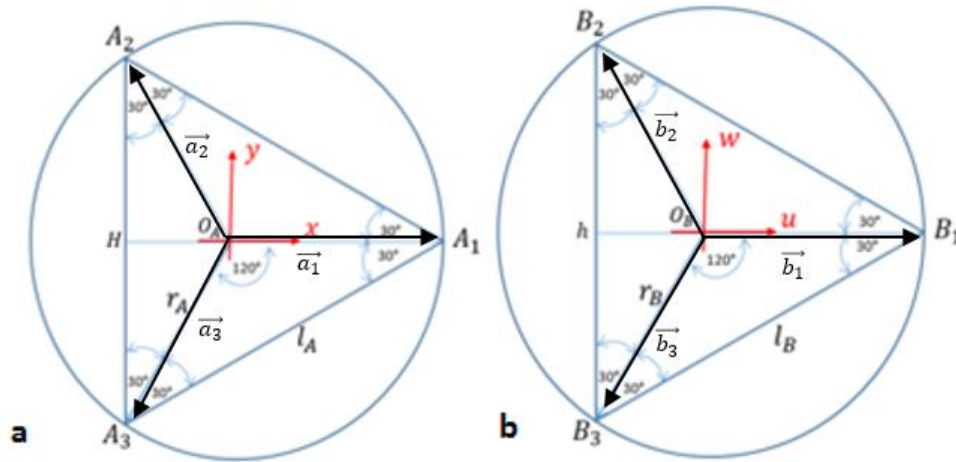


Figura 33. Coordenadas de (a) Plataforma giratoria (b) Base móvil para la cabina

Tabla 4.

Descripción de las variables de la base móvil y plataforma

Variable	Descripción
O_A, O_B	Son los puntos de origen del sistema A para la plataforma y el sistema B para la base
A_1, A_2, A_3	Vértices del triángulo equilátero de la plataforma
B_1, B_2, B_3	Vértices del triángulo equilátero de la base móvil
$\vec{a}_1, \vec{a}_2, \vec{a}_3$	Vectores de posición de los vértices A_i con respecto al sistema coordenado A de la plataforma giratoria
$\vec{b}_1, \vec{b}_2, \vec{b}_3$	Vectores de posición de los vértices B_i con respecto al sistema coordenado B de la base móvil

Con el concepto de un triángulo equilátero inscrito en una circunferencia, los términos r_A y r_B , son los radios de la plataforma giratoria y de la base móvil, el valor de esto se los calcula mediante las expresiones planteadas en (25) y (26).

$$r_A = \frac{l_A}{\sqrt{3}} \quad (25)$$

$$r_B = \frac{l_B}{\sqrt{3}} \quad (26)$$

Donde l_A y l_B son los lados de la plataforma y la base móvil respectivamente, los vértices de ambos triángulos se definen mediante los respectivos radios y centros de la circunferencia O_A, O_B ,

y como es una figura de tres lados dentro de la circunferencia los vértices deben estar separados 120° grados entre sí, los vectores \vec{a}_i, \vec{b}_i determinan la posición de los vértices A_i y B_i en función de los términos r_A, r_B , llegando a las expresiones (27), (28), (29) para la plataforma y (30), (31), (32) para la base móvil.

$$\vec{a}_1 = \begin{bmatrix} a_{1x} \\ a_{1y} \\ a_{1z} \end{bmatrix} = \begin{bmatrix} r_A \\ 0 \\ 0 \end{bmatrix} \quad (27)$$

$$\vec{a}_2 = \begin{bmatrix} a_{2x} \\ a_{2y} \\ a_{2z} \end{bmatrix} = \begin{bmatrix} r_A \cos 120^\circ \\ r_A \sen 120^\circ \\ 0 \end{bmatrix} = \begin{bmatrix} -\frac{1}{2}r_A \\ \frac{\sqrt{3}}{2}r_A \\ 0 \end{bmatrix} \quad (28)$$

$$\vec{a}_3 = \begin{bmatrix} a_{3x} \\ a_{3y} \\ a_{3z} \end{bmatrix} = \begin{bmatrix} r_A \cos 240^\circ \\ r_A \sen 240^\circ \\ 0 \end{bmatrix} = \begin{bmatrix} -\frac{1}{2}r_A \\ -\frac{\sqrt{3}}{2}r_B \\ 0 \end{bmatrix} \quad (29)$$

$$\vec{b}_1 = \begin{bmatrix} b_{1x} \\ b_{1y} \\ b_{1z} \end{bmatrix} = \begin{bmatrix} r_B \\ 0 \\ 0 \end{bmatrix} \quad (30)$$

$$\vec{b}_2 = \begin{bmatrix} b_{2x} \\ b_{2y} \\ b_{2z} \end{bmatrix} = \begin{bmatrix} r_B \cos 120^\circ \\ r_B \sen 120^\circ \\ 0 \end{bmatrix} = \begin{bmatrix} -\frac{1}{2}r_B \\ \frac{\sqrt{3}}{2}r_B \\ 0 \end{bmatrix} \quad (31)$$

$$\vec{b}_3 = \begin{bmatrix} b_{3x} \\ b_{3y} \\ b_{3z} \end{bmatrix} = \begin{bmatrix} r_B \cos 240^\circ \\ r_B \sen 240^\circ \\ 0 \end{bmatrix} = \begin{bmatrix} -\frac{1}{2}r_B \\ -\frac{\sqrt{3}}{2}r_B \\ 0 \end{bmatrix} \quad (32)$$

Definidas las coordenadas de los vectores de posición de los vértices de ambas bases, la siguiente parte es realizar el análisis cinemático del manipulador R- 3RPS, mediante la separación

en la estructura 3 RPS y la plataforma giratoria, modelando cada una, para después relacionar ambos modelos inversos y encontrar el definitivo, en la Figura 34 se muestra a los manipuladores separados.

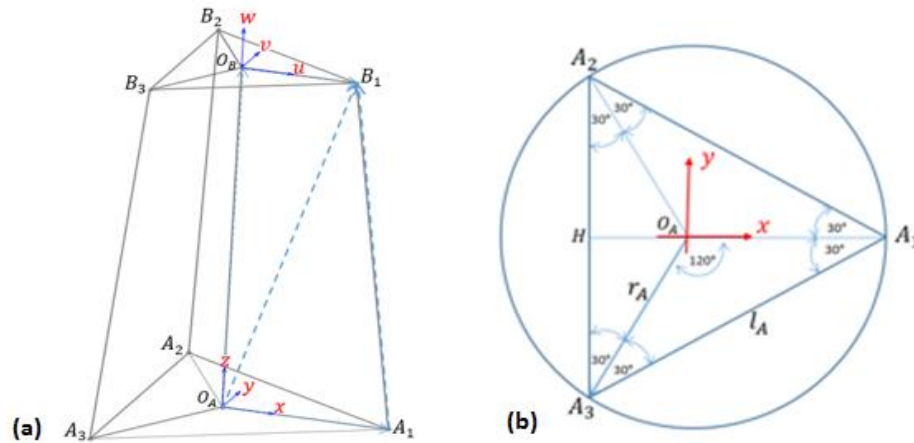


Figura 34. (a) Manipulador 3RPS (b) Plataforma giratoria

3.3.1.1 Cinemática del manipulador 3 RPS

Dentro del manipulador 3 RPS se encuentra la base móvil para la cabina, con O_B punto de origen del sistema coordenado B, el mismo que rota alrededor de sus ejes u, v, w , ante el movimiento de las articulaciones activas de los 3 brazos, las cuales son las rotatorias y prismáticas, con un total de 6 articulaciones entre los brazos. Las rotaciones de la base móvil producen traslaciones dependientes de las rotaciones, se requiere transformar el sistema B en un sistema cartesiano A en función de los ejes x, y, z , para lo cual se utiliza el método de matriz de rotación que permite la representación de la orientación del sistema B respecto al A (Ospina, 2012) (Ruiz Hidalgo, y otros, 2017) (Ramana Babu, Ramachandra Raju, & Ramji, 2013), mediante la descomposición de los ejes u, v, w en componentes x, y, z , dejando todo en función de vectores unitarios de los ejes del sistema B con subíndices del sistema A, distribuidos de forma matricial como se muestra en (33).

$$R_{3RPS_B}^A = \begin{bmatrix} u_x & v_x & w_x \\ u_y & v_y & w_y \\ u_z & v_z & w_z \end{bmatrix} \quad (33)$$

Con la expresión obtenida en (33) se encuentra la orientación del sistema B en función de los coordenadas cartesianas, la matriz descrita se puede relacionar con la cinemática de vuelo de una aeronave, en el capítulo anterior se mencionó algunos métodos para modelar el movimiento de un avión, a partir de manipuladores robóticos entre los que destacaban los ángulos de Euler y los cuaterniones, la elección dependió de rango de funcionamiento de las articulaciones, como el manipulador tiene rango menor a 90° el método adecuado para trabajar es Euler, esta metodología trabaja con los ángulos (roll (ϕ), pitch (θ), yaw (ψ)), que asignan a (alabeo, cabeceo, guiñada), respectivamente, pero con la acotación que el yaw (ψ), no es la guiñada del mecanismo debido a este es dependiente del alabeo y cabeceo, la guiñada debe ser independiente de las variables mencionadas, por lo cual a esta variable se la denomina yaw_{3RPS} , con esta aclaración, se define la matriz $R_{3RPS_B}^A$ en función de los ángulos de Euler como se determinó en el estado del arte, quedando la expresión, como se muestra en (34).

$$R_{3RPS_B}^A = \begin{bmatrix} c\psi c\theta & c\psi s\theta s\phi - s\psi c\phi & c\psi s\theta c\phi + s\psi s\phi \\ s\psi c\theta & s\psi s\theta s\phi + c\psi c\phi & s\psi s\theta c\phi - c\psi s\phi \\ s\theta & -c\theta s\phi & c\theta c\phi \end{bmatrix} \quad (34)$$

Donde c y s representan coseno y seno respectivamente, relacionando ambas matrices descritas se obtienen las siguientes expresiones:

$$u_x = c\psi c\theta \quad (35)$$

$$u_y = s\psi c\theta \quad (36)$$

$$u_z = s\theta \quad (37)$$

$$v_x = c\psi s\theta s\phi - s\psi c\phi \quad (38)$$

$$v_y = s\psi s\theta s\phi - c\psi c\phi \quad (39)$$

$$v_z = -c\theta s\phi \quad (40)$$

$$w_x = c\psi s\theta c\phi + s\psi s\phi \quad (41)$$

$$w_y = s\psi s\theta c\phi - c\psi s\phi \quad (42)$$

$$w_z = c\theta c\phi \quad (43)$$

Con las expresiones de relación entre los vectores unitarios de la matriz de rotación y los ángulos de Euler, se define la expresión vectorial que permite encontrar el movimiento de los vértices de la base móvil, en función del alabeo y cabeceo, tomando a la base inferior como fija, en la Figura 35, se muestra como se designaron las variables de trabajo para el manipulador 3 RPS (Ramana Babu, Ramachandra Raju, & Ramji, 2013).

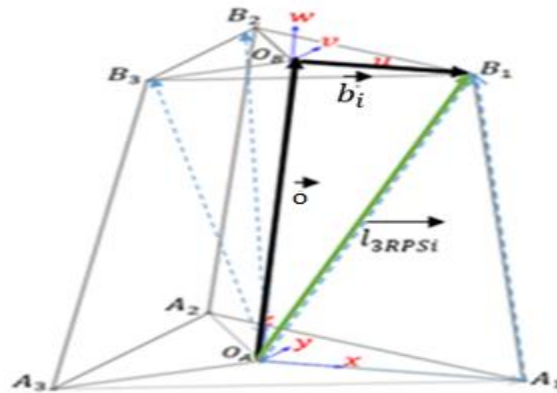


Figura 35. Variables de trabajo del manipulador 3RPS

Tabla 5.

Descripción de las variables del manipulador 3 RPS

Variable	Descripción
\vec{o}	Vector posición que indica la ubicación del punto O_B con respecto al punto O_A , con coordenadas (o_x, o_y, o_z)
\vec{b}_i	Vector posición que indica la ubicación de los puntos B_i con respecto al punto O_B , gobernado por la matriz de rotación.
\vec{l}_{3RPSi}	Vector resultante de la algebra vectorial entre los vectores \vec{o} y \vec{b}_i , que representa posición de los vértices B_i con respecto al punto O_A

Con las variables definidas en la Figura 35 y en la Tabla 5, se establece la expresión vectorial que define el movimiento de los vértices de la base móvil del manipulador 3RPS. Donde i es el subíndice que indica el número de vértice donde se encuentra cada brazo que va de 1 a 3, como se mencionó en la Tabla 3, el término \vec{b}_i , es gobernado por la matriz de rotación (34), dejando la expresión vectorial de la siguiente manera:

$$\vec{l}_{3RPSi} = \vec{o} + R_{3RPSB}^A * \vec{b}_i \quad (44)$$

Mediante la operación obtenida en (44), se define el movimiento de los vértices B_i , con respecto al punto O_A , mediante los vectores \vec{l}_{3RPSi} , reemplazando (30), (31), (32), (33) en (44), y adecuando las ecuaciones se obtienen las coordenadas de los vectores \vec{l}_{3RPSi} , como se muestra a continuación:

$$\vec{l}_{3RPS1} = \begin{bmatrix} l_{3RPS1x} \\ l_{3RPS1y} \\ l_{3RPS1z} \end{bmatrix} = \begin{bmatrix} o_x + r_B * u_x \\ o_y + r_B * u_y \\ o_z + r_B * u_z \end{bmatrix} \quad (45)$$

$$\vec{l}_{3RPS2} = \begin{bmatrix} l_{3RPS2x} \\ l_{3RPS2y} \\ l_{3RPS2z} \end{bmatrix} = \begin{bmatrix} o_x - \frac{1}{2}r_B * u_x + \frac{\sqrt{3}}{2}r_B * v_x \\ o_y - \frac{1}{2}r_B * u_y + \frac{\sqrt{3}}{2}r_B * v_y \\ o_z - \frac{1}{2}r_B * u_z + \frac{\sqrt{3}}{2}r_B * v_z \end{bmatrix} \quad (46)$$

$$\vec{l}_{3RPS3} = \begin{bmatrix} l_{3RPS3x} \\ l_{3RPS3y} \\ l_{3RPS3z} \end{bmatrix} = \begin{bmatrix} o_x - \frac{1}{2}r_B * u_x - \frac{\sqrt{3}}{2}r_B * v_x \\ o_y - \frac{1}{2}r_B * u_y - \frac{\sqrt{3}}{2}r_B * v_y \\ o_z - \frac{1}{2}r_B * u_z - \frac{\sqrt{3}}{2}r_B * v_z \end{bmatrix} \quad (47)$$

En las expresiones (45), (46), (47), falta encontrar las expresiones que definan las componentes o_x, o_y, o_z , del vector o , la variable o_z , es la altura de referencia del punto O_B , mientras que las demás

expresiones se encuentran de las restricciones del manipulador que se desarrolla en el siguiente punto.

3.3.1.1.1 Restricciones de movimiento del manipulador 3 RPS

En las especificaciones de funcionamiento se detalló que los tres brazos del manipulador se mueven en un plano, en el primero lo realiza solo en el plano x-z, con una componente $y=0$, mientras el segundo brazo se ubica a 120° del primero lo que indica que tendrá movimiento en el plano x-z, pero tiene una componente en y constante positiva, el tercer brazo se localiza a 240° del primero tendrá movilidad en el mismo plano que el segundo pero con una componente en y constante negativa, la determinación de la variables en y, parte de la ubicación de los vértices B_i en el plano x-y, combinado con el concepto de la pendiente entre los puntos B_i con el punto O_B , gobernado por los vectores l_{3RPSi} , se expresa las siguientes condiciones:

$$l_{3RPS1y} = 0 \quad (48)$$

$$l_{3RPS2y} = \tan(120^\circ) * l_{3RPS2x} = -\sqrt{3} * l_{3RPS2x} \quad (49)$$

$$l_{3RPS3y} = \tan(240^\circ) * l_{3RPS3x} = \sqrt{3} * l_{3RPS3x} \quad (50)$$

Reemplazando las expresiones (45). (46), (47) en (48), (49), (50) respectivamente se encuentran las siguientes ecuaciones:

$$o_y = -r_B * u_y \quad (51)$$

$$o_x = \frac{1}{2} r_B * (u_x - v_y) \quad (52)$$

$$v_x = u_y \quad (53)$$

A partir de la expresión (53), se puede despejar el ángulo yaw_{3RPS} (ψ) en función de los ángulos roll (Φ) y pitch (θ), sustituyendo (36) y (38) en (56), encontrando la siguiente ecuación:

$$\psi = \text{atan} \left(\frac{\text{sen}(\theta) * \text{sen}(\phi)}{\text{cos}(\phi) + \text{cos}(\theta)} \right) \quad (54)$$

Con las expresiones obtenidas se completa la parte del manipulador 3 RPS. En el siguiente punto se completa el modelo inverso con la plataforma giratoria.

3.3.1.2 Modelo de la plataforma giratoria

Para completar el modelo cinemático inverso se requiere aumentar la rotación alrededor del eje z generada por el movimiento de articulación rotacional colocada en el centro O_A de la plataforma giratoria, para esta expresión se utiliza la misma matriz de rotación de Euler pero con la definición del ángulo alrededor del eje z, denominado yaw (γ), quedando la matriz $R_A(\gamma)$ de la siguiente manera:

$$R_A(\gamma) = \begin{bmatrix} \text{cos}\gamma & -\text{sen}\gamma & 0 \\ \text{sen}\gamma & \text{cos}\gamma & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad (55)$$

La variable yaw (γ), es la guiñada del mecanismo R-3RPS, en la Figura 36 y Tabla 5 se detallan las variables para el desarrollo del modelo de la plataforma giratoria.

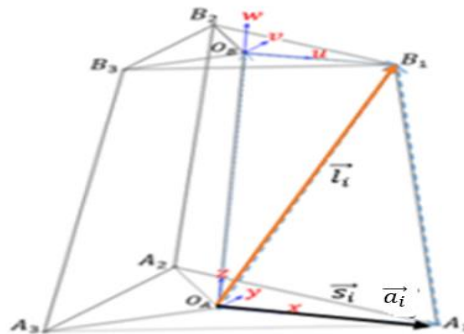


Figura 36. Vectores para posición de la plataforma giratoria

Tabla 6.

Descripción de las variables de la plataforma giratoria

Variable	Descripción
\vec{l}_i	Vector posición que representa la ubicación del punto l_{3RPSi} a partir de la rotación de la plataforma

CONTINÚA



\vec{s}_i	Vector posición que representa la ubicación de los vectores a_i por medio de la plataforma
\vec{a}_i	Vector posición que representa ubicación de los vértices A_i con respecto al punto O_A

En las descripciones de las variables de la plataforma giratoria, se muestra que el movimiento de la misma produce el movimiento de los puntos l_{3RPSi} definidos en el manipulador 3RPS y los vértices A_i , por lo que se define las expresiones vectoriales que describen este movimiento, como se muestra a continuación:

$$\vec{l}_i = R_A(\gamma) * \overrightarrow{l_{3RPSi}} \quad (56)$$

$$\vec{s}_i = R_A(\gamma) * \vec{a}_i \quad (57)$$

Se definen las coordenadas de los vectores \vec{l}_i y \vec{s}_i , reemplazando (27), (28), (29), (45), (46), (47) en (56) y (57), de la siguiente manera:

$$\vec{l}_1 = \begin{bmatrix} l_{1x} \\ l_{1y} \\ l_{1z} \end{bmatrix} = \begin{bmatrix} l_{3RPS1x} * \cos(\gamma) - l_{3RPS1y} * \sen(\gamma) \\ l_{3RPS1x} * \sen(\gamma) - l_{3RPS1y} * \cos(\gamma) \\ o_z + r_B * u_z \end{bmatrix} \quad (58)$$

$$\vec{l}_2 = \begin{bmatrix} l_{2x} \\ l_{2y} \\ l_{2z} \end{bmatrix} = \begin{bmatrix} l_{3RPS2x} * \cos(\gamma) - l_{3RPS2y} * \sen(\gamma) \\ l_{3RPS2x} * \sen(\gamma) - l_{3RPS2y} * \cos(\gamma) \\ o_z - \frac{1}{2} r_B * u_z + \frac{\sqrt{3}}{2} r_B * v_z \end{bmatrix} \quad (59)$$

$$\vec{l}_3 = \begin{bmatrix} l_{3x} \\ l_{3y} \\ l_{3z} \end{bmatrix} = \begin{bmatrix} l_{3RPS3x} * \cos(\gamma) - l_{3RPS3y} * \sen(\gamma) \\ l_{3RPS3x} * \sen(\gamma) - l_{3RPS3y} * \cos(\gamma) \\ o_z - \frac{1}{2} r_B * u_z - \frac{\sqrt{3}}{2} r_B * v_z \end{bmatrix} \quad (60)$$

$$\vec{s}_1 = \begin{bmatrix} s_{1x} \\ s_{1y} \\ s_{1z} \end{bmatrix} = \begin{bmatrix} r_A * \cos(\gamma) \\ r_A * \sen(\gamma) \\ 0 \end{bmatrix} \quad (61)$$

$$\vec{s}_2 = \begin{bmatrix} s_{2x} \\ s_{2y} \\ s_{3z} \end{bmatrix} = \begin{bmatrix} -\frac{1}{2}r_A * \cos(\gamma) - \frac{\sqrt{3}}{2}r_A * \text{sen}(\gamma) \\ -\frac{1}{2}r_A * \text{sen}(\gamma) + \frac{\sqrt{3}}{2}r_A * \cos(\gamma) \\ 0 \end{bmatrix} \quad (62)$$

$$\vec{s}_3 = \begin{bmatrix} s_{3x} \\ s_{3y} \\ s_{3z} \end{bmatrix} = \begin{bmatrix} -\frac{1}{2}r_A * \cos(\gamma) + \frac{\sqrt{3}}{2}r_A * \text{sen}(\gamma) \\ -\frac{1}{2}r_A * \text{sen}(\gamma) - \frac{\sqrt{3}}{2}r_A * \cos(\gamma) \\ 0 \end{bmatrix} \quad (63)$$

Definidas las variables de movimiento de los vértices de la base móvil y la plataforma giratoria se establece las variables dependientes e independientes del modelo para identificar las entradas del mismo.

Tabla 7.

Variables dependientes e independientes del modelo cinemático inverso

Variable	Tipo
\mathbf{o}_x	Dependiente (φ, θ, γ)
\mathbf{o}_y	Dependiente (θ, γ)
ψ	Dependiente (φ, θ)
θ	Independiente (cabeceo)
φ	Independiente (alabeo)
γ	Independiente (guiñada)
\mathbf{o}_z	Independiente (referencia de altura fija en 1647.39 mm)

Se establece que las entradas del modelo cinemático inverso son las variables independientes ($\varphi, \theta, \gamma, \mathbf{o}_z$), con la consideración para este modelo de mantener la altura \mathbf{o}_z constante y las salidas son las longitudes de los actuadores prismáticos, ángulo de rotación de las articulaciones rotacionales a lo largo de los 3 brazos, en la Figura 37 y en la Tabla 8 se muestra la ubicación de las variables de las articulaciones activa en el manipulador R-3 RPS.

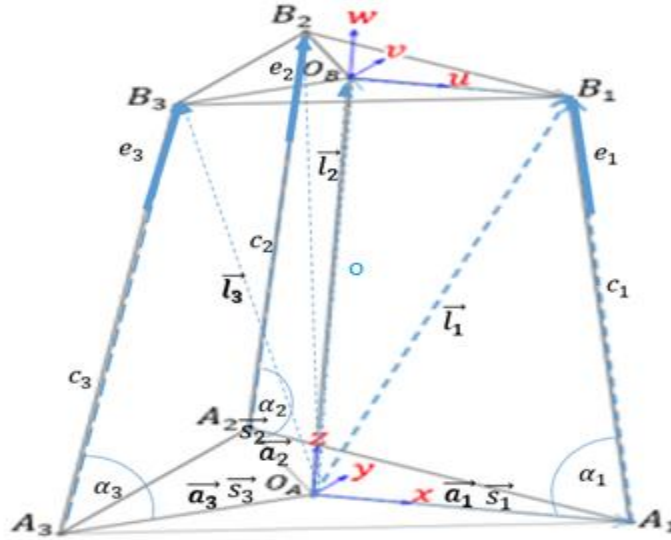


Figura 37. Variables de las articulaciones activas en manipulador R-3RPS

Tabla 8.

Descripción de las articulaciones activas en el manipulador R-3RPS

Variable	Descripción
c_1, c_2, c_3	Longitudes totales de los brazos
e_1, e_2, e_3	Longitudes variables determinadas por los desplazamientos de los actuadores prismáticos
$\alpha_1, \alpha_2, \alpha_3$	Ángulos formados entre la plataforma giratoria y los brazos, que determinan el desplazamiento angular de las juntas rotatorias inferiores de cada brazo.

Para el cálculo de las longitudes de los brazos es necesario aplicar la ecuación de distancia entre dos puntos determinados por los vectores \vec{l}_i y \vec{s}_i , debido a que ambas variables son respecto al punto de origen $O_A(0, 0, 0)$, las coordenadas de los vectores corresponden a los puntos l_i y s_i , encontrando las longitudes mediante las siguientes expresiones:

$$c_1 = \sqrt{(l_{1x} - a_{1x})^2 + (l_{1y} - a_{1y})^2 + (l_{1z} - a_{1z})^2} \quad (64)$$

$$c_2 = \sqrt{(l_{2x} - a_{2x})^2 + (l_{2y} - a_{2y})^2 + (l_{2z} - a_{2z})^2} \quad (65)$$

$$c_3 = \sqrt{(l_{3x} - a_{3x})^2 + (l_{3y} - a_{3y})^2 + (l_{3z} - a_{3z})^2} \quad (66)$$

Reemplazando (58), (59), (60), (61), (62), (63) en (64), (65), (66) dejando las expresiones de la siguiente manera:

$$c_1 = \sqrt{d_1 + 2d_2 - 2r_A o_x - 2r_A r_B u_x} \quad (67)$$

$$c_2 = \sqrt{d_1 - d_2 + d_3 + r_A(o_x - \sqrt{3}o_y) - d_6 d_4} \quad (68)$$

$$c_3 = \sqrt{d_1 - d_2 - d_3 + r_A(o_x + \sqrt{3}o_y) - d_6 d_5} \quad (69)$$

Siendo las constantes

$$d_1 = o_x^2 + o_y^2 + o_z^2 + r_A^2 + r_B^2 \quad (70)$$

$$d_2 = r_B(o_x u_x + o_y u_y + o_z u_z) \quad (71)$$

$$d_3 = \sqrt{3}r_B(o_x v_x + o_y v_y + o_z v_z) \quad (72)$$

$$d_4 = u_x + 3v_y - \sqrt{3}(v_x + u_y) \quad (73)$$

$$d_5 = u_x + 3v_y + \sqrt{3}(v_x + u_y) \quad (74)$$

$$d_6 = \frac{1}{2}r_A r_B \quad (75)$$

La longitud requerida es la del actuador prismático por lo cual a magnitudes obtenidas se resta el valor fijo de los actuadores que es 1191.2 mm, quedando de la siguiente manera:

$$e_1 = c_1 - 1191.2 \quad (76)$$

$$e_2 = c_2 - 1191.2 \quad (77)$$

$$e_3 = c_3 - 1191.2 \quad (78)$$

Para el cálculo de los ángulos de la parte inferior de los brazos se usa la ley de cosenos entre los vértices superiores e inferiores con el punto O_B de la siguiente manera:

$$\alpha_1 = \text{acos}((l_1^2 - c_1^2 - a_1^2) / -2a_1 c_1) \quad (79)$$

$$\alpha_2 = \text{acos}((l_2^2 - c_2^2 - a_2^2) / -2a_2c_2) \quad (80)$$

$$\alpha_3 = \text{acos}((l_3^2 - c_3^2 - a_3^2) / -2a_3c_3) \quad (81)$$

Con los datos obtenidos de las articulaciones se finaliza el modelo de posición por cinemática inversa, con lo se puede realizar las primeras comparaciones con la simulación en ANSYS, en los puntos l_i encontrados, en la siguiente sección se desarrolla el modelo de posición por cinemática directa.

3.3.2 Cinemática directa

En esta sección se encuentra los vectores posición de los vértices de la base móvil con respecto al punto O_A , a partir del ingreso de los desplazamientos de las articulaciones activas provenientes del modelo inverso, para lo cual se utilizan dos métodos para desarrollar la cinemática directa, debido a que la resolución directa en robot paralelos es compleja y produce varias soluciones, se determinaron metodologías que facilitan resultados únicos para esta aplicación. Los métodos utilizados son vectorial combinado con análisis geométrico y convención de Denavit Hartenberg ambos se comparan con los resultados de la cinemática inversa con el objetivo de elegir uno de los métodos directos planteados que más se aproxime a los resultados de verificación, debido que en los siguientes puntos se obtienen las velocidades y aceleraciones en los puntos de comparación y se requiere que los resultados de ambos modelos de posición sean lo más semejantes entre sí.

3.3.2.1 Método vectorial

En este método se utiliza una combinación de algebra de vectores con geometría y funciones trigonométricas para encontrar la ubicación de los vértices de la base móvil con el uso de los desplazamientos de las articulaciones activas como parámetros de entrada, en la Figura 38 y Tabla 9 se definen las variables utilizadas en este modelo.

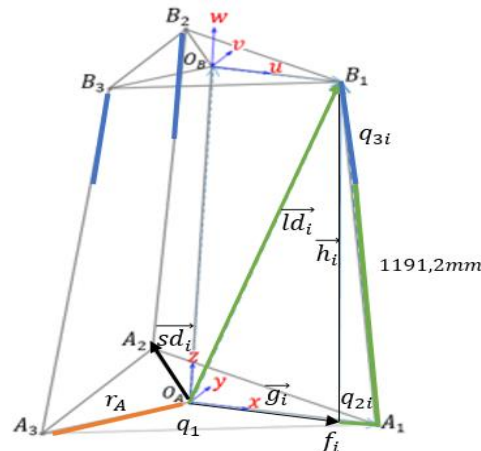


Figura 38. Variables para el manipulador R-3RPS por el método vectorial.

Tabla 9.

Descripción de las variables de trabajo del método vectorial

Variable	Descripción
O_A	Punto de origen de coordenadas (0, 0, 0)
q_1	Ángulo desplazado por la plataforma giratoria
q_{2i}	Ángulo desplazado por la junta rotatoria de cada brazo
q_{3i}	Desplazamiento lineal del actuador prismático de cada brazo
f_i	Punto de intersección el radio r_A y la proyección perpendicular de la longitud total de cada brazo.
\vec{h}_i	Vector posición que representa la ubicación de los vértices B_i con respecto al punto f_i
\vec{g}	Vector posición que representa la ubicación del punto f_i con respecto al punto O_A
ld_i	Vector posición que representa la ubicación de los vértices B_i con respecto al punto O_A
sd_i	Vector posición que representa la ubicación de los vértices A_i con respecto al punto O_A

El subíndice i determina el número de brazo que varía entre 1 y 3, para encontrar el punto f_i se requiere descomponer a la longitud total de cada brazo mediante el triángulo rectángulo conformado por los puntos A_i , f_i , B_i y el ángulo q_{2i} , como se muestra en la Figura 39, para encontrar las componentes r_{xi} y r_{zi} , con el uso de funciones trigonométricas.

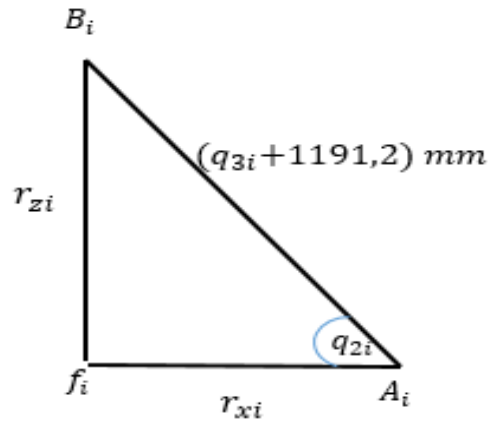


Figura 39. Componentes de la longitud total de los brazos.

Las componentes de la longitud total se obtienen mediante las siguientes expresiones:

$$r_{xi} = (q_{3i} + 1191,2)\cos q_{2i} \text{ mm} \quad (82)$$

$$r_{zi} = (q_{3i} + 1191,2)\sen q_{2i} \text{ mm} \quad (83)$$

Para obtener el vector \vec{g}_i que gobierna la ubicación del punto f_i , se empieza por el concepto de representación de un vector en coordenadas polares mediante (radio, ángulo), el radio se determina con el término r_A que es la distancia que separa a los vértices A_i con el punto O_A , a esta magnitud se resta la componente r_{xi} y el ángulo está definido por la separación angular entre cada vértice añadiendo el desplazamiento q_1 , obteniendo las siguientes expresiones en coordenadas polares:

$$\vec{g}_1 = ((r_A - r_{x1}), q_1) \quad (84)$$

$$\vec{g}_2 = ((r_A - r_{x2}), (120^\circ + q_1)) \quad (85)$$

$$\vec{g}_3 = ((r_A - r_{x3}), (240^\circ + q_1)) \quad (86)$$

Transformando los vectores \vec{g}_i en coordenadas cartesianas y reemplazando (82) se obtiene:

$$\vec{g}_1 = \begin{bmatrix} (r_A - (q_{31} + 1191,2)\cos q_{21})\cos q_1 \\ (r_A - (q_{31} + 1191,2)\cos q_{21})\sen q_1 \\ 0 \end{bmatrix} \quad (87)$$

$$\vec{g}_2 = \begin{bmatrix} (r_A - (q_{32} + 1191,2)\cos q_{22})\cos(120^\circ + q_1) \\ (r_A - (q_{32} + 1191,2)\cos q_{22})\sen(120^\circ + q_1) \\ 0 \end{bmatrix} \quad (88)$$

$$\vec{g}_3 = \begin{bmatrix} (r_A - (q_{33} + 1191,2)\cos q_{23})\cos(240^\circ + q_1) \\ (r_A - (q_{33} + 1191,2)\cos q_{23})\sin(240^\circ + q_1) \\ 0 \end{bmatrix} \quad (89)$$

El vector \vec{h}_i depende la componente r_{zi} , determinando las siguientes expresiones:

$$\vec{h}_1 = \begin{bmatrix} 0 \\ 0 \\ (q_{31} + 1191,2)\sin q_{21} \end{bmatrix} \quad (90)$$

$$\vec{h}_2 = \begin{bmatrix} 0 \\ 0 \\ (q_{32} + 1191,2)\sin q_{22} \end{bmatrix} \quad (91)$$

$$\vec{h}_3 = \begin{bmatrix} 0 \\ 0 \\ (q_{33} + 1191,2)\sin q_{23} \end{bmatrix} \quad (92)$$

Para encontrar el vector \vec{ld}_i que representa la ubicación del vértice B_i con respecto al punto O_A , se utiliza la siguiente expresión vectorial:

$$\vec{ld}_i = \vec{g}_i + \vec{h}_i \quad (93)$$

Reemplazando (87), (88), (89), (90), (91), (92) en (93) obteniendo las siguientes ecuaciones:

$$\vec{ld}_1 = \begin{bmatrix} ld_{1x} \\ ld_{1y} \\ ld_{1z} \end{bmatrix} = \begin{bmatrix} (r_A - (q_{31} + 1191,2)\cos q_{21})\cos q_1 \\ (r_A - (q_{31} + 1191,2)\cos q_{21})\sin q_1 \\ (q_{31} + 1191,2)\sin q_{21} \end{bmatrix} \quad (94)$$

$$\vec{ld}_2 = \begin{bmatrix} ld_{2x} \\ ld_{2y} \\ ld_{2z} \end{bmatrix} = \begin{bmatrix} (r_A - (q_{32} + 1191,2)\cos q_{22})\cos(120^\circ + q_1) \\ (r_A - (q_{32} + 1191,2)\cos q_{22})\sin(120^\circ + q_1) \\ (q_{32} + 1191,2)\sin q_{22} \end{bmatrix} \quad (95)$$

$$\vec{ld}_3 = \begin{bmatrix} ld_{3x} \\ ld_{3y} \\ ld_{3z} \end{bmatrix} = \begin{bmatrix} (r_A - (q_{33} + 1191,2)\cos q_{23})\cos(240^\circ + q_1) \\ (r_A - (q_{33} + 1191,2)\cos q_{23})\sin(240^\circ + q_1) \\ (q_{33} + 1191,2)\sin q_{23} \end{bmatrix} \quad (96)$$

Con la obtención de las coordenadas de los vectores \overrightarrow{ld}_i se cumple con el objetivo de la cinemática directa para esta aplicación en el siguiente punto se desarrolla el otro método directa para comparar resultados.

3.3.2.2 Convención Denavit Hartenberg

Este método permite representar la cinemática de estructuras robóticas conformadas por articulaciones y eslabones mediante sistemas matriciales. El funcionamiento de esta metodología se basa en la colocación de los sistemas de coordenadas en cada articulación de acuerdo al sentido de movimiento de la misma con la ayuda de la regla de la mano derecha, para modelar la cinemática utiliza la parametrización de las coordenadas en cuatro variables que se detallan en la Tabla 10.

Tabla 10.

Parámetros Denavit Hartenberg

Parámetro	Descripción
θ_i	Ángulo de articulación
d_i	Distancia de articulación
α_i	Ángulo de enlace
a_i	Distancia de enlace

Cada articulación de los brazos del manipulador determina de un grado de libertad, como se trabaja con las activas cada uno tiene tres grados de libertad, además se utiliza las variables del método vectorial ($q_1, q_{2i}, q_{3i}, r_A, ld_i$) y el punto de origen igualmente es el O_A con la ubicación de ejes coordenados x, y, z con respecto al método anterior, para desarrollar este método se lo gráfica con la herramienta Robotics de MATLAB, como se muestra en la Figura 40 y los parámetros Denavit Hartenberg para cada brazo en la Tabla 11, Tabla 12 y Tabla 13 respectivamente.

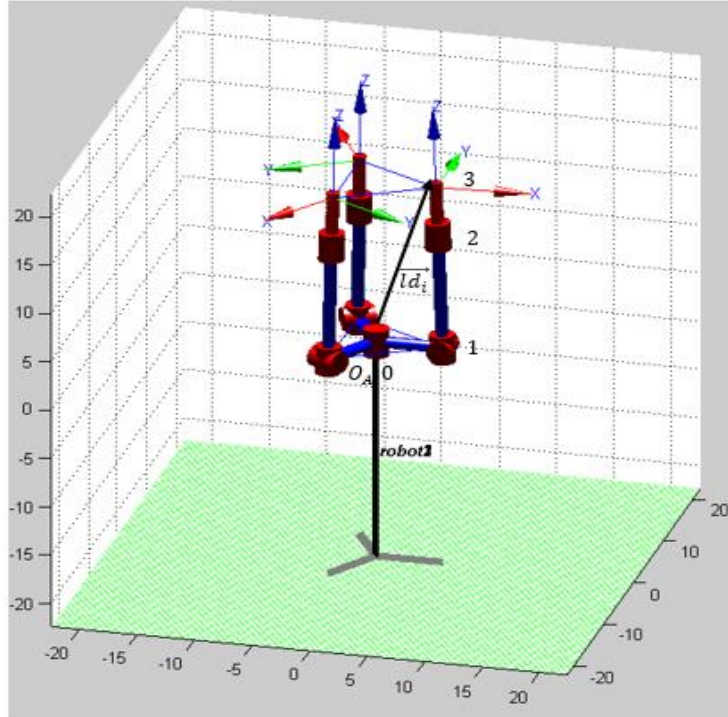


Figura 40. Manipulador R-3RPS graficado en Robotics de MATLAB

Tabla 11.

Parámetros Denavit Hartenberg para el brazo 1

Grados de libertad	θ_i	d_i	α_i	a_i
1	q_1	0	90°	r_A
2	q_{21}	1191.2	-90°	0
3	0°	q_{31}	0°	0

Tabla 12.

Parámetros Denavit Hartenberg para el brazo 2

Grados de libertad	θ_i	d_i	α_i	a_i
1	$q_1 + 120^\circ$	0	90°	r_A
2	q_{22}	1191.2	-90°	0
3	0°	q_{32}	0°	0

Tabla 13.

Parámetros Denavit Hartenberg para el brazo 3

Grados de libertad	θ_i	d_i	α_i	a_i
1	$q_1 + 240^\circ$	0	90°	r_A
2	q_{23}	1191.2	-90°	0
3	0°	q_{33}	0°	0

El siguiente paso es reemplazar los parámetros definidos en la matriz de transformación de coordenadas que se expresa de la siguiente manera:

$$T = \begin{bmatrix} \cos\theta_i & -\cos\alpha_i \sin\theta_i & \sin\alpha_i \sin\theta_i & a_i \cos\alpha_i \\ \sin\theta_i & \cos\alpha_i \cos\theta_i & -\sin\alpha_i \cos\theta_i & a_i \sin\alpha_i \\ 0 & \sin\alpha_i & \cos\alpha_i & d_i \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (97)$$

Mediante el reemplazo de los parámetros de las tablas y multiplicación de matrices con respecto al punto O_A , se obtiene las coordenadas de los vectores posición \vec{ld}_i , expresado de la siguiente manera:

$$\vec{ld}_1 = T_0^3 = T_0^1 x T_1^2 x T_2^3 x O_A \quad (98)$$

Realizando las operaciones matriciales se llega a las coordenadas del vector posición mencionado, como se muestra a continuación:

$$\vec{ld}_1 = \begin{bmatrix} ld_{1x} \\ ld_{1y} \\ ld_{1z} \\ 1 \end{bmatrix} = \begin{bmatrix} r_A \cos q_{21} \cos q_1 \\ r_A \sin q_{21} \cos q_1 \\ (q_{31} + 1191,2) + r_A \sin q_1 \sin q_{21} \\ 1 \end{bmatrix} \quad (99)$$

$$\vec{ld}_2 = \begin{bmatrix} ld_{2x} \\ ld_{2y} \\ ld_{2z} \\ 1 \end{bmatrix} = \begin{bmatrix} r_A \cos q_{22} \cos(120^\circ + q_1) \\ r_A \sin q_{22} \cos(120^\circ + q_1) \\ (q_{32} + 1191,2) + r_A \sin(120^\circ + q_1) \sin q_{22} \\ 1 \end{bmatrix} \quad (100)$$

$$\vec{ld}_3 = \begin{bmatrix} ld_{3x} \\ ld_{3y} \\ ld_{3z} \\ 1 \end{bmatrix} = \begin{bmatrix} r_A \cos q_{23} \cos(240^\circ + q_1) \\ r_A \sin q_{23} \cos(240^\circ + q_1) \\ (q_{33} + 1191,2) + r_A \sin(240^\circ + q_1) \sin q_{23} \\ 1 \end{bmatrix} \quad (101)$$

Con las coordenadas de los vectores posición obtenidas se compara resultados con el método vectorial, para la elección de la opción que más se acerque a la cinemática inversa.

La convención de Denavit Hartenberg es un método que parametriza el movimiento de las articulaciones de un manipulador en cuatro variables, por esta razón en cada grado de libertad

cambia la orientación de los ejes coordenados lo que provoca que la ubicación del efector final no se relacione con la posición de los ejes del punto de origen, para la verificación del funcionamiento de este se utiliza herramientas específicas como Robotics en donde puede comprobar la ubicación del efector final de manera visual, por eso al momento de comprobar los resultados de este método con respecto a los ejes coordenados del punto O_A , se alejan con respecto al modelo inverso, pero al utilizar la herramienta y sobreponiendo los puntos de la cinemática inversa se puede observar la proximidad de ambos resultados, pero para las secciones de velocidad y aceleración se requiere de las posiciones referenciadas al mismo sistema coordenado ya que se van relacionar los resultados de ambos modelos para encontrar las magnitudes mencionadas en los actuadores, por este motivo el método conveniente es el vectorial para utilizar como modelo directo. La diferencia del método de los ángulos de Euler con la convención de Denavit Hartenberg que el método inverso busca la transformación del sistema de la base móvil en función del sistema de la plataforma giratoria dejando el modelo inverso referenciado en un solo sistema coordenado, mientras que Denavit Hartenberg realiza transformaciones en base a las articulaciones que cambia la posición y sentido de los ejes coordenados iniciales.

Con la elección del método directo en las siguientes secciones el objetivo es encontrar las velocidades y aceleraciones de los actuadores a partir de los puntos encontrados.

3.4 Velocidad

El objetivo en este punto es encontrar las expresiones de las velocidades de los actuadores en función de velocidades de los ángulos de Euler y datos de posición de ambos modelos, debido a que las especificaciones de las ilusiones de trabajo son en función de parámetros de posición y velocidad de la cinemática inversa como datos de entrada. Para desarrollar este punto se puede comenzar de dos opciones la primera es derivar las expresiones de los desplazamientos de las

articulaciones obtenidas en el modelo inverso y la otra es derivar los puntos encontrados en ambos modelos para relacionarlos para la expresión requerida.

La primera opción no se utiliza debido a que el derivar las expresiones obtenidas resulta una operación compleja de realizar por la cantidad de variables que manejan en cada ecuación y además no permite el análisis de singularidades de movimiento del manipulador, por esta razón se usa la opción de relacionar ambos modelos.

Para la determinación de velocidades de los actuadores activos y análisis de singularidades con la opción elegida, se utiliza la matriz Jacobiana de velocidad inversa y directa, esta matriz permite obtener la velocidad espacial instantánea de los efectores finales del manipulador mediante datos de posición (Vázquez Hernández & Cuenca Jiménez, 2009) (Hernando, 2017), velocidad articulares o cartesianas dependiendo si el modelo es directo o inverso. La matriz Jacobiana se encuentra derivando expresiones de posición (p) con respecto al tiempo y relacionando las velocidades lineales, angulares ya sea en el espacio articular $J_q(q)\dot{q}$ (102) o en el espacio cartesiano

$$J_x(\vartheta)\dot{\theta} \quad (103)$$

$$J_q(q)\dot{q} = \begin{bmatrix} \frac{\partial p}{\partial q_1} & \frac{\partial p}{\partial q_2} & \frac{\partial p}{\partial q_3} \\ \frac{\partial p}{\partial q_1} & \frac{\partial p}{\partial q_2} & \frac{\partial p}{\partial q_3} \\ \frac{\partial p}{\partial q_1} & \frac{\partial p}{\partial q_2} & \frac{\partial p}{\partial q_3} \\ \frac{\partial p}{\partial q_1} & \frac{\partial p}{\partial q_2} & \frac{\partial p}{\partial q_3} \end{bmatrix} \begin{bmatrix} \dot{q}_1 \\ \dot{q}_2 \\ \dot{q}_3 \end{bmatrix} \quad (102)$$

$$J_x(\vartheta)\dot{\theta} = \begin{bmatrix} \frac{\partial p}{\partial x} & \frac{\partial p}{\partial y} & \frac{\partial p}{\partial \theta} \\ \frac{\partial p}{\partial x} & \frac{\partial p}{\partial y} & \frac{\partial p}{\partial \theta} \\ \frac{\partial p}{\partial x} & \frac{\partial p}{\partial y} & \frac{\partial p}{\partial \theta} \\ \frac{\partial p}{\partial x} & \frac{\partial p}{\partial y} & \frac{\partial p}{\partial \theta} \end{bmatrix} \begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{\theta} \end{bmatrix} \quad (103)$$

Con la descripción de la opción escogida, el siguiente paso es derivar las expresiones de los vectores posición de los vértices de la base móvil encontrados en cada modelo.

3.4.1 Velocidad del modelo inverso

En este caso se deriva con respecto al tiempo los vectores posición \vec{l}_i para encontrar los vectores de velocidad \vec{vl}_i , mediante la siguiente expresión:

$$\vec{vl}_i = \frac{d\vec{l}_i}{dt} \quad (104)$$

Derivando los vectores posición se encuentran los vectores de velocidad mostrados a continuación:

$$\vec{vl}_i = \begin{bmatrix} vl_{ix} \\ vl_{iy} \\ vl_{iz} \end{bmatrix} = \begin{bmatrix} \dot{l}_{ix} \\ \dot{l}_{iy} \\ \dot{l}_{iz} \end{bmatrix} \quad (105)$$

El subíndice i indica el número de brazo, las expresiones $\dot{l}_{ix}, \dot{l}_{iy}, \dot{l}_{iz}$ son funciones dependientes de las variables de velocidad $(\dot{\phi}, \dot{\theta}, \dot{\gamma})$, que son datos de entrada conocidos en las especificaciones de las ilusiones de vuelo, a partir de estos parámetros se conforma la matriz jacobiana de velocidad del modelo inverso, como las expresiones desarrolladas son extensas se las denomina k_{ij} , donde j es el subíndice de posicionamiento de términos dentro de la matriz, además las expresiones k son dependientes de las variables de posición (ϕ, θ, γ) , que son términos independientes, p_z también es de la misma naturaleza de las variables mencionadas pero al considerarlo a una referencia de altura constante al derivar se hace 0, la formación de la matriz queda de la siguiente manera:

$$\vec{vl}_i = \begin{bmatrix} k_{i1} & k_{i4} & k_{i7} \\ k_{i2} & k_{i5} & k_{i8} \\ k_{i3} & k_{i6} & k_{i9} \end{bmatrix} \begin{bmatrix} \dot{\phi} \\ \dot{\theta} \\ \dot{\gamma} \end{bmatrix} \quad (106)$$

A esta matriz se la denomina con la nomenclatura $J_x(\vartheta)$, esta expresión permite analizar la presencia de singularidades en el movimiento del manipulador.

$$\overrightarrow{v}l_i = J_x(\vartheta) \begin{bmatrix} \dot{\phi} \\ \dot{\theta} \\ \dot{\gamma} \end{bmatrix} \quad (107)$$

3.4.1.1 Singularidades de velocidad en el modelo inverso

Las singularidades son interminaciones que producen que el manipulador pierda o gane movilidad o también grados de libertad para el análisis de estas expresiones usa el determinante de la matriz $J_x(\vartheta)$ evaluando las siguientes condiciones:

$$\det(J_x(\vartheta)) = 0 \quad (108)$$

$$\det(J_x(\vartheta)) = \infty \quad (109)$$

Si no se cumplen ninguna de las condiciones el movimiento no tiene indeterminaciones, el análisis de la matriz se lo realiza en puntos críticos como son los ángulos 90° , 180° , 270° para el caso del manipulador R-3RPS, las variables analizadas son el alabeo y cabeceo ya que la guiñada por las expresiones obtenidas no produce singularidades, los ángulos mencionados bajo las especificaciones de funcionamiento no alcanzan los puntos críticos por esta razón se los analiza en $\pm 45^\circ$ donde las funciones seno, coseno tienen el mismo valor pero el signo depende del cuadrante ubicado y también se verifica los ángulos en 0° . En las pruebas realizadas con los ángulos mencionados se obtuvo determinantes de la matriz diferentes de 0, por este motivo se determina que el modelo de velocidad inversa no tiene singularidades.

3.4.2 Velocidad del modelo directo

En este punto se obtiene el vector velocidad $\overrightarrow{v}l_i$ a partir de la derivación con respecto al tiempo, los vectores posición $\overrightarrow{l}d_i$, de la siguiente manera:

$$\overrightarrow{vld}_i = \frac{d\overrightarrow{ld}_i}{dt} \quad (110)$$

Derivando los vectores posición se encuentran los vectores de velocidad mostrados a continuación:

$$\overrightarrow{vld}_i = \begin{bmatrix} vld_{ix} \\ vld_{iy} \\ vld_{iz} \end{bmatrix} = \begin{bmatrix} \dot{ld}_{ix} \\ \dot{ld}_{iy} \\ \dot{ld}_{iz} \end{bmatrix} \quad (111)$$

El subíndice i indica el número de brazo, las expresiones $\dot{ld}_{ix}, \dot{ld}_{iy}, \dot{ld}_{iz}$ son funciones dependientes de las variables de velocidad $(\dot{q}_1, \dot{q}_{2i}, \dot{q}_{3i})$, para estos los parámetros de entrada son los vectores de velocidad inversa para encontrar las variables de velocidad de las articulaciones activas, en este caso también se forma la matriz jacobiana directa formada por los términos kv_{ij} que son referenciados a las variables de posición directa (q_1, q_{2i}, q_{3i}) , expresada de la siguiente forma:

$$\overrightarrow{vld}_i = \begin{bmatrix} kv_{i1} & kv_{i4} & kv_{i7} \\ kv_{i2} & kv_{i5} & kv_{i8} \\ kv_{i3} & kv_{i6} & kv_{i9} \end{bmatrix} \begin{bmatrix} \dot{q}_1 \\ \dot{q}_{2i} \\ \dot{q}_{3i} \end{bmatrix} \quad (112)$$

A esta matriz se la denomina con la nomenclatura $J_q(q)$, con la aclaración que los términos de la matriz directa son dependientes de la cinemática inversa ya los desplazamientos (q_{2i}, q_{3i}) son de funciones de los ángulos de alabeo, cabeceo y guiñada, mientras que q_1 es asociado al giro de la plataforma, se analiza el determinante de la matriz para evaluar singularidades en el modelo directo.

$$\overrightarrow{vld}_i = J_q(q) \begin{bmatrix} \dot{q}_1 \\ \dot{q}_{2i} \\ \dot{q}_{3i} \end{bmatrix} \quad (113)$$

3.4.2.1 Singularidades de velocidad en el modelo directo

Para evaluar las singularidades se utiliza la determinante de la matriz $J_q(q)$ con los mismos valores de interminación de la cinemática inversa cero e infinito, como esta matriz es dependiente del modelo inverso se analiza en los mismos valores de la matriz jacobiana inversa en alabeo, cabeceo en $\pm 45^\circ$ y 0° . La pruebas en los valores mencionados mostraron el determinante en cualquiera de los casos es diferente de cero, por este motivo no hay singularidades en el modelo de velocidad directo.

3.4.3 Determinación de la velocidad de los actuadores

Con la determinación de la no presencia de singularidades tanto en el modelo inverso y directo, se encuentran las expresiones para encontrar las velocidades de los actuadores mediante la siguiente igualdad:

$$\overrightarrow{vl}_i = \overrightarrow{vld}_i \quad (114)$$

El vector de velocidad \overrightarrow{vl}_i son datos conocidos por lo que con la expresión planteada en (114), se busca hallar las variables $(\dot{q}_1, \dot{q}_{2l}, \dot{q}_{3l})$, como se observa el término \dot{q}_1 , no está en función del número de brazo, debido a que es el giro de la plataforma y es el mismo para los tres brazos, por esta razón se comprobaba los resultados mediante la obtención de ese valor en los tres puntos, \dot{q}_{1l} el cual debe ser igual, con esta aclaración se muestra la expresión a resolver a continuación:

$$\begin{bmatrix} vl_{ix} \\ vl_{iy} \\ vl_{iz} \end{bmatrix} = \begin{bmatrix} kv_{i1} & kv_{i4} & kv_{i7} \\ kv_{i2} & kv_{i5} & kv_{i8} \\ kv_{i3} & kv_{i6} & kv_{i9} \end{bmatrix} \begin{bmatrix} \dot{q}_{1l} \\ \dot{q}_{2l} \\ \dot{q}_{3l} \end{bmatrix} \quad (115)$$

Resolviendo (115) mediante determinantes se obtienen las siguientes expresiones:

$$q_{1i} = \frac{\det \begin{pmatrix} vl_{ix} & kv_{i4} & kv_{i7} \\ vl_{iy} & kv_{i5} & kv_{i8} \\ vl_{iz} & kv_{i6} & kv_{i9} \end{pmatrix}}{\det(J_q(q))} \quad (116)$$

$$q_{2i} = \frac{\det \begin{pmatrix} kv_{i1} & vl_{ix} & kv_{i7} \\ kv_{i2} & vl_{iy} & kv_{i8} \\ kv_{iz} & vl_{iz} & kv_{i9} \end{pmatrix}}{\det(J_q(q))} \quad (117)$$

$$q_{3i} = \frac{\det \begin{pmatrix} kv_{i1} & kv_{i4} & vl_{ix} \\ kv_{i2} & kv_{i5} & vl_{iy} \\ kv_{iz} & kv_{i6} & vl_{iz} \end{pmatrix}}{\det(J_q(q))} \quad (118)$$

3.5 Aceleración

En este punto se realiza un procedimiento igual al de velocidad, deriva las expresiones de los vectores de velocidad con respecto al tiempo de ambos modelos, para encontrar las expresiones de las aceleraciones de los actuadores. En este caso la matriz Jacobiana de aceleración varia con respecto a la velocidad en que aumenta un término tanto en el espacio articular (119) como en el cartesiano (120), que son componente de velocidad asociados a la matriz Jacobiana de velocidad derivada (Pérez Menéndez, 2014), este término en el análisis se lo considera como variables sin singularidades debido que se basa en parámetros ya evaluados en velocidad, mientras la matriz asociada a términos de aceleración es la misma matriz Jacobiana de velocidad.

$$\overline{ad} = J_q(q)(\ddot{q}) + J_q(\dot{q})(\dot{q}) \quad (119)$$

$$\vec{a} = J_x(\vartheta)(\ddot{\theta}) + J_x(\dot{\vartheta})(\dot{\theta}) \quad (120)$$

3.5.1 Aceleración del modelo inverso

Se deriva con respecto al tiempo los vectores de velocidad $v\vec{l}_i$ para encontrar los vectores de aceleración $\vec{a}l_i$, mediante la siguiente expresión:

$$\vec{a}l_i = \frac{d v\vec{l}_i}{dt} \quad (121)$$

Derivando los vectores de velocidad se encuentran los vectores de aceleración mostrados a continuación:

$$\vec{a}l_i = \begin{bmatrix} al_{ix} \\ al_{iy} \\ al_{iz} \end{bmatrix} = \begin{bmatrix} v\dot{l}_{ix} \\ v\dot{l}_{iy} \\ v\dot{l}_{iz} \end{bmatrix} = \begin{bmatrix} \ddot{l}_{ix} \\ \ddot{l}_{iy} \\ \ddot{l}_{iz} \end{bmatrix} \quad (122)$$

El subíndice i indica el número de brazo, las expresiones $v\dot{l}_{ix}, v\dot{l}_{iy}, v\dot{l}_{iz}$ son funciones dependientes de las variables de aceleración $(\ddot{\phi}, \ddot{\theta}, \ddot{\gamma})$, que son datos que se calculan mediante ecuaciones de movimiento, parámetros de posición y velocidad, a partir de estos términos se conforma la matriz jacobiana de aceleración del modelo inverso, pero al construir la misma aparecen términos de velocidad en función de las variables $(\dot{\phi}\dot{\theta}, \dot{\phi}\dot{\gamma}, \dot{\gamma}\dot{\theta}, \dot{\phi}^2, \dot{\theta}^2, \dot{\gamma}^2)$, que se denota mediante el vector \overrightarrow{compv}_i , los términos de la matriz son expresiones extensas por lo que se las denomina ka_{ij} , donde j es el subíndice de posicionamiento de términos dentro de la matriz, además las expresiones ka son dependientes de las variables de posición (ϕ, θ, γ) , que son términos independientes, la formación de la expresión con la matriz jacobiana de aceleración queda de la siguiente manera:

$$\vec{a}l_i = \begin{bmatrix} ka_{i1} & ka_{i4} & ka_{i7} \\ ka_{i2} & ka_{i5} & ka_{i8} \\ ka_{i3} & ka_{i6} & ka_{i9} \end{bmatrix} \begin{bmatrix} \ddot{\phi} \\ \ddot{\theta} \\ \ddot{\gamma} \end{bmatrix} + \begin{bmatrix} compv_{ix} \\ compv_{iy} \\ compv_{iz} \end{bmatrix} \quad (123)$$

A esta matriz se la denomina con la nomenclatura $J_x(\vartheta)$, esta expresión permite analizar la presencia de singularidades de aceleración del manipulador.

$$\overrightarrow{al}_i = J_x(\vartheta) \begin{bmatrix} \ddot{\phi} \\ \ddot{\theta} \\ \ddot{\gamma} \end{bmatrix} + \begin{bmatrix} compv_{ix} \\ compv_{iy} \\ compv_{iz} \end{bmatrix} \quad (124)$$

3.5.1.1 Singularidades de aceleración en el modelo inverso

Al tener la misma matriz de velocidad inversa y el vector \overrightarrow{compv}_i no se evalúa debido a que contiene componentes que ya se probaron en el punto de singularidades de velocidad en el modelo inverso, se concluye que no hay singularidades de aceleración en este modelo.

3.5.2 Aceleración del modelo directo

Se deriva con respecto al tiempo los vectores de velocidad \overrightarrow{vld}_i para encontrar los vectores de aceleración \overrightarrow{ald}_i , mediante la siguiente expresión:

$$\overrightarrow{ald}_i = \frac{d\overrightarrow{vld}_i}{dt} \quad (125)$$

Derivando los vectores de velocidad se encuentran los vectores de aceleración mostrados a continuación:

$$\overrightarrow{ald}_i = \begin{bmatrix} ald_{ix} \\ ald_{iy} \\ ald_{iz} \end{bmatrix} = \begin{bmatrix} vld_{ix} \\ vld_{iy} \\ vld_{iz} \end{bmatrix} = \begin{bmatrix} ld_{ix} \\ ld_{iy} \\ ld_{iz} \end{bmatrix} \quad (126)$$

El subíndice i indica el número de brazo, las expresiones $vld_{ix}, vld_{iy}, vld_{iz}$ son funciones dependientes de las variables de aceleración ($\ddot{q}_1, \ddot{q}_{2i}, \ddot{q}_{3i}$), que son datos que se buscan encontrar mediante la relación con los vectores de aceleración del modelo inverso, a partir de estos términos se conforma la matriz jacobiana de aceleración del modelo directo, pero al construir la misma aparecen términos de velocidad en función de las variables ($\dot{q}_1 \dot{q}_{2i}, \dot{q}_1 \dot{q}_{3i}, \dot{q}_{2i} \dot{q}_{3i}, \dot{q}_1^2, \dot{q}_{2i}^2, \dot{q}_{3i}^2$),

que se denota mediante el vector $\overrightarrow{compvd}_l$, los términos de la matriz son expresiones extensas por lo que se las denomina kad_{ij} , donde j es el subíndice de posicionamiento de términos dentro de la matriz, además las expresiones kad son dependientes de las variables de desplazamiento de las articulaciones (q_1, q_{2l}, q_{3l}) , que son términos independientes, la formación de la expresión con la matriz jacobiana de aceleración queda de la siguiente manera:

$$\overrightarrow{ald}_l = \begin{bmatrix} kad_{i1} & kad_{i4} & kad_{i7} \\ kad_{i2} & kad_{i5} & kad_{i8} \\ kad_{i3} & kad_{i6} & kad_{i9} \end{bmatrix} \begin{bmatrix} \ddot{q}_1 \\ \ddot{q}_{2l} \\ \ddot{q}_{3l} \end{bmatrix} + \begin{bmatrix} compvd_{ix} \\ compvd_{iy} \\ compvd_{iz} \end{bmatrix} \quad (127)$$

A esta matriz se la denomina con la nomenclatura $J_q(q)$, esta expresión permite analizar la presencia de singularidades de aceleración en las articulaciones activas del manipulador.

$$\overrightarrow{ald}_l = J_q(q) \begin{bmatrix} \ddot{q}_1 \\ \ddot{q}_{2l} \\ \ddot{q}_{3l} \end{bmatrix} + \begin{bmatrix} compvd_{ix} \\ compvd_{iy} \\ compvd_{iz} \end{bmatrix} \quad (128)$$

3.5.2.1 Singularidades de aceleración en el modelo directo

Al tener la misma matriz de velocidad directa y el vector $\overrightarrow{compvd}_l$ no se evalúa debido a que contiene componentes que ya se probaron en el punto de singularidades de velocidad en el modelo directo, se concluye que no hay singularidades de aceleración en este modelo.

3.5.3 Determinación de la aceleración de los actuadores

Sin la presencia de singularidades de aceleración tanto en el modelo inverso y directo, se encuentran las expresiones para encontrar las aceleraciones de los actuadores mediante la siguiente igualdad:

$$\overrightarrow{al}_l = \overrightarrow{ald}_l \quad (129)$$

El vector de aceleración \overrightarrow{al}_l son datos calculados a partir de ecuaciones de movimiento, velocidad y posición de los ángulos de Euler, por lo que con la expresión planteada en (129), se

busca hallar las variables $(\ddot{q}_1, \ddot{q}_{2l}, \ddot{q}_{3l})$, como se observa el término \ddot{q}_1 , no está en función del número de brazo, debido a que es el giro de la plataforma y es el mismo para los tres brazos, por esta razón se comprobara los resultados mediante la obtención de ese valor en los tres puntos, \ddot{q}_{1l} , el cual debe ser igual, con esta aclaración se muestra la expresión a resolver a continuación:

$$\begin{bmatrix} al_{ix} \\ al_{iy} \\ al_{iz} \end{bmatrix} = \begin{bmatrix} kad_{i1} & kad_{i4} & kad_{i7} \\ kad_{i2} & kad_{i5} & kad_{i8} \\ kad_{i3} & kad_{i6} & kad_{i9} \end{bmatrix} \begin{bmatrix} \ddot{q}_1 \\ \ddot{q}_{2l} \\ \ddot{q}_{3l} \end{bmatrix} + \begin{bmatrix} compvd_{ix} \\ compvd_{iy} \\ compvd_{iz} \end{bmatrix} \quad (130)$$

Resolviendo (130) mediante determinantes se obtienen las siguientes expresiones:

$$\ddot{q}_{1l} = \frac{\det \begin{pmatrix} al_{ix} - compvd_{ix} & kad_{i4} & kad_{i7} \\ al_{iy} - compvd_{iy} & kad_{i5} & kad_{i8} \\ al_{iz} - compvd_{iz} & kad_{i6} & kad_{i9} \end{pmatrix}}{\det(J_q(q))} \quad (131)$$

$$\ddot{q}_{2l} = \frac{\det \begin{pmatrix} kad_{i1} & al_{ix} - compvd_{ix} & kad_{i7} \\ kad_{i2} & al_{iy} - compvd_{iy} & kad_{i8} \\ kad_{i3} & al_{iz} - compvd_{iz} & kad_{i9} \end{pmatrix}}{\det(J_q(q))} \quad (132)$$

$$\ddot{q}_{3l} = \frac{\det \begin{pmatrix} kad_{i1} & kad_{i4} & al_{ix} - compvd_{ix} \\ kad_{i2} & kad_{i5} & al_{iy} - compvd_{iy} \\ kad_{i3} & kad_{i6} & al_{iz} - compvd_{iz} \end{pmatrix}}{\det(J_q(q))} \quad (133)$$

3.6 Control cinemático por interpolación de trayectorias

Con los modelos de posición, velocidad y aceleración tanto inverso como directo desarrollados el siguiente paso es probarlos mediante las trayectorias de las ilusiones vestibulares de vuelo, que están especificadas a través de los ángulos de alabeo, cabeceo y guiñada. Para realizar este procedimiento se utiliza la interpolación de trayectorias que permite integrar puntos intermedios dentro de las mismas, a este método en varios trabajos se lo denomina control cinemático, debido que a partir de los puntos colocados se puede encontrar y manejar variables de posición, velocidad

y aceleración de los actuadores en un manipulador robótico. Las especificaciones de cada ilusión son parámetros de la cinemática inversa, por lo que las trayectorias de posición, velocidad y aceleración son en base a estas variables que mediante los modelos cinemáticos obtenidos se encuentran los parámetros de los actuadores, también se consideran conceptos de control de actuadores para generar las trayectorias de trabajo.

La interpolación de trayectorias utiliza las herramientas llamadas interpoladores que son funciones polinomiales de diferentes órdenes que permiten integrar puntos en un periodo de tiempo, los interpoladores permiten definir la forma y convergencia de las trayectorias, para la generación de caminos se requiere las condiciones iniciales y finales de posición, velocidad y aceleración de cada trayectoria.

Las ilusiones de vuelo que se van a generar mediante este método son la barrena y coriolis, las cuales se van a dividir en tramos que tendrán cada uno condiciones iniciales y finales, además en cada uno de ellos se define el tipo de movimiento sea constante o acelerado, en algunos tramos no se encuentra especificada la velocidad, por esta razón se trabaja con la velocidad máxima, como los parámetros son basados en los ángulos de Euler para calcular los datos faltantes como aceleración y tiempo se utilizan las ecuaciones de movimiento circular uniforme o variado dependiendo de los requerimientos de cada tramo. La velocidad máxima especificada en la Tabla 3 es de $90^\circ/\text{s}$ para alabeo (Φ), cabeceo (θ), pero la misma es muy elevada para los desplazamientos angulares que con la velocidad mencionada trabaja en el orden de los milisegundos generado un movimiento brusco y acelerado, por esta razón se usa la décima parte de la magnitud mencionada siendo igual a $9^\circ/\text{s}$, En el caso de la guiñada (γ), se utiliza la velocidad $80.24^\circ/\text{s}$, ya que el desplazamiento es de 360° para cada ilusión y esto permite trabajar en segundos, además se requiere que la misma genere un movimiento giratorio constante.

Para los actuadores prismáticos en cada tramo se considera el concepto de control que ante un cambio de sentido de movimiento la velocidad y aceleración debe llegar a 0, con las aclaraciones mencionadas se especifica las condiciones de posición de cada tramo de las ilusiones barrena y coriolis, tipo de movimiento en las Tablas 14 y 15.

Tabla 14.

Condiciones de posición de los tramos de la ilusión barrena

Tramo/Variable	1-2	3-4	5	6-7	8
ϕ_o	0°	0°	0°	0°	0°
ϕ_f	0°	0°	0°	15°-20°	15°-20°
θ_o	0°	20°-30°	0°	0°	0°
θ_f	20°-30°	0°	0°	-20°- -30°	-20°- -30°
γ_o	0°	0°	0°	0°	0°
γ_f	0°	0°	0°	0°	360°
tipo	variado	variado	nulo	variado	constante

Tabla 15.

Condiciones de posición de los tramos de la ilusión coriolis

Tramo/Variable	1-2	3
ϕ_o	0°	0°
ϕ_f	30°-45°	30°-45°
θ_o	0°	0°
θ_f	0°	0°
γ_o	0°	0°
γ_f	0°	360°
tipo	variado	constante

Antes de definir el tipo de movimiento, se hizo pruebas con todos los tramos con el movimiento circular uniforme, pero al probarlo en los modelo cinemáticos obtenidos se pudo observar que el comportamiento de velocidad y aceleración de los actuadores prismáticos fue en sentido contrario a lógica de movilidad, por esta razón se utilizó el movimiento circular variado para posición de roll y pitch, en las Tablas 16, 17 se detalla las condiciones de velocidad angular inicial y final.

Tabla 16.*Condiciones de velocidad de los tramos de la ilusión barrena*

Tramo/Variable	1	2	3	4	5	6	7	8
$\dot{\phi}_o$	0°	0°	0°	0°	0°	0°	9°	0°
$\dot{\phi}_f$	0°	0°	0°	0°	0°	9°	0°	0°
$\dot{\theta}_o$	0°	9°	0°	-9°	0°	0°	-9°	0°
$\dot{\theta}_f$	9°	0°	-9°	0°	0°	-9°	0°	0°
$\dot{\gamma}$	0°	0°	0°	0°	0°	0°	0°	80.241°

Tabla 17.*Condiciones de velocidad de los tramos de la ilusión coriolis*

Tramo/Variable	1	2	3
$\dot{\phi}_o$	0°	9°	0°
$\dot{\phi}_f$	9°	0°	0°
$\dot{\theta}_o$	0°	0°	0°
$\dot{\theta}_f$	0°	0°	0°
$\dot{\gamma}$	0°	0°	4.5°

Definidas las condiciones de posición y velocidad para cada tramo, los datos faltantes son la aceleración y tiempo de cada trama para definir el periodo total de cada ilusión, para la obtención de los parámetros se utilizan las ecuaciones de movimiento circular variado para alabeo y cabeceo, mientras que para los datos de guiñada se utiliza el movimiento circular uniforme debido a que en el tramo que se activa la guiñada es constante en velocidad. A partir de las siguientes ecuaciones de movimiento se encuentra los parámetros requeridos:

$$t = \frac{2\vartheta}{(w_o + w_f)} \quad (134)$$

$$a = \frac{w_f - w_o}{t} \quad (135)$$

$$t = \frac{\vartheta}{w} \quad (136)$$

Los términos ϑ , a son el desplazamiento y aceleración angular respectivamente, las ecuaciones (134), (135) son utilizadas para alabeo y cabeceo debido a son gobernadas por el movimiento circular variado donde aparecen los términos de velocidad angular inicial y final (w_o, w_f), para la guiñada el movimiento es constante por lo se usa la ecuación (136) para encontrar el tiempo, mientras que la aceleración para esta variable es 0, reemplazando los parámetros correspondientes en las ecuaciones se obtienen los datos restantes que muestra en las Tablas 18 y 19.

Tabla 18.*Condiciones de aceleración y tiempo de los tramos de la ilusión barrena*

Tramo/Variable	1-2	3-4	5	6-7	8
$\ddot{\vartheta}$ ($^{\circ}/s^2$)	0	0	0	1.35 - 4.07	0
$\ddot{\theta}$ ($^{\circ}/s^2$)	2.7 - 4.05	-2.7 - -4.05	0	-2.7 - -4.07	0
t (s)	4.44 -6.66	4.44 - 6.66	2	4.44 - 6.66	5

Tabla 19.*Condiciones de aceleración y tiempo de los tramos de la ilusión coriolis*

Tramo/Variable	1	2	3
$\ddot{\vartheta}$ ($^{\circ}/s^2$)	0.9- 2.7	0.9 - 2.7	0
$\ddot{\theta}$ ($^{\circ}/s^2$)	0	0	0
t (s)	3.33 - 5	3.33 -5	80

Como se observa en las condiciones de posición, aceleración y tiempo son rango planteados en los datos debido a que se puede dar combinaciones en los ángulos de alabeo y cabeceo, para la ilusión barrena y de solo alabeo en la coriolis, en la barrena se pude dar 4 casos pero en el caso de alabeo en 20° y cabeceo en 30° no se utiliza debido a una restricción mecánica que afecta a las juntas esféricas, mientras en la coriolis se producen 2 combinaciones, en las Tablas 20 y 21 se

detalla los valores de posición, aceleración y tiempo para las combinaciones planteadas de las ilusiones:

Tabla 20.

Condiciones de las combinaciones de la ilusión barrena

Nombre/Variable	Barrena 1	Barrena 2	Barrena 3
ϕ ($^{\circ}$)	15	15	20
θ ($^{\circ}$)	± 20	± 30	± 20
$\ddot{\phi}$ ($^{\circ}/s^2$)	± 3.05	± 1.35	± 4.05
$\ddot{\theta}$ ($^{\circ}/s^2$)	± 4.05	± 2.7	4.05
t (s)	20.32	26.98	20.32

Tabla 21.

Condiciones de las combinaciones de la ilusión coriolis

Tramo/Variable	Coriolis 1	Coriolis 2
ϕ ($^{\circ}$)	30	45
θ ($^{\circ}$)	0	0
$\ddot{\phi}$ ($^{\circ}/s^2$)	1.35	0.9
$\ddot{\theta}$ ($^{\circ}/s^2$)	0	0
t (s)	86.66	90

El tiempo indicado es el tiempo total con las variables planteadas en cada ilusión, las observaciones de los datos mostrados son:

- En barrena 3 se muestra que la aceleración en alabeo como en cabeceo es la misma debido que ambos términos son iguales en valor absoluto en el tramo que se activan los dos.
- En barrena 2 se muestra la reducción de la aceleración del alabeo con respecto al cabeceo debido a que el ángulo de alabeo es de 15° y el cabeceo es -30° , si se hace el que ambos ángulos tengan las mismas aceleraciones el alabeo se realiza más lo produce un cambio brusco de orientación de los actuadores prismáticos que pueden provocar fallos en los mismos, por esta razón se requiere tanto alabeo como cabeceo terminen al mismo tiempo por eso se reduce la aceleración del alabeo a la mitad.

- En barrena 1 se realiza las mismas consideraciones que en barrena 2 pero la diferencia es que la aceleración en alabeo se reduce a un 75% del cabeceo.

Con las condiciones de posición, velocidad y aceleración de las trayectorias de las ilusiones planteadas, se definen en el siguiente punto los interpoladores de cada uno de los tramos.

3.6.1 Interpoladores

Para las condiciones de posición, velocidad y aceleración se define dos interpoladores el cuadrático y cúbico, se expresa los dos tipos de interpoladores con el objetivo de observar cómo se presentan los resultados ante diferentes formas de trayectoria, para cada interpolador se utiliza periodos de 0.5556 segundos para encontrar los puntos en las trayectorias.

3.6.1.1 Interpolador Cuadrático

Para el desarrollo de este interpolador se realizó una combinación de las ecuaciones circular uniforme y variado con la cinemática diferencial, que define la dependencia de cada ecuación con respecto al estado anterior representada por la variable k que va desde 0 hasta el número total de puntos interpolados, con lo que quedan las siguientes expresiones:

$$\ddot{\phi}(k+1) = \ddot{\theta}(k+1) = a \quad (137)$$

$$\dot{\gamma}(k+1) = 0 \quad (138)$$

$$\dot{\phi}(k+1) = \dot{\phi}(k) + \ddot{\phi}(k) * t \quad (139)$$

$$\dot{\theta}(k+1) = \dot{\theta}(k) + \ddot{\theta}(k) * t \quad (140)$$

$$\dot{\gamma}(k+1) = w \quad (141)$$

$$\phi(k+1) = \phi(k) + \dot{\phi}(k) * t + 0.5 * \ddot{\phi}(k) * t^2 \quad (142)$$

$$\theta(k+1) = \theta(k) + \dot{\theta}(k) * t + 0.5 * \ddot{\theta}(k) * t^2 \quad (143)$$

$$\gamma(k+1) = \gamma(k) + \dot{\gamma}(k) * t \quad (144)$$

El número de puntos interpolados se detalla en las Tablas 22 y 23.

Tabla 22.

Número de puntos de las combinaciones de la ilusión barrena

Tramo	1-2	3-4	5	6-7	8	Total
Número de puntos						
Barrena 1	8	8	2	8	5	31
Barrena 2	12	12	2	12	5	43
Barrena 3	8	8	2	8	5	31

Tabla 23.

Número de puntos de las combinaciones de la ilusión coriolis

Tramo	1-2	3	Total
Número de puntos			
Coriolis 1	16	80	96
Coriolis 2	22	80	102

3.6.1.2 Interpolador cúbico

Este interpolador se mantiene las ecuaciones de guiñada del cuadrático debido a que el movimiento es constante, pero para alabeo y cabeceo se parte de las siguientes ecuaciones:

$$\ddot{\phi}(k+1) = \ddot{\phi}(k) + 6 * a_{31} * t + 2 * a_{21} \quad (145)$$

$$\ddot{\theta}(k+1) = \ddot{\theta}(k) + 6 * a_{32} * t + 2 * a_{22} \quad (146)$$

$$\dot{\phi}(k+1) = \dot{\phi}(k) + 3 * a_{31} * t^2 + 2 * a_{21} * t + a_{11} \quad (147)$$

$$\dot{\theta}(k+1) = \dot{\theta}(k) + 3 * a_{32} * t^2 + 2 * a_{22} * t + a_{12} \quad (148)$$

$$\phi(k+1) = \phi(k) + a_{31} * t^3 + a_{21} * t^2 + a_{11} * t \quad (149)$$

$$\theta(k+1) = \theta(k) + a_{32} * t^3 + a_{22} * t^2 + a_{12} * t \quad (150)$$

El objetivo de este interpolador es obtener los valores de los coeficientes a, pero también al utilizar este interpolador permite trabajar con trayectorias más suaves para el manipulador, con la misma cantidad de puntos interpolados del interpolador cuadrático, en las Tablas 24, 25, 26, 27 y 28 se muestran los valores obtenidos de los coeficientes para cada combinación de la ilusión:

Tabla 24.*Valores de los coeficientes para el interpolador cúbico de barrena 1*

Tramo	1,4	2,3	5	6	7	8
Coeficientes						
a_{31}	0	0	0	1.367	-1.367	0
a_{32}	1.823	-1.823	0	-1.823	1.823	0
a_{21}	0	0	0	0	3.04	0
a_{22}	0	4.05	0	0	-4.05	0
a_{11}	0	0	0	0	0	0
a_{12}	0	0	0	0	0	0

Tabla 25.*Valores de los coeficientes para el interpolador cúbico de barrena 2*

Tramo	1,4	2,3	5	6	7	8
Coeficientes						
a_{31}	0	0	0	0.405	-0.405	0
a_{32}	0.81	-0.81	0	-0.81	0.81	0
a_{21}	0	0	0	0	1.35	0
a_{22}	0	2.7	0	0	-2.7	0
a_{11}	0	0	0	0	0	0
a_{12}	0	0	0	0	0	0

Tabla 26.*Valores de los coeficientes para el interpolador cúbico de barrena 3*

Tramo	1,4	2,3	5	6	7	8
Coeficientes						
a_{31}	0	0	0	1.823	-1.823	0
a_{32}	1.823	-1.823	0	-1.823	1.823	0
a_{21}	0	0	0	0	4.05	0
a_{22}	0	4.05	0	0	-4.05	0
a_{11}	0	0	0	0	0	0
a_{12}	0	0	0	0	0	0

Tabla 27.*Valores de los coeficientes para el interpolador cúbico de coriolis 1*

Tramo	1	2	3
Coeficientes			
a_{31}	0.81	-0.81	0
a_{32}	0	0	0
a_{21}	0	2.7	0
a_{22}	0	0	0
a_{11}	0	0	0
a_{12}	0	0	0

Tabla 28.*Valores de los coeficientes para el interpolador cúbico de coriolis 2*

Tramo	1	2	3
Coeficientes			
a_{31}	0.36	-0.36	0
a_{32}	0	0	0
a_{21}	0	1.8	0
a_{22}	0	0	0
a_{11}	0	0	0
a_{12}	0	0	0

Con los coeficientes calculados se tienen definidos los interpoladores cúbicos a continuación se muestra el esquema de funcionamiento a partir de trayectorias, interpoladores y modelos obtenidos:

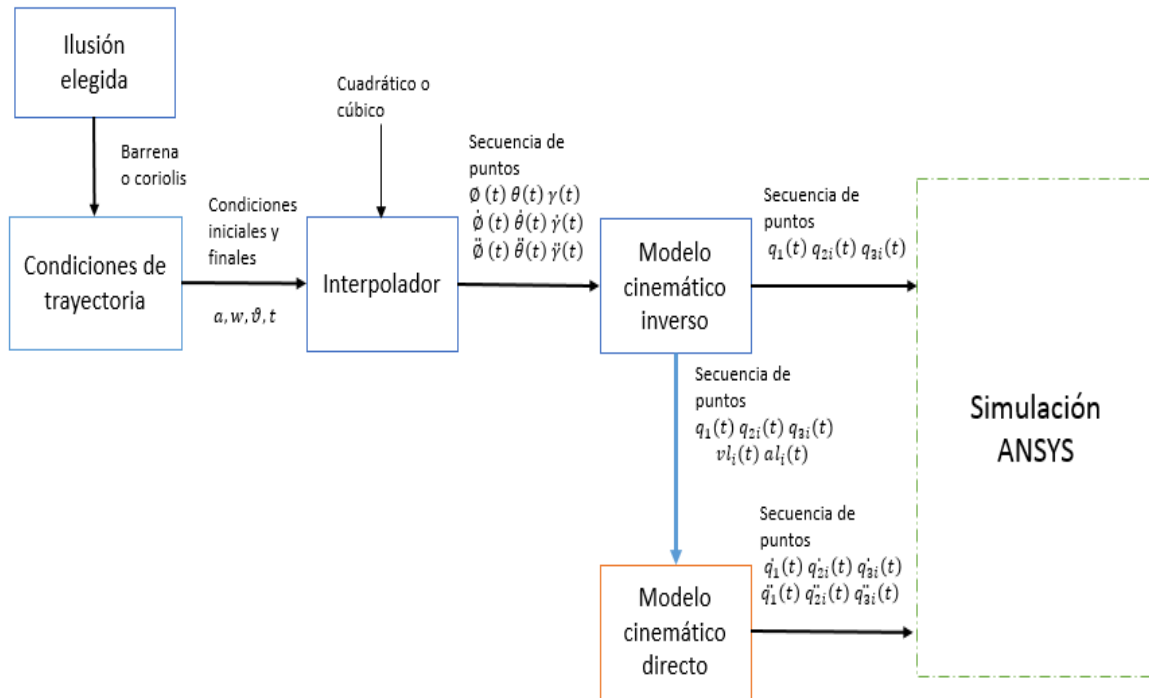


Figura 41. Esquema de funcionamiento de trayectorias con los modelos obtenidos

Con el esquema planteado en la Figura 41 a continuación se muestra dos ejemplos de gráficas de las variables de los actuadores prismáticos en el brazo 1 en la ilusión barrena 3 mediante los dos tipos de interpoladores cuadrático y cúbico, en la Figuras 42 y 43 se puede observar las formas de las trayectorias de salida.

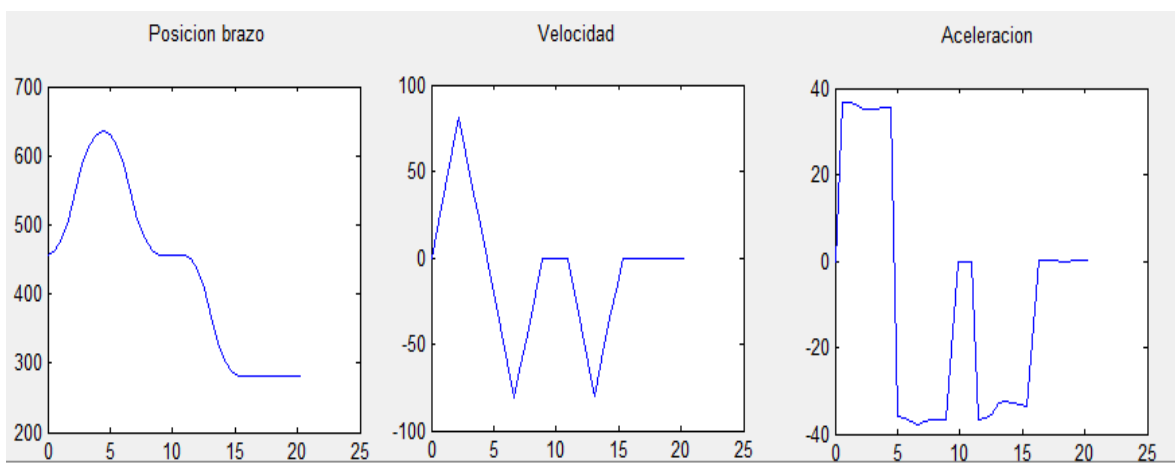


Figura 42. Variables de salida del actuador prismático con interpolador cuadrático

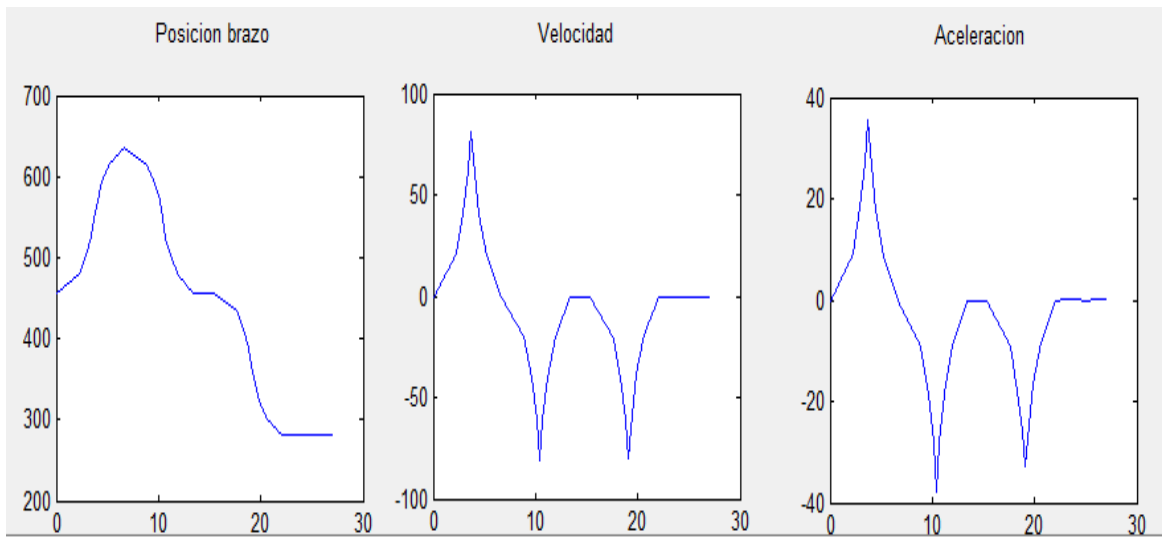


Figura 43. Variables de salida del actuador prismático con interpolador cúbico

Como se observa cada tipo de interpolador genera la forma de la trayectoria de salida de los actuadores, para representar el modelo obtenido y los interpoladores, se realiza un algoritmo e interfaz en MATLAB, que se detalla en el siguiente capítulo.

CAPITULO IV

4. INTERFAZ EN MATLAB DE LOS MODELOS CINEMÁTICOS OBTENIDOS

En este capítulo se presenta el desarrollo del algoritmo e interfaz que representa los modelos cinemáticos obtenidos, además de la interpolación de trayectorias de las ilusiones para generar las variables de los actuadores activos que son entrada para las simulación en ANSYS, el software para el desarrollo punto es MATLAB debido a que los algoritmos se pueden representar mediante las plantillas de funciones y la interfaz con el uso de la herramienta de este programa denominada GUIDE.

Se plantea el desarrollo de tres interfaces, la primera es para comprobar el modelo cinemático inverso, la segunda es para comprobar el modelo cinemático directo a partir de los datos enviados por la interfaz inversa como relación maestro- esclavo y la tercera se comprueba los interpoladores de las trayectorias de las ilusiones barrena y coriolis, teniendo como datos de entrada las condiciones de posición, velocidad, aceleración de los ángulos de alabeo, cabeceo y guiñada, a continuación se detalla cada interfaz con su respectivo algoritmo:

4.1 Interfaz del modelo cinemático inverso con datos estáticos

En esta interfaz se desarrolla una herramienta para visualizar el movimiento y los parámetros de salida son las longitudes de los brazos, la velocidad y aceleración en función de las coordenadas x, y, z de los vértices de la base móvil, teniendo como entrada los ángulos de alabeo, cabeceo, guiñada en ($^{\circ}$), atura de referencia en milímetros, velocidades angulares de alabeo, cabeceo, guiñada en (rad/s) y aceleraciones angulares de las mismas magnitudes en (rad/s²).

4.1.1 Parámetros de entrada y salida

En la Tabla 29 se muestra las entradas y salidas planteadas para la interfaz

Tabla 29.

Parámetros de entrada y salida de la interfaz del modelo inverso

Entradas	Salidas
ϕ, θ, γ	$q_{3i}, (q_{3i} + 1191.2)$
$\dot{\phi}, \dot{\theta}, \dot{\gamma}$	$\vec{vl}_1, \vec{vl}_2, \vec{vl}_3$
$\ddot{\phi}, \ddot{\theta}, \ddot{\gamma}$	$\vec{al}_1, \vec{al}_2, \vec{al}_3$
Altura de referencia	Gráfica del manipulador

4.1.2 Diseño de la pantalla de la interfaz

Como se mencionó se utiliza la herramienta GUIDE de MATLAB, por lo cual se plantea la ubicación de los parámetros dentro de la pantalla mediante la siguiente disposición:

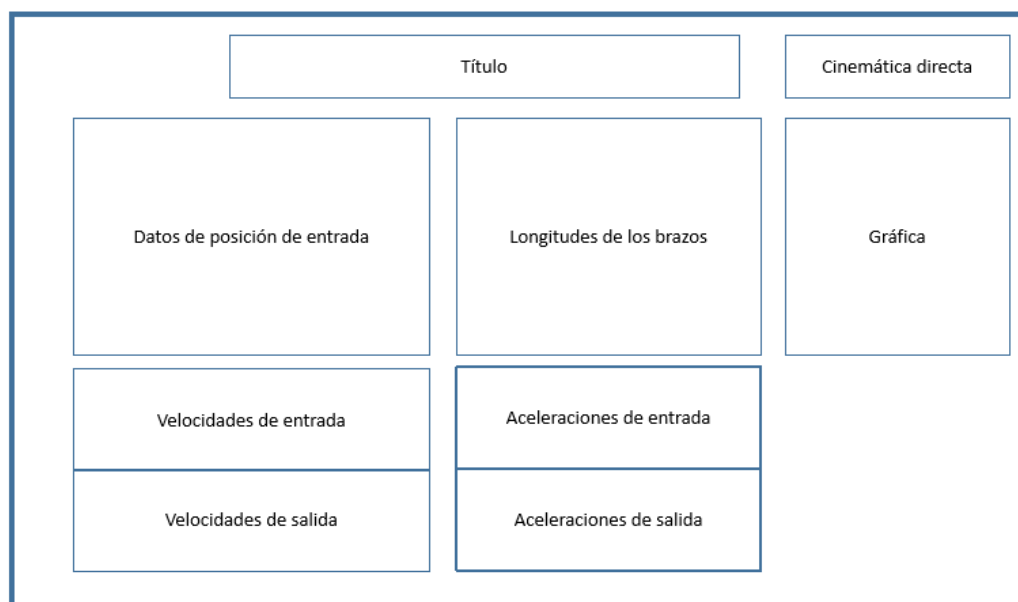


Figura 44. Distribución de los parámetros en la pantalla de la interfaz inversa

A continuación se detalla una descripción de los bloques de variables en la pantalla.

- **Título:** En esta área va el nombre de la interfaz que es modelo cinemático inverso

- **Datos de posición de entrada:** Al iniciar la herramienta los parámetros comienzan en los datos especificados en la Tabla 1, en este espacio se permite el ingreso y variación de los ángulos de alabeo, cabeceo y guiñada, además del ingreso de la altura de referencia en milímetros, en este punto se varía la longitud de carrera de los pistones.
- **Longitudes de los brazos:** En este espacio se encuentran dos columnas, en la cual la primera muestra la longitud total de los brazos, en la segunda se visualiza la carrera de los pistones que cambia respecto a los ángulos ingresados, también hay tres filas que indica los términos mencionados para cada brazo.
- **Gráfica:** Se muestra el manipulador y como varía respecto a los ángulos y altura ingresada.
- **Velocidades de entrada:** En esta área se ingresa las velocidades angulares de alabeo, cabeceo y guiñada en (rad/s).
- **Velocidades de salida:** En este espacio se visualiza las velocidades en (mm/s) de los vértices de la base móvil distribuida en tres filas que representan las componentes x, y, z y cuatro columnas que indica las tres primeras cada brazo y la última es la velocidad del punto central.
- **Aceleraciones de entrada:** En esta área se ingresa las aceleraciones angulares de alabeo, cabeceo y guiñada en (rad/s²).
- **Aceleraciones de salida:** En este espacio se visualiza las aceleraciones en (mm/s²) de los vértices de la base móvil distribuida en tres filas que representan las componentes x, y, z y cuatro columnas que indica las tres primeras cada brazo y la última es la aceleración del punto central.

4.1.3 Algoritmo de funcionamiento de la interfaz inversa

Se describe el algoritmo de funcionamiento a continuación:

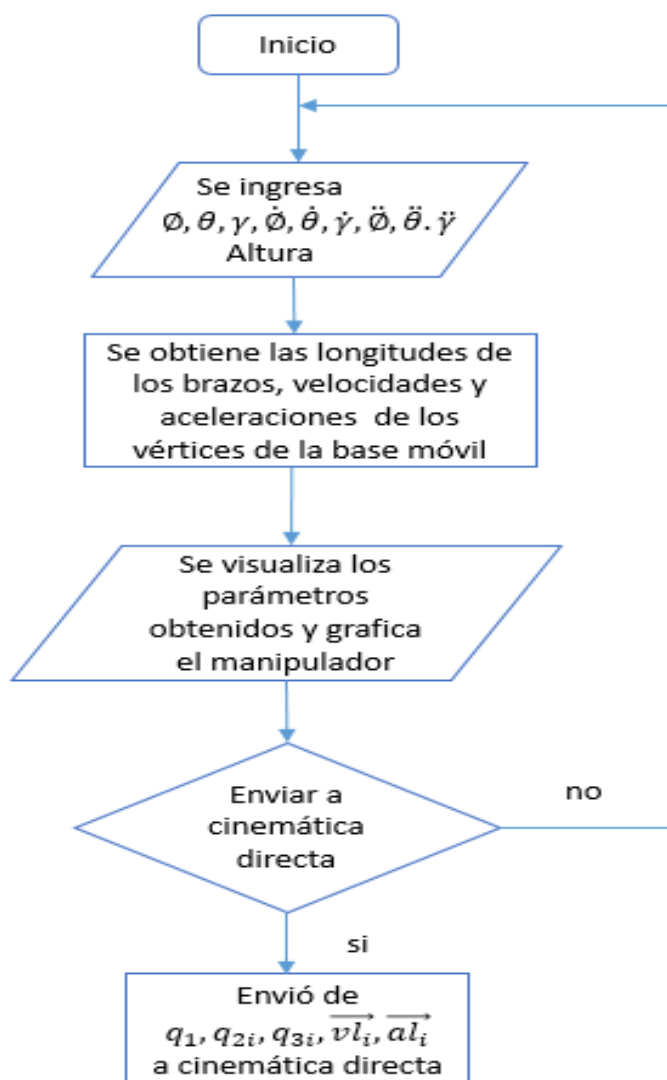


Figura 45. Algoritmo de funcionamiento de la interfaz inversa

Con el algoritmo descrito, la distribución de pantalla planteada se desarrolla en la herramienta de MATLAB la interfaz, como se muestra en la Figura 46.

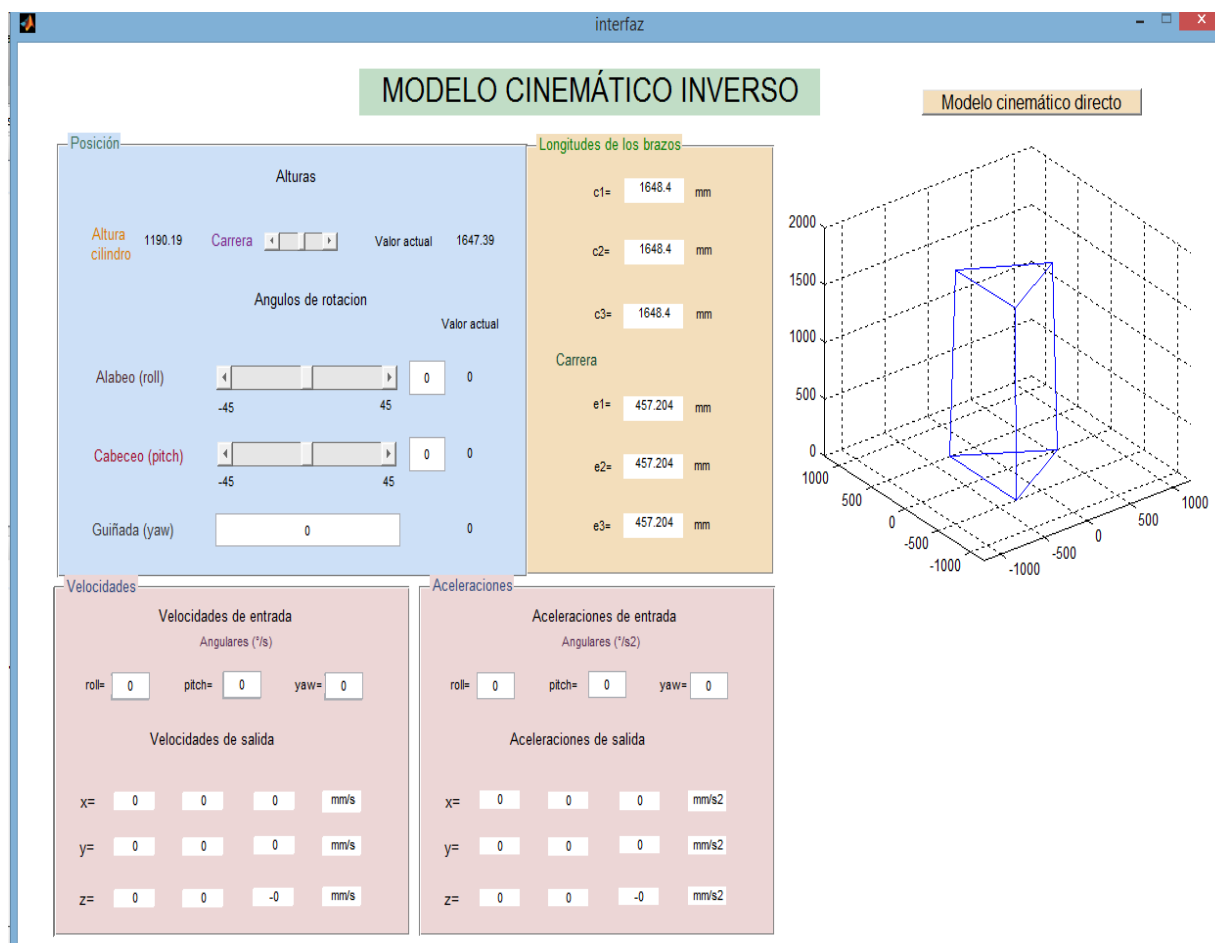


Figura 46. Interfaz inversa en MATLAB

4.2 Interfaz del modelo cinemático directo con datos estáticos

Esta interfaz es dependiente de la inversa debido a que solo se activa cuando la cinemática inversa envía parámetros a la directa, las entradas son las longitudes de los brazos, la velocidad y aceleración en función de las coordenadas x , y , z de los vértice de la base móvil, mientras que las salidas son las velocidades, aceleraciones de los actuadores, en este caso la interfaz es solo de visualización, el sentido de no dejar ingresar parámetros en esta interfaz es para evitar movimiento inadecuados que alteren las condiciones cinemáticas del manipulador.

4.2.1 Datos de entrada y salida

En la Tabla 30, se muestra los parámetros de entrada y salida cuando se activa esta interfaz.

Tabla 30.*Parámetros de entrada y salida de la interfaz del modelo directo*

Entradas	Salidas (Visualizadas)
q_{3i}, q_{2i}, q_1	q_{3i}, q_{2i}, q_1
$\vec{vl}_1, \vec{vl}_2, \vec{vl}_3$	$\vec{vld}_1, \vec{vld}_2, \vec{vld}_3, \dot{q}_1, \dot{q}_2, \dot{q}_3$
$\vec{al}_1, \vec{al}_2, \vec{al}_3$	$\vec{ald}_1, \vec{ald}_2, \vec{ald}_3, \ddot{q}_1, \ddot{q}_2, \ddot{q}_3$

4.2.2 Diseño de la pantalla de la interfaz

La interfaz es solo de visualización, por lo que queda la siguiente distribución de pantalla.

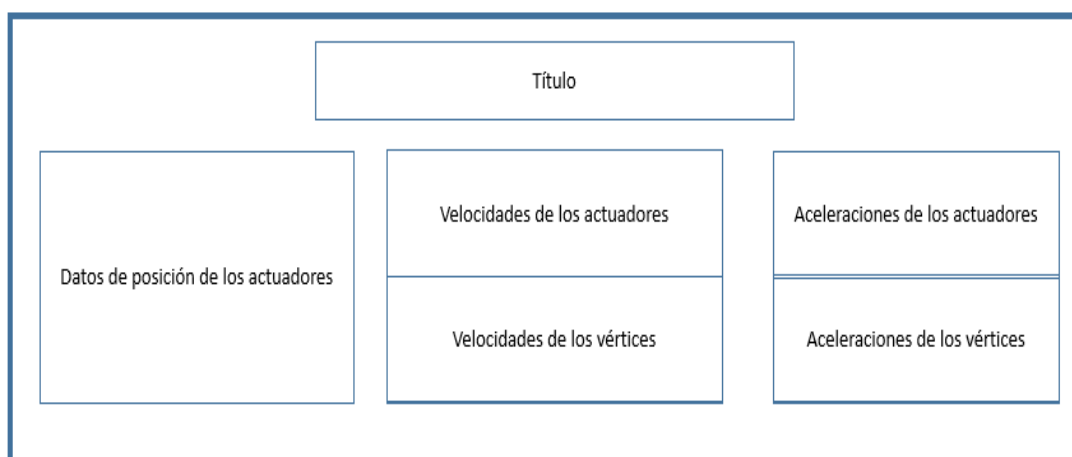


Figura 47. Distribución de los parámetros en la pantalla de la interfaz directa

A continuación se detalla una descripción de los bloques de variables en la pantalla.

- **Título:** En esta área va el nombre de la interfaz que es cinemática directa.
- **Datos de posición de los actuadores:** En este espacio se visualiza las longitudes de los brazos, los ángulos inferiores y el giro de la plataforma, además se añade dos columnas donde se observa las coordenadas del punto central de base móvil para las comparaciones de los dos métodos planteados en cinemática directa.
- **Velocidades de los actuadores:** En esta área se visualiza las magnitudes de velocidad de los actuadores calculados a partir de los datos de entrada.

- **Velocidades de los vértices:** Se visualiza las velocidades de los vértices de la base móvil con respecto a los parámetros de los actuadores calculados por las funciones de cálculo basados en el modelo cinemático directo.
- **Aceleraciones de los actuadores:** En esta área se visualiza las magnitudes de aceleración de los actuadores calculados a partir de los datos de entrada.
- **Aceleraciones de los vértices:** Se visualiza las aceleraciones de los vértices de la base móvil con respecto a los parámetros de los actuadores calculados por las funciones de cálculo basados en el modelo cinemático directo.

4.2.3 Algoritmo de funcionamiento de interfaz directa

A continuación se describe el algoritmo de funcionamiento de la interfaz directa:

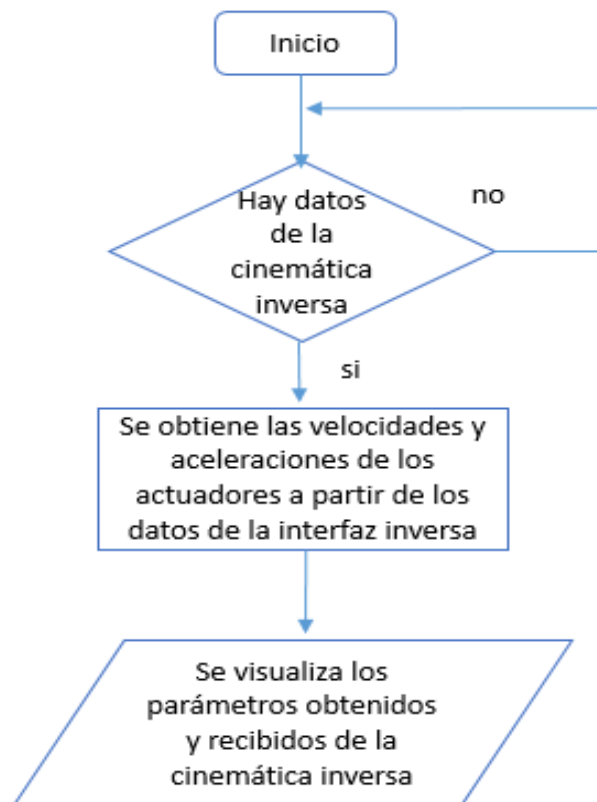


Figura 48. Algoritmo de funcionamiento de la interfaz directa

Con el algoritmo planteado, se desarrolla la interfaz que se muestra a continuación:



Figura 49. Interfaz directa en MATLAB

Las dos interfaces mencionadas se las desarrollaron con el objetivo de verificar datos de posición ante diferentes ángulos de alabeo, cabeceo y guiñada de forma puntual con la simulación en ANSYS, pero estas interfaces se consideraron como herramienta complementaria para la interfaz definitiva que se detalla en el siguiente punto.

4.3 Interfaz de simulación de movimiento a partir de las trayectorias de las ilusiones

En este interfaz se combina las dos interfaces anteriores con las condiciones de las ilusiones barrena y coriolis con sus respectivas variaciones y los interpoladores que permiten integrar punto de control para generar el movimiento que recrea las ilusiones mencionadas a partir de los modelos cinemáticos obtenidos. Para esta interfaz los parámetros de entrada son la ilusión, el interpolador, el brazo y el actuador que se quiera visualizar sus variables cinemáticas, las salidas serán las trayectorias de posición, velocidad y aceleración del actuador elegido.

4.3.1 Parámetro de entrada y salida de la interfaz de simulación de movimiento

En la Tabla 31 se muestra los parámetros entrada y salida de esta interfaz.

Tabla 31.

Parámetros de entrada y salida de la interfaz de simulación

Entradas	Salidas
Altura de referencia, ilusión, interpolador, brazo y articulación	Espacio de trabajo realizado por cada brazo
$\phi(t), \theta(t), \gamma(t)$	$q_{3i}(t), q_{2i}(t), q_{1i}(t)$
$\dot{\phi}(t), \dot{\theta}(t), \dot{\gamma}(t)$	$\dot{q}_{1i}(t), \dot{q}_{2i}(t), \dot{q}_{3i}(t)$
$\ddot{\phi}(t), \ddot{\theta}(t), \ddot{\gamma}(t)$	$\ddot{q}_{1i}(t), \ddot{q}_{2i}(t), \ddot{q}_{3i}(t)$
Altura de referencia (mm)	Gráfica del movimiento del manipulador

4.3.2 Diseño de la pantalla de la interfaz simulación de movimiento

En la distribución de los parámetros en la pantalla de la interfaz se considera las opciones de ilusión, el interpolador, el número de brazo, la articulación, además se busca la visualización de las trayectorias de posición, velocidad y aceleración de entrada, las gráficas de las mismas magnitudes en la salida referenciadas a los actuadores elegidos.

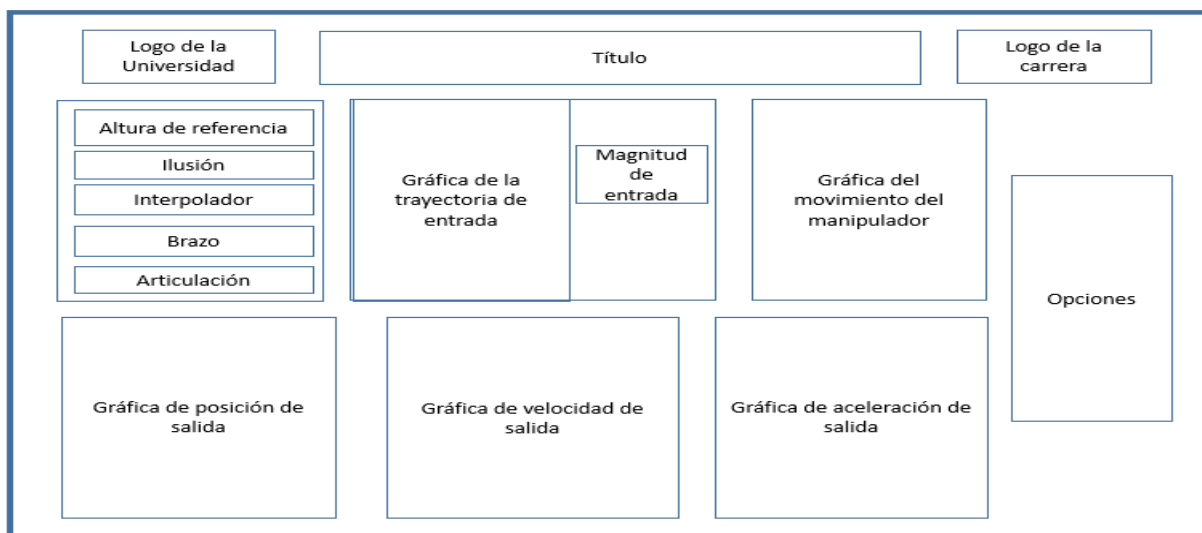


Figura 50. Distribución de los parámetros en la pantalla de la interfaz de simulación

A continuación se detalla una descripción de los bloques de variables en la pantalla.

- **Título:** En esta área va el nombre de la interfaz que es Modelamiento cinemático del simulador de vuelo basado en el manipulador R-3RPS.
- **Altura de referencia:** Se visualiza la altura de referencia del manipulador en mm.
- **Ilusión:** En un seleccionador que permite elegir entre las ilusiones de trabajo que son barrena1, barrena 2, barrena3, coriolis1 y coriolis2.
- **Interpolador:** Se escoge el tipo de interpolador que puede ser cuadrático y cúbico, al seleccionar alguno de ellos con la ilusión escogida se crea la trayectoria de entrada en posición, velocidad y aceleración.
- **Brazo:** Se escoge el brazo de trabajo.
- **Articulación:** Se escoge entre la plataforma giratoria, las articulaciones rotatorias y primaticas de los brazos.
- **Gráfica de la trayectoria de entrada:** Se visualiza las trayectorias de alabeo, cabeceo y guiñada, tanto en posición, velocidad y aceleración
- **Magnitud de entrada:** Se selecciona el tipo de trayectoria que se dese visualizar que puede ser posición, velocidad y aceleración.
- **Gráfica del movimiento del manipulador:** Se visualiza el movimiento del manipulador ante las trayectorias de entrada.
- **Opciones:** Se comprende de dos botones el primero es movimiento que permite visualizar la movilidad del manipulador.
- **Gráficas de posición, velocidad y aceleración de salida:** Se visualiza las gráficas de las magnitudes mencionadas de los actuadores ante la ilusión y el interpolador escogido.

4.3.3 Algoritmo de funcionamiento de la interfaz

A continuación se detalla el algoritmo de funcionamiento de la interfaz.

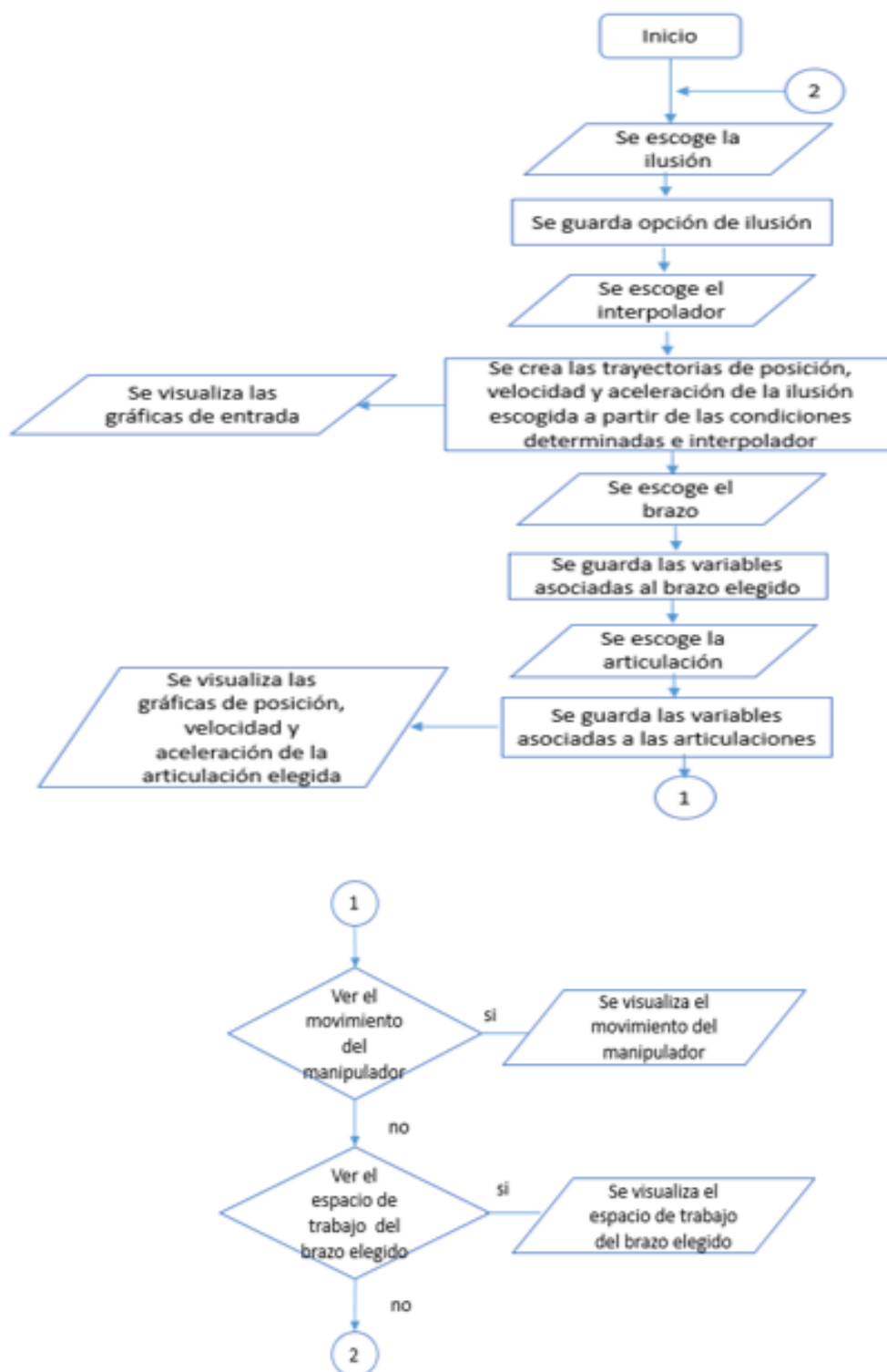


Figura 51. Algoritmo de funcionamiento de la interfaz

Con el algoritmo planteado se desarrolla la interfaz de simulación de movimiento que queda la siguiente manera:

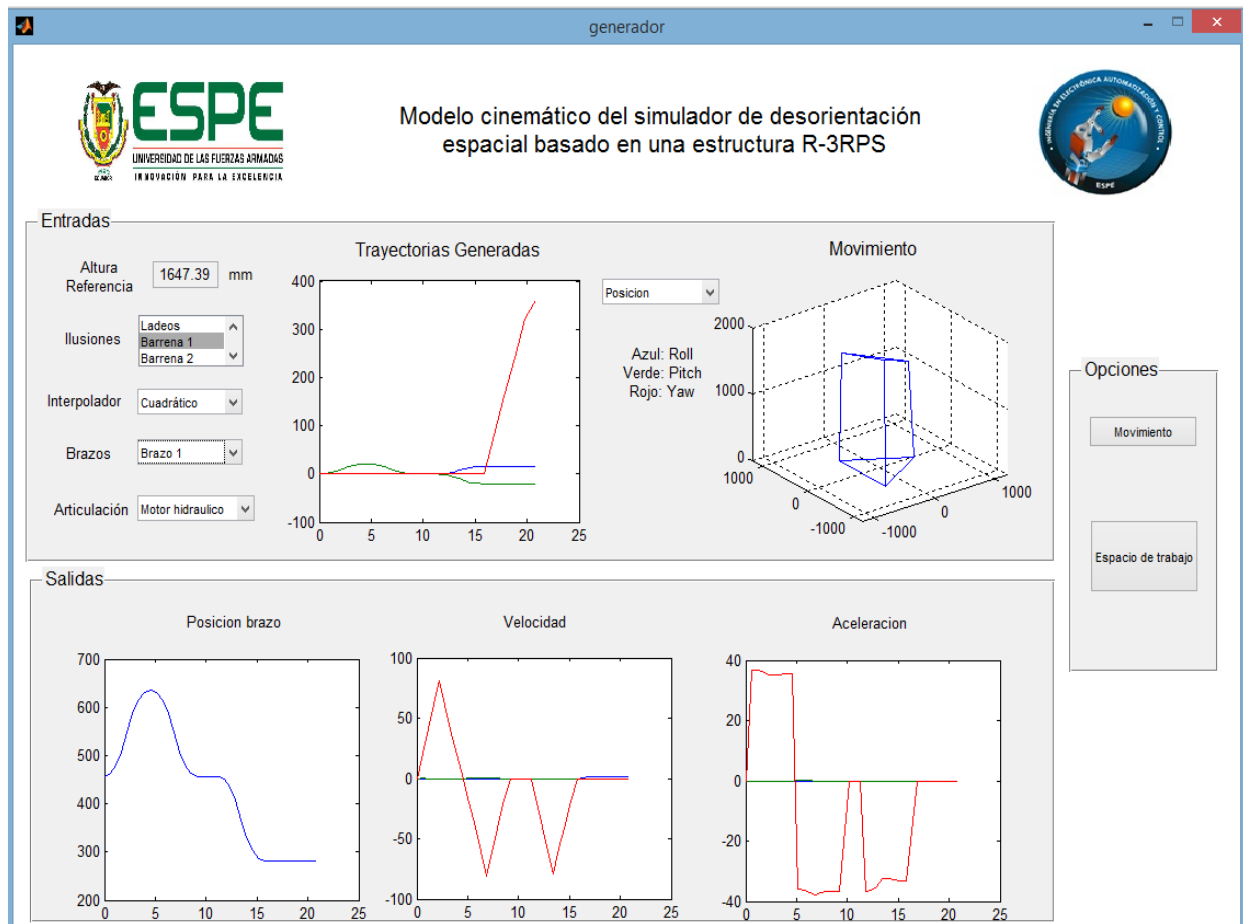


Figura 52. Interfaz de simulación de movimiento en MATLAB

Con la interfaz desarrollada el siguiente es verificar los resultados obtenidos con la misma con relación a la simulación en ANSYS, tomando en cuenta que la simulación genera los datos de entrada para la simulación en ANSYS. En el siguiente capítulo se obtiene los resultados de las interfaces con datos estáticos, la simulación de movimiento y el errores entre la simulación de MATLAB con la de ANSYS.

CAPITULO V

5. RESULTADOS

En este capítulo se compara los resultados obtenidos de la interfaz del modelo cinemático inverso con la simulación en ANSYS, por medio de datos estáticos, la siguiente verificación es de los métodos directos planteados con la cinemática inversa igualmente por los parámetros estáticos. Para las trayectorias de las ilusiones se realiza la comparación de las longitudes, de los brazos, la velocidad y aceleración total en los vértices de base móvil con el cálculo de los respectivos errores.

Para el cálculo de los errores absoluto y porcentual se emplea las siguientes fórmulas:

$$e_a = \left| \frac{\text{valor simulado} - \text{valor teórico}}{\text{valor teórico}} \right| \quad (151)$$

$$e_p = \left| \frac{\text{valor simulado} - \text{valor teórico}}{\text{valor teórico}} \right| \times 100 \quad (152)$$

El valor teórico es el entregado por la interfaz de MATLAB, mientras que el valor simulado es el entregado por ANSYS.

5.1 Comprobación de los resultados de la interfaz inversa con ANSYS.

En esta sección se realiza la verificación de los parámetros entregados por el modelo cinemático inverso mediante la interfaz con los datos obtenidos en ANSYS en donde se ingresó los desplazamientos de los actuadores prismáticos y de la plataforma giratoria en cada brazo que son la salida del modelo en MATLAB, para la comprobación se realizó la combinación de los tres ángulos alabeo, cabeceo y guiñada, encontrando el error absoluto máximo en cada tramo de prueba,

el parámetro que se utilizó para verificación fue la longitud total de cada brazo obtenida en ANSYS, a partir de la aplicación de la fórmula de distancia entre dos puntos, y la longitud total mostrada en MATLAB, en la Tabla 32 se muestra los errores absoluto y porcentuales máximo en cada trama.

Tabla 32.

Errores absolutos y porcentuales máximos entre el modelo inverso y ANSYS

Rango de Números de dato	Alabeo	Cabeceo	Guiñada	e_{amax}	$e_{pmax}\%$
0-9	0-45°	0°	0°	0	0
10-18	0°	0° - 45°	0°	0.0001	0.01
19-27	0-45°	5°	0° - 360°	0.00004	0.004
28-36	0-45°	10°	0°-360°	0.00003	0.003
37-45	0-45°	15°	0°-360°	0.00006	0.006
46-54	0-45°	20°	0°-360°	0.00007	0.007
55-63	0-45°	25°	0°-360°	0.00004	0.004
64-72	0-45°	30°	0°-360°	0.00005	0.005
73-81	0-45°	35°	0°-360°	0.00006	0.006
82-90	0-45°	40°	0°-360°	0.00004	0.004
91-99	0-45°	45°	0°-360°	0.0159	1.59

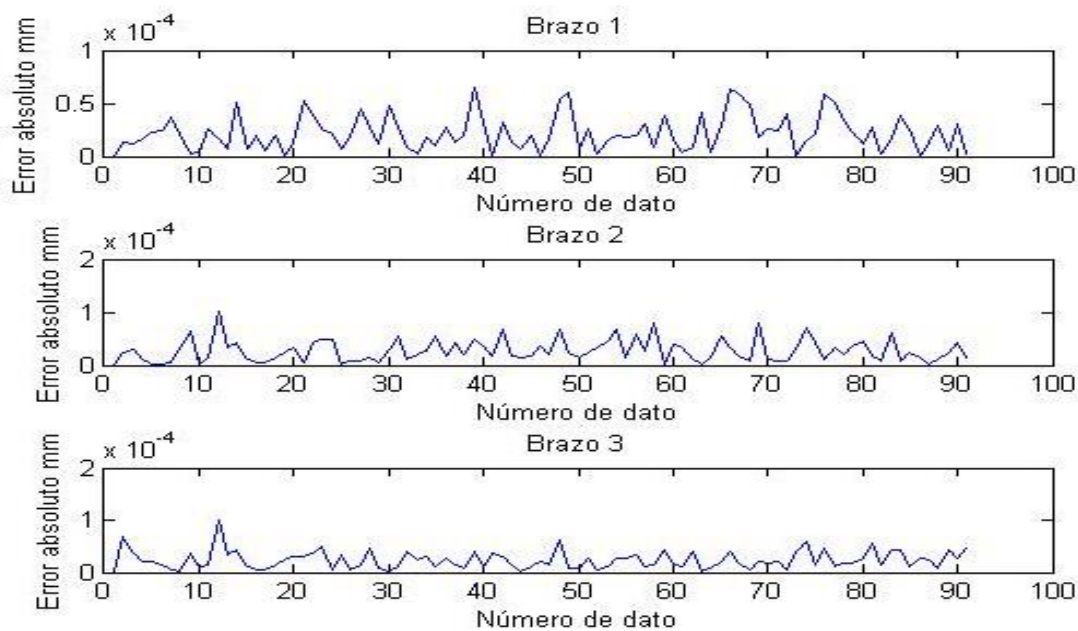


Figura 53. Gráfica del error absoluto en cada brazo hasta número de dato 90

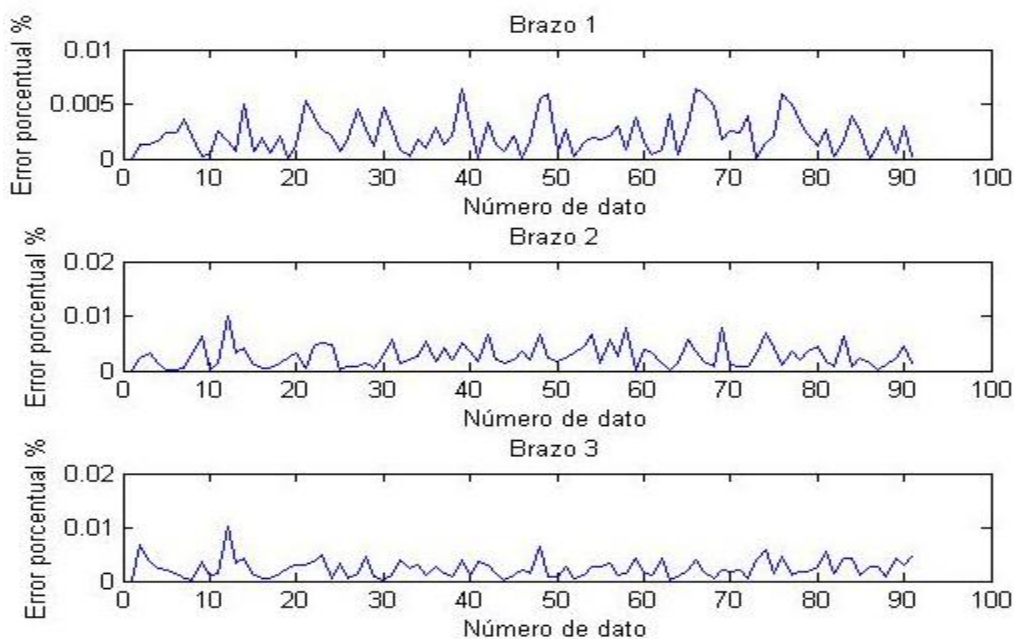


Figura 54. Gráfica del error porcentual en cada brazo hasta número de dato 90

Como se muestra en las Figuras 53 y 54, se muestra las gráficas del error absoluto, porcentual, de los datos del 0 al 90, en cada brazo respectivamente, en donde se observa que los máximos errores absoluto y porcentual no llegan a ser valores considerables, para ser restringidos, en el caso de los valores de los datos a partir de 91 el error sube pero sigue manteniéndose en límites de confiabilidad.

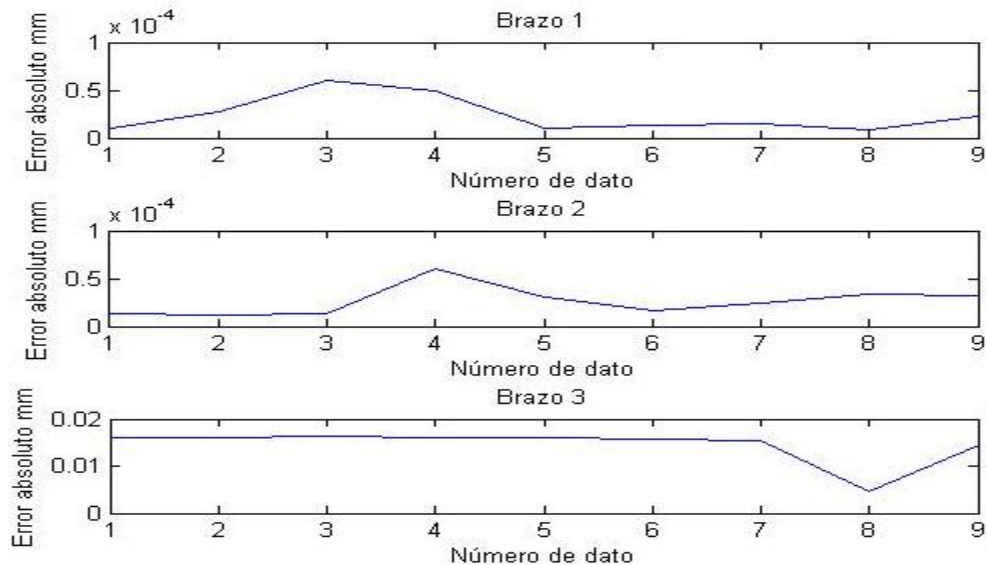


Figura 55. Gráfica del error absoluto en cada brazo a partir del número de dato 91

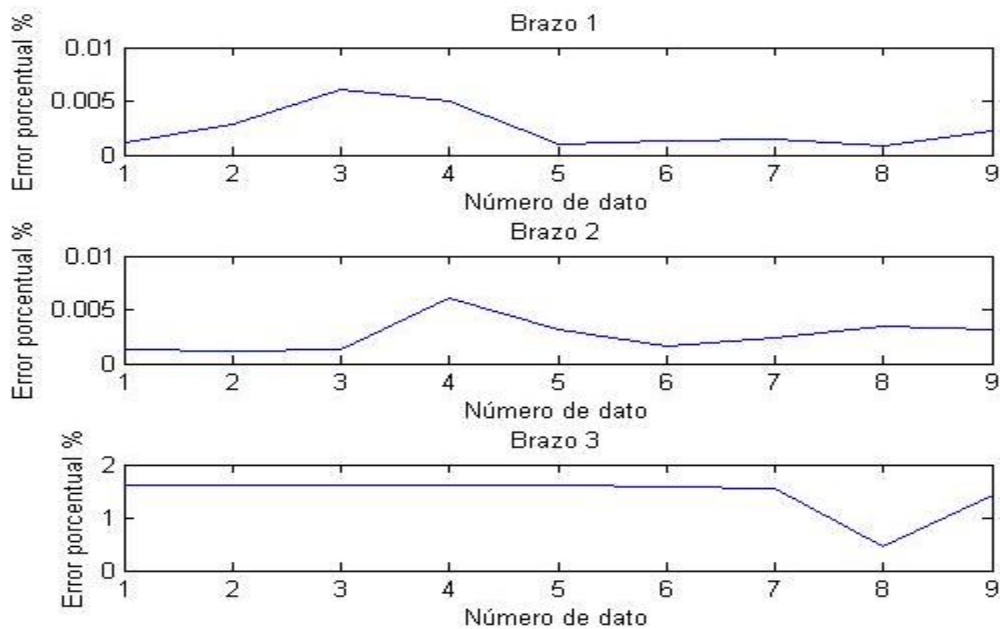


Figura 56. Gráfica del error porcentual en cada brazo a partir del número de dato 91

Los valores mostrados llegan a un máximo 1.59% que comprende a los datos de alabeo en 20° y cabeceo 45° en este caso los puntos comprendidos en este rango se lo considera inestables debido a que los mismos producen un esfuerzo alto en el manipulador que puede llegar a romper las juntas

esféricas por esta razón establece una condición de precaución donde se garantiza que las juntas estén propensas a dañarse, la condición planteada es:

$$alabeo + cabeceo \leq 45^\circ \quad (153)$$

Con la condición planteada permite descartar varias combinaciones que puede romper las juntas esféricas, con los puntos hábiles se encuentra el espacio de trabajo del manipulador con la guiñada tomada cada 5° hasta 360° , se obtiene las longitudes máxima y mínima de cada brazo, validadas con la condición anterior se encuentra el siguiente gráfico:

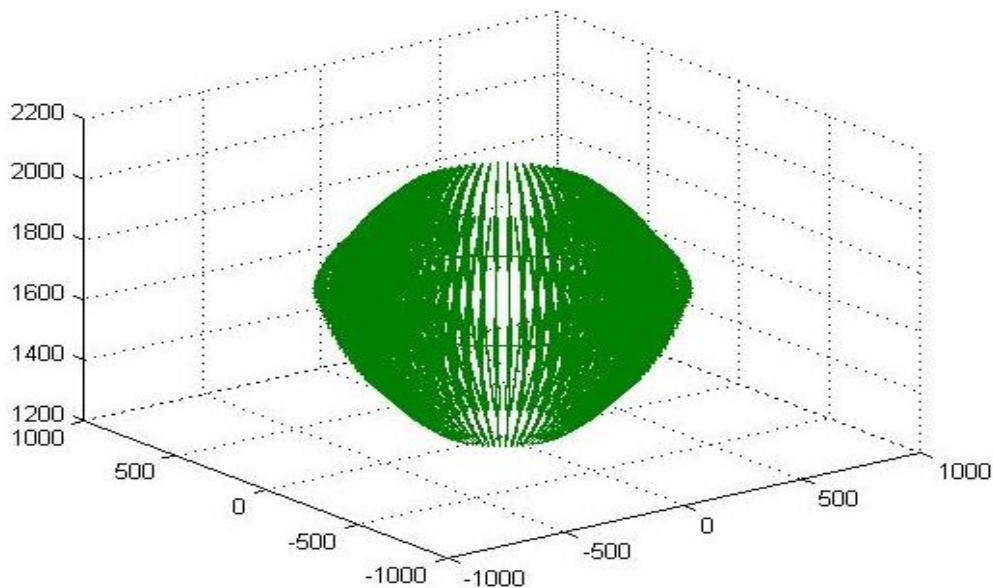


Figura 57. Espacio de trabajo del manipulador R-3RPS

Se encuentran las longitudes máximas de 2030.75 mm y las mínimas de 1346.8, para cada brazo, mediante los errores mostrados y la condición planteada se observa que los datos entre los softwares usados son confiables dentro del campo de solo posición por datos estáticos debido a que

el porcentaje de error no llega ni al 1%, los siguientes resultados a verificar son entre los modelos cinemáticos directo e inverso para la elección del método directo respecto a posición.

5.2 Comprobación de resultados entre los modelos inverso y directo

Como se mencionó en el desarrollo del modelo directo se comparó entre dos métodos para la elección de uno de ellos, se verificó el porcentaje de error de ambos métodos con respecto al modelo inverso para eso se utilizó el tramo que menor error tenía con respecto a los comprobados en ANSYS que fue el tramo de 0-9, en ese rango se los verificó obteniendo los siguientes errores porcentuales:

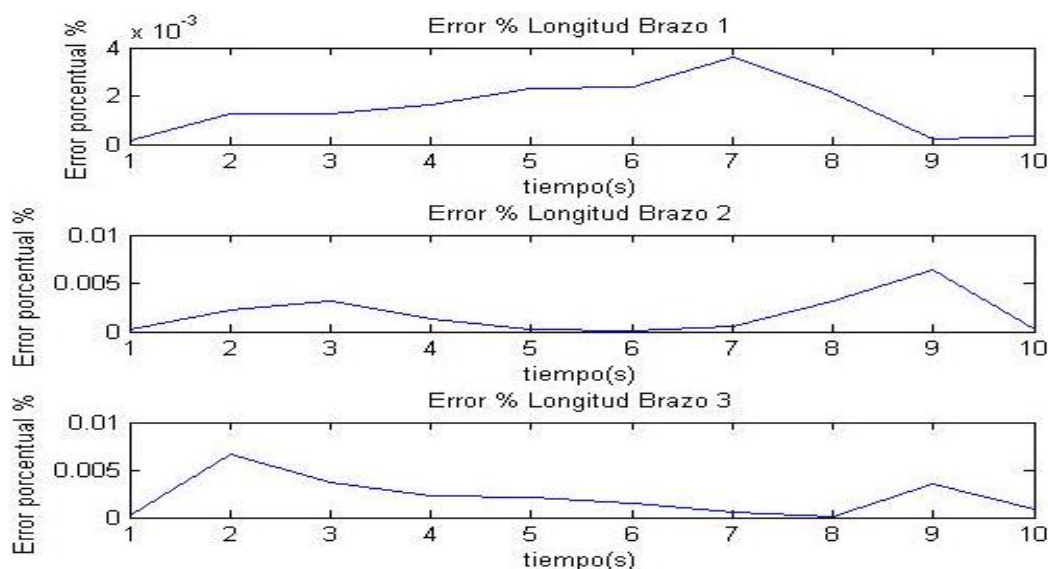


Figura 58. Error porcentual método vectorial

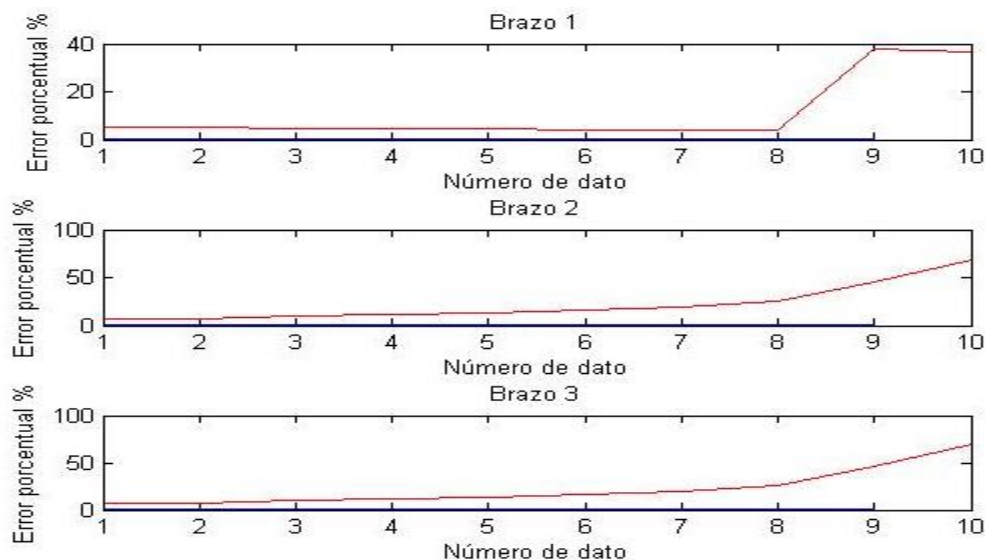


Figura 59. Error porcentual Denavit Hartenberg

Como se observa el método vectorial tiene un error máximo de 0.004% mientras que en Denavit Hartenberg, el error máximo es de 40% esto tiene sentido debido a que este método es parametrizado en cuatro variables, pero se requiere que el método sea lo más cercano a los resultados del modelo inverso por esta razón se escogió el método vectorial. El error provocado en Denavit Hartenberg no determina que este método este incorrecto sino que para verificarlo se requiere de herramientas específicas para comprobar como es el Robotics de MATLAB, en la siguiente imagen se muestra la prueba de la herramienta mencionada con alabeo en 20° y cabeceo en 25° sobrepuesto con las coordenadas de los vértices de la cinemática inversa.

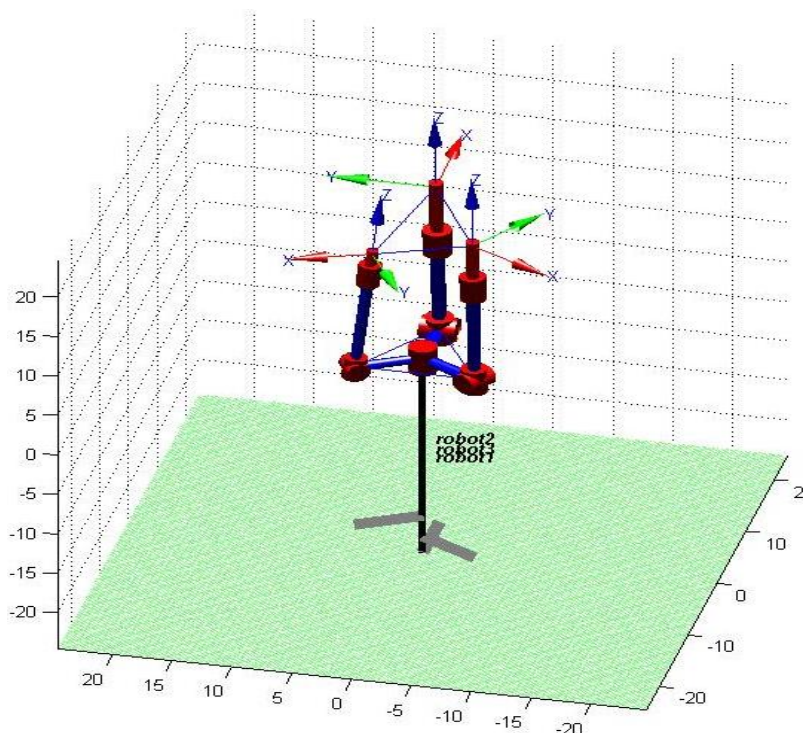


Figura 60. Gráfica con Robotics

Como se observa los vértices por cinemática inversa con la estructura del manipulador coinciden comprobando los parámetros planteados en el método de Denavit Hartenberg. En la verificación de los puntos por posición se presentó un error máximo de 1.59% en datos inestables y 0.008 % en datos ya condicionados por la restricción de precaución, dando la viabilidad de posición entre el modelo teórico en MATLAB y la simulación en ANSYS. Los resultados presentados son referentes a posición de los brazos y la plataforma del manipulador mediante datos estáticos, para comprobar lo que es velocidad y aceleración se utiliza el control cinemático por interpolación de trayectorias que combina ambos modelos, para se comprueba mediante el ingreso de las velocidades de los actuadores en ANSYS.

5.3 Comprobación de las trayectorias de las ilusiones dinámicas de vuelo.

Para verificar las trayectorias de las ilusiones se ingresó en ANSYS, las velocidades de los actuadores mediante puntos interpolados con respecto al tiempo, las variantes probadas fueron la

barrena 1 y coriolis, los puntos para comprobación fueron los vértices de la base la base móvil donde se analiza posición mediante la longitud de cada brazo, velocidades y aceleración total en los puntos mencionados además se comprueba posición y velocidad mediante los interpoladores cúbico y cuadrático, la aceleración se comprobó con el cuadrático debido a las diferentes formas de gráficas entre ambos interpoladores para esta magnitud.

5.3.1 Comprobación de la Coriolis 1

En esta ilusión actúa el alabeo hasta 30° y guiñado una vuelta, donde se genera los desplazamientos, velocidades, aceleraciones de los actuadores prismáticos y de la plataforma, la velocidad de los actuadores fue el dato de entrada en el software ANSYS, obtiene la posición, velocidad y aceleración en los vértices de la base móvil, mediante los resultados obtenidos se calcula el error porcentual de cada una de la magnitudes, en la Figura 61 se muestra los resultados en posición, en la Figura 62 se muestra la velocidad, en la Figura 63 se muestra la aceleración con respecto al interpolador cuadrático.

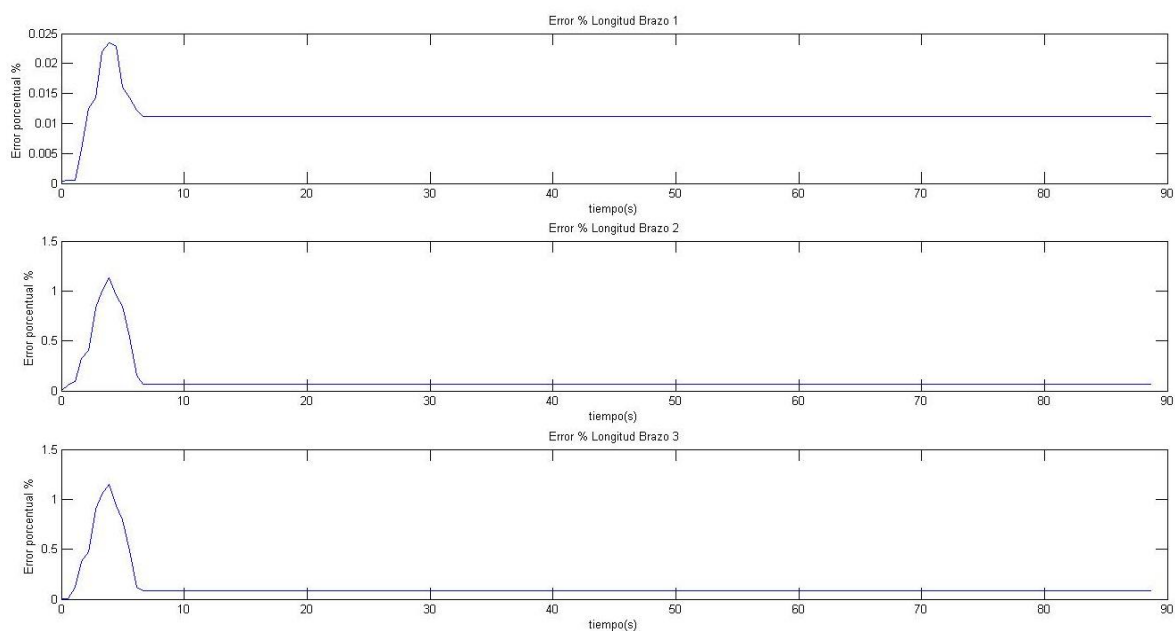


Figura 61. Error porcentual en posición coriolis1 por interpolador cuadrático

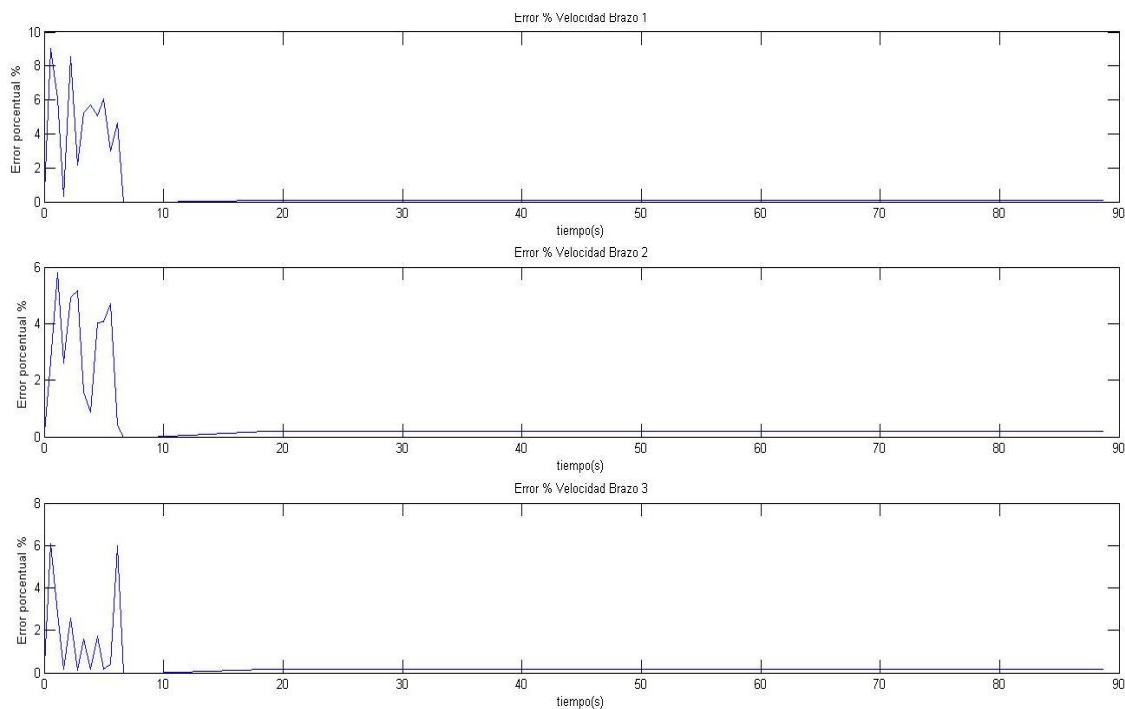


Figura 62. Error porcentual en velocidad coriolis1 por interpolador cuadrático

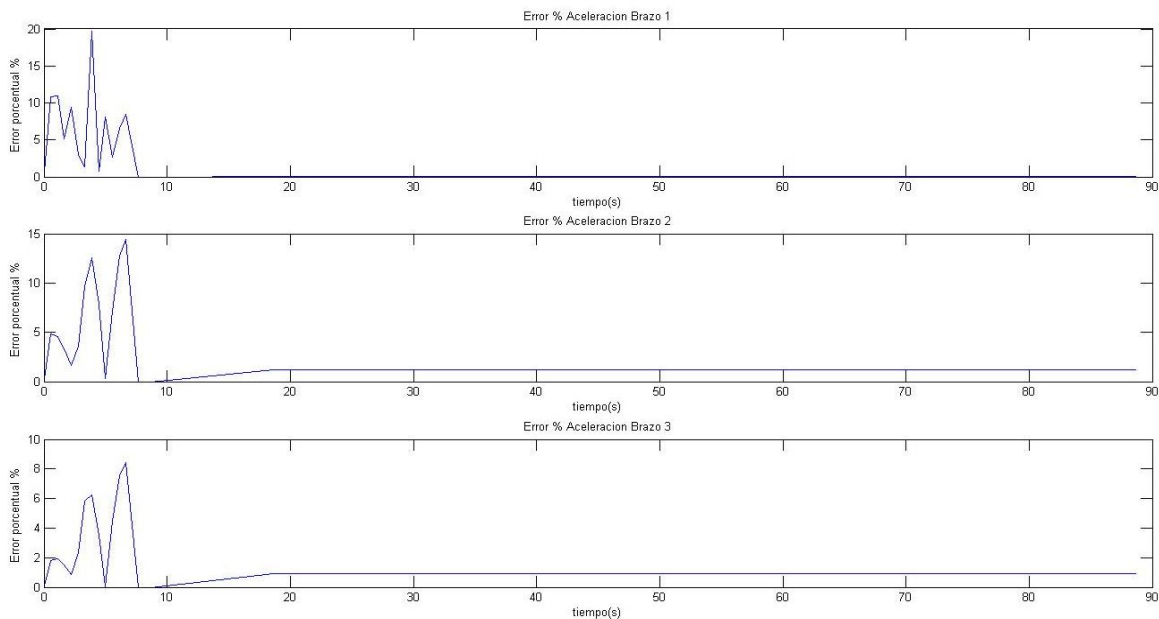


Figura 63. Error porcentual en aceleración coriolis1 por interpolador cuadrático

Como se observa los errores en posición, velocidad y aceleración varían respecto al ángulo de orientación tratado, se observa que cuando hay alabeo los errores son mayores con respecto cuando

existe la guiñada, en la Tabla 33 se especifica el porcentaje de error máximo en cada magnitud y orientación.

Tabla 33.

Errores porcentuales máximos ilusión coriolis 1 por interpolador cuadrático

Magnitud - Orientación	e_{pmax} %	Brazo
Posición-Alabeo	1.145	3
Posición-Guiñada	0.008	3
Velocidad-Alabeo	9.03	1
Velocidad-Guiñada	0.19	2
Aceleración-Alabeo	14.38	2
Aceleración-	1.17	2

En el caso de posición se observa un error máximo de 1.145% lo que indica una alta viabilidad de datos de ambos softwares para esta magnitud, en velocidad ya se muestra un valor que bordea el 9% en el brazo 1 esto se produce debido a que en el alabeo las velocidades son velocidades bajas por que una pequeña variación entre las mismas genera un error alto pero se encuentra dentro de la tolerancia de un error menor a 10%, para la aceleración en alabeo se encuentra un error de 14.38%, el cual se produce en el punto anterior de cambiar de alabeo a guiñada en el brazo, lo que indica al existir alabeo en donde el movimiento se prioriza entre los brazo 2 y 3, la junta 2 es la más propensa a fallar en el control de aceleración, en ese punto se debe considerar el error producido al momento de probar en la estructura real. El valor que varía es 3 mm/s^2 , con respecto al modelo planteado.

A continuación se muestra los resultados de las mismas magnitudes obtenidas por el interpolador cuadrático, pero con respecto al cúbico, a excepción de la aceleración debido a la convergencia y forma de cada interpolador, si se analiza esta magnitud se producen errores altos con respecto al cuadrático, por esta razón solo se analiza posición y velocidad.

En las Figura 64 y 65 se muestra los errores porcentuales de posición y velocidad entre el interpolador cúbico y ANSYS.

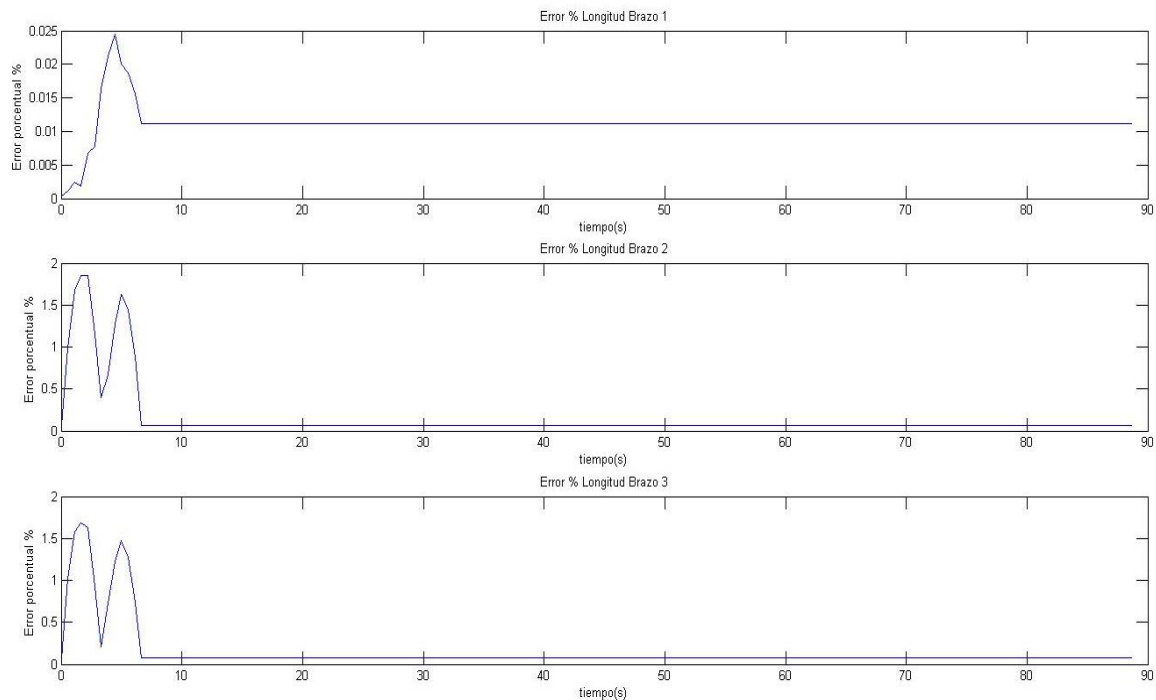


Figura 64. Error porcentual en posición coriolis1 por interpolador cúbico

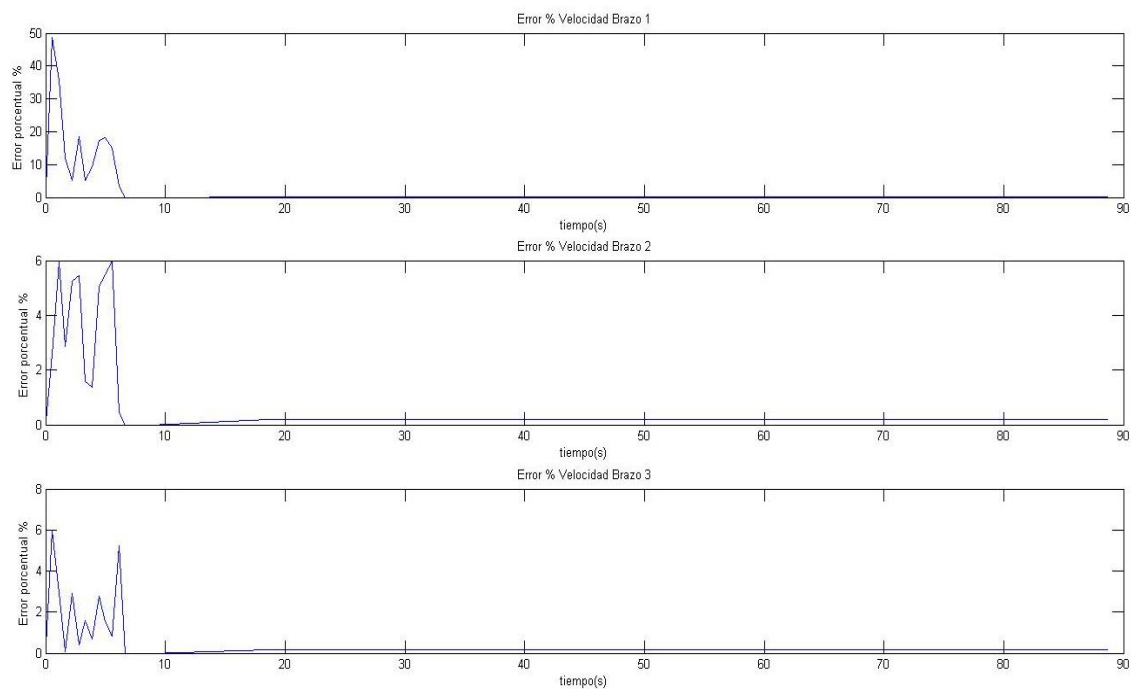


Figura 65. Error porcentual en velocidad coriolis1 por interpolador cúbico

En la Tabla 34 se muestra los errores porcentuales máximos en posición y velocidad donde puede visualizar en que brazo se produce el mayor error.

Tabla 34.

Errores porcentuales máximos ilusión coriolis 1 por interpolador cúbico

Magnitud - Orientación	$e_{pmax}\%$	Brazo
Posición-Alabeo	1.857	2
Posición-Guiñada	0.008	3
Velocidad-Alabeo	48.52	1
Velocidad-Guiñada	0.19	2

Se observa errores bajos en posición, pero en velocidad aparece un error de 48.52% debido a la misma razón del error en el cuadrático es porque el valor de la velocidad en el brazo 1 es bajo en el orden de los 0.33 mm/s como máxima velocidad por esta razón un pequeño cambio te produce un error alto, pero en los brazos 2 y 3 se observa errores menores al 10% por lo que los resultados en estos se considera aceptables.

A continuación se muestra las gráficas de posición, velocidad y aceleración en los vértices de la base móvil con los dos interpoladores y los resultados de ANSYS.

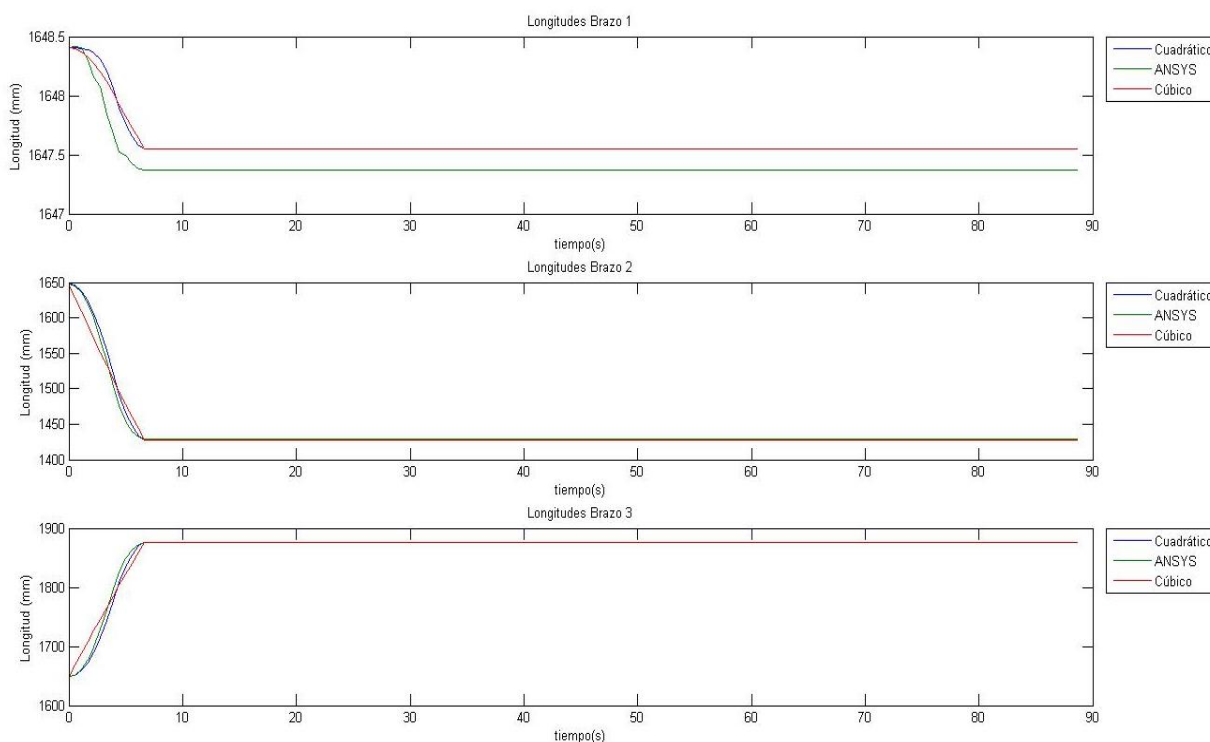


Figura 66. Posición mediante los interpoladores y ANSYS

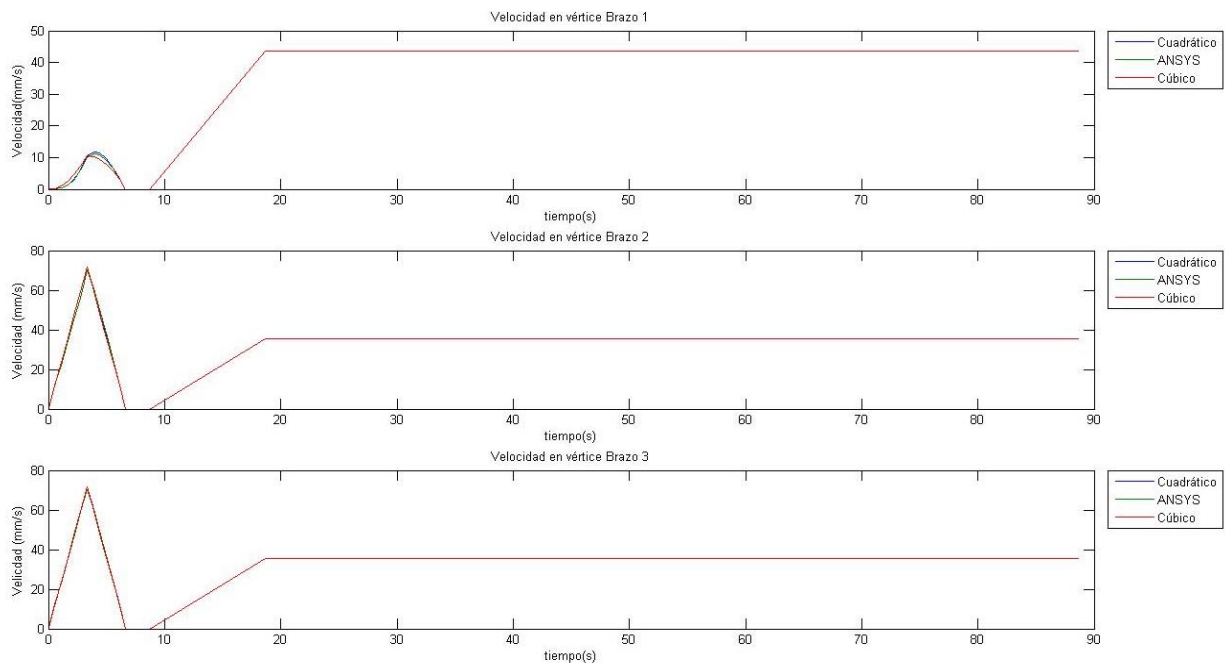


Figura 67. Velocidad mediante los interpoladores y ANSYS

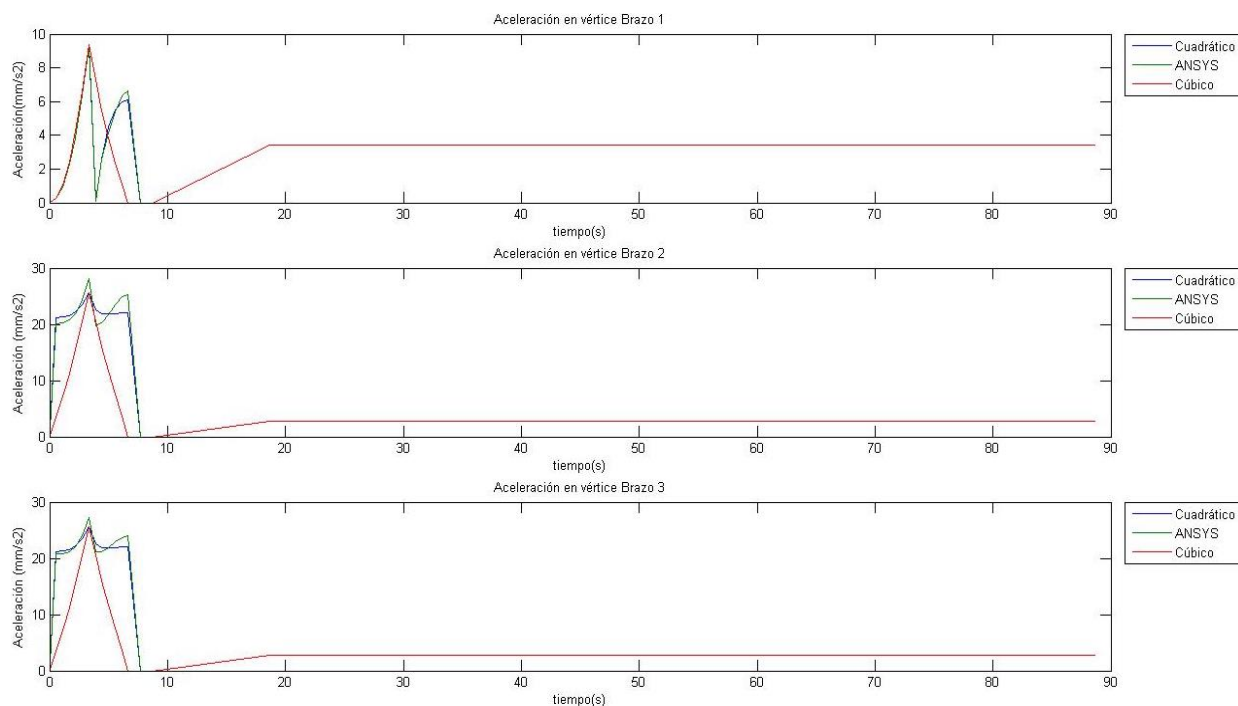


Figura 68. Aceleración mediante los interpoladores y ANSYS

En la gráfica de aceleración se observa las diferentes formas que tienen entre los interpoladores, en los resultados por el interpolador cúbico se muestra un error mayor con respecto al cuadrático, en la formas de gráficas de posición y velocidad se observa una trascendencia lineal en el interpolador cúbico por lo que se puede determinar que para cinemática basta con un interpolador cuadrático.

5.3.2 Barrena 1

En esta ilusión se produce las tres orientaciones, alabeo hasta 15° , cabeceo hasta 20° y guiñada una vuelta, en este caso se va analizar tanto el interpolador cuadrático como cúbico en un solo punto, igualmente se visualiza posición y velocidad con respecto a los dos interpoladores y aceleración solo en relación al cuadrático. En este caso hay un tramo donde actúan dos orientaciones que son alabeo y cabeceo, donde se analiza el efecto que se produce en los brazos del manipulador. A continuación se muestra los errores porcentuales en posición, velocidad de ANSYS con respecto a los dos interpoladores y la aceleración solo con respecto al interpolador cuadrático.

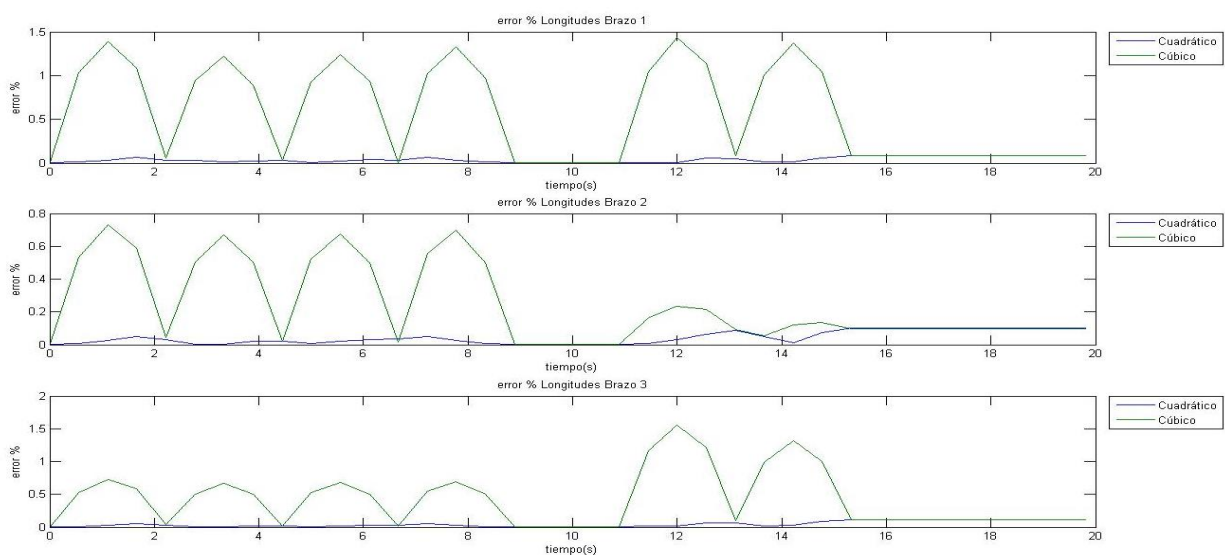


Figura 69. Error porcentual en posición barrena 1 por los dos interpoladores

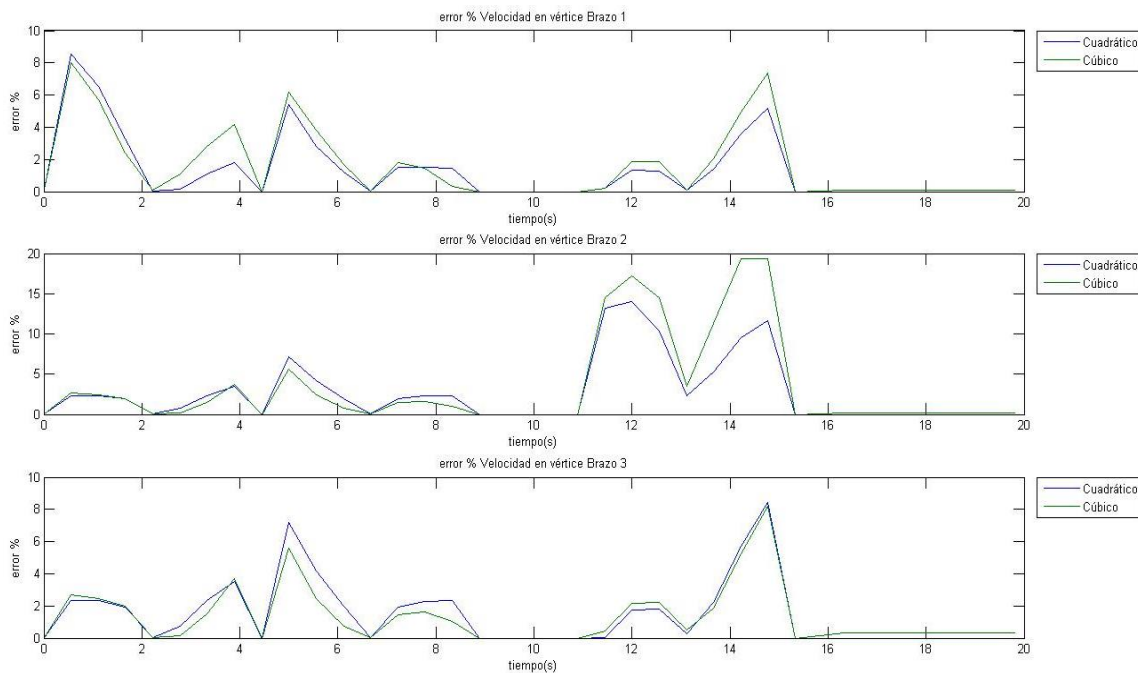


Figura 70. Error porcentual en velocidad barrena 1 por los dos interpoladores

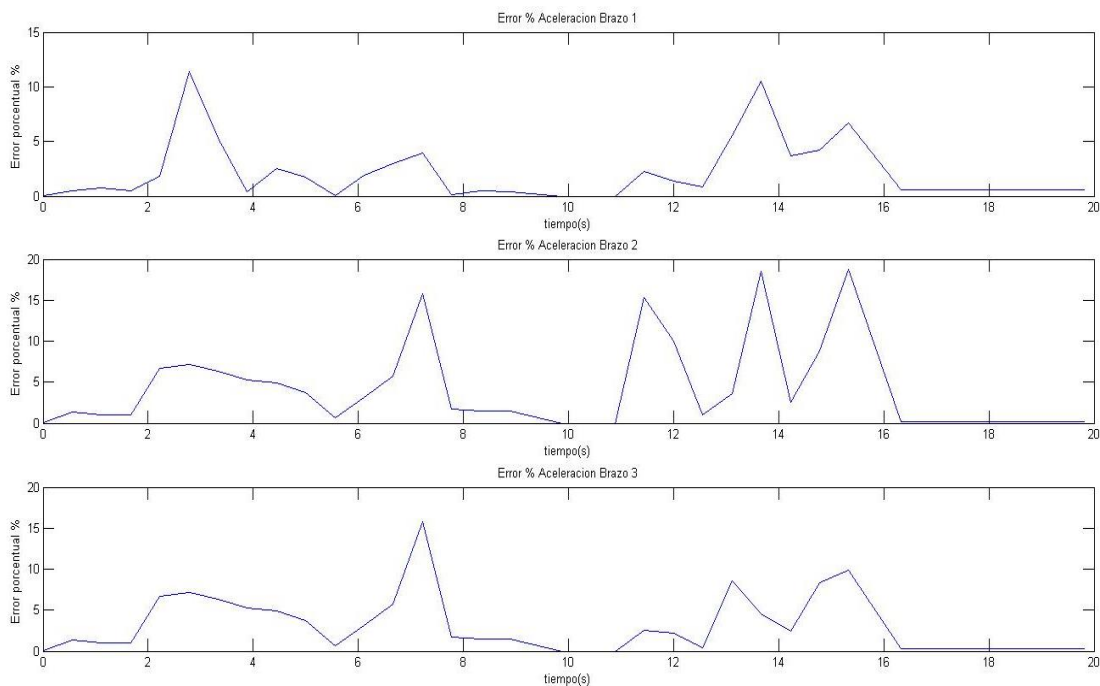


Figura 71. Error porcentual en posición barrena 1 por interpolador cuadrático

En la Tabla 35 se muestra los errores porcentuales máximos en las diferentes orientaciones de la ilusión barrena 1.

Tabla 35.*Errores porcentuales máximos ilusión coriolis 1 por interpolador cuadrático*

Magnitud -Orientación	$e_{pmax2}\%$	$e_{pmax3}\%$	Brazo
Posición-Cabeceo	0.07	1.38	1
Posición-Alabeo-Cabeceo	0.064	1.56	3
Posición-Guiñada	0.11	0.1	3
Velocidad-Cabeceo	8.55	8	1
Velocidad-Alabeo-cabeceo	14	19.36	2
Velocidad-Guiñada	0.32	0.32	3
Aceleración-Cabeceo	11.4	-	1
Aceleración-Alabeo-cabeceo	18.53	-	2
Aceleración-Guiñada	0.55	-	1

Como se observa en posición tanto en los dos interpoladores el error es mínimo, en velocidad se observa que al momento que hay las dos orientaciones alabeo y cabeceo se produce un error de 14% en el cuadrático y de casi 20% en el cúbico la explicación de este incremento en el error se debe a que el cabeceo se produce para abajo lo que genera la subida del pistón del brazo 2 mientras que el alabeo produce un descenso en el mismo brazo lo que crea un efecto dinámico que se ve reflejado en ANSYS mediante alguna fuerza en el brazo, cinemática mente se observa que el brazo llega a los puntos topes de control con un error bajo en los intermedios es donde se produce el incremento del error, por el efecto mencionado esto se soluciona con el jacobiano de fuerzas que permitiría agregar la parte dinámica al modelo, pero este tema ya queda como trabajo futuro debido a que el objetivo de este trabajo es comprobar la cinemática que como se observa está dentro de los errores tolerables. Otra opción del error puede ser la etapa de arranque de cada actuador debido a que el incremento de los errores se produce en los puntos intermedios iniciales, esto se solucionaría con un controlador para trabajar con esa etapa de los actuadores.

A continuación se muestra las gráficas de posición, velocidad y aceleración de los vértices de la base móvil por interpoladores y ANSYS.

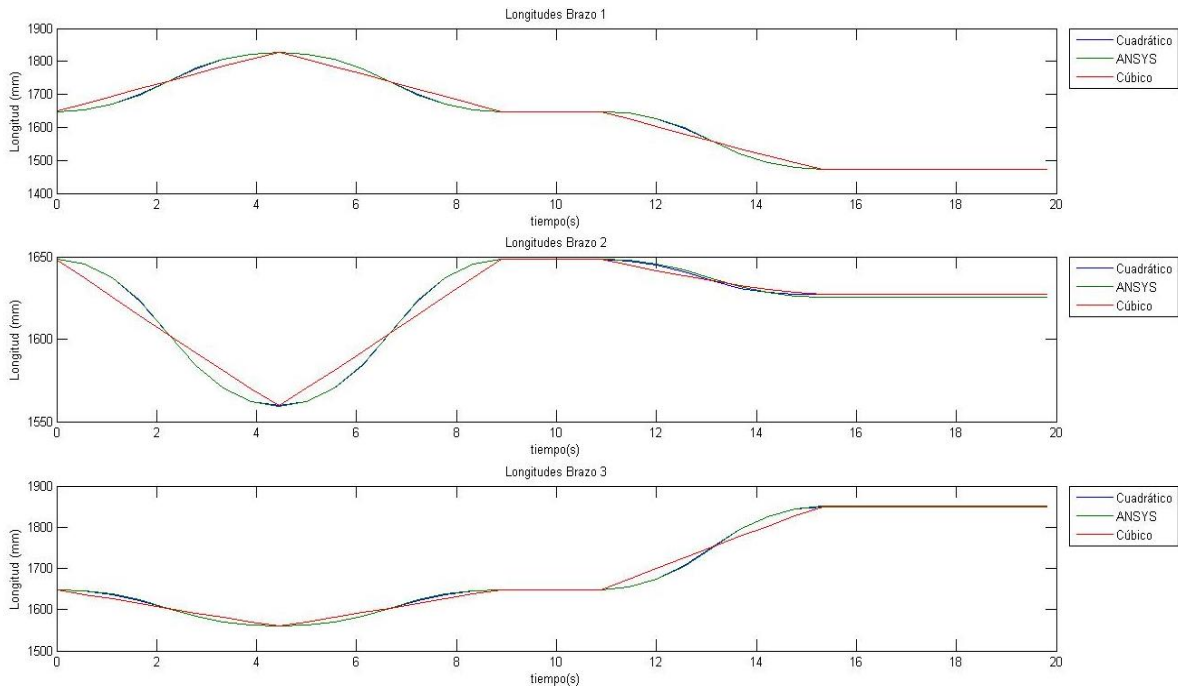


Figura 72. Posición barrena 1 mediante los interpoladores y ANSYS

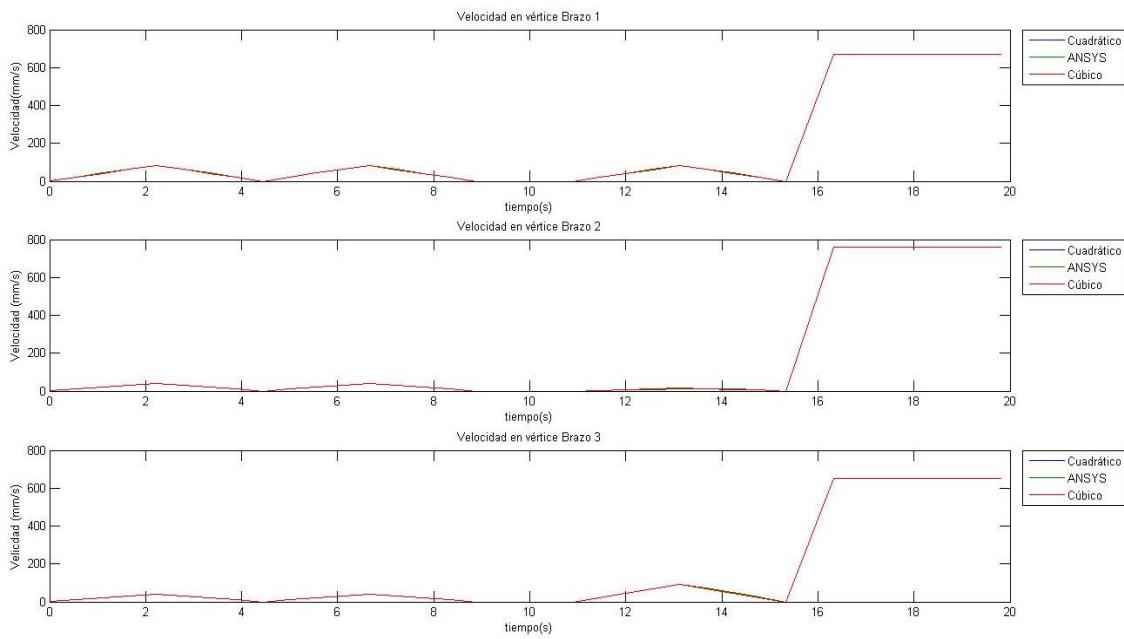


Figura 73. Velocidad barrena 1 mediante los interpoladores y ANSYS

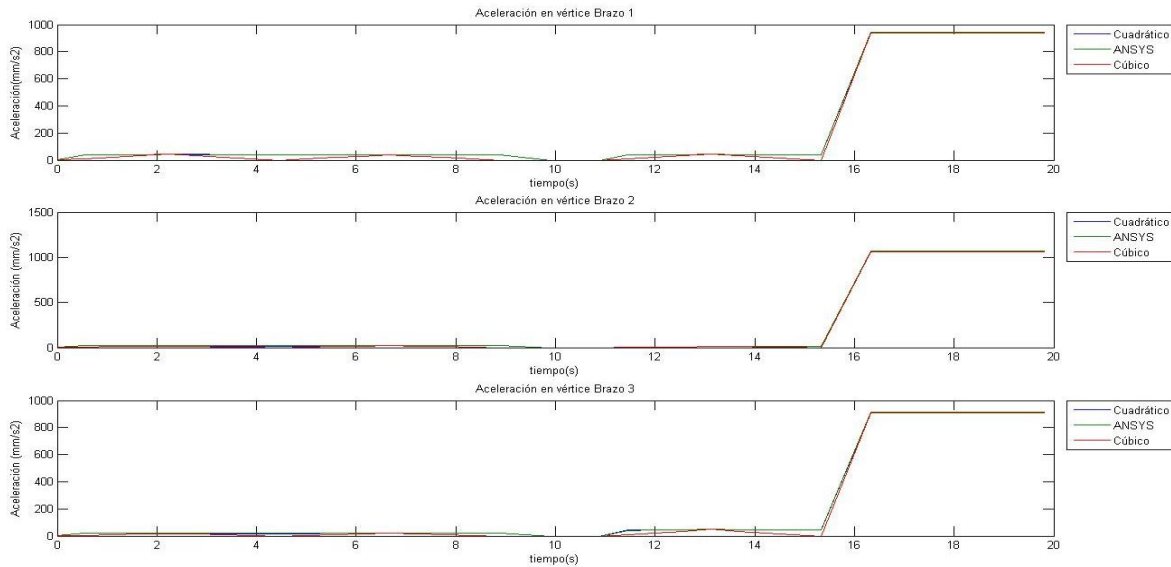


Figura 74. Aceleración barrena 1 mediante los interpoladores y ANSYS

Como se observa en posición y velocidad el interpolador cúbico sigue teniendo un comportamiento lineal cuando deberían ser curva de segundo y tercer orden respectivamente, por lo que se concluye que el resultado con menor se encuentra mediante el interpolador cuadrático. En esta ilusión se produce dos orientaciones al mismo tiempo lo que genera efectos sobre los brazo del manipulador principalmente en el brazo 2.

CAPÍTULO VI

6. CONCLUSIONES Y RECOMENDACIONES

6.1 Conclusiones

En este proyecto se desarrolló el modelo cinemático inverso del manipulador R-3RPS, usado para el simulador de desorientación espacial, a partir del método de los ángulos de Euler combinado con álgebra vectorial, con el fin de encontrar la posición de los vértices de la base móvil referencia al sistema de coordenadas de la plataforma giratoria, en donde se mantuvo la altura de referencia fija en 1647.39 mm, mediante las restricciones de movimiento se obtuvo las traslaciones en x, y del punto central de la base móvil, con el modelo obtenido se encontró las longitudes de los brazos y los ángulos de inferiores rotatorios, que son parámetros de entrada para la cinemática directa.

El modelo cinemático directo se obtuvo mediante la elección de uno de los dos métodos planteados, el primero era una combinación de vectores con geometría y el segundo fue la convención de Denavit Hartenberg. El método escogido fue el vectorial debido a que sus datos se aproximaban a la cinemática inversa, este modelo funciona en una relación maestro esclavo con el inverso debido a que recibe parámetro de entrada del modelo mencionado y calcula la posición de los vértices de la base móvil a partir de los datos de entrada.

Se obtuvo las velocidades y aceleraciones de ambos modelos mediante derivación de los mismos respecto al tiempo, se formó las matrices jacobianas cinemáticas para analizar las singularidades de movimiento tanto en velocidad y aceleración de ambos modelos.

Se realizó el control cinemático a partir de dos interpoladores que permitieron integrar puntos intermedios a las trayectorias de las ilusiones barrena y coriolis, a partir de los puntos ingresados se formó un esquema de combinación de ambos para encontrar una secuencia de desplazamientos,

velocidad y aceleración de los actuadores que sirvieron como datos de entrada a la simulación en ANSYS.

Se desarrolló tres interfaces en MATLAB, las dos primeras permitieron comprobar los resultados de los modelo cinemáticos en posición con respecto a la simulación en ANSYS, mientras que la tercera interfaz integró la combinación de ambos modelos, con las especificaciones de las trayectorias y los interpoladores, obteniendo como salida las variables de los actuadores prismáticos, rotatorios y de la plataforma.

Se verificó los resultados de los modelos cinemáticos obtenidos mediante datos estáticos, comprobando la simulación en ANSYS, definiendo el espacio de trabajo del manipulador, con el control cinemático se comprobó las trayectorias de las ilusiones barrena y coriolis en ANSYS, donde se obtuvo errores bajo en posición, tolerables en velocidad siendo estos menores al 10% y en aceleración se produjo un incremento de error al momento de la activación del alabeo en el brazo 2 llegando al 20% con el interpolador cuadrático, mientras que con el cúbico se produjo un comportamiento lineal , en donde se debe obtener curvas de segundo y tercer orden dando por descartado a este interpolador en cinemática.

Con el control cinemático se generó una base de datos de las ilusiones dinámicas trabajadas, para la estructura real en los puntos interpolados, complementado este estudio con la parte dinámico se puede realizar el control del manipulador mediante el análisis de los puntos de control entregados en este trabajo, se consideró en los actuadores prismático que para un cambio de sentido de movimiento tiene que llegar la velocidad a 0 para realizar esta acción.

6.2 Recomendaciones

En este trabajo se consideró el análisis cinemático lo que implica que no se tomó en cuenta las fuerzas, por esta razón en el estudio dinámico puede cambiar el sentido de movimiento de los

actuadores en base a la gravedad y momentos de inercias, por esto es recomendable trabajar con las matrices jacobianas para determinar el sentido y restricciones de movimiento, para este caso específico se usa para singularidades cinemáticas.

El software ANSYS presenta la opción de ingresar fuerzas si se requiere probar cinemática es recomendable omitir estas fuerzas en el espacio de trabajo.

REFERENCIAS BIBLIOGRÁFICAS

- Alcocer, W. (2012). *Obtención de parámetros aerodinámicos de una aeronave radiocontrolada de ala fija*. México DF: Instituto Nacional Politécnico.
- ANSYS. (2018). *Engineering Simulation & 3D design Software (ANSYS)*. Obtenido de Engineering Simulation & 3D design Software (ANSYS): <https://www.ansys.com/>
- Arias, R. (2013). *Diseño, construcción y control de una plataforma Stewart con 6 grados de libertad que funcione como simulador de vuelo*. Quito: Escuela Politécnica Nacional.
- Aristizabal, D., & Giraldo, J. (2015). *Control de una plataforma Stewart*. Pereira: Universidad Tecnológica de Pereira.
- Asif, U. (2012). *Design of a Parallel Robot with a Large Workspace for the Functional Evaluation of Aircraft Dynamics beyond the Nominal Flight Envelope*. International Journal of Advanced Robotic Systems, 1-13.
- Barraza, A., Rúa, J., Sosa, J., Díaz, J., Yime, E., & Roldán, J. (2016). *Modelado dinámico del manipulador serial Mitsubishi Movemaster RV-M1 usando SolidWorks*. Revista UIS Ingenierías, 49-62.
- Beard, R., & McLain, T. (2013). *Implementing Dubins Airplane Paths on Fixed-wing UAVs*. Brigham Young University.
- Bi, Z., & Jin, Y. (2011). *Kinematic modeling of Exechon parallel kinematic machine*. Robotics and Computer-Integrated Manufacturing, 27, 186-193.
- Bingul, Z., & Karahan, O. (2012). *Dynamic Modeling and Simulation of Stewart Platform*. En S. Kucuk, Serial and Parallel Robot Manipulators - Kinematics, Dynamics, Control and Optimization (págs. 19-42). Rijeka: Intech.
- Bozan, A. (2015). *Applying a model-based observer to quantitatively assess spatial disorientation and loss of energy state awareness*. Georgia: Georgia Institute of Technology.
- Brusola Fernández-Portolés, P. (2016). *Diseño y validación de sistemas de control para aeronaves basado en las herramientas software FlightGear y Matlab. Aplicación al diseño de pilotos automáticos*. Valencia: Universidad Politécnica de Valencia.
- Caetano, J. V., Weehuizen, M. B., de Visser, C. C., de Croon, G., de Wagter, C., Remes, B., & Mulder, M. (2014). *Rigid vs. Flapping: The Effects of Kinematic Formulations in Force Determination of a Free Flying Flapping Wing Micro Air Vehicle*. 2014 International Conference on Unmanned Aircraft Systems, ICUAS 2014 - Conference Proceedings, 345-356.
- Caetano, J., de Viesser, C., de Croon, G., Remes, B., de Wagter, C., Verboom, J., & Mulder, M. (2013). *Linear Aerodynamic Model Identification of a Flapping Wing MAV Based on Flight Test Data*. International Journal of Micro Air Vehicles, 273-286.

- Campos, A., de Faveri, G., Furtado, C., Reis, A., & Garcia, A. (2013). *Inverse kinematics for general 6-rus parallel robots applied on udesc-ceart flight simulator*. 22nd International Congress of Mechanical Engineering (COBEM 2013), 1-9.
- Cantón Romero, J. J. (2016). *Hispaviación*. Obtenido de Hispaviación: <http://www.hispaviacion.es/desorientacion-espacial/>
- Castañeda, H. (2013). *Control robusto por modos deslizantes de orden superior aplicado a vehículos aéreos no tripulados*. Nuevo León, México: Universidad Autónoma De Nuevo León.
- Castro, A., & Zubieta, C. (2014). *Simulador de vuelo utilizando plataformas paralelas de seis grados de libertad*. Bogotá: Universidad Militar Nueva Granada.
- Chablat, D., Jha, R., Rouillier, F., & Moroz, G. (2014). *Non-singular assembly mode changing trajectories in the workspace for the 3-RPS parallel robot*. 14th International Symposium on Advances in Robot Kinematics, 1-10.
- Chaparro Altamirano, D. (2013). *Análisis, Diseño y Control de un Manipulador Paralelo 3-SPS-IS Utilizado en Aplicaciones de Seguridad*. Monterrey: Instituto Tecnológico y de Estudios Superiores de Monterrey.
- CICTE. (2017). *Construcción de un simulador de desorientación espacial para contribuir a la seguridad aérea y entrenamiento de los pilotos de las FF.AA.* Sangolquí: Universidad de la Fuerzas Armadas ESPE.
- Cunliffe, C. (2009). *Desorientacion espacial*. Fuerza Aérea de Chile - Centro de Medicina Aeroespacial.
- Dančuo, Z., Kvirgić, V., Rašuo, B., & Vidaković, J. (2013). *On dynamics of a spatial disorientation trainer for pilot training*. Forth Serbian (29th Yu) Congress on Theoretical and Applied Mechanics, (págs. 1-6). Vrnjačka Banja.
- Escuela de Aviación Naval. (2016). *Simulador de desorientación espacial*. Armada del Ecuador.
- Estevez, J. (2014). *Simulador de vuelo: realidad aumentada*. Barcelona: Universidad Politecnica de Catalunya.
- Fernández, L., & Sotomayor, L. (2016). *Análisis cinemático inverso y directo del robot paralelo*. Quito: Escuela Politécnica Nacional.
- Flórez Vergara, D. E., & Castro Riveros, F. C. (2014). *Planeacion y ejecucion de trayectorias en un robot delta*. Bogotá D.C: Universidad Militar Nueva Granada.
- Flórez-Vergara, E., D., Castro-Riveros, F. C., & Castillo-Esteba, R. A. (2016). *Planeación y ejecución de trayectorias en el espacio de trabajo para un robot Delta*. Fac. Ing., vol. 25 (42), 75–88.
- Gallardo, J., Rico, J., & Orozco, H. (2004). *Un Algoritmo para Resolver la Cinemática Directa de Plataformas Gough-Stewart Tipo 6-3*. Computación y Sistemas 8, 132-149.

- Garcia, H., Pereda, F., Giron-Sierra, J., & Espinosa, F. (2012). *UAV attitude estimation using Unscented Kalman Filter and TRIAD*. IEEE Transactions on Industrial Electronics, Volume 59, Issue 11, 4465-4474.
- Gastaldi, L., Pastorelli, S., & Sorli, M. (2009). *Vestibular apparatus: Dynamic model of the semicircular canals WIT Transactions*. Biomedicine and Health, 8th International Conference on Modelling in Medicine and Biology, 223-234.
- González, G. (2008). *Plataforma de movimiento de seis grados de libertad aplicable a entrenador de vuelo*. Lima: Universidad Ricardo Palma.
- Gonzales-Villela, V. J., Martínez-Zamudio, P., López-Parra, M., Cruz-López, I. C., Flores-Méndez, J. d., & Díaz-Hernández, O. (2013). *Análisis, instrumentación y manipulación de un robot paralelo híbrido*. Memorias del xix congreso internacional anual de la Somim, Pachuca.
- González, N., & Reinoso, E. (2011). *Estudio, diseño y construcción de una plataforma robótica didáctica tipo Stewart aplicada al estudio de controladores difusos*. Cuenca: Universidad Politécnica Salesiana.
- Gouasmi, M., Ouali, M., Fernini, B., & Meghatria, M. (2012). *Kinematic Modelling and Simulation of a 2-R Robot Using SolidWorks and Verification by MATLAB/Simulink*. International Journal of Advanced Robotic Systems, 1-13.
- Hernández, L., Rubio, E., Prieto, P., Hernández, A., & Izaguirre, E. (2011). *Control Desacoplado de Plataforma Neumática de 3-GDL utilizada como Simulador de Movimiento*. Revista Iberoamericana de Automática e Informática industrial 8, 345–356.
- Hernando, M. (2017). *Capítulo 6. Cinemática diferencial*. En M. Hernando, Apuntes de Robótica (págs. 1-33). Madrid: Universidad Politécnica de Madrid.
- Huang, Y., Wang, J., & Tham, D. M. (2017). *Kinematics Modelling and ADAMS MATLAB/Simulink Co-Simulation for Automated Aerobridge Docking Process*. 3rd International Conference on Control, Automation and Robotics (ICCAR), 304-309.
- Iriarte, E., & Noguera, P. (2012). *Controlador para robot delta-lineal con generación de trayectorias continuas mediante protocolo abierto*. Seminario de Inteligencia Artificial y Robótica - UTN - FRBA, 1-8.
- Izaguirre, E. (2012). *Control cinemático en el espacio de tareas de robot paralelo neumático en aplicación de simulador de movimiento*. Santa Clara: Universidad Central “Marta Abreu” de las Villas.
- Krause, G., Zapico, E., & Rodríguez, S. (2012). *Simulador de vuelo de vehículos aeroespaciales de seis grados de libertad*. Mecánica Computacional 31, 2961-2977.
- Kvrgic, V., Visnjic, Z., Cvijanovic, V., Divnic, D., & Mitrovic, S. (2015). *Dynamics and control of a spatial disorientation trainer*. Robotics and Computer-Integrated Manufacturing 2015, 1-28.

- Limache, A., Rojas, P., & Murillo, M. (2010). *Diseño de un moderno simulador de vuelo en tiempo real*. International Center of Computational Methods in Engineering (CIMEC), 4955-4966.
- Lombaerts, T., Looye, G., Chu, Q., & Mulder, J. (2012). *Design and simulation of fault tolerant flight control based on a physical approach*. Aerospace Science and Technology 23, 151-171.
- López, Y. (2008). *Modelo Cinemático para Robot Paralelo de tres grados de libertad*. Santa Clara: Universidad Central “Marta Abreu” de Las Villas.
- Lukács, L., & Lantos, B. (2012). *Data fusion and primary image processing for aircraft identification*. Electrical Engineering and Computer Science 56/3, 83-94.
- Lynge, K., Blanke, M., & Arne Johansen, T. (2015). *IFAC-PapersOnLine* 48, 541-546.
- Malamud, E., & Scarpin, G. (2014). *Desarrollo de un simulador de vuelo de seis grados de libertad del FMA IA-63 Pampa*. Córdoba: Instituto Universitario Aeronáutico.
- Matlalcuatzi, E., Alexandrov, V., Matlalcuatzi, F., Altamirano, L., Reyes, M., Moctezuma, J., & Hernandez, O. (2014). *Diseño del simulador dinámico para pilotos como un sistema biomecatrónico*. Memorias del XVI Congreso Latinoamericano de Control Automático (págs. 1-6). Cancún, México: CLCA.
- Mauro, S., Gastaldi, L., Pastorelli, S., & Sorli, M. (2016). *Dynamic Flight Simulation with a 3 d.o.f. Parallel Platform*. International Journal of Applied Engineering Research ISSN 0973-4562, 9436-9442.
- OACI. (2012). *Manual de medicina aeronáutica*. Recuperado el 20 de Julio de 2017, de http://www.icao.int/publications/Documents/8984_cons_es.pdf
- Oliveira, T., & Encarnação, P. (2013). *Ground Target Tracking Control System for Unmanned Aerial Vehicles*. Journal of Intelligent and Robotic Systems, 373-387.
- Ospina, D. (2012). *Cinematica y Simulacion de una Plataforma Robotica Paralela 3-RPS*. III Congreso internacional de ingeniería mecatrónica. Cali, Colombia.
- Pastorelli, S., & Battezzato, A. (2009). *Singularity analysis of a 3 degrees-of-freedom parallel manipulator*. Proceedings of the 5th International Workshop on Computational Kinematics, 331-340.
- Patterson, F. R., & Chandler, J. F. (2010). *Effect of acceleration frequency on spatial orientation mechanisms*. Florida: Naval Aerospace Medical Research Laboratory.
- Peddle, I. (2005). *Autonomous Flight of a model aircraft*. University of Stellenbosch.
- Pérez Menéndez, F. (2014). *Desarrollo e un sistema de control para un manipulador de seis grados de libertad*. Oviedo: Universidad de Oviedo.
- Ramana Babu, S., Ramachandra Raju, V., & Ramji, K. (2013). *Design for optimal performance of 3-rps parallel manipulator using evolutionary algorithms*. Transactions of the Canadian Society for Mechanical Engineering, Vol. 37, No. 2, 135-160.

- Ramírez Rodríguez, D., & Méndez Moreno, L. (2011). *Modelamiento dinámico de mecanismos paralelos con actuadores prismáticos utilizando SimMechanics de Matlab*. *Intekhnia* 6, 75 - 85.
- Ran, D., & Chuangchuang, S. (2015). *Path Planning of Spatial Rigid Motion with Constrained Attitude*. *Journal of Guidance, Control, and Dynamics*, 1356-1365.
- Rodriguez, J. (2012). *Modelo cinemático y control de un brazo robótico imprimible*. Madrid: Universidad Carlos III de Madrid.
- Ruiz Hidalgo, N. C., Blanco Ortega, A., Abúndez Pliego, A., Colín Ocampo, J., Alcocer Rosado, W. M., & Arias Montiel, M. (2017). *Dinámica y control de un robot paralelo 3-rps*. *Pistas Educativas* Vol. 39 No. 125, 518-542.
- Salehinia, Y., Najafi, F., Sadati, H., Shiee, M., & Salehinia, S. (2013). *Solving Forward Kinematics Problem of Stewart Robot Using Soft Computing*. *RSI/ISM International Conference on Robotics and Mechatronics*, 1-6.
- Schopferer, S., & Pfeifer, T. (2015). *Performance-Aware Flight Path Planning for Unmanned Aircraft in Uniform Wind Fields*. *International Conference on Unmanned Aircraft System (ICUAS)*, 1-11.
- Small, R., Wickens, C., Oster, A., Keller, J., & French, J. (2004). *Multisensory integration for pilot spatial orientation (MIPSO)*. United States Air Force Research Laboratory.
- VanEykeren, L., & Chu, Q. (2014). *Sensor fault detection and isolation for aircraft control systems by kinematic relations*. *Control Engineering Practice* 31, 200–210.
- Vargas, W. L., Riaño, C. A., & Pineda, L. M. (2005). *Ambientes Virtuales Para la enseñanza de la Mecánica de Fluidos: Algunos ejemplos simplificados aplicando ANSYS*. *Ciencia e Ingeniería Neogradina* 15, 94-115.
- Vázquez Hernández, J., & Cuenca Jiménez, F. (2009). *Cinemática Inversa y Análisis Jacobiano del Robot Paralelo Hexa*. *Memorias del xv congreso internacional anual de la Somim* (págs. 800- 810). CD. Obregón, Sonora. México: Somim.
- Villarroel, J. (2011). *Diseño e implementación de un controlador de posición para una plataforma de stewart con acomodación activa*. Barcelona: Universitat Politècnica de Catalunya.