

DISEÑO E IMPLEMENTACIÓN DE UNA PLATAFORMA ROBÓTICA TIPO DELTA

Autores

LUCAS MATEO URREA MANTILLA
SERGIO ALEJANDRO MEDINA PAPAGAYO

UNIVERSIDAD MILITAR NUEVA GRANADA
FACULTAD DE INGENIERÍA
PROGRAMA DE INGENIERÍA EN MECATRÓNICA
BOGOTÁ, D.C.
2012

DISEÑO E IMPLEMENTACIÓN DE UNA PLATAFORMA ROBÓTICA TIPO DELTA

Autores

LUCAS MATEO URREA MANTILLA
SERGIO ALEJANDRO MEDINA PAPAGAYO

Trabajo de Grado para optar por el título de
Ingeniero en Mecatrónica

Director

Ing. RICARDO ANDRÉS CASTILLO ESTEPA MS.c

UNIVERSIDAD MILITAR NUEVA GRANADA
FACULTAD DE INGENIERÍA
PROGRAMA DE INGENIERÍA EN MECATRÓNICA
BOGOTÁ, D.C.
2012

Nota de aceptación:

El trabajo de grado titulado “DISEÑO E IMPLEMENTACIÓN DE UNA PLATAFORMA ROBÓTICA TIPO DELTA”, elaborado y presentado por los estudiantes Lucas Mateo Urrea Mantilla y Sergio Alejandro Medina Papagayo, como requisito para optar al título de Ingeniero en Mecatrónica, fue aprobado por el jurado calificador.

Prof. Ing. Ricardo Castillo M.Sc

Firma del jurado

Firma del jurado

Bogotá 14 de Septiembre de 2012

DEDICATORIA

Dedicamos este trabajo de grado a Dios principalmente. Además, agradecemos a nuestros padres Patricia Mantilla, Jorge Urrea, Vilma Papagayo y Oswaldo Medina por el apoyo y la confianza depositada en todos estos años y por todos sus esfuerzos para sacar a sus hijos adelante.

AGRADECIMIENTOS

Los autores agradecen a:

La universidad Militar Nueva Granada, especialmente al programa de ingeniería en Mecatrónica y a todos sus docentes por creer en este proyecto y por dar todo su apoyo, no solo económico sino también intelectual para que el proyecto pudiera ser desarrollado exitosamente.

También agradecen al director del proyecto, el ingeniero Ricardo Andrés Castillo quien es un entusiasta por la investigación en el área de la robótica en la universidad y en especial, en los robots paralelos. Además, agradecen todo su tiempo y dedicación, sus consejos oportunos y su compromiso con el proyecto.

A Gustavo Andrés Gil, amigo incondicional, quien siempre estuvo pendiente de la evolución del proyecto y quien en todo momento dio sus puntos de vista, tanto en las etapas de diseño e implementación como en la etapa de redacción de este libro, enriqueciendo de forma significativa esta investigación. A Edward Chaparro, amigo que estuvo pendiente en la fase de implementación del robot y que gracias a su ayuda, se pudieron mecanizar unas piezas en la fresa CNC de la universidad.

Y a muchas otras personas que de forma indirecta, permitieron que el proyecto se desarrollara satisfactoriamente.

Lucas Mateo Urrea Mantilla
Sergio Alejandro Medina Papagayo

RESUMEN

El objetivo principal de esta investigación es diseñar e implementar un prototipo de un robot Delta, solucionando los problemas de espacio de trabajo reducido y gran número de singularidades, optimizando las dimensiones de sus eslabones para maximizar su espacio de trabajo y su destreza. Adicionalmente, se desarrollan los modelos cinemáticos directo e inverso y se implementan en una interfaz gráfica de usuario para controlar los movimientos del robot desde un computador.

SolidWorks permitió diseñar y validar, por el método de elementos finitos, cada uno de los componentes del robot. Su modelo cinemático se desarrolló geométricamente y la optimización del espacio de trabajo y la destreza del robot se llevó a cabo por medio de seis algoritmos genéticos multi-objetivos: Random Weight Genetic Algorithm, Fixed Weight Genetic Algorithm, Weight Genetic Algorithm, Non-dominated Sorting Genetic Algorithm, Niched Pareto Genetic Algorithm y Multiplied Objectives Genetic Algorithm, cada uno de ellos implementado por tres métodos de selección diferentes : ruleta, truncamiento y el muestreo universal estocástico. La interfaz gráfica de usuario se implementó en Matlab que permite transmitir datos del computador al robot por el puerto serial.

El resultado del trabajo es un robot Delta de 4 grados de libertad para el laboratorio de robótica de la universidad Militar Nueva Granada, en donde las dimensiones de sus eslabones garantizan un buen espacio de trabajo y un índice global de destreza mayor a 0,47.

Palabras Claves:

Robot Delta, espacio de trabajo, destreza, cinemática directa e inversa, interfaz gráfica de usuario, algoritmos genéticos multi-objetivos.

ABSTRACT

The main aim of the research is to design and to implement a Delta robot prototype solving the problems of low workspace and many singularities, optimizing its links dimensions in order to maximizing its workspace and its dexterity. Also, the direct and inverse kinematic models were developed and implemented in a graphic user interface for controlling the movements' robot from a computer.

SolidWorks let to design and to validate by finite elements method each components robot. Its kinematic model was geometrically developed and its workspace and dexterity optimization were developed by six multi-objective genetic algorithm: Random Weight Genetic Algorithm, Fixed Weight Genetic Algorithm, Weight Genetic Algorithm, Non-dominated Sorting Genetic Algorithm, Niched Pareto Genetic Algorithm y Multiplied Objectives Genetic Algorithm, each one was implemented by three different selection methods: roulette selection, truncate selection and stochastic universal sampling selection. The graphic user interface was implemented in Matlab and it let the data transmission from a computer to the robot using the serial port.

The result of the research is a Delta robot of four degrees of freedom from the robotic lab to the Militar University. The links dimensions of the robot guarantee a good workspace and a global dexterity index greater than 0,47.

Key words:

Delta Robot, workspace, dexterity, direct and inverse kinematic, graphic user interface, multi-objective genetic algorithms.

CONTENIDO

CAPÍTULO 1: INTRODUCCIÓN	14
INTRODUCCIÓN	14
PLANTEAMIENTO DEL PROBLEMA	15
OBJETIVOS	18
JUSTIFICACIÓN	18
METODOLOGÍA	19
CAPÍTULO 2: MARCO CONCEPTUAL	21
2.1 ¿QUÉ ES UN ROBOT?.....	21
2.2 CLASIFICACIÓN DE LOS ROBOTS	21
2.2.1 Robots Manipuladores.....	21
2.2.2 Robots Móviles	22
2.3 ESTRUCTURA MECÁNICA DE UN ROBOT MANIPULADOR	23
2.4 CARACTERÍSTICAS DE UN ROBOT MANIPULADOR.....	24
2.4.1 Espacio de trabajo.....	24
2.4.2 Capacidad de carga	24
2.4.3 Velocidad.....	25
2.4.4 Precisión, repetibilidad y resolución	25
2.5 CLASIFICACIÓN DE LOS ROBOTS MANIPULADORES.....	25
2.5.1 Manipuladores en Serie.....	25
2.5.2 Manipulador en Paralelo.....	26
2.6 CINEMÁTICA DE UN ROBOT MANIPULADOR	28
2.6.1 Cinemática Directa	28
2.6.2 Cinemática Inversa.....	28
2.7 ALGORITMOS GENÉTICOS	29
2.7.1 Operadores Genéticos	30
2.8 PROBLEMAS DE OPTIMIZACIÓN MULTI-OBJETIVO (MOP)	31

2.8.1 Niched Pareto Genetic Algorithm (NPGA)	33
2.8.2 Non-Dominated Sorting Genetic Algorithm (NSGA)	33
2.8.3 Random-Weight Genetic Algorithm (RWGA).....	34
2.9 MÉTODO DE ELEMENTOS FINITOS	36
CAPÍTULO 3: ESTADO DEL ARTE.....	37
3.1 HISTORIA DE LOS ROBOTS PARALELOS.....	37
3.2 ¿POR QUÉ LOS ROBOTS PARALELOS?	39
3.3 APLICACIONES EN LA ACTUALIDAD.....	40
CAPÍTULO 4: CINEMÁTICA DEL ROBOT DELTA Y ESPACIO DE TRABAJO	42
4.1 GEOMETRÍA DEL ROBOT DELTA	42
4.2 CINEMÁTICA INVERSA	45
4.2.1 Variable de articulación θ_{11}	47
4.2.2 Variable de articulación θ_{12}	47
4.2.3 Variable de articulación θ_{13}	47
4.3 CINEMÁTICA DIRECTA	48
4.4 MATRIZ JACOBIANA	51
4.4.1 Singularidades de la cinemática inversa	53
4.4.2 Singularidades de la cinemática directa	53
4.4.3 Combinación de singularidades	54
4.4.4 Destreza del robot delta	54
4.5 ESPACIO DE TRABAJO DEL ROBOT DELTA.....	54
4.5.1 Optimización del espacio de trabajo del robot.....	57
4.5.2 Resultados del algoritmo de optimización	60
CAPÍTULO 5: DISEÑO E IMPLEMENTACIÓN DEL ROBOT DELTA.....	73
5.1 DISEÑO MECÁNICO	73

5.1.1 Bases fija y móvil.....	73
5.1.2 Eslabones.....	76
5.1.3 Junturas universales.....	76
5.1.4 Estructura soporte	77
5.1.5 Selección de materiales	78
5.1.6 Diseño asistido por computadora (CAD)	79
5.1.7 Resultados de Análisis por elementos finitos	80
5.1.8 Implementación de la estructura mecánica del robot	84
5.2 DISEÑO ELECTRÓNICO.....	86
5.2.1 Selección de los actuadores.....	86
5.2.2 Diseño del circuito electrónico.....	87
5.2.3 Resultados del circuito electrónico	88
5.3 SOFTWARE DE CONTROL DE MOVIMIENTOS DEL ROBOT	89
5.3.1 Comunicación computador-robot	89
5.3.2 GUI (Interfaz gráfica de usuario)	91
5.3.3 Resultados de la GUI	91
5.4 RESUMEN	95
CAPÍTULO 7: CONCLUSIONES Y TRABAJOS FUTUROS	96
7.1 CONCLUSIONES	96
7.2 TRABAJOS FUTUROS	97
BIBLIOGRAFÍA	99

ÍNDICE DE FIGURAS

Figura 1. Robot Paralelo tipo Delta. CAD en SolidWorks.	14
Figura 2. Robot manipulador en serie.	16
Figura 3. Solución de la cinemática inversa planteada por Tsai en 1966.	17
Figura 4. Solución cinemática inversa planteada por Tsai en 1966.	17
Figura 5. Componentes de un manipulador.	21
Figura 6. Efector final. CAD SolidWorks.	22
Figura 7. Robot móvil.	23
Figura 8. Configuraciones de un robot con ruedas.	23
Figura 9. Manipulador en serie. CAD SolidWorks.	25
Figura 10. Manipulador paralelo.	26
Figura 11. Robot paralelo planar.	27
Figura 12. Robot Delta.	27
Figura 13. GA básico.	29
Figura 14. Espacio de Criterio de un MOP con dos funciones objetivo.	33
Figura 15. Diagrama de bloques de un GA multi-objetivo.	35
Figura 16. Primer mecanismo paralelo (US Patent No. 1789680).	37
Figura 17. Primer robot paralelo industrial (US Patent No. 2286571).	37
Figura 18. Robot Delta inventado de Clavel (US Patent No. 4976582).	39
Figura 19. SurgiScope system by ISIS.	40
Figura 20. Dispositivo Falcon de la empresa Novint.	41
Figura 21. Estructura del robot paralelo RUU.	42
Figura 22. Modelo geométrico del robot Delta.	43
Figura 23. Descripción de los ángulos: a) vista frontal y b) vista lateral.	43
Figura 24. Forma geométrica de a) la base fija y b) la plataforma móvil.	45
Figura 25. Representación vectorial de la primera cadena cinemática.	46
Figura 26. Cadena cinemática i en las coordenadas $X_i Y_i Z_i$	50
Figura 27. Espacio de trabajo del Robot Delta, método de Monte Carlo.	55
Figura 28. Restricciones del ángulo θ_{i3}	56
Figura 29. Restricción del ángulo θ_{i2}	56
Figura 30. Diagrama de flujo del programa del espacio de trabajo del robot.	57
Figura 31. Espacio de solución del problema.	58

Figura 32. Diagrama de flujo de la evaluación de las funciones objetivo para un individuo	60
Figura 33. Tiempo de evaluación de las funciones objetivo para un individuo según el incremento de las variables de los actuadores	60
Figura 34. Conjunto de soluciones del algoritmo RWGA, MATLAB.....	61
Figura 35. Conjunto de soluciones del algoritmo MOFGA, MATLAB.....	61
Figura 36. Conjunto de soluciones del algoritmo FWGA, MATLAB.....	61
Figura 37. Conjunto de soluciones del algoritmo AWGA, MATLAB.....	62
Figura 38. Conjunto de soluciones del algoritmo NSGA, MATLAB.....	62
Figura 39. Conjunto de soluciones del algoritmo NPGA, MATLAB.....	62
Figura 40. Conjunto de soluciones con el método de truncamiento, MATLAB.	63
Figura 41. Evolución de las dimensiones de los eslabones del robot: algoritmo RWGA método de truncamiento, MATLAB.....	63
Figura 42. Evolución de las dimensiones de los eslabones del robot: algoritmo MOFGA método de truncamiento, MATLAB.....	63
Figura 43. Evolución de las dimensiones de los eslabones del robot: algoritmo FWGA método de truncamiento, MATLAB.....	64
Figura 44. Evolución de las dimensiones de los eslabones del robot: algoritmo AWGA método de truncamiento, MATLAB.....	64
Figura 45. Evolución de las dimensiones de los eslabones del robot: algoritmo NSGA método de truncamiento, MATLAB.....	64
Figura 46. Evolución de las dimensiones de los eslabones del robot: algoritmo NPGA método de truncamiento, MATLAB.....	65
Figura 47. Conjunto de soluciones con el método SUS, MATLAB.....	65
Figura 48. Evolución de las dimensiones de los eslabones del robot: algoritmo RWGA método de muestreo universal estocástico, MATLAB.	66
Figura 49. Evolución de las dimensiones de los eslabones del robot: algoritmo MOFGA método de muestreo universal estocástico, MATLAB.	66
Figura 50. Evolución de las dimensiones de los eslabones del robot: algoritmo FWGA método de muestreo universal estocástico, MATLAB.....	66
Figura 51. Evolución de las dimensiones de los eslabones del robot: algoritmo AWGA método de muestreo universal estocástico, MATLAB.	67
Figura 52. Evolución de las dimensiones de los eslabones del robot: algoritmo NSGA método de muestreo universal estocástico, MATLAB.	67
Figura 53. Evolución de las dimensiones de los eslabones del robot: algoritmo NPGA método de muestreo universal estocástico, MATLAB.	67
Figura 54. Conjunto de soluciones con el método de la ruleta, MATLAB.	68
Figura 55. Evolución de las dimensiones de los eslabones del robot: algoritmo RWGA método de la ruleta, MATLAB.....	68

Figura 56. Evolución de las dimensiones de los eslabones del robot: algoritmo MOFGA método de la ruleta, MATLAB.....	69
Figura 57. Evolución de las dimensiones de los eslabones del robot: algoritmo FWGA método de la ruleta, MATLAB	69
Figura 58. Evolución de las dimensiones de los eslabones del robot: algoritmo AWGA método de la ruleta, MATLAB.....	69
Figura 59. Evolución de las dimensiones de los eslabones del robot: algoritmo NSGA método de la ruleta, MATLAB.....	70
Figura 60. Evolución de las dimensiones de los eslabones del robot: algoritmo NPGA método de la ruleta, MATLAB.....	70
Figura 61. Tiempo en segundos la de ejecución de los algoritmos, EXCEL.....	70
Figura 62. Frontera de Pareto del conjunto final de soluciones, MATLAB.....	71
Figura 63. Espacio de trabajo del robot Delta, MATLAB.....	72
Figura 64. Vista superior del espacio de trabajo del robot, MATLAB.....	72
Figura 65. Vista del espacio de trabajo del robot en plano XZ, MATLAB.	72
Figura 66. Vista del espacio de trabajo del robot en plano YZ, MATLAB.	72
Figura 67. Base fija del robot, CAD SolidWorks.....	74
Figura 68. Vista superior de la base fija, CAD SolidWorks.	74
Figura 69. Explosión del soporte de los actuadores de la base fija, CAD SolidWorks.....	74
Figura 70. Base móvil, CAD SolidWorks.....	75
Figura 71. Soportes de la base móvil, CAD SolidWorks.	75
Figura 72. Explosión de un soporte de la base móvil, CAD SolidWorks.	75
Figura 73. Una cadena cinemática del robot, CAD SolidWorks.	76
Figura 74. Primer eslabón de la cadena cinemática, CAD SolidWorks.....	76
Figura 75. Explosión de las juntas en el primer eslabón, CAD SolidWorks.	77
Figura 76. Explosión de las juntas universales, CAD SolidWorks.	77
Figura 77. Estructura soporte, CAD SolidWorks.	78
Figura 78. Robot Delta, CAD SolidWorks.	79
Figura 79. Componentes del Robot Delta, CAD SolidWorks.	80
Figura 80. Fuerzas y reacciones del eje principal, SolidWorks.	81
Figura 81. Mallado del eje principal, SolidWorks.	81
Figura 82. Deformaciones en el eje principal, SolidWorks.	81
Figura 83. Desplazamientos en el eje principal, SolidWorks.	82
Figura 84. Tensiones en el eje principal, SolidWorks.	82
Figura 85. Fuerzas y reacciones del primer eslabón, SolidWorks.	82
Figura 86. Mallado del primer eslabón, SolidWorks.....	83
Figura 87. Deformaciones del eslabón, SolidWorks.	83

Figura 88. Desplazamientos del eslabón, SolidWorks	83
Figura 89. Tensiones del eslabón, SolidWorks	84
Figura 90. Base fija del primer prototipo.	84
Figura 91. Base fija del prototipo final.	84
Figura 92. Junturas universales del primer prototipo.	85
Figura 93. Junturas universales del prototipo final.	85
Figura 94. Base móvil del primer prototipo.	85
Figura 95. Base móvil del prototipo final.	85
Figura 96. Robot Delta, primer prototipo.	86
Figura 97. Robot Delta, prototipo final.	86
Figura 98. Servomotor Hitec Hs755 HB.....	87
Figura 99. Descripción gráfica de un PWM.....	87
Figura 100. Diseño del PCB, EAGLE.....	88
Figura 101. Diseño PCB vs Circuito implementado.	88
Figura 102. Circuito electrónico.	88
Figura 103. Señales PWM para posicionar el motor en 0° , -90° y 90°	89
Figura 104. Módulo XBee y sus conexiones del XBee	90
Figura 105. Tarjeta EXPLORER para el XBee.....	90
Figura 106. Comunicación entre el computador y el robot.....	90
Figura 107. GUI, Matlab.....	91
Figura 108. Ventana de la cinemática directa del robot Delta, Matlab.....	93
Figura 109. Ventana de la cinemática inversa del robot Delta, Matlab.....	93
Figura 110. Ventana del espacio de trabajo del robot Delta, Matlab.....	94
Figura 111. Ventana de ayuda sobre el simulador del robot Delta, Matlab.....	94
Figura 112. CAD del robot Delta importado de SolidWorks a Simulink.....	98

CAPÍTULO 1: INTRODUCCIÓN

INTRODUCCIÓN

Un manipulador paralelo (Figura 1) es un mecanismo compuesto de una plataforma móvil y una plataforma fija, conectadas por al menos dos cadenas cinemáticas, siendo una cadena cinemática la unión de dos o más eslabones (Kong and Gosselin, 2007). Generalmente, el número de grados de libertad del robot es igual al número de cadenas cinemáticas que lo conforman porque cada una de ellas es accionada por un actuador (Tsai, 1999).

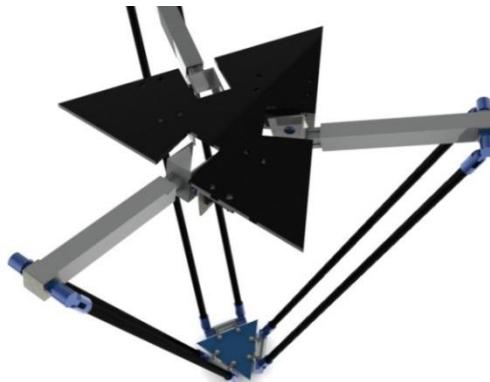


Figura 1. Robot Paralelo tipo Delta. CAD en SolidWorks.

Actualmente el programa de ingeniería en Mecatrónica de la Universidad Militar Nueva Granada no cuenta con un manipulador para desarrollar prácticas que permitan ampliar los conocimientos de los estudiantes sobre la cinemática de un robot paralelo, y de esta forma, estar a la vanguardia en los desarrollos industriales ya que hoy en día, los robots paralelos juegan un papel muy importante en aplicaciones de pick and place.

Por este motivo, el objetivo principal de este trabajo de grado, es diseñar e implementar un prototipo funcional de un robot Delta para el laboratorio de robótica del programa de ingeniería en Mecatrónica, debido a que los robots paralelos que se ofrecen en la industria son costosos y las posibilidades de desarrollar trabajos futuros con ellos son reducidas, debido a que no son plataformas abiertas.

Esta investigación se ha desarrollado por medio del método científico, el cual se compone de las siguientes etapas:

- Planteamiento del problema.
- Formulación de los objetivos.
- Metodología.
- Análisis de resultados y conclusiones.

Este documento está dividido en ocho capítulos. El primer capítulo es la introducción y contiene la motivación del proyecto, la problemática alrededor del tema de investigación, los objetivos: general y específicos, los beneficios del proyecto y la metodología empleada.

El capítulo 2 contiene el marco conceptual con el cual se busca contextualizar al lector sobre los diferentes temas relevantes dentro de la investigación.

El capítulo 3 contiene una breve descripción del estado del arte de los robots paralelos, sus aplicaciones actuales en el mundo y los trabajos realizados en Colombia.

En el capítulo 4 se describe la geometría del robot Delta y el desarrollo matemático para la solución de su cinemática directa, inversa y matriz Jacobiana. Finalmente se plantea un algoritmo para optimizar el espacio de trabajo del robot.

El capítulo 5 contiene la etapa de diseño e implementación del robot Delta y se encuentra dividido en tres secciones. La primera sección muestra el diseño mecánico de cada componente del robot, el diseño asistido por computadora (CAD), el análisis por el método de elementos finitos y la implementación del prototipo. La segunda sección muestra la selección de los actuadores y el diseño e implementación del circuito electrónico del robot. La tercera sección muestra la implementación de la interfaz gráfica de usuario que permite comunicar el robot con un computador para controlar sus movimientos.

El capítulo 6 contiene los resultados obtenidos con el desarrollo del proyecto.

En el capítulo 7 se presentan las conclusiones y trabajos futuros del proyecto.

El capítulo 8 contiene la bibliografía.

Finalmente se da un espacio para los anexos.

PLANTEAMIENTO DEL PROBLEMA

Un robot es una máquina con actuadores, sensores y un control, que puede llegar a realizar las tareas de los seres humanos (Martínez Angulo et al., 2005).

En sus inicios, los robots desempeñaban tareas sencillas y monótonas en ambientes de trabajo peligrosos para el hombre, reduciendo los costos y tiempos de manufactura de los productos y aumentando su calidad y productividad (Tsai, 1999). Actualmente, los robots son esenciales en los procesos industriales, siendo útiles en operaciones como (Carvajal, 2009): pick and place, paletización, despaletización e inserción.

Existen otros casos en donde el efecto final del robot deja de ser una pinza o garra para ser una herramienta. El tipo de herramienta depende de la operación que realiza el robot, siendo las más comunes (Groover, 1989): soldadura, pintura, operaciones de mecanizado, cortes, control de calidad de productos, entre otras.

Hasta hace unos años, los manipuladores en serie eran los encargados de realizar las tareas nombradas anteriormente. Un manipulador en serie se forma de

una sola cadena cinemática y su estructura mecánica se asemeja a la del brazo humano (Figura 2).



Figura 2. Robot manipulador en serie.

En estos robots, los actuadores se encuentran ubicados a lo largo de su estructura mecánica lo que genera un aumento en la inercia que el robot debe vencer para moverse, es decir, entre más actuadores tenga el robot, mayor es la carga en él, afectando la estabilidad de todo el mecanismo. Adicional a esto, cada actuador aumenta la probabilidad de generar un error de posición y orientación del efecto final (Zhang, 2009).

Por esta razón, surgieron los manipuladores paralelos, en donde dos o más cadenas cinemáticas soportan la plataforma móvil aumentando la rigidez, la estabilidad, la capacidad de carga, la precisión de la posición y orientación del efecto final, la velocidad y la aceleración de los movimientos del robot (Lou et al., 2004). Estas ventajas hacen a estos robots adecuados en aplicaciones como:

- Pick and place para operaciones de envasado, empaquetado o control de calidad en la industria de alimentos.
- Robots de micro-movimientos.
- Simuladores de movimiento.
- Interfaces hápticas.

Sin embargo, comparados con los manipuladores en serie, los manipuladores paralelos tienen tres desventajas: dificultad en el desarrollo de su cinemática, menor espacio de trabajo y aumento de las singularidades, lo que entorpece el desempeño del robot en su entorno (Lou et al., 2004).

A través de los años, se han propuesto diferentes soluciones al desarrollo de la cinemática de un robot Delta. En 1996, Tsai planteó que los eslabones que van unidos a los actuadores generan una circunferencia en el espacio del robot mientras que los eslabones que sostienen la plataforma móvil generan una esfera en el mismo espacio de trabajo, como se aprecia en la Figura 3 (Tsai et al., 1996). Los puntos de intersección de la circunferencia con la esfera son las posibles soluciones de la cinemática inversa del robot. En el mismo paper, se plantea que al generar tres esferas (Figura 4), cuyos centros son los puntos de contacto entre los dos eslabones de cada cadena cinemática, la intersección de las tres esferas

es la solución al problema de la cinemática directa del robot. Otra forma de solucionar la cinemática del robot es por medio de un análisis de su geometría (Hong and Yamamoto, 2009) o empleando un software como ADAMS (Jingjun et al., 2009).

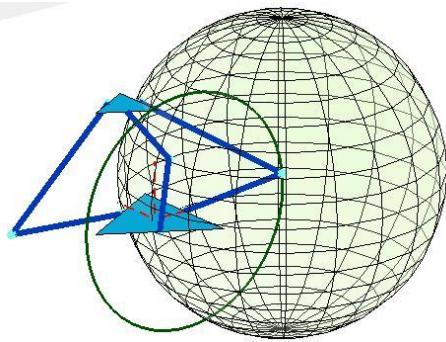


Figura 3. Solución de la cinemática inversa planteada por Tsai en 1966.

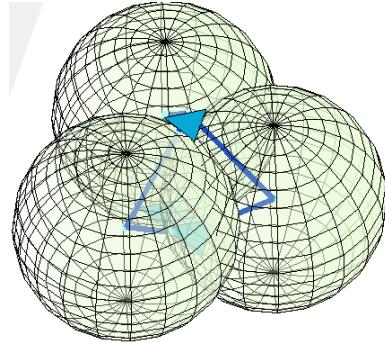


Figura 4. Solución cinemática inversa planteada por Tsai en 1966.

Para solucionar el problema del espacio de trabajo, se han desarrollado nuevas formas de robots paralelos, en donde se combinan articulaciones de revolución con prismáticas (Stan, Manic, et al., 2008) o las articulaciones prismáticas remplazan a las de revolución (Xin-Jun Liu and Jongwon Kim, 2002; Liu and Kim, 2005). Estas nuevas configuraciones logran un mayor espacio de trabajo pero se reduce la velocidad de movimiento del robot, debido a que los actuadores lineales no pueden alcanzar las velocidades que alcanzan los actuadores de revolución.

También se han implementado algoritmos para maximizar el volumen de trabajo de los robots paralelos usando: algoritmos genéticos (Liu et al., 2004; Laribi et al., 2007; Stan et al., 2007; Stan, Maties, et al., 2008; Martínez et al., 2010), el método SQP (Ceccarelli and Lanni, 2004), el método CRS (Yunjiang et al., 2005), entre otros.

Así ante lo expuesto, con el desarrollo de este proyecto se responderán las siguientes preguntas:

¿Es posible diseñar e implementar un prototipo de un robot Delta para el laboratorio de robótica de la universidad Militar sin superar una inversión no mayor a \$1'000.000?

¿Es posible emplear un algoritmo genético para optimizar las dimensiones de los eslabones del robot con el fin de maximizar su espacio de trabajo y reducir sus singularidades en dicho espacio de trabajo?

¿Es posible implementar las ecuaciones de la cinemática del robot en una interfaz gráfica de usuario para posicionar y orientar el efecto final del robot desde un computador?

OBJETIVOS

Objetivo General

Diseñar e implementar una plataforma robótica paralela tipo Delta, posibilitando el movimiento de su elemento terminal (efector final) hasta una posición y orientación deseada.

Objetivos Específicos

- Desarrollar un software basado en optimización multi-objetivo para el cálculo de las dimensiones de los eslabones del robot.
- Diseñar e implementar la arquitectura mecánica de la plataforma.
- Diseñar e implementar un circuito electrónico que permita el accionamiento de la plataforma.
- Desarrollar los modelos cinemático directo e inverso del robot.
- Desarrollar un software de comando de movimiento del robot, empleando la cinemática del mismo.

JUSTIFICACIÓN

La robótica es una de las principales aplicaciones de la ingeniería en Mecatrónica. Por eso, es importante que los estudiantes de Mecatrónica de la universidad Militar Nueva Granada estén a la vanguardia de los desarrollos de esta rama de la ingeniería.

Por cuestiones de tiempo, en la materia de robótica del programa, únicamente se trabaja con los manipuladores en serie y los robots móviles. En vista que en la actualidad, los robots paralelos y en especial los tipo Delta, están dominando la industria en aplicaciones de pick and place gracias a sus notorias ventajas frente a los manipuladores en serie, es importante que los estudiantes de ingeniería en Mecatrónica tengan conocimientos sobre esta clase de robots y puedan realizar prácticas para entender mejor su funcionamiento.

La finalidad de este trabajo de grado es diseñar e implementar un prototipo de un robot Delta, optimizando las dimensiones de sus eslabones con el fin de maximizar su espacio de trabajo y minimizar el número de singularidades del robot. Adicionalmente, se desarrollará la cinemática del robot por análisis geométrico y se implementará en una interfaz gráfica de usuario para que se puedan controlar los movimientos del robot desde un computador.

También se espera solucionar los tres problemas relacionados con los robots paralelos: espacio de trabajo reducido, cinemática difícil de desarrollar y aumento de las singularidades. Académicamente hablando, el proyecto espera que los estudiantes de ingeniería en Mecatrónica amplíen sus conocimientos acerca de manipuladores paralelos.

Este proyecto beneficiará a la Universidad Militar Nueva Granada y especialmente a los usuarios del laboratorio de robótica, quienes podrán realizar investigaciones, desarrollos y/o avances en áreas como:

- La industria: en proyectos que impliquen la tele-operación del robot Delta o su cooperación en células de manufactura. Por ejemplo, en aplicaciones de pick and place o control de calidad de productos que se desplazan en una banda transportadora.
- Realidad virtual: adaptar el robot como un dispositivo háptico y/o sensor de fuerza/torque para la interacción con ambientes virtuales.
- Medicina: robot de micro-movimientos.

METODOLOGÍA

A continuación se presenta la metodología propuesta para el desarrollo del trabajo de grado.

1 Definir el problema

- ✓ Definir el tema de investigación y delimitar el proyecto: ¿Qué aspectos se van a tratar?
- ✓ Definir el objetivo general y los objetivos específicos.
- ✓ Justificar la realización del proyecto: ¿Por qué es importante el estudio? ¿Qué información aportará? ¿A quiénes beneficia? ¿Qué utilidad tendrá? ¿Qué problemas podría resolver?
- ✓ Planteamiento de la metodología: designar las actividades que permitan el cumplimiento de los objetivos.

2 Recolección de información

- ✓ Revisar el estado del arte de los robots paralelos, recopilando información acerca de los robots Delta que se han construido en el mundo y en Colombia.
- ✓ Leer artículos relacionados con el diseño óptimo, modelado cinemático, circuitos de comando, control y simulación de robots paralelos tipo Delta.

3 Generar alternativas de solución y seleccionar la mejor alternativa

4 Diseño detallado

4.1 Diseño mecánico

- ✓ Obtener las dimensiones de los eslabones por medio de algoritmos genéticos.
- ✓ Diseñar los componentes del robot: plataforma fija, plataforma móvil, eslabones y juntas universales.
- ✓ Realizar el CAD del robot.

- ✓ Seleccionar el material de cada componente.
- ✓ Analizar el robot por el método de elementos finitos.
- ✓ Hacer los rediseños pertinentes.
- ✓ Desarrollar la cinemática directa e inversa del robot.

4.2 Diseño electrónico

- ✓ Seleccionar los actuadores del robot.
- ✓ Diseñar el circuito electrónico.

4.3 Diseño del software

- ✓ Implementar la cinemática directa e inversa del robot en una interfaz gráfica de usuario (GUI) para poder controlar los movimientos del robot desde un computador.

5 Manufactura y ensamble

- ✓ Construir las partes que conforman el robot y mecanizar las piezas que sean necesarias.
- ✓ Hacer la PCB con el circuito electrónico.
- ✓ Ensamblar el robot.

6 Pruebas finales y análisis de resultados

- ✓ Hacer pruebas de comunicación entre el robot y el computador.
- ✓ Hacer pruebas del movimiento del robot usando la cinemática directa e inversa.
- ✓ Hacer pruebas de precisión y exactitud del posicionamiento y la orientación de la plataforma móvil en el espacio de trabajo.

CAPÍTULO 2: MARCO CONCEPTUAL

2.1 ¿QUÉ ES UN ROBOT?

Existen muchas definiciones de lo que es un robot, pero una de las más concretas es la del RIA (*Robotics Industries Association*, antes *Robot Institute of America*), que define a un robot como: “reprogrammable multifunctional manipulator designed to move material, parts, tools, or specialized devices through variable programmed motions for the performance of variety of tasks”.

Por otro lado, la IFR (*International Federation of Robotics*) define al robot industrial como: “an automatically controlled, reprogrammable, multipurpose manipulator programmable in three or more axes, which may be either fixed in place or mobile for use in industrial automation applications”.

Como se aprecia en las dos definiciones anteriores, un robot es diseñado para ser controlado por un computador, o un dispositivo similar, con el fin de tener un dispositivo que pueda desarrollar múltiples tareas con tan solo cambiar su programación (Niku, 2010).

2.2 CLASIFICACIÓN DE LOS ROBOTS

2.2.1 Robots Manipuladores

Los robots manipuladores cuentan con una base fija, eslabones, articulaciones y un efecto final o herramienta de trabajo (Figura 5).

Los *eslabones* de un robot son cada uno de los cuerpos rígidos que lo componen y las *articulaciones* son los elementos que permiten un movimiento relativo entre dos eslabones (Jazar, 2010).

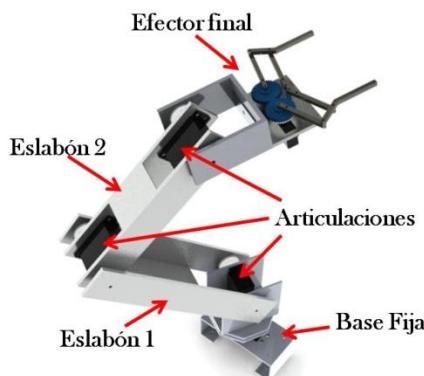


Figura 5. Componentes de un manipulador.

El movimiento de cada articulación puede ser de desplazamiento, de giro o una combinación de ambos, de este modo, son posibles seis tipos de

articulaciones (Spong et al., 2005): articulación de revolución, prismáticas, cilíndricas, helicoidales, esférica y planares. Las articulaciones más usadas en robótica son las de revolución (permite una rotación relativa entre dos eslabones, alrededor de un eje) y las prismáticas (permite un movimiento lineal relativo entre dos eslabones, a lo largo de un eje).

Cada uno de los movimientos independientes que puede realizar cada articulación con respecto a la anterior, se conoce como *grados de libertad* (DOF, por sus siglas en inglés Degrees of Freedom). El número de grados de libertad del robot es la suma de los grados de libertad de las articulaciones que lo componen. Para posicionar y orientar un cuerpo en el espacio se requieren seis DOF, tres para la posición y tres para la orientación (Barrientos, 2007).

Hunt planteó que los grados de libertad de un robot espacial se determinan por:

$$F = \lambda(n - j - 1) + \sum_{i=1}^j f_i \quad (2.1)$$

donde F son los DOF del mecanismo; λ los DOF del espacio en el cual el mecanismo funciona ($\lambda = 6$ para mecanismos espaciales y $\lambda = 3$ para mecanismos planares); n el número de cuerpos rígidos del mecanismo; j el número de articulaciones y f_i los DOF de la articulación i (Hunt, 1990).

Finalmente, el *efector final* o *herramienta de trabajo* (**Error! Reference source not found.**) es el elemento que permite que el robot realice el trabajo para el que fue diseñado: pintar, soldar, ensamblar, organizar, etc. (Jazar, 2010).

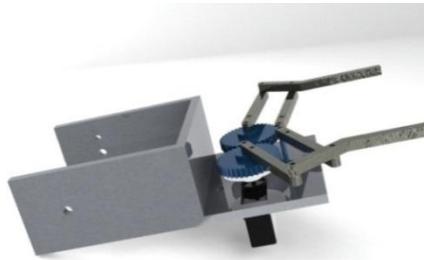


Figura 6. Efecto final. CAD SolidWorks.

2.2.2 Robots Móviles

Son robots que con una base que cuenta con un sistema de locomoción que les permiten un desplazamiento libre en su entorno (Figura 7), siendo las ruedas el sistema de locomoción más común en los robots (Siciliano et al., 2011).



Figura 7. Robot móvil.

Los sistemas de locomoción con ruedas más comunes son (Ollero Baturone, 2001; Siciliano et al., 2011): el diferencial (Figura 8.a), el síncrono (Figura 8.b), el triciclo (Figura 8.c), el ackerman (Figura 8.d), el skid steer (Figura 8.e), el omnidireccional y las pistas de deslizamiento.

Otras configuraciones de los robots móviles son los robots con patas, robots redundantes, robots submarinos y robots aéreos.

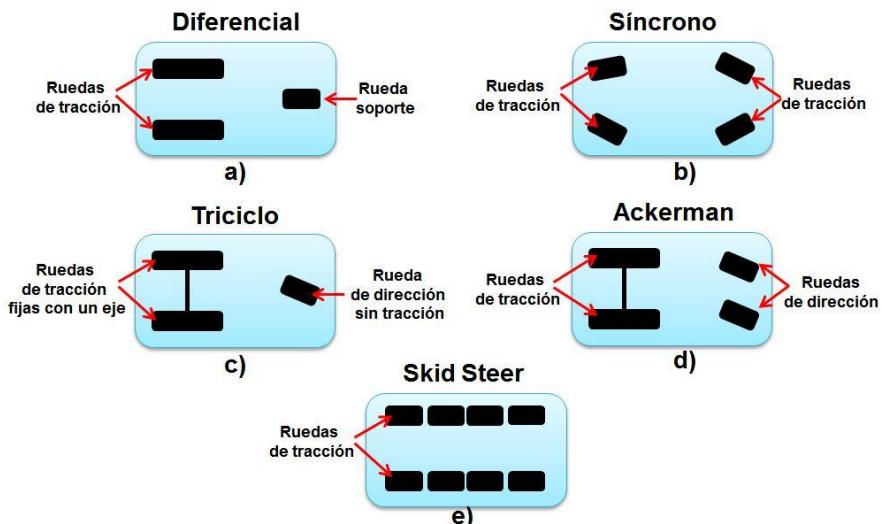


Figura 8. Configuraciones de un robot con ruedas.

2.3 ESTRUCTURA MECÁNICA DE UN ROBOT MANIPULADOR

Generalmente, un robot está compuesto por una estructura mecánica con sensores, actuadores y un sistema de control (Sciavicco and Siciliano, 2000).

La estructura de un robot se asemeja al esqueleto de un ser vivo, ya que es la responsable de soportar todo sus componentes.

Los sensores son los sentidos del robot ya que le permiten desenvolverse adecuadamente en su entorno (Martínez Angulo et al., 2005). El robot necesita los sensores para recibir información de su espacio de trabajo y para monitorear sus propios componentes (Bekey, 2005).

El sistema de control procesa la información de los sensores y envía señales a los actuadores para generar el movimiento de los elementos del robot (Niku, 2010). Los actuadores utilizados en robótica pueden emplear energía neumática, hidráulica o eléctrica (Barrientos, 2007):

- Los actuadores neumáticos utilizan aire comprimido como fuente de energía y pueden ser: cilindros neumáticos (simples o de doble efecto) o motores neumáticos (de aletas rotativas o de pistones axiales). Su desventaja es la falta de precisión en el posicionamiento.
- Actuadores hidráulicos: su fuente de energía son aceites minerales a una determinada presión. Son más precisos que los actuadores neumáticos. Son útiles en aplicaciones de fuerzas y pares elevados.
- Actuadores eléctricos: son los más usados en los robots industriales debido a sus características de control, sencillez y precisión. Su gran desventaja es la potencia limitada. Se pueden clasificar en: motores de corriente continua (controlados por inducido o por excitación) y motores de corriente alterna (síncronos o asíncronos).

Un sistema de control de un robot puede realizar funciones básicas o inteligentes (Bonasso et al., 1998):

- Funciones básicas: permiten controlar las articulaciones del robot y generar trayectorias.
- Funciones inteligentes: son las que permiten que el robot cuente con una autonomía parcial o total en operaciones como planificación de tareas o de movimientos, percepción sensorial, y la reacción ante la presencia de obstáculos y condiciones no previstas dentro de su espacio de trabajo.

2.4 CARACTERÍSTICAS DE UN ROBOT MANIPULADOR

La geometría de la estructura del robot, sus componentes, el análisis de su cinemática, los actuadores y el sistema de control determinan las características de desempeño del robot (Lewis et al., 2003).

2.4.1 Espacio de trabajo

Es la porción del entorno al que puede acceder el efecto final del manipulador, es decir, es el volumen total barrido por el efecto final cuando el manipulador ejecuta todos los movimientos posibles (Sciavicco and Siciliano, 2000). El espacio de trabajo está limitado por el tamaño, forma y tipo de los eslabones del robot y las limitaciones de movimiento de las articulaciones y el sistema de control (Spong et al., 2005).

2.4.2 Capacidad de carga

Característica condicionada por el tamaño, configuración y los actuadores del robot. Es una de las mayores influencias en la respuesta dinámica del robot

(Ollero Baturone, 2001), por eso, se debe considerar el peso de las piezas a manipular y el propio peso del robot (Barrientos, 2007).

2.4.3 Velocidad

La velocidad de movimiento de un robot puede calcularse por la velocidad de cada una de sus articulaciones o por la velocidad media de su efecto final. Cuando los movimientos del robot son rápidos y cortos los fabricantes no indican la velocidad sino el tiempo empleado en realizar un movimiento típico (Barrientos, 2007). La velocidad a la que puede moverse un robot es inversamente proporcional a la carga que transporta.

2.4.4 Precisión, repetibilidad y resolución

- Resolución: mínimo incremento que puede aceptar la unidad de control del robot.
- Precisión: distancia entre el punto programado y el valor medio de los puntos realmente alcanzados al repetir el movimiento varias veces con condiciones de carga y temperatura nominales.
- Repetibilidad: capacidad de regresar a una misma posición en repetidas ocasiones bajo las mismas condiciones de temperatura, carga, etc. (Ollero B, 2007).

2.5 CLASIFICACIÓN DE LOS ROBOTS MANIPULADORES

2.5.1 Manipuladores en Serie

Son aquellos que están formados por una cadena cinemática abierta, es decir, solo existe una posibilidad para unir el efecto final con la base fija, como se aprecia en la Figura 9 (Zhang, 2009).



Figura 9. Manipulador en serie. CAD SolidWorks.

Estos robots, generalmente, se clasifican es según su espacio de trabajo o su sistema de coordenadas (Jazar, 2010):

- Configuración cartesiana: su espacio de trabajo es un cubo y son robots apropiados para tareas de cargas pesadas.

- Configuración cilíndrica: configuración en la que la primera articulación es de revolución y proporciona una rotación sobre la base, mientras que las otras dos articulaciones son prismáticas, lo que genera que su espacio de trabajo sea un cilindro.
- Configuración esférica: cuenta con dos articulaciones de revolución que son ortogonales y una articulación prismática. Su nombre se debe a que la posición del efecto final con respecto a la base está dada en coordenadas esféricas.
- Configuración antropomórfica: es una configuración apropiada para robots industriales y cuenta únicamente con articulaciones de revolución. El manipulador de la Figura 9 es un robot antropomórfico.
- Configuración SCARA: robots que cuenta con dos articulaciones de revolución y una prismática, se diferencian de los robots de configuración esféricos por la posición de sus articulaciones.

2.5.2 Manipulador en Paralelo

Un manipulador paralelo (Figura 10) está formado por cadenas cinemáticas cerradas, es decir, existe más de una posibilidad para unir la plataforma móvil y la plataforma fija (Zhang, 2009). Los robots paralelos han despertado un gran interés en aplicaciones industriales, en el entretenimiento o la medicina, gracias a sus ventajas frente a los robots en serie: su rigidez, precisión en el posicionamiento y altas velocidades (Ottaviano and Ceccarelli, 2000).



Figura 10. Manipulador paralelo¹.

Generalmente, estos robots presentan las siguientes características (Merlet, 2010):

- Al menos dos cadenas cinemáticas conectan la plataforma móvil con la fija.
- Cada cadena cinemática cuenta con al menos un actuador.
- El número de actuadores es el mismo número de grados de libertad la plataforma móvil.

¹ Inmotion, 6 DOF Hydraulic Square Frame, [Online] <http://www.inmotionsimulation.com>

- Cuando los actuadores están inmóviles, la plataforma móvil permanece en su posición.

- La carga manipulada por el robot se distribuye en las cadenas cinemáticas.

Por otro lado, los manipuladores paralelos se pueden clasificar según las características de sus movimientos en (Merlet, 2010):

- Robots planares (Figura 11): son robots que cuentan con tres grados de libertad, dos de posición y uno de orientación. Sus articulaciones pueden ser de revolución, prismáticas o una combinación de las dos.

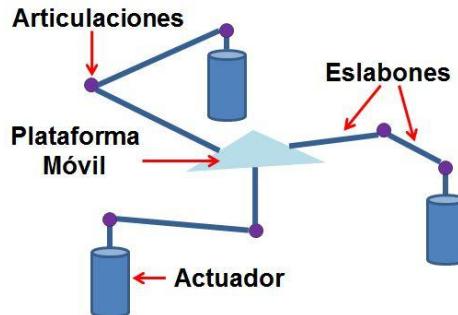


Figura 11. Robot paralelo planar.

- Robots espaciales: se pueden clasificar según sus grados de libertad en:

- ✓ Robots de tres DOF: el robot más popular es el propuesto por Clavel y se conoce como robot Delta (Figura 12). Sus tres grados de libertad le permiten posicionar su plataforma móvil en el espacio.

Generalmente, estos robots cuentan con articulaciones esféricas o universales, aunque también es común encontrar robots paralelos con articulaciones prismáticas o una combinación de prismáticas y de revolución. Cabe aclarar que en la industria existen robots Delta con 4 DOF para posicionar y orientar su plataforma móvil en el espacio.



Figura 12. Robot Delta².

- ✓ Robots de más de tres grados de libertad: son robots que cuentan con más de tres cadenas cinemáticas. Existen muchos tipos de

² Adept, Quattro S650HS, [Online] <http://www.adept.com/products/robots>

configuraciones, las cuales varían dependiendo al tipo de articulación que utilicen. Su aplicación más significativa es la de simuladores de movimiento.

2.6 CINEMÁTICA DE UN ROBOT MANIPULADOR

La cinemática de un robot es el estudio de su movimiento sin considerar las fuerzas que lo producen (Ollero Baturone, 2001).

2.6.1 Cinemática Directa

El problema de la cinemática directa consiste en determinar la posición y orientación del efecto final del robot desde un sistema de coordenadas de referencia (Carvajal, 2009).

Dado que un robot se puede considerar como una cadena cinemática formada por eslabones, se puede establecer un sistema de referencia fijo situado en la base del robot y describir la localización de cada uno de los eslabones con respecto a dicho sistema de referencia (Barrientos, 2007).

- **Cinemática directa de los manipuladores en serie:** una convención comúnmente usada para seleccionar marcos de referencia es la de Denavit-Hartenberg, en donde cada transformación homogénea A_i^j es representada como un producto de cuatro transformaciones básicas:

$$A_i^j = \text{Rot}_Z(\theta_i) \text{Trans}_Z(d_i) \text{Trans}_X(a_i) \text{Rot}_X(\alpha_i) \quad (2.2)$$

$$A_i^j = \begin{bmatrix} \text{Cos}(\theta_i) & -\text{Sen}(\theta_i) & \text{Sen}(\theta_i)\text{Sen}(\alpha_i) & \alpha_i\text{Cos}(\theta_i) \\ \text{Sen}(\theta_i) & \text{Cos}(\theta_i) & -\text{Cos}(\theta_i)\text{Sen}(\alpha_i) & \alpha_i\text{Sen}(\theta_i) \\ 0 & \text{Sen}(\alpha_i) & \text{Cos}(\alpha_i) & d_i \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (2.3)$$

donde θ_i , d_i , a_i y α_i son parámetros asociados con el eslabón i y el eslabón j del robot (Siciliano et al., 2011).

- **Cinemática directa de manipuladores en paralelo:** se pueden buscar tres ecuaciones que relacionen la posición de la plataforma móvil con los ángulos de sus eslabones y solucionar el problema algebraicamente (Hong and Yamamoto, 2009) o por medio de métodos numéricos iterativos (Merlet, 1993; Pierrot et al., 2009).

2.6.2 Cinemática Inversa

La cinemática inversa consiste en encontrar las posibles configuraciones del robot conociendo la posición y orientación de su efecto final (Carvajal, 2009). Su solución es importante para el planteamiento de las trayectorias y su complejidad se basa en la no linealidad de las ecuaciones a desarrollar y sus múltiples soluciones.

El procedimiento para encontrar la cinemática inversa depende de la configuración del robot y se puede resolver geométricamente, tanto para manipuladores en serie como en paralelo.

2.7 ALGORITMOS GENÉTICOS

Antes de hablar de algoritmos genéticos (GA, por sus siglas en inglés Genetic Algorithms) es necesario entender el concepto de optimización. La optimización es un proceso en donde se establece un modelo inicial y sus parámetros se redefinen progresivamente para determinar el máximo o el mínimo valor de alguna función (Hopgood, 2011). Por simplicidad, se llamará *función objetivo* a aquella función que es objeto de optimización.

Los GAs fueron desarrollados por John Holland y son procedimientos inspirados en la evolución de las especies para resolver problemas de optimización, en donde una población “evoluciona” hacia una población óptima.

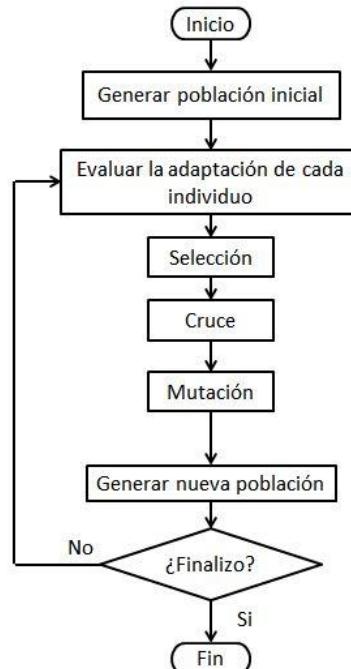


Figura 13. GA básico.

La figura 13 muestra el diagrama de flujo de un algoritmo genético simple. El proceso comienza con una población inicial, generada aleatoriamente. Cada individuo de la población se conoce como cromosoma y representa una solución al problema. Los cromosomas evolucionan a través de iteraciones sucesivas llamadas generaciones. Durante cada generación los cromosomas son evaluados, asignándole a cada uno un valor de adaptación. Los mejores cromosomas son seleccionados para reproducirse, surgiendo una nueva generación, descendencia. Este proceso se realiza por medio de los operadores de cruce y mutación. Después de varias generaciones, el algoritmo converge al mejor cromosoma,

quien representará una solución óptima del problema (Affenzeller et al., 2009; Gen et al., 2010; Maulik et al., 2011).

2.7.1 Operadores Genéticos

Son procedimientos que permiten generar nuevas poblaciones. Primero se seleccionan los padres para luego generar las descendencias por medio del cruce y la mutación.

Selección:

En un GA la función de adaptación asigna una calificación a cada individuo de la población, la cual indica la calidad de la solución representada por el individuo. La selección determina cuales son los individuos que se reproducirán (Affenzeller et al., 2009). Los métodos de selección más comunes son:

- Ruleta: es el método más conocido, propuesto por Holland (Holland, 1992), y la idea es determinar la probabilidad de selección de cada cromosoma (Gen and Cheng, 1999). La probabilidad se obtiene dividiendo la adaptabilidad del cromosoma sobre la suma de la adaptabilidad de todos los cromosomas de la población:

$$P_i = \frac{f_i}{\sum f_i} \quad (2.4)$$

De esta forma cada cromosoma es representado por un espacio proporcional de su adaptabilidad (Affenzeller et al., 2009). La selección se hace generando un número aleatorio entre 0-1.

- Selección universal estocástica: método similar al de la ruleta, con la diferencia que todos los padres están separados a una misma distancia: $\frac{1}{N_{padres}}$. Primero se debe seleccionar un parente cuya posición en la ruleta sea un número aleatorio entre el intervalo $[0, \frac{1}{N_{padres}}]$, el segundo parente está ubicado a una distancia $\frac{1}{N_{padres}}$ del primero, este proceso se realiza hasta seleccionar N_{padres} . Los individuos son seleccionados con la probabilidad expresada en (2.4) (RuiJun et al., 2007).
- Truncamiento: Se clasifican todos los cromosomas de acuerdo a su adaptabilidad y se escoge un umbral T de tal forma que el $T\%$ de los mejores cromosomas son seleccionados como padres. Generalmente ese porcentaje está entre el 10% y el 50% (Thierens and Goldberg, 1994).

Cruce:

Es la operación para producir la descendencia. Si los padres codifican su información en números reales, es común utilizar los métodos de cruce discreto, intermedio o de línea (Mühlenbein and Schlierkamp-Voosen, 1993).

Sean $x = (x_1, x_2, \dots, x_n)$ y $y = (y_1, y_2, \dots, y_n)$ los padres, las descendencias son:

- Cruce discreto: la descendencia es calculada por

$$z_i = \{x_i\} \text{ o } \{y_i\} \quad (2.5)$$

en donde x_i y y_i tienen la misma probabilidad de ser escogidos.

- Cruce intermedio: la descendencia es calculada por

$$z_i = x_i + \alpha_i(y_i - x_i) \quad (2.6)$$

en donde α_i se escoge aleatoriamente entre el rango $[-0.25, 1.25]$.

- Cruce de línea: la descendencia es calculada por

$$z_i = x_i + \alpha(y_i - x_i) \quad (2.7)$$

en donde α se escoge aleatoriamente entre el rango $[-0.25, 1.25]$.

Mutación:

Los cromosomas de la población se alteran aleatoriamente, con pequeñas variaciones. Generalmente, las descendencias son las que se mutan (Hopgood, 2011). Se selecciona una variable x_i con una probabilidad $p_m = \frac{1}{n}$, siendo n el número de variables que conforman el cromosoma. El nuevo valor z_i es calculado por

$$z_i = x_i \pm \text{rango} * \delta \quad (2.8)$$

donde **rango** es un número aleatorio en el rango $[-0.1, 0.1]$ y δ es una pequeña desviación y se calcula:

$$\delta = \sum_{i=0}^{15} \alpha_i 2^{-i} \quad \text{donde } \alpha_i \in [0, 1] \quad (2.9)$$

El signo + o - de (2.10) es escogido con una probabilidad de 0.5 (Mühlenbein and Schlierkamp-Voosen, 1993).

2.8 PROBLEMAS DE OPTIMIZACIÓN MULTI-OBJETIVO (MOP)

En un problema de optimización simple se busca optimizar una única función y se obtiene una única solución, pero la mayoría de los problemas de optimización del mundo real son problemas en donde se necesita optimizar más de una función a la vez. Estos problemas se conocen como MOP, por sus siglas en inglés *Multi-objective Optimization Problems*. Para los MOPs no existe una única solución que sea óptima con respecto a todos los objetivos, existe un conjunto de posibles soluciones, conocidas como soluciones óptimas de Pareto. Entonces, la meta de un MOP es encontrar la mayor cantidad de soluciones (Haupt and Haupt, 2004).

Antes de profundizar en los MOPs es importante tener en cuenta algunos conceptos (Coello et al., 2007):

- **Variable de decisión:** es una cantidad numérica que desea ser encontrada en un problema de optimización. Un vector x de n variables de decisión se representa como $x = [x_1, x_2, \dots, x_n]^T$.
- **Restricciones:** son aquellas que permiten delimitar un problema de optimización y conocer si una variable de decisión es aceptable o no. Pueden ser expresadas en forma de igualdades o desigualdades matemáticas y se debe tener en cuenta que el número de restricciones de igualdad debe ser menor al número de variables de decisión para que el problema no se considere sobre restringido.
- **Función objetivo:** es un criterio que se expresa en función de las variables de decisión y permite evaluar si una solución es buena o no. En un MOP se busca maximizar, minimizar o maximizar y minimizar las funciones objetivo. Una $f(x)$ de k funciones objetivo se representa como $f(x) = [f_1(x), f_2(x), \dots, f_k(x)]^T$.

El objetivo de un MOP es encontrar un vector x de n variables de decisión que cumplan con las restricciones y optimicen k funciones objetivo. Generalmente, estas funciones objetivo se encuentran en conflicto entre sí, por lo tanto no se obtiene una única solución sino un conjunto de posibles soluciones. Un MOP se puede representar matemáticamente como (Gen et al., 2010):

$$\begin{aligned} & \max\{z_1 = f_1(x), z_2 = f_2(x), \dots, z_k = f_k(x)\} \\ & \text{sujeto a: } g_i(x) \leq 0 \quad i = 1, 2, \dots, m \\ & \quad h(x) > 0 \end{aligned} \tag{2.10}$$

siendo x el vector de n variables de decisión. Todos los vectores de decisión que satisfacen las restricciones forman la región factible S en el espacio de decisión y se representa:

$$S = \{x \in R^n | g_i(x) \leq 0, i = 1, 2, \dots, m, h(x) > 0\} \tag{2.11}$$

Adicionalmente, Z es el conjunto de imágenes de todos los elementos de S y es usado para representar la región factible en el espacio de criterio (Figura 14):

$$Z = \{z \in R^q | z_1 = f_1(x), z_2 = f_2(x), \dots, z_k = f_k(x), x \in S\} \tag{2.12}$$

Un vector x es una solución eficiente en S si su imagen en Z es una solución no dominada (Gen and Cheng, 1999). Para dos vectores de decisión x y y se dice que x domina a y si:

- $f_i(x) < f_i(y)$, si se busca maximizar la función objetivo f_i .
- $f_i(x) > f_i(y)$, si se busca minimizar la función f_i .

A través del tiempo se han implementado muchos algoritmos que permiten resolver problemas de múltiples objetivos. En este trabajo solo se nombrarán algunos algoritmos basados en algoritmos genéticos.

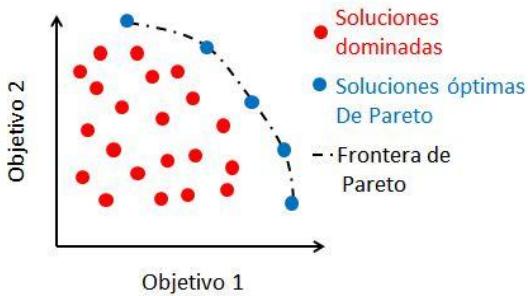


Figura 14. Espacio de Criterio de un MOP con dos funciones objetivo.

2.8.1 Niched Pareto Genetic Algorithm (NPGA)

Algoritmo basado en un torneo de dominancia de Pareto. Se seleccionan dos individuos al azar y se selecciona un conjunto de contrincantes, el cual es comparado con los dos individuos seleccionados. Si uno de los dos individuos domina a sus contrincantes y el segundo individuo es dominado por al menos un contrincante, el primero gana el torneo y se selecciona para reproducción (Horn et al., 1994). Por el contrario, si los dos individuos son dominados o son dominantes en necesario recurrir a la función de *sharing* explicada en la ecuación (2.17).

2.8.2 Non-Dominated Sorting Genetic Algorithm (NSGA)

Los individuos de cada generación se agrupan por capas. En la primera capa se encuentran los individuos no dominados de toda la población actual, asignándole a cada uno el mismo valor de adaptabilidad. Para la siguiente capa, se eliminan los individuos de la primera capa y se encuentran los individuos no dominados dentro de la población restante, a estos individuos se les asigna un valor de adaptación menor que a los de la primera capa. Este proceso se realiza hasta asignarle un valor de adaptabilidad a cada individuo de la población (Srinivas and Deb, 1994).

En (Mohanty, 2006) para calcular la adaptabilidad de cada individuo, primero se asigna un valor ficticio f_i , luego se calcula la distancia euclidiana entre el individuo i y el resto de los individuos j de la capa

$$d_{ij} = \sqrt{\sum_{p=1}^P (x_p^{(i)} - x_p^{(j)})^2} \quad (2.14)$$

donde x es la variable de decisión, P el número de variables de decisión y **min** y **max** hacen referencia a los límites mínimo y máximo de las variables de decisión. Entonces, el valor de adaptabilidad del individuo es:

$$f'_i = \frac{f_i}{m_i} \quad (2.15)$$

dónde:

$$m_i = \sum_{j=1}^{n_k} Sh(d_{ij}) \quad (2.16)$$

Siendo $Sh(d_{ij}) = \begin{cases} 1 - \left(\frac{d_{ij}}{\sigma_{share}}\right) & \text{si } d_{ij} < \sigma_{share} \\ 0 & \text{otro caso} \end{cases}$

donde n_k es el número de individuos en la k-ésima capa y σ_{share} es la máxima distancia permitida entre dos individuos en el espacio de solución (Srinivas and Deb, 1994). El valor de σ_{share} puede ser:

$$\sigma_{share} = \frac{r}{\sqrt[p]{q}} = \frac{\sqrt{\sum_{p=1}^P (x_p^{(max)} - x_p^{(min)})^2}}{\sqrt[p]{q}} \quad (2.17)$$

donde r es el volumen de una P-dimensional esfera de radio σ_{share} y q es el número de soluciones óptimas de Pareto que el algoritmo espera encontrar (Coello et al., 2007).

2.8.3 Random-Weight Genetic Algorithm (RWGA)

En (Murata and Ishibuchi, 1995) se plantea un algoritmo genético cuyo objeto es determinar todas las soluciones no dominadas en un problema de optimización con más de un objetivo. Para evaluar cada individuo de la población, se le asigna una función adaptación que corresponde a la suma de todas las funciones objetivo, cada una de ellas multiplicada por un peso:

$$f(x) = \omega_1 f_1(x) + \omega_2 f_2(x) + \dots + \omega_n f_n(x) \quad (2.18)$$

donde ω_n es el peso de la n-ésima función objetivo, satisfaciendo las condiciones:

$$\omega_i \geq 0 \quad y \quad \omega_i = \frac{random_i}{\sum_{j=1}^n random_j} \quad \text{para } i = 1, 2, \dots, n \quad (2.19)$$

siendo $random_i$ y $random_j$ son números aleatorios positivos. La idea del algoritmo es que los pesos de las funciones objetivo no sean fijos para poder buscar todas las posibles soluciones no dominadas del problema.

En (Murata et al., 1996) plantea otra forma de calcular los pesos y da como ejemplo un problema de dos funciones objetivo:

$$\omega_1 = \frac{k-1}{N_{padres}-1} \quad y \quad \omega_2 = 1 - \omega_1 \quad (2.20)$$

donde N_{padres} es el número de padres seleccionados en cada generación y k es un número aleatorio entre 1 y N_{padres} .

Los pesos también pueden ir ajustándose en base a los valores máximos y mínimos de cada función objetivo en cada generación (Gen and Cheng, 1999):

$$\omega_1 = \frac{1}{f_1(x)_{max} - f_1(x)_{min}} \quad y \quad \omega_2 = \frac{1}{f_2(x)_{max} - f_2(x)_{min}} \quad (2.21)$$

donde $f_n(x)_{max}$ y $f_n(x)_{min}$ corresponden al máximo valor y al mínimo valor de la función objetivo n respectivamente.

Los métodos anteriores se desarrollan de forma similar, teniendo en cuenta que la base de todos son los algoritmos genéticos; la diferencia radica en la forma como se asigna el valor de adaptación a cada individuo.

La Figura 15 muestra el diagrama de bloques de un algoritmo genético multiobjetivo. Primero se genera una población inicial de forma aleatoria. Luego, se evalúa cada individuo de la población en las funciones objetivo, seleccionando algunos individuos que formarán un conjunto tentativo de soluciones óptimas de Pareto (se seleccionan los mejores individuos de cada función objetivo). Se encuentra la función de adaptabilidad de cada individuo. Se escogen los padres por medio de la selección y se generan descendencias por medio de cruce y mutación. Se seleccionan aleatoriamente algunos individuos del conjunto tentativo de Pareto para que forme parte de la nueva población. Este proceso se realiza hasta obtener un número determinado de generaciones o hasta que se cumpla alguna condición.

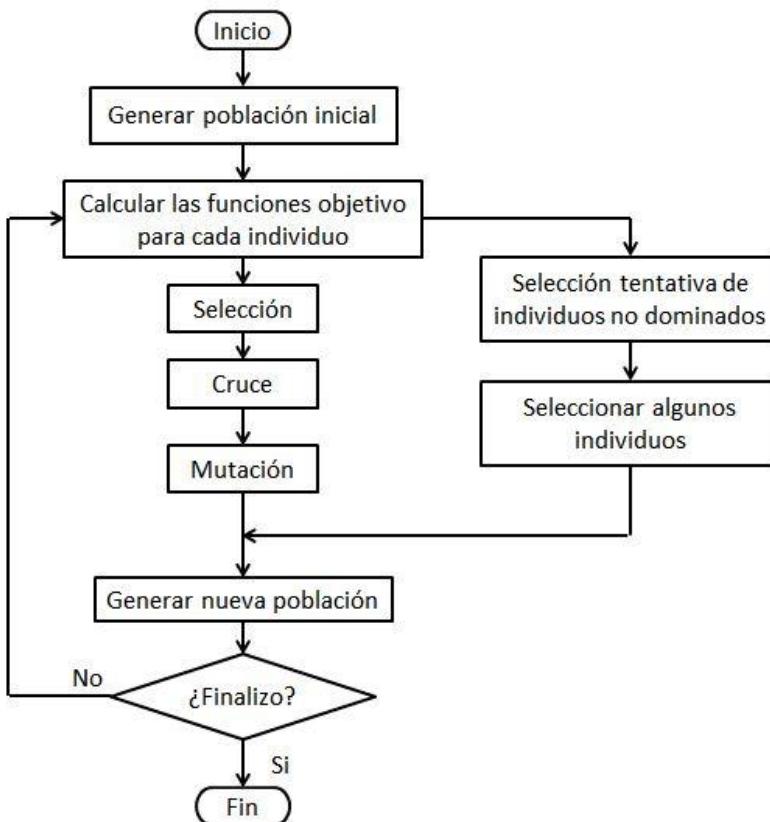


Figura 15. Diagrama de bloques de un GA multi-objetivo.

2.9 MÉTODO DE ELEMENTOS FINITOS

Con el fin de analizar sistemas en ingeniería, se han desarrollado modelos matemáticos, basados en las ecuaciones diferenciales, para describir dichos sistemas. La solución exacta de las ecuaciones representa el comportamiento detallado del sistema bajo unas condiciones iniciales (Moaveni, 2007). Por esta razón se desarrollaron métodos numéricos para solucionar los modelos matemáticos de los sistemas.

El método de elementos finitos (FEM, por sus siglas en inglés Finite Element Method) es un método numérico para obtener soluciones aproximadas de ecuaciones diferenciales ordinarias y parciales.

En el mundo, millones de ingenieros y científicos usan el FEM para predecir el comportamiento de sistemas mecánicos, térmicos, eléctricos y químicos. El método les permite hacer un análisis de diseño y de desempeño del sistema (Fish and Belytschko, 2007).

El FEM requiere la división del dominio del problema en muchos subdominios. Cada subdominio se conoce como elemento finito. Los elementos finitos se conectan entre sí por medio de nodos; el conjunto de todos los elementos finitos forma el dominio del problema (Kwon and Bang, 2000).

La aplicación del FEM en un problema determinado involucra los siguientes pasos (Bhatti, 2005):

1. Desarrollo de ecuaciones de cada elemento.
2. Discretizar el dominio de soluciones en una malla de elementos finitos.
3. Ensamble de las ecuaciones en la matriz de rigidez.
4. Aplicar condiciones de frontera, condiciones iniciales y cargas.
5. Solución de nodos desconocidos.
6. Cálculo de la solución y las cantidades relacionadas sobre cada elemento.
7. Obtener otra información de interés.

Los pasos del 1 al 5 corresponden a la fase de pre procesamiento, el paso 6 es la fase de solución y el paso 7 es la fase de post procesamiento (Moaveni, 2007).

Con el FEM se pueden resolver problemas unidimensionales, bidimensionales y tridimensionales. Existen programas como ANSYS, SolidWorks o Catia que se especializan en hacer el análisis de elementos finitos a estructuras en 3D.

CAPÍTULO 3: ESTADO DEL ARTE

3.1 HISTORIA DE LOS ROBOTS PARALELOS

En 1928, James E. Gwinnet diseñó una plataforma rotatoria con inclinaciones en cualquier dirección. El invento era destinado a la industria del entretenimiento y debía ser usado con una imagen en movimiento (Figura 16); las inclinaciones en la plataforma permitían hacer énfasis en la acción que aparecía en la imagen, por ejemplo las inclinaciones laterales en un avión (Gwinnett, 1931). Este invento se consideró como el primer mecanismo paralelo.

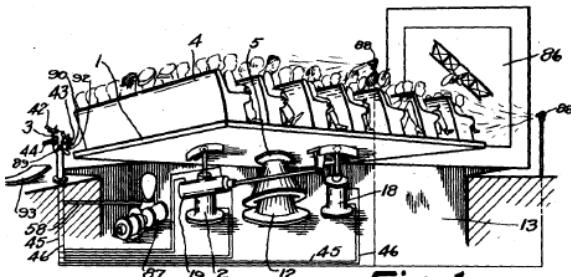
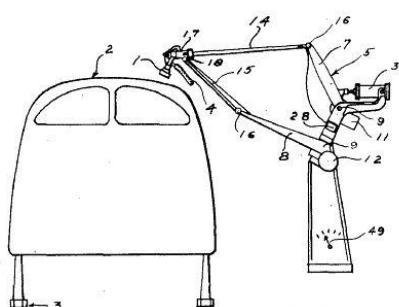


Figura 16. Primer mecanismo paralelo (US Patent No. 1789680).

En 1942, Willard LV Pollard Jr. patentó un mecanismo para controlar el movimiento y la posición de una pistola de spray con respecto a una superficie curva o irregular, gracias a sus 5 grados de libertad (Pollard, 1942). Como se aprecia en la Figura 17, una de las aplicaciones del mecanismo era pintar la carrocería de un automóvil.



En 1953, John Serafin patentó un simulador para el entrenamiento de vuelo en el cual la posición del simulador era controlado por un dispositivo mecánico (Serafin, 1953).

En 1962, el ingeniero Klaus Cappel propuso un robot similar al de Gough, el cual patentó en 1967 para ser usado como un simulador de movimiento que producía los movimientos de proa, popa, lateral, de elevación, roll, pitch y yaw, o una combinación de cualquiera de ellos (Cappel, 1967).

En 1965, Stewart publicó un artículo en el que describía un mecanismo controlado por seis motores. La plataforma de seis grados de libertad era capaz de moverse en tres direcciones lineales y tres angulares, individualmente o en cualquier combinación (Stewart, 1965). El mecanismo contaba con aplicaciones como:

- Vehículo para representar un cuerpo en el espacio sometido a todas las fuerzas que puede tener en el viaje.
- Simulador de los movimientos aleatorios a los que se somete un barco.
- Base del diseño de una nueva máquina.
- Simulador de vuelo de un helicóptero. Siendo esta aplicación la más relevante.

En 1978, Hunt sugirió que los robots paralelos, cuya finalidad era servir como mecanismos simuladores de vuelo, se usarían como robots manipuladores y destacó que dichos robots debían ser estudiados más a fondo en el contexto de aplicaciones robóticas por sus ventajas frente a los robots en serie (Aracil et al., 2006).

En 1979, McCallion y Truong propusieron usar la plataforma Stewart como un manipulador paralelo dentro de una célula de ensamblaje robotizada debido a que los sensores en los eslabones, que permiten posicionar el efecto final, presentan menos errores que en un manipulador en serie (Aracil et al., 2006).

En 1983, Hunt se basó en la teoría Screw para estudiar la estructura cinemática de los brazos de los robots paralelos (Hunt, 1983).

En 1984, Yang y Lee desarrollaron un diseño para la plataforma de Stewart, en donde trataron de analizar el espacio de trabajo de la plataforma, mostrando que el espacio de trabajo y la maniobrabilidad de un manipulador paralelo es limitada, en comparación con el manipulador en serie.

En 1989 Reymond Clavel patentó un dispositivo para mover y posicionar un elemento en el espacio. El dispositivo podía ser considerado como una nueva clase de robot industrial con cuatro grados de libertad, tres para la posición y uno para la orientación. Los actuadores se ubicaban en la base fija y transmitían el movimiento hasta una base móvil, a través de los eslabones, por medio de juntas universales (Figura 18). El movimiento de la base móvil era paralelo a la base fija. El robot permitía transportar piezas ligeras de un lugar a otro a altas velocidades. Clavel fue el primero en hablar del robot delta y en plantear su campo de trabajo en la industria (Clavel, 1990).

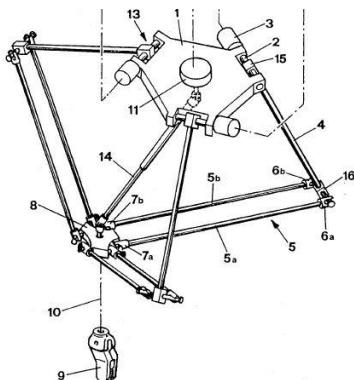


Figura 18. Robot Delta inventado de Clavel (US Patent No. 4976582).

En 1993, Merlet avanzó en la investigación de los robots paralelos al plantear dos métodos para resolver la cinemática directa de la plataforma de Stewart, desarrollar el Jacobiano y determinar el espacio de trabajo del robot (Merlet, 1993); en 1996 presentó un método para diseñar la plataforma paralela de Gough, en pro de obtener un mejor desempeño en su espacio de trabajo (Merlet, 1996) y en el 2003 comenzó a hablar de robots paralelos modulares, los cuales son mecanismos que pueden adaptar su geometría de acuerdo a la tarea que realizan (Merlet, 2003).

3.2 ¿POR QUÉ LOS ROBOTS PARALELOS?

Dado que un robot manipulador en serie mueve su extremo con aceleraciones elevadas, es de gran importancia reducir al máximo su momento de inercia. Del mismo modo, los pares estáticos que deben vencer los actuadores dependen directamente de la distancia de las masas del actuador. Es por ello que se procura que los actuadores, por lo general pesados, estén lo más cerca posible a la base del robot (Barrientos, 2007). Por esta razón, en aplicaciones como envasado, empaquetado o control de calidad de productos dentro de células manufactureras, los manipuladores en serie son ineficientes. Por esta razón, las industrias solucionaron el problema integrando los robots Delta en sus procesos. Estos robots tienen la ventaja que la relación de su masa frente a una carga a soportar es mucho menor, por lo que admiten mayores aceleraciones durante su movimiento, por lo tanto, mayores velocidades. A diferencia de los robots en serie, el robot Delta no debe mover sus actuadores porque estos están ubicados en la base fija (Zabala and Ros, 2007).

Además, al considerarse el robot como una cadena cinemática cerrada, es decir, todos sus eslabones están conectados tanto a la base fija como a la base móvil, la carga se distribuye equitativamente en los eslabones aumentando su rigidez y permitiendo mayor precisión en sus movimientos (Zhang, 2009).

3.3 APLICACIONES EN LA ACTUALIDAD

Actualmente, es común encontrar a los robots paralelos en aplicaciones como:

- Simuladores de movimiento: empresas como Bosch Rexroth, Force Dynamics, CAE, Frasca International, ANCO Engineers, Mitsubishi, Simtec Simulation Technology, Flight Safety International, entre otras, ofrecen simuladores de movimiento (especialmente simuladores de vuelo) basados en mecanismos paralelos.
- Dispositivos médicos: la compañía ISIS (Instrumental Surgical Instruments and Systems) fabricó el SurgiScope (Figura 19), el cual es un robot paralelo que cuenta con siete grados de libertad y es utilizado en neurocirugías.



Figura 19. SurgiScope system by ISIS³.

Otra aplicación en la medicina la presentó la compañía Medical Intelligence con la plataforma de posicionamiento de seis grados de libertad HexaPOD, que permite acomodar al paciente apropiadamente en una cirugía. También se han utilizado a los robots paralelos como suspensiones en las camillas de las ambulancias (Bruzzone and Molfino, 2003); este mecanismo permite reducir las vibraciones que se transmiten al paciente por las irregularidades del terreno y las aceleraciones y velocidades que experimenta la ambulancia, las cuales son perjudiciales para los pacientes. En el LARM (Laboratory of Robotic and Mechatronics) de la universidad de Cassino están experimentando la combinación de un manipulador en serie y uno en paralelo para aplicaciones en medicina (Carbone and Ceccarelli, 2005).

- Sensor de Fuerza/torque: la plataforma de Stewart se convirtió en el primer mecanismo paralelo usado como sensor de fuerza/torque (Dwarakanath et al., 1999; Hongguang et al., 2004; Jiantao et al., 2006). Actualmente, estructuras como la propuesta por Clavel están siendo usadas como dispositivos hápticos por su alta rigidez, baja inercia y por permitir la transmisión de una amplia gama de fuerzas. Un claro ejemplo de esto es el

³ ISIS, SurgiScope solution, [online] <http://www.isis-robotics.com/en/surgiscope-solution.html>

dispositivo Falcon (Figura 20) de la empresa Novint, el cual es un dispositivo de interacción con los computadores que permite “tocar” objetos con diferentes texturas que están en un ambiente virtual.



Figura 20. Dispositivo Falcon de la empresa Novint⁴.

- Pick and place: industrias como Fanuc Robotics, Kawasaki Heavy Industries, Festo, Adept Technology, PWR-Pack, AEMK systems, BOSCH y ABB Automation han implementado robots paralelos de tres o cuatro grados de libertad en tareas de “pick and place” gracias a su velocidad y flexibilidad. Estos robots son empleados en aplicaciones de selección en línea propias de los sectores electrónicos y de alimentos.

Los robots paralelos también tienen un espacio en la academia. En Colombia se han construido robots Delta con fines académicos en pro de una optimización en su diseño mecánico, cinemática, dinámica, control y espacio de trabajo (Angel, Sebastian, Saltaren, and Aracil, 2005; Angel, Sebastian, Saltaren, Aracil, et al., 2005; Silva et al., 2005; Vivas and Poignet, 2006; Vivas, 2007; Martínez et al., 2010; Peña Cortés et al., 2011).

En la universidad Militar Nueva Granada, se han desarrollado dos proyectos relacionados con robots paralelos. El primero es el diseño y construcción de una plataforma de Stewart (Gómez and Guacaneme, 2011), cuyo principal problema es su reducido espacio de trabajo; y el segundo es el desarrollo de un entorno virtual para interactuar con una plataforma de robot paralelo (Castro and Arangure, 2010).

⁴ Novint, Novint Falcon, [Online] <http://home.novint.com/index.php/products/novintfalcon>

CAPÍTULO 4: CINEMÁTICA DEL ROBOT DELTA Y ESPACIO DE TRABAJO

4.1 GEOMETRÍA DEL ROBOT DELTA

Para poder obtener un modelo matemático que describa la cinemática del robot, primero se debe analizar su geometría. Como se observa en la Figura 21, el robot Delta está formado por una base fija, en donde se ubicaron tres servomotores (R_1 , R_2 y R_3) a una misma distancia del centro de la base y separados 120° entre sí; una base móvil conectada a la base fija por medio de tres cadenas cinemáticas, cada una compuesta por dos eslabones, los cuales se conectan entre sí por medio de juntas universales (U). Adicionalmente, una segunda plataforma móvil se une a la primera y cuenta con un cuarto servomotor R_4 para permitir la orientación del robot.

El sistema de referencia XYZ se ubica en el centro de la base fija, con el eje X perpendicular al eje del servomotor R_1 y el eje Z perpendicular a la base.

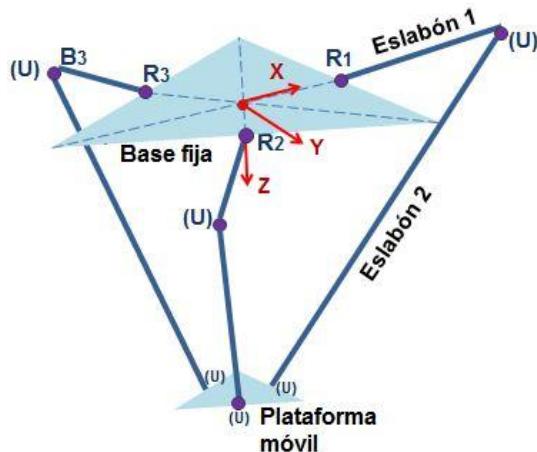


Figura 21. Estructura del robot paralelo RUU.

Los grados de libertad del robot se pueden encontrar empleando la ecuación (2.1). Como el robot es un mecanismo espacial, entonces $\lambda = 6$; adicionalmente, está formado por seis eslabones, una plataforma fija y dos móviles ($n = 9$); cuenta con cuatro articulaciones de revolución y seis articulaciones universales ($j = 10$); cada articulación de revolución cuenta con un grado de libertad y cada articulación universal cuenta con dos grados de libertad ($\sum_{i=1}^j f_i = 16$); Entonces los grados de libertad del robot Delta son $F = 6(9 - 10 - 1) + 16 = 4$.

De esta forma, el robot Delta cuenta con tres grados de libertad para posicionar y un grado de libertad para orientar su efecto final en el espacio de trabajo.

Como se mencionó anteriormente, el robot consta de tres cadenas cinemáticas ($i = 1, 2$ y 3), cada una de ellas formadas por dos eslabones, cuyas dimensiones

son L_1 y L_2 respectivamente. En la Figura 22 se puede apreciar que el eslabón L_1 se une a la base fija por medio de la articulación A_i y al eslabón L_2 por la articulación B_i . El eslabón L_2 se une a la plataforma móvil por medio de la articulación C_i . Finalmente, β_i es el ángulo que hay entre el eslabón L_1 y el L_2 .

El centro de la plataforma móvil representa la posición del robot Delta $P = [Px \; Py \; Pz]^T$.

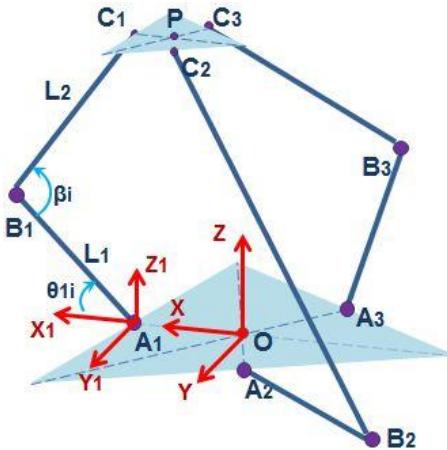


Figura 22. Modelo geométrico del robot Delta.

La Figura 23 define los ángulos de las articulaciones asociados con la i -esima cadena cinemática, donde \overrightarrow{OP} es el vector posición del centro de la plataforma móvil, θ_{1i} es el ángulo que se forma entre X_{i1} y el vector $\overrightarrow{A_iB_i}$, θ_{2i} es el ángulo que se forma entre la proyección de $\overrightarrow{A_iB_i}$ y la intersección del plano del paralelogramo y el plano $X_{i1} - Z_{i1}$ y θ_{3i} es el ángulo que se forma de la dirección de Y_{i1} hasta $\overrightarrow{B_iC_i}$.

En total hay nueve ángulos de articulaciones (θ_{1i} , θ_{2i} y θ_{3i} para $i = 1, 2$ y 3) asociados al manipulador, siendo θ_{11} , θ_{12} y θ_{13} las variables de las articulaciones de los actuadores.

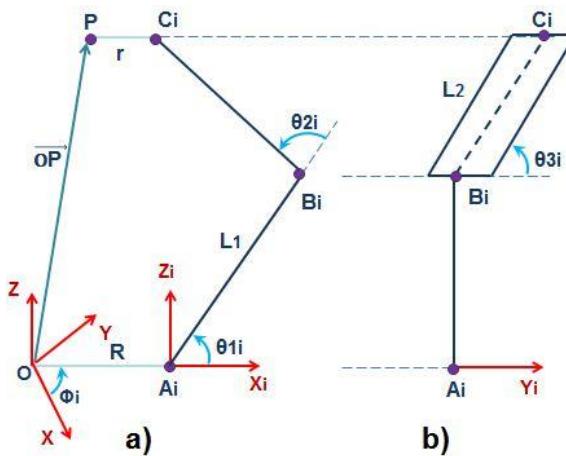


Figura 23. Descripción de los ángulos: a) vista frontal y b) vista lateral.

Una matriz de transformación \mathbf{T} representa la orientación y la posición de un sistema $O'X'Y'Z'$ rotado y trasladado con respecto a un sistema de referencia $OXYZ$. Esta matriz es útil para conocer las coordenadas del vector $\vec{r} = [r_x, r_y, r_z]^T$ en el sistema de referencia, a partir del vector $\vec{r}' = [r'_x, r'_y, r'_z]^T$ en el sistema $O'X'Y'Z'$ (Barrientos, 2007).

$$\begin{bmatrix} r_x \\ r_y \\ r_z \\ 1 \end{bmatrix} = \mathbf{T} \begin{bmatrix} r'_{x'} \\ r'_{y'} \\ r'_{z'} \\ 1 \end{bmatrix} \quad (4.1)$$

donde la matriz \mathbf{T} es:

$$\mathbf{T} = \begin{bmatrix} R_{3x3} & P_{3x1} \\ 0 & 1 \end{bmatrix} = \begin{bmatrix} \text{Rotación} & \text{Traslación} \\ 0 & 1 \end{bmatrix} \quad (4.2)$$

El sistema $A_iX_iY_iZ_i$ está trasladado R a lo largo de OX y rotado ϕ_i alrededor de OZ , con respecto al sistema de referencia $OXYZ$. Para conocer los vectores \overrightarrow{OA}_i , \overrightarrow{OB}_i y \overrightarrow{OC}_i , conociendo los vectores $\overrightarrow{A_iB}_i$ y $\overrightarrow{A_iC}_i$, se utiliza una traslación seguida de una rotación. Si se traslada un vector con coordenadas (x, y, z) , seguida de una rotación sobre el eje Z, la matriz \mathbf{T} es:

$$\mathbf{T}(P, (Z, \alpha)) = \begin{bmatrix} \cos(\alpha) & -\sin(\alpha) & 0 & x\cos(\alpha) - y\sin(\alpha) \\ \sin(\alpha) & \cos(\alpha) & 0 & x\sin(\alpha) + y\cos(\alpha) \\ 0 & 0 & 1 & z \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (4.3)$$

Aplicando la ecuación (4.1) con la transformación homogénea (4.3) se obtienen los vectores \overrightarrow{OA}_i , \overrightarrow{OB}_i y \overrightarrow{OC}_i :

$$\overrightarrow{OA}_i = \begin{bmatrix} c\phi_i & -s\phi_i & 0 & Rc\phi_i \\ s\phi_i & c\phi_i & 0 & Rs\phi_i \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 0 \\ 0 \\ 0 \\ 1 \end{bmatrix} = \begin{bmatrix} Rc\phi_i \\ Rs\phi_i \\ 0 \\ 1 \end{bmatrix} \quad (4.4)$$

$$\overrightarrow{OB}_i = \begin{bmatrix} c\phi_i & -s\phi_i & 0 & Rc\phi_i \\ s\phi_i & c\phi_i & 0 & Rs\phi_i \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} L_1c\theta_{1i} \\ 0 \\ L_1s\theta_{1i} \\ 1 \end{bmatrix} = \begin{bmatrix} c\phi_i(R + L_1c\theta_{1i}) \\ s\phi_i(R + L_1c\theta_{1i}) \\ L_1s\theta_{1i} \\ 1 \end{bmatrix} \quad (4.5)$$

$$\overrightarrow{OC}_i = \begin{bmatrix} c\phi_i & -s\phi_i & 0 & Rc\phi_i \\ s\phi_i & c\phi_i & 0 & Rs\phi_i \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} P_Xc\phi_i + P_Ys\phi_i + r - R \\ P_Yc\phi_i - P_Xs\phi_i \\ P_Z \\ 1 \end{bmatrix} = \begin{bmatrix} P_X + rc\phi_i \\ P_Y + rs\phi_i \\ P_Z \\ 1 \end{bmatrix} \quad (4.6)$$

Adicionalmente, se pueden encontrar los vectores \overrightarrow{BA}_i , \overrightarrow{BC}_i y \overrightarrow{AC}_i :

$$\overrightarrow{BA}_i = \overrightarrow{OA}_i - \overrightarrow{OB}_i = \begin{bmatrix} -c\phi_iL_1c\theta_{1i} \\ -s\phi_iL_1c\theta_{1i} \\ -L_1s\theta_{1i} \end{bmatrix} \quad (4.7)$$

$$\overrightarrow{B_l C_l} = \overrightarrow{O C_l} - \overrightarrow{O B_l} = \begin{bmatrix} P_x - c\phi_i(R + L_1 c\theta_{1i} - r) \\ P_y - s\phi_i(R + L_1 c\theta_{1i} - r) \\ P_z - L_1 s\theta_{1i} \end{bmatrix} \quad (4.8)$$

$$\overrightarrow{A_l C_l} = \overrightarrow{O C_l} - \overrightarrow{O A_l} = \begin{bmatrix} P_x + c\phi_i(r - R) \\ P_y + s\phi_i(r - R) \\ P_z \end{bmatrix} \quad (4.9)$$

Por facilidad de expresión, en las ecuaciones (4.4) a (4.9) se denota el coseno de un ángulo $\cos(x)$ como cx y $\sin(x)$ como sx .

Geométricamente, la base fija y la plataforma móvil son dos triángulos equiláteros de lados L y l respectivamente. Como se aprecia en la Figura 24.a la distancia del centro de la base a A_i es R . En la Figura 24.b se aprecia que la distancia del punto P a C_i es r .

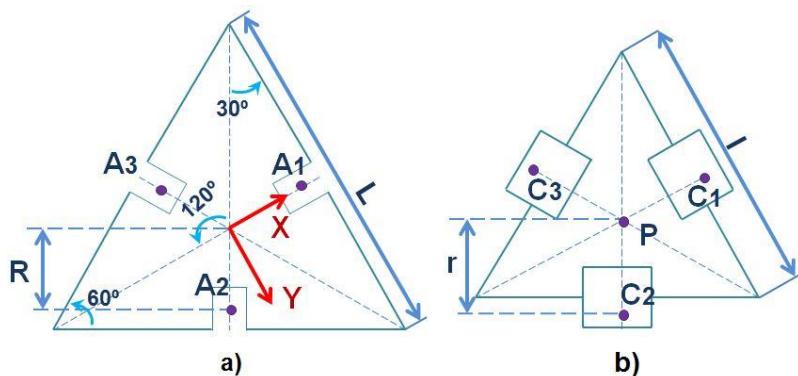


Figura 24. Forma geométrica de a) la base fija y b) la plataforma móvil.

4.2 CINEMÁTICA INVERSA

La cinemática inversa permite conocer las variables de los actuadores si se conoce la posición del efecto final, en este caso, la posición de la plataforma móvil. Solucionar la cinemática inversa le permite al sistema de control enviarle órdenes a los actuadores para llevar el efecto final de un robot hasta un punto deseado en el espacio.

Gracias a la simetría del robot Delta, cada cadena cinemática puede ser analizada independientemente.

Es posible formar un triángulo con vértices $A_i B_i C_i$ como se aprecia en la Figura 25 y se puede encontrar la magnitud $A_i C_i$ por medio del teorema del coseno:

$$A_i C_i^2 = L_1^2 + L_2^2 - 2L_1 L_2 \cos(\beta_i) \quad (4.10)$$

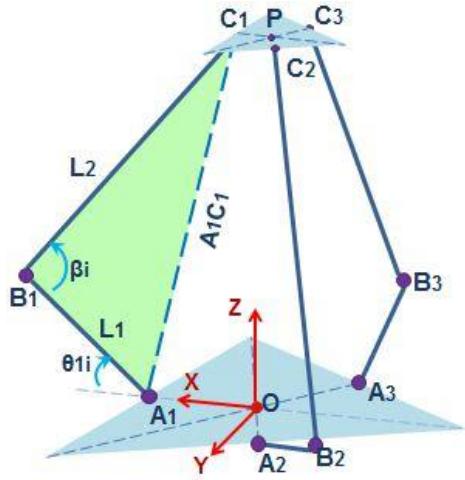


Figura 25. Representación vectorial de la primera cadena cinemática.

En (4.10) la única incógnita es $\cos(\beta_i)$, la cual se puede encontrar usando la definición del producto escalar:

$$\overrightarrow{B_l A_l} * \overrightarrow{B_l C_l} = (B_l A_l)(B_l C_l) \cos(\beta_i) \quad (4.11)$$

Como $B_l A_l = L_1$ y $B_l C_l = L_2$ entonces:

$$\cos(\beta_i) = \frac{\overrightarrow{B_l A_l} * \overrightarrow{B_l C_l}}{L_1 L_2} \quad (4.12)$$

Remplazando (4.12) en (4.10) se obtiene:

$$A_l C_l^2 = L_1^2 + L_2^2 - \overrightarrow{B_l A_l} * \overrightarrow{B_l C_l} \quad (4.13)$$

donde:

$$2\overrightarrow{B_l A_l} * \overrightarrow{B_l C_l} = 2(L_1^2 - L_1 c \theta_{1i} (P_X c \emptyset_i + P_Y s \emptyset_i + r - R) - L_1 s \theta_{1i} P_Z) \quad (4.14)$$

Remplazando (4.14) en (4.13) se obtiene:

$$A_l C_l^2 = L_2^2 - L_1^2 + 2L_1 c \theta_{1i} (P_X c \emptyset_i + P_Y s \emptyset_i + r - R) + 2L_1 s \theta_{1i} P_Z \quad (4.15)$$

La magnitud del vector de la ecuación (4.9) ($\overrightarrow{AC_l}$) es:

$$A_l C_l^2 = (P_X + c \emptyset_i (r - R))^2 + (P_Y + s \emptyset_i (r - R))^2 + P_Z^2 \quad (4.16)$$

Igualando (4.16) y (4.15) se obtiene una ecuación de la forma:

$$a_i \sin \theta_{1i} + b_i \cos \theta_{1i} = c_i \quad (4.17)$$

donde:

$$ca_i = 2P_Z L_1$$

$$cb_i = 2L_1 (P_X c \emptyset_i + P_Y s \emptyset_i + r - R)$$

$$cc_i = (P_X + c \emptyset_i (r - R))^2 + (P_Y + s \emptyset_i (r - R))^2 + P_Z^2 + L_1^2 - L_2^2$$

Es posible rescribir (4.17) teniendo en cuenta la identidad trigonométrica de suma de senos y cosenos:

$$\theta_{1i} = \sin^{-1} \left(\frac{cc_i}{\sqrt{ca_i^2 + cb_i^2}} \right) - \tan^{-1} \left(\frac{cb_i}{ca_i} \right) \quad (4.18)$$

4.2.1 Variable de articulación θ_{11}

Para encontrar el ángulo θ_{11} , i toma el valor de uno y el ángulo ϕ_1 es de 0° , entonces:

$$ca_1 = 2L_1 P_Z$$

$$cb_1 = 2L_1(P_X + r - R)$$

$$cc_1 = (P_X + r - R)^2 + P_Y^2 + P_Z^2 + L_1^2 - L_2^2$$

$$\theta_{11} = \sin^{-1} \left(\frac{cc_1}{\sqrt{ca_1^2 + cb_1^2}} \right) - \tan^{-1} \left(\frac{cb_1}{ca_1} \right) \quad (4.19)$$

4.2.2 Variable de articulación θ_{12}

Para encontrar el ángulo θ_{12} , i toma el valor de dos y el ángulo ϕ_2 es de 120° , entonces:

$$ca_2 = 2P_Z L_1$$

$$cb_2 = L_1(-P_X + \sqrt{3}P_Y + 2r - 2R)$$

$$cc_2 = \left(P_X - \frac{1}{2}(r - R) \right)^2 + \left(P_Y + \frac{\sqrt{3}}{2}(r - R) \right)^2 + P_Z^2 + L_1^2 - L_2^2$$

$$\theta_{12} = \sin^{-1} \left(\frac{cc_2}{\sqrt{ca_2^2 + cb_2^2}} \right) - \tan^{-1} \left(\frac{cb_2}{ca_2} \right) \quad (4.20)$$

4.2.3 Variable de articulación θ_{13}

Para encontrar el ángulo θ_{13} , i toma el valor de tres y el ángulo ϕ_3 es de 240° , entonces:

$$ca_3 = 2P_Z L_1$$

$$cb_3 = L_1(-2P_X - \sqrt{3}P_Y + 2r - 2R)$$

$$cc_3 = \left(P_X - \frac{1}{2}(r - R) \right)^2 + \left(P_Y - \frac{\sqrt{3}}{2}(r - R) \right)^2 + P_Z^2 + L_1^2 - L_2^2$$

$$\theta_{13} = \sin^{-1} \left(\frac{cc_3}{\sqrt{ca_3^2 + cb_3^2}} \right) - \tan^{-1} \left(\frac{cb_3}{ca_3} \right) \quad (4.21)$$

4.3 CINEMÁTICA DIRECTA

En la sección anterior se definió el vector $\overrightarrow{BC_i}$, ecuación (5.8), cuya magnitud es L_2 :

$$BC_i = (P_X - c\phi_i(R + L_1c\theta_{1i} - r))^2 + (P_Y - s\phi_i(R + L_1c\theta_{1i} - r))^2 + (P_Z - L_1s\theta_{1i})^2 = L_2^2 \quad (4.22)$$

De (4.22) se obtiene un sistema de tres ecuaciones no lineales que relaciona las variables de las articulaciones (θ_{11}, θ_{12} y θ_{13}) con la posición del efecto móvil (P_X, P_Y, P_Z):

$$f_1 = (P_X - R - L_1c\theta_{11} + r)^2 + (P_Y)^2 + (P_Z - L_1s\theta_{11})^2 - L_2^2 \quad (4.23)$$

$$f_2 = \left(P_X + \frac{1}{2}(R - r + L_1c\theta_{12}) \right)^2 + \left(P_Y - \frac{\sqrt{3}}{2}(R - r + L_1c\theta_{12}) \right)^2 + (P_Z - L_1s\theta_{12})^2 - L_2^2 \quad (4.24)$$

$$f_3 = \left(P_X + \frac{1}{2}(R - r + L_1c\theta_{13}) \right)^2 + \left(P_Y + \frac{\sqrt{3}}{2}(R - r + L_1c\theta_{13}) \right)^2 + (P_Z - L_1s\theta_{13})^2 - L_2^2 \quad (4.25)$$

Las ecuaciones (4.23) a (4.25) se pueden escribir de la forma:

$$f_1 = P_X^2 + 2a_1P_X + P_Y^2 + P_Z^2 - 2a_2P_Z - a_3 \quad (4.26)$$

dónde:

$$a_1 = r - R - L_1c\theta_{11}$$

$$a_2 = L_1s\theta_{11}$$

$$a_3 = L_2^2 - a_1^2 - a_2^2$$

$$f_2 = P_X^2 + b_1P_X + P_Y^2 - \sqrt{3}b_1P_Y + P_Z^2 - 2b_2P_Z - b_3 \quad (4.27)$$

dónde:

$$b_1 = R - r + L_1c\theta_{12}$$

$$b_2 = L_1s\theta_{12}$$

$$b_3 = L_2^2 - b_1^2 - b_2^2$$

$$f_3 = P_X^2 + c_1P_X + P_Y^2 + \sqrt{3}c_1P_Y + P_Z^2 - 2c_2P_Z - c_3 \quad (4.28)$$

dónde:

$$c_1 = R - r + L_1 c \theta_{13}$$

$$c_2 = L_1 s \theta_{13}$$

$$c_3 = L_2^2 - c_1^2 - c_2^2$$

Se resuelven las ecuaciones (4.26) a (4.28) para encontrar P_X y P_Y . Primero se restan las ecuaciones (4.27) y (4.28) de la ecuación (4.26) respectivamente. De las dos ecuaciones resultantes, se despeja P_X y se igualan las ecuaciones para poder encontrar P_Y y luego P_X , obteniendo:

$$P_Y = \frac{A + BP_Z}{C} \quad (4.29)$$

dónde:

$$A = \frac{a_3 - b_3}{2a_1 - b_1} - \frac{a_3 - c_3}{2a_1 - c_1}$$

$$B = \frac{2(c_2 - a_2)}{2a_1 - c_1} - \frac{2(b_2 - a_2)}{2a_1 - b_1}$$

$$C = \frac{\sqrt{3}b_1}{2a_1 - b_1} + \frac{\sqrt{3}c_1}{2a_1 - c_1}$$

$$P_X = D - EP_Z \quad (4.30)$$

dónde:

$$D = \frac{a_3 - b_3}{2a_1 - b_1} - \frac{\sqrt{3}b_1 A}{C(2a_1 - b_1)}$$

$$E = \frac{2(b_2 - a_2)}{2a_1 - b_1} + \frac{\sqrt{3}b_1 B}{C(2a_1 - b_1)}$$

Sustituyendo (4.29) y (4.30) en (4.26) se obtiene:

$$aP_Z^2 + bP_Z + c = 0 \quad (4.31)$$

dónde:

$$a = E^2 + \left(\frac{B}{C}\right)^2 + 1$$

$$b = 2 \left(\frac{AB}{C^2} - DE - a_1 E - a_2 \right)$$

$$c = D^2 + \left(\frac{A}{C}\right)^2 + 2a_1 D - a_3$$

La ecuación (4.31) tiene dos posibles soluciones, pero como el sistema de referencia del robot Delta es el centro de la plataforma fija, P_Z no puede tomar valores negativos, entonces:

$$P_Z = \frac{-b + \sqrt{b^2 - 4ac}}{2a} \quad (4.32)$$

La cinemática directa e inversa solo son válidas para puntos a los que pueda acceder la plataforma móvil sin que sus componentes colisionen. Para detectar una colisión entre componentes del robot es necesario encontrar los ángulos β_i , θ_{2i} y θ_{3i} .

En la Figura 26 se aprecia que el vector $\overrightarrow{A_i C_i}$ es igual a la suma de los vectores $\overrightarrow{A_i B_i}$ y $\overrightarrow{B_i C_i}$, visto desde el sistema de coordenadas $X_i Y_i Z_i$. De la ecuación (4.6) se observa que el vector $\overrightarrow{A_i C_i}$ se puede expresar como:

$$\overrightarrow{A_i C_i} = \begin{bmatrix} P_X c\phi_i + P_Y s\phi_i + r - R \\ P_Y c\phi_i - P_X s\phi_i \\ P_Z \end{bmatrix} \quad (4.33)$$

De la Figura 26 se observa que:

$$\overrightarrow{A_i B_i} + \overrightarrow{B_i C_i} = \begin{bmatrix} L_1 c\theta_{1i} + L_2 c\theta_{3i} c(180^\circ - \theta_{1i} - \theta_{2i}) \\ L_2 s\theta_{3i} \\ L_1 s\theta_{1i} + L_2 c\theta_{3i} s(180^\circ - \theta_{1i} - \theta_{2i}) \end{bmatrix} \quad (4.34)$$

Igualando (4.33) y (4.34) se obtiene el valor de los ángulos θ_{3i} y θ_{2i} :

$$\theta_{3i} = \sin^{-1} \left(\frac{P_Y c\phi_i - P_X s\phi_i}{L_2} \right) \quad (4.35)$$

$$\theta_{2i} = 180^\circ - \sin^{-1} \left(\frac{P_Z - L_1 s\theta_{1i}}{L_2 c\theta_{3i}} \right) - \theta_{1i} \quad (4.36)$$

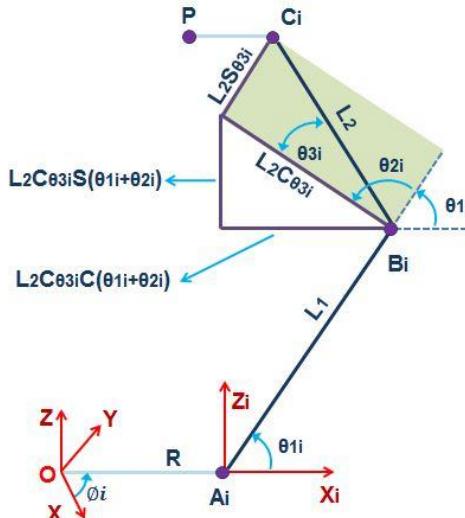


Figura 26. Cadena cinemática i en las coordenadas $X_i Y_i Z_i$.

Para encontrar el ángulo β_i es necesario conocer la posición de la plataforma móvil con respecto a la plataforma fija, para encontrar la magnitud del vector $\overrightarrow{A_i C_i}$ y poder β_i despejar de la ecuación (4.10):

$$\beta_i = \cos^{-1} \left(\frac{L_1^2 + L_2^2 - A_i C_i^2}{2L_1 L_2} \right) \quad (4.37)$$

Se considera que no existen colisiones entre los componentes del robot si satisfacen las siguientes restricciones:

- $10^\circ \leq \beta_i \leq 180^\circ$
- $10^\circ \leq \theta_{2i} \leq 180^\circ$
- $-45^\circ \leq \theta_{3i} \leq 45^\circ$

La restricción del ángulo θ_{3i} depende de las junturas universales. Por lo tanto, el intervalo puede aumentar o disminuir una vez se mecanicen las junturas.

4.4 MATRIZ JACOBIANA

Las cinemáticas directa e inversa no tienen en cuenta las fuerzas o pares que actúan sobre el robot. La cinemática diferencial relaciona las velocidades angulares y lineales del efecto final con las velocidades en las articulaciones.

Gosselin y Angeles plantean que una cadena cinemática cerrada se caracteriza por un vector de entrada θ n-dimensional, que corresponde a las articulaciones activas, y un vector de salida x m-dimensional, que corresponde a las coordenadas cartesianas de la plataforma móvil. La relación entre esos dos vectores es (Gosselin and Angeles, 1990):

$$F(\theta, x) = 0 \quad (4.38)$$

donde F es una función implícita n-dimensional de θ y x , y $\mathbf{0}$ es un vector cero de n dimensiones.

La rigidez de un manipulador paralelo en un punto de su espacio de trabajo se puede caracterizar por su matriz de rigidez. Esta matriz, relaciona las fuerzas y torques en el efecto final con sus desplazamientos angulares y lineales (Gosselin, 1990). Diferenciando (4.38) con respecto al tiempo se obtiene una relación entre las velocidades de la entrada y la salida en un manipulador paralelo:

$$\dot{\theta} = J\nu \quad (4.39)$$

donde $\dot{\theta}$ es el vector de las velocidades en las articulaciones activas, J es la matriz Jacobiana y ν es un vector de seis dimensiones que contiene las velocidades lineal y angular del efecto final:

$$\nu = \begin{bmatrix} v_p \\ \omega_p \end{bmatrix} \quad (4.40)$$

Gosselin y Angeles separan la matriz Jacobiana en dos matrices, una matriz asociada con la cinemática directa (J_x) y otra matriz asociada con la cinemática inversa (J_θ) (Gosselin and Angeles, 1990), donde:

$$J_x = \frac{\partial F}{\partial x} \quad y \quad J_\theta = \frac{\partial F}{\partial \theta} \quad (4.41)$$

Entonces, la matriz Jacobiana de un robot paralelo queda definida como $J = (J_\theta)^{-1}J_x$, la cual corresponde a la matriz Jacobiana inversa de un manipulador en serie. De esta forma, la ecuación (4.39) se puede escribir:

$$\dot{\theta} = (J_{\theta})^{-1} J_x \dot{x} \quad (4.42)$$

donde $\dot{\theta}$ es el vector de las velocidades de los actuadores y \dot{x} es el vector de la velocidad del punto P en la plataforma móvil en el sistema de referencia XYZ . Entonces, la ecuación (4.42) puede escribirse:

$$\begin{bmatrix} \dot{\theta}_{11} \\ \dot{\theta}_{12} \\ \dot{\theta}_{13} \end{bmatrix} = (J_{\theta})^{-1} J_x \begin{bmatrix} V_{PX} \\ V_{PY} \\ V_{PZ} \end{bmatrix} \quad (4.43)$$

Referente a la Figura 26, una ecuación del lazo cerrado de la i -ésima cadena cinemática puede ser:

$$\overrightarrow{OP} + \overrightarrow{PC_i} = \overrightarrow{OA_i} + \overrightarrow{A_iB_i} + \overrightarrow{B_iC_i} \quad (4.44)$$

Derivando la ecuación (4.44) con respecto al tiempo y expresando el resultado en el sistema $X_iY_iZ_i$:

$$\overrightarrow{V_p} = \overrightarrow{\omega_{1i}} \times \overrightarrow{A_iB_i} + \overrightarrow{\omega_{2i}} \times \overrightarrow{B_iC_i} \quad (4.45)$$

donde $\overrightarrow{V_p}$ es la velocidad lineal de la plataforma móvil y ω_{ji} es la velocidad angular del j -ésimo eslabón de la i -ésima cadena cinemática.

De la expresión (4.45) las velocidades $\overrightarrow{\omega_{2i}}$ no son de interés debido a que son las velocidades de las articulaciones pasivas (junturas universales), por lo tanto, se eliminan multiplicando ambos lados de la ecuación por $\overrightarrow{B_iC_i}$:

$$\overrightarrow{B_iC_i} * \overrightarrow{V_p} = \overrightarrow{B_iC_i} (\overrightarrow{\omega_{1i}} \times \overrightarrow{A_iB_i}) + \overrightarrow{B_iC_i} (\overrightarrow{\omega_{2i}} \times \overrightarrow{B_iC_i})$$

donde $\overrightarrow{B_iC_i} (\overrightarrow{\omega_{2i}} \times \overrightarrow{B_iC_i}) = \mathbf{0}$ por la propiedad de *cancelación por ortogonalidad* del producto cruz. Entonces:

$$\overrightarrow{B_iC_i} * \overrightarrow{V_p} = \overrightarrow{B_iC_i} (\overrightarrow{\omega_{1i}} \times \overrightarrow{A_iB_i}) \quad (4.46)$$

dónde:

$$\begin{aligned} \overrightarrow{B_iC_i} &= \begin{bmatrix} L_2 C \theta_{3i} C(\theta_{1i} + \theta_{2i}) \\ L_2 S \theta_{3i} \\ L_2 C \theta_{3i} S(\theta_{1i} + \theta_{2i}) \end{bmatrix}, \overrightarrow{V_p} = \begin{bmatrix} V_{px} C \emptyset i + V_{py} S \emptyset i \\ V_{py} C \emptyset i - V_{px} S \emptyset i \\ V_{pz} \end{bmatrix}, \overrightarrow{\omega_{1i}} = \begin{bmatrix} 0 \\ -\dot{\theta}_{1i} \\ 0 \end{bmatrix} \text{ y} \\ \overrightarrow{A_iB_i} &= \begin{bmatrix} L_1 C \theta_{1i} \\ 0 \\ L_1 S \theta_{1i} \end{bmatrix} \end{aligned}$$

Resolviendo (4.46) y simplificando se obtiene:

$$j_{xi1} V_{px} + j_{xi2} V_{py} + j_{xi3} V_{pz} = j_{\theta i} \dot{\theta}_{1i} \quad (4.47)$$

dónde:

$$j_{xi1} = C \theta_{3i} C(\theta_{1i} + \theta_{2i}) C \emptyset i - S \theta_{3i} S \emptyset i$$

$$j_{xi2} = C \theta_{3i} C(\theta_{1i} + \theta_{2i}) S \emptyset i + S \theta_{3i} C \emptyset i$$

$$j_{xi3} = C \theta_{3i} S(\theta_{1i} + \theta_{2i})$$

$$j_{\theta i} = L_1 S \theta_{2i} C \theta_{3i}$$

La expresión (4.47) representa un sistema de tres ecuaciones que se puede escribir de la forma $\mathbf{J}_x \dot{\mathbf{x}} = \mathbf{J}_\theta \dot{\theta}$:

$$\begin{bmatrix} j_{x11} & j_{x12} & j_{x13} \\ j_{x21} & j_{x22} & j_{x23} \\ j_{x31} & j_{x32} & j_{x33} \end{bmatrix} \begin{bmatrix} V_{px} \\ V_{py} \\ V_{pz} \end{bmatrix} = \begin{bmatrix} j_{\theta1} & 0 & 0 \\ 0 & j_{\theta2} & 0 \\ 0 & 0 & j_{\theta3} \end{bmatrix} \begin{bmatrix} \dot{\theta}_{11} \\ \dot{\theta}_{12} \\ \dot{\theta}_{13} \end{bmatrix} \quad (4.48)$$

Entonces, la matriz Jacobiana del robot Delta es:

$$\mathbf{J} = (\mathbf{J}_\theta)^{-1} \mathbf{J}_x = \left(\begin{bmatrix} j_{\theta1} & 0 & 0 \\ 0 & j_{\theta2} & 0 \\ 0 & 0 & j_{\theta3} \end{bmatrix} \right)^{-1} \begin{bmatrix} j_{x11} & j_{x12} & j_{x13} \\ j_{x21} & j_{x22} & j_{x23} \\ j_{x31} & j_{x32} & j_{x33} \end{bmatrix} \quad (4.49)$$

Dependiendo de cuál matriz es singular, un mecanismo de lazo cerrado puede tener una configuración singular de cinemática directa, una configuración singular de cinemática inversa, o ambas.

La singularidad es uno de los mayores problemas en un robot paralelo debido a que en esas configuraciones el robot no puede ser controlado y existen infinitas fuerzas/torques en sus junturas, lo que puede generar su ruptura (Merlet, 2007).

4.4.1 Singularidades de la cinemática inversa

Se dice que existe una singularidad de la cinemática inversa cuando el determinante de \mathbf{J}_θ tiende a cero:

$$\det(\mathbf{J}_\theta) = (S\theta_{21}C\theta_{31})(S\theta_{22}C\theta_{32})(S\theta_{23}C\theta_{33}) = 0 \quad (4.50)$$

Esto implica que movimientos infinitesimales de la plataforma móvil a lo largo de ciertas direcciones no pueden realizarse, es decir, el manipulador es capaz de resistir fuerzas y torques en esas direcciones. Por lo tanto el manipulador pierde uno o más DOF. Generalmente, estas singularidades ocurren en los límites del espacio de trabajo del manipulador (Tsai, 1999).

Para que se cumpla (4.50) $S\theta_{2i} = 0$ ó $C\theta_{3i} = 0$, por lo tanto, esta singularidad ocurre cuando se satisface alguna de las siguientes condiciones:

- ✓ $\theta_{2i} = 0^\circ$ o 180°
- ✓ $\theta_{3i} = 90^\circ$ o 270°

4.4.2 Singularidades de la cinemática directa

Se dice que existe una singularidad de la cinemática inversa cuando el determinante de \mathbf{J}_x tiende a cero:

$$\det(\mathbf{J}_x) = 0 \quad (4.51)$$

La plataforma móvil posee movimientos infinitesimales en algunas direcciones mientras los actuadores están bloqueados completamente, es decir, la velocidad de la plataforma móvil es diferente de cero a pesar que las velocidades de los actuadores son cero (Merlet, 1992).

4.4.3 Combinación de singularidades

Se dice que las singularidades son combinadas cuando los determinantes de J_θ y de J_x tienden a cero. Este tipo de singularidad es común en robots espaciales. La plataforma móvil puede experimentar algunos movimientos infinitesimales mientras los actuadores están fijos o por el contrario permanecer inmóvil mientras los actuadores presentan movimientos infinitesimales.

4.4.4 Destreza del robot delta

La matriz Jacobiana permite identificar las configuraciones no deseadas en un robot Delta (singularidades). Adicionalmente, permite conocer el nivel de destreza de cada configuración del robot por medio del número de condición de la matriz Jacobiana:

$$\lambda = \|J\| * \|J^{-1}\| \quad (4.52)$$

debido a que la el número de condición de una matriz se define como la norma-2 de la matriz multiplicada por la norma-2 de la matriz inversa (Chapra and Canale, 2001; Merlet, 2010).

La destreza local de un robot paralelo se define como la inversa del número de condición de la matriz Jacobiana (Silva et al., 2005):

$$\varsigma = \frac{1}{\lambda}, \quad 0 \leq \varsigma \leq 1 \quad (4.53)$$

Si el valor de ς es cero es porque el robot está en una configuración singular y si el valor de ς es uno, la configuración del robot se conoce como isotrópica (Badescu et al., 2002).

En vista que el índice ς permite analizar únicamente cada configuración del robot paralelo, Gosselin y Angeles en 1991, plantearon un índice que permite medir la destreza de un robot en todo su espacio de trabajo (índice global):

$$\eta = \frac{\int_W \varsigma dW}{\int_W dW} \quad (4.54)$$

donde el numerador $\int_W \varsigma dW$ representa la sumatoria de las destrezas locales del robot Delta en todo su espacio de trabajo y el denominador $\int_W dW$ es el valor del espacio de trabajo del robot.

4.5 ESPACIO DE TRABAJO DEL ROBOT DELTA

El espacio de trabajo de un robot se considera como la región en el plano cartesiano tridimensional que puede ser alcanzada por un punto de su efecto final (Laribi et al., 2007), en el caso de un robot Delta, es la región en el espacio tridimensional que puede alcanzar un punto de su plataforma móvil. Al inicio, los investigadores solo se preocuparon por encontrar el espacio de trabajo de los robots paralelos, es por ello, que en 1994 Merlet presentó un algoritmo para

determinar si la trayectoria de un manipulador paralelo entre dos puntos se podía realizar⁵, con el fin de verificar que las trayectorias del manipulador estuvieran dentro de su espacio de trabajo.

Diseñar un robot Delta para un máximo espacio de trabajo no garantiza que el robot vaya a ser óptimo para aplicaciones prácticas y es factible que resulte un manipulador con características cinemáticas no deseadas como lo son una baja destreza y maniobrabilidad (Stamper, 1997). Por este motivo, esta investigación se centra en diseñar un robot Delta que logre alcanzar la mayor cantidad de puntos en el espacio.

Existen básicamente dos métodos para encontrar el espacio de trabajo de un manipulador paralelo: el método geométrico y el método de discretización (Merlet, 2010), siendo este último el más usado. El método de discretización, que se basa en métodos numéricos, consiste en discretizar el espacio en tres dimensiones, resolviendo la cinemática inversa para cada punto y verificando las restricciones que limitan dicho espacio de trabajo (Ottaviano and Ceccarelli, 2000). El método numérico más conocido para encontrar el espacio de trabajo de un manipulador paralelo es el método de Monte Carlo (Stamper et al., 1997) y consiste en:

- 1) Definir un volumen de trabajo (V) que encierre el volumen de trabajo del robot Delta.
- 2) Generar un gran número (n_{total}) de puntos, seleccionados aleatoriamente, que se encuentren dentro del Volumen V .
- 3) Utilizar la cinemática inversa para evaluar cada punto y determinar si se encuentra dentro del espacio de trabajo del robot. Una vez se evalúan todos los puntos, se obtiene el número total de puntos dentro del espacio de trabajo del robot (n_{in}).
- 4) El volumen del espacio de trabajo de robot (Figura 27) se calcula multiplicando el volumen de trabajo V con el número total de puntos que alcanzo el robot y dividiendo por el número total de puntos seleccionados

$$V' = \frac{n_{in}}{n_{total}} V.$$

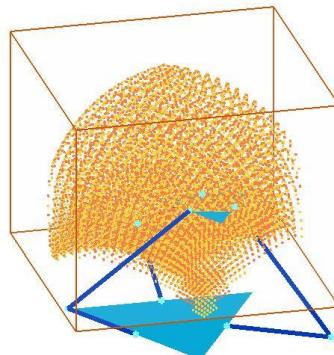


Figura 27. Espacio de trabajo del Robot Delta, método de Monte Carlo.

⁵ Merlet, J.P., 1994. Trajectory verification of parallel manipulators in the workspace. In the 1994 IEEE International Conference on Robotics and Automation. San Diego, CA, pp. 2166 –2171.

Como se aprecia en las Figuras 28 y 29, el espacio de trabajo del robot Delta depende de:

- Las dimensiones del robot (L_1, L_2, R y r).
- Las restricciones de los actuadores (θ_{11}, θ_{12} y θ_{13}).
- Las restricciones de las junturas universales ($\theta_{21}, \theta_{22}, \theta_{23}, \theta_{31}, \theta_{32}$ y θ_{33}).

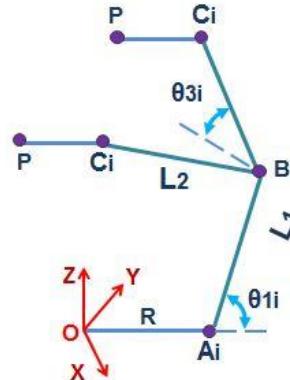


Figura 28. Restricciones del ángulo θ_{i3} .

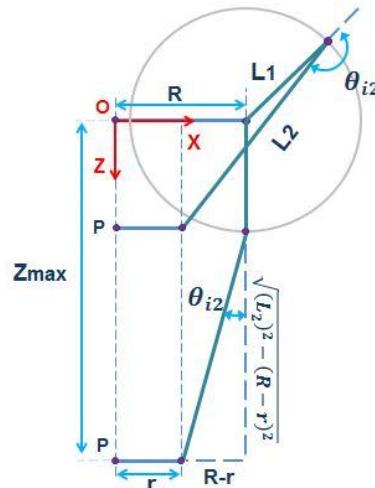


Figura 29. Restricción del ángulo θ_{i2} .

Adicionalmente, en la Figura 29 se observa que la plataforma móvil alcanza su altura máxima cuando las variables de los tres actuadores son 90°:

$$P_{Zmax} = L_1 + \sqrt{(L_2)^2 - (R - r)^2} \quad (4.55)$$

La exactitud del volumen de trabajo del robot Delta, depende de la probabilidad de que los puntos seleccionados aleatoriamente puedan ser alcanzados por el robot. Por esta razón, en este trabajo de grado se plantea un algoritmo para calcular la cantidad exacta de los puntos en el espacio que es capaz de alcanzar la plataforma móvil del robot. El algoritmo se basa en la solución de la cinemática directa para todas las posibles combinaciones de los actuadores, obteniendo en cada caso un punto en el espacio. El punto pertenece al espacio de trabajo del robot sí y solo sí:

- $\theta_{2i} \in [10^\circ, 180^\circ]$ y $\theta_{3i} \in [-45^\circ, 45^\circ]$ para $i = 1, 2$ y 3 .

La figura 30 muestra el diagrama de flujo del algoritmo.

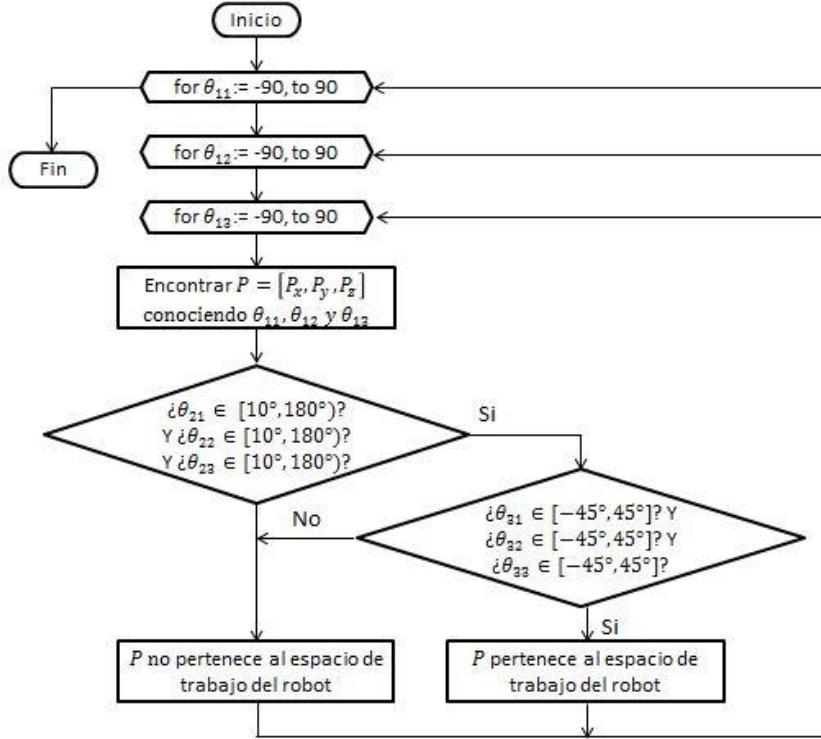


Figura 30. Diagrama de flujo del programa del espacio de trabajo del robot.

El paso incremental de las variables de los actuadores (θ_{11} , θ_{12} y θ_{13}) depende de la sensibilidad de los actuadores. Entre menor sea este incremento, más exacto es el espacio de trabajo que se obtiene.

Para encontrar la destreza global del robot, primero se calcula el índice local de destreza de cada punto que pertenece a su espacio de trabajo, luego se suman todos los índices locales y se dividen por la cantidad de puntos que el efecto final del robot es capaz de alcanzar en el espacio.

4.5.1 Optimización del espacio de trabajo del robot

En este trabajo se busca encontrar un conjunto de vectores de decisión, cuyos elementos sean las dimensiones de los eslabones L_1 y L_2 , que maximicen el espacio de trabajo y la destreza del robot Delta. De esta forma, las funciones objetivo son la cantidad de puntos que forman el espacio de trabajo del robot y el índice global de destreza planteado por Gosselin y Angeles.

Las dimensiones de los eslabones deben satisfacer las siguientes restricciones:

- $L_1 < L_2$
- $100 < L_1 < 350$

- $360 < L_2 < 600$
- $L_1 + L_2 < 900$

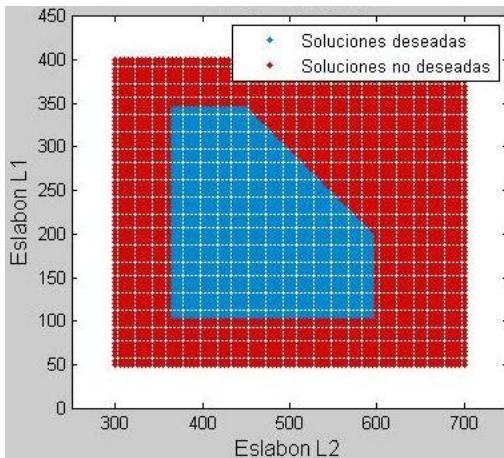


Figura 31. Espacio de solución del problema.

En la Figura 31 se aprecia el espacio de solución del problema en donde los puntos azules representan las posibles dimensiones de los eslabones y los puntos rojos son dimensiones que no cumplen con las restricciones planteadas anteriormente.

Para encontrar el conjunto formado por las mejores soluciones, se plantearon seis métodos diferentes basados en algoritmos genéticos:

- RWGA: Random Weight Genetic Algorithm.
- FWGA: Fixed Weight Genetic Algorithm.
- AWGA: Adaptive Weight Genetic Algorithm.
- NSGA: Non-dominated Sorting Genetic Algorithm.
- NPGA: Niched Pareto Genetic Algorithm.
- MOFGA: Multiplied Objectives Genetic Algorithm.

Los primeros cinco algoritmos se explican con detalle en la sección 2.8. El último algoritmo es un método propuesto en este trabajo en donde el valor de adaptabilidad de cada individuo se encuentra multiplicando los dos valores de las funciones objetivo con el fin preservar los individuos con la mayor cantidad de puntos que pueden alcanzar y el índice global de destreza más alto.

Gracias a que los seis algoritmos se basan en algoritmos genéticos, se deben seguir los nueve pasos siguientes para poder encontrar el conjunto de soluciones del problema:

Paso 1: Generar una población inicial de N individuos, verificando que los individuos cumplan con las restricciones de los eslabones L_1 y L_2 .

Paso 2: Evaluar cada individuo en las dos funciones objetivo, es decir, obtener el índice de destreza global y la cantidad de puntos que forman el espacio de trabajo del robot Delta para cada solución.

Paso 3: Se forma un conjunto tentativo de soluciones óptimas de Pareto, donde la mitad corresponde a los diez individuos de la población actual con la mayor cantidad de puntos en el espacio que es capaz de alcanzar la plataforma móvil y la otra mitad son los diez individuos con el mejor índice global de destreza.

Paso 4: Se le asigna un valor de adaptabilidad a cada individuo.

Paso 5: Por medio del operador de selección se escogen algunos individuos como padres. Entre mayor sea la adaptabilidad de un individuo mayor es la probabilidad de ser escogido como parente. Para tener más opciones de soluciones, se emplearon tres métodos de selección (truncamiento, muestreo universal estocástico y ruleta) para cada algoritmo.

Paso 6: Por medio del operador de cruce intermedio se generan N nuevos individuos (descendencias).

Paso 7: Por medio del operador de mutación, expuesto en la sección 2.7.1, se escogen algunos individuos aleatoriamente para generarles pequeños cambios.

Paso 8: Se seleccionan algunos individuos aleatoriamente de la población para ser cambiados por individuos del conjunto tentativo de Pareto.

Paso 9: Se repiten los pasos del 2 al 9 hasta completar el número de generaciones deseado.

El paso 2 se puede apreciar en la Figura 32 y describe la forma como se obtiene el índice de destreza global y la cantidad de puntos que forman el espacio de trabajo del robot para cada individuo de la población. Como se puede observar, el incremento de las variables de los actuadores (θ_{11} , θ_{12} y θ_{13}) influye significativamente en el tiempo que tarda esta parte del algoritmo en ejecutarse. Entre mayor sea este tiempo, mayor es el tiempo total que tardará cada algoritmo genético multiobjetivo en encontrar un conjunto de soluciones del problema.

Entre menor sea el incremento de las variables de los actuadores, mayor es la exactitud del índice de destreza global y el espacio de trabajo de cada individuo y mayor es el tiempo que tarda el algoritmo en encontrar estos dos valores. Como se aprecia en la Figura 33, evaluar un individuo en las dos funciones objetivo con un incremento de 2° tarda 45 segundos aproximadamente. Si cada población tiene 50 individuos, cada generación tardaría aproximadamente 45 minutos evaluando sus individuos en las funciones objetivo, lo que permite pensar que la solución de cada algoritmo puede tardar días. Este tiempo es elevado en vista que en total se busca implementar 18 algoritmos. Por esta razón, se usará un incremento de 10° para garantizar que la ejecución de los 18 algoritmos no tarde más de un día. Adicionalmente, el número de individuos de cada población será de 50 y el número de generaciones por algoritmo será de 100.

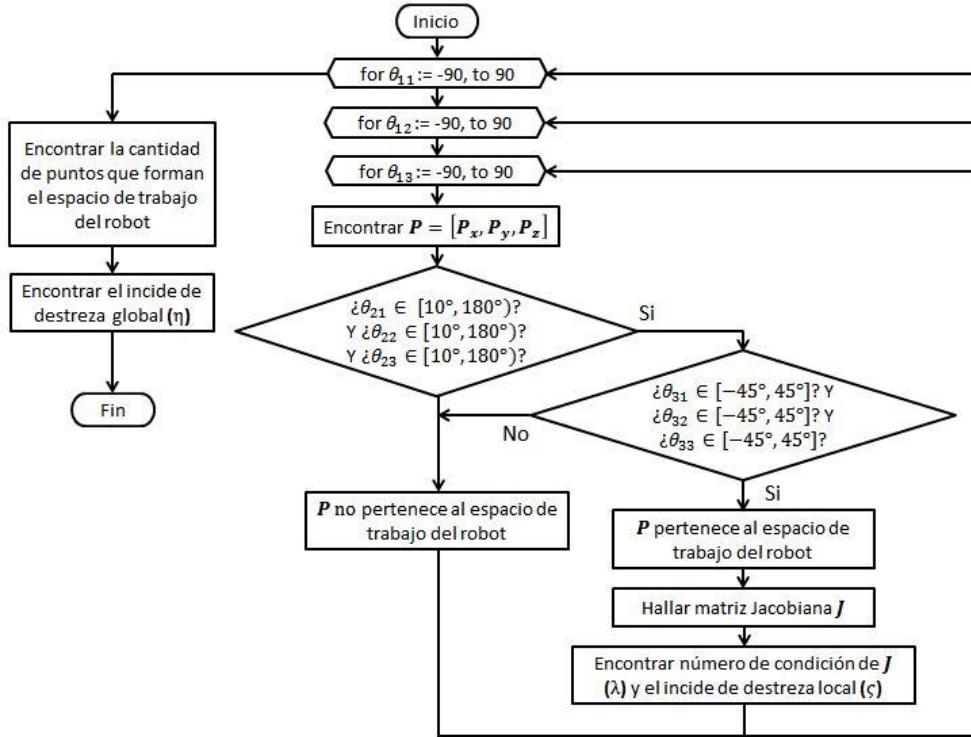


Figura 32. Diagrama de flujo de la evaluación de las funciones objetivo para un individuo.

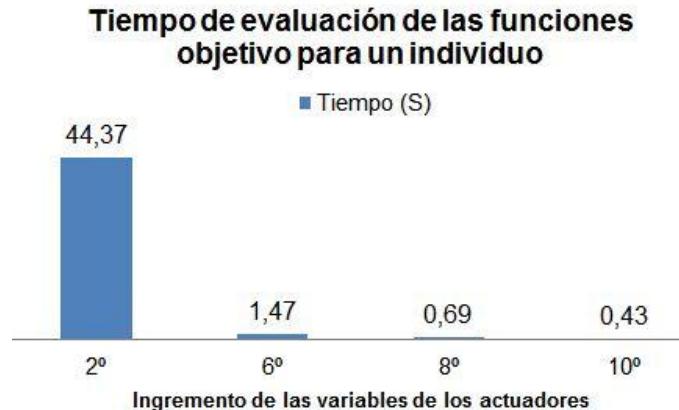


Figura 33. Tiempo de evaluación de las funciones objetivo para un individuo según el incremento de las variables de los actuadores.

4.5.2 Resultados del algoritmo de optimización

Una vez se ejecutaron todos los algoritmos, los resultados se agruparon según el algoritmo implementado, obteniendo seis conjuntos de posibles soluciones (Figuras 34 a 39).

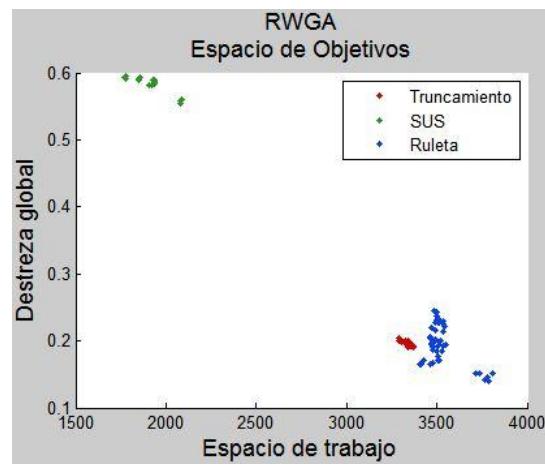


Figura 34. Conjunto de soluciones del algoritmo RWGA, MATLAB.

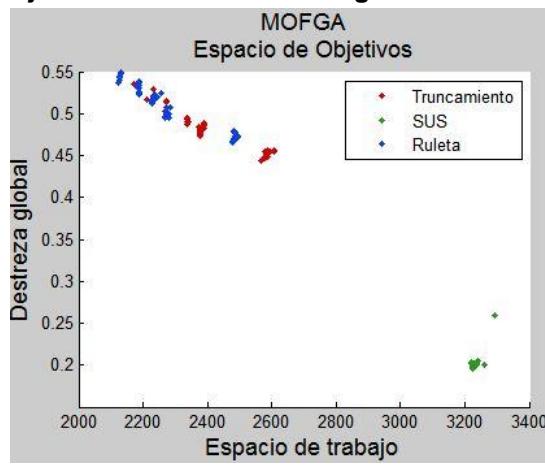


Figura 35. Conjunto de soluciones del algoritmo MOFGA, MATLAB.

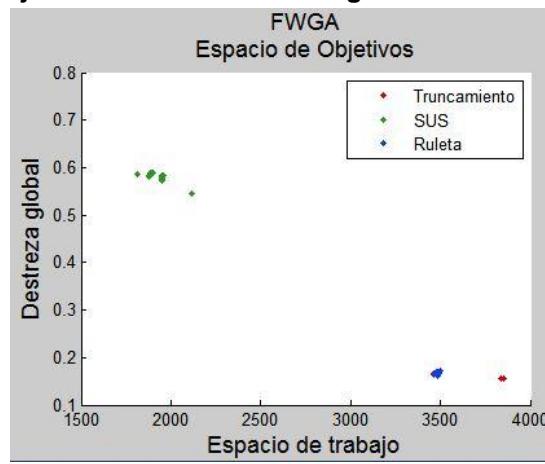


Figura 36. Conjunto de soluciones del algoritmo FWGA, MATLAB.

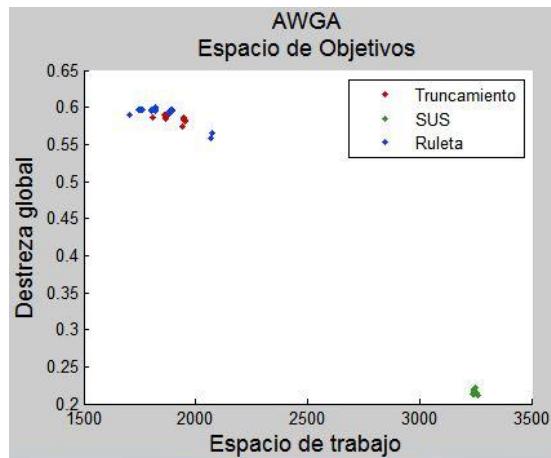


Figura 37. Conjunto de soluciones del algoritmo AWGA, MATLAB.

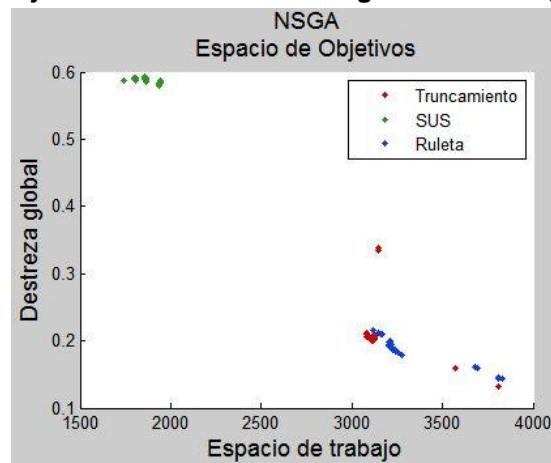


Figura 38. Conjunto de soluciones del algoritmo NSGA, MATLAB.

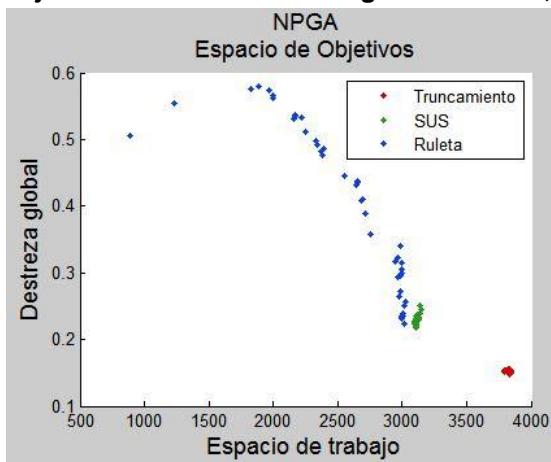


Figura 39. Conjunto de soluciones del algoritmo NPGA, MATLAB.

Posteriormente, se agruparon los resultados por el método de selección para obtener solo tres conjuntos de posibles soluciones.

La Figura 40 muestra el conjunto de soluciones de los algoritmos empleando el método de truncamiento como operador de selección.

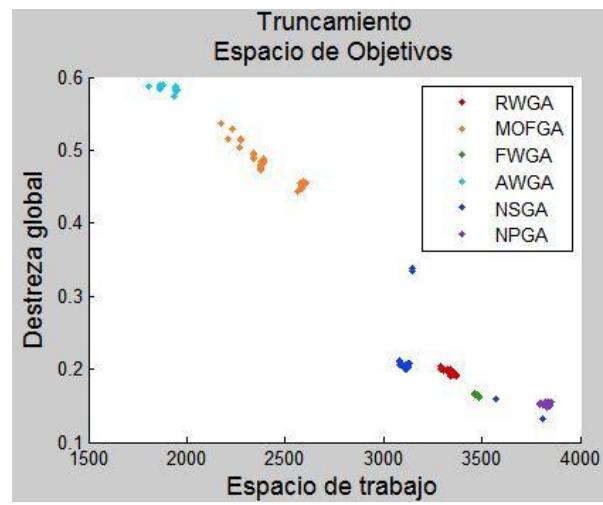
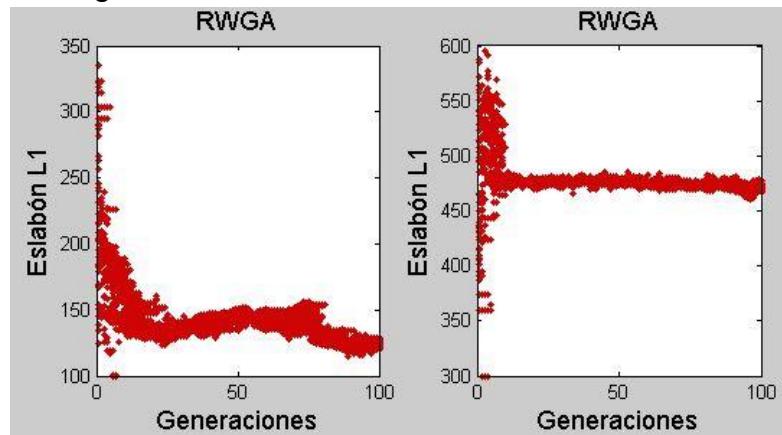
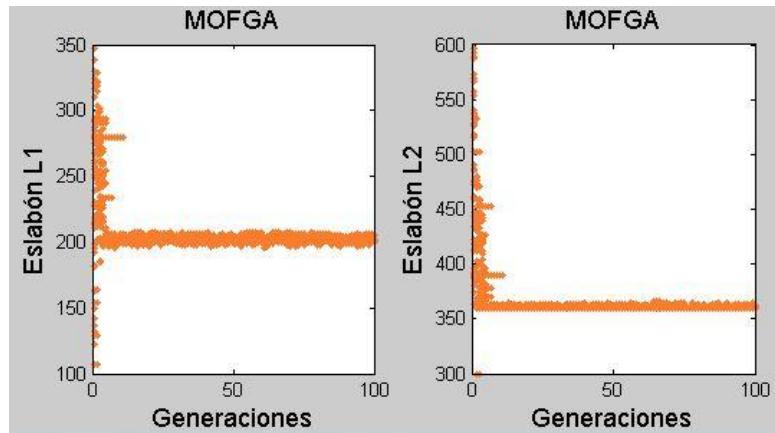


Figura 40. Conjunto de soluciones con el método de truncamiento, MATLAB.

Las Figuras 41 a 46 muestran la evolución de las dimensiones de los eslabones a través de las generaciones, empleando el método de truncamiento en cada algoritmo genético.



**Figura 41. Evolución de las dimensiones de los eslabones del robot: algoritmo RWGA
método de truncamiento, MATLAB.**



**Figura 42. Evolución de las dimensiones de los eslabones del robot: algoritmo MOFGA
método de truncamiento, MATLAB.**

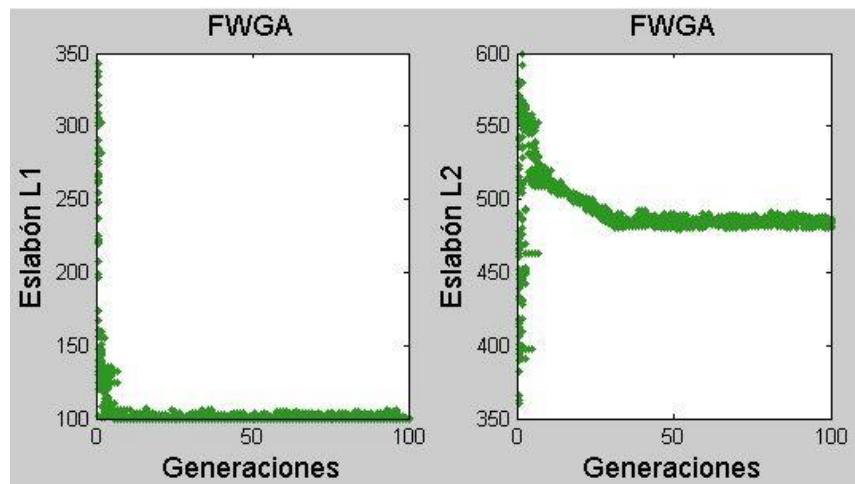


Figura 43. Evolución de las dimensiones de los eslabones del robot: algoritmo FWGA
método de truncamiento, MATLAB.

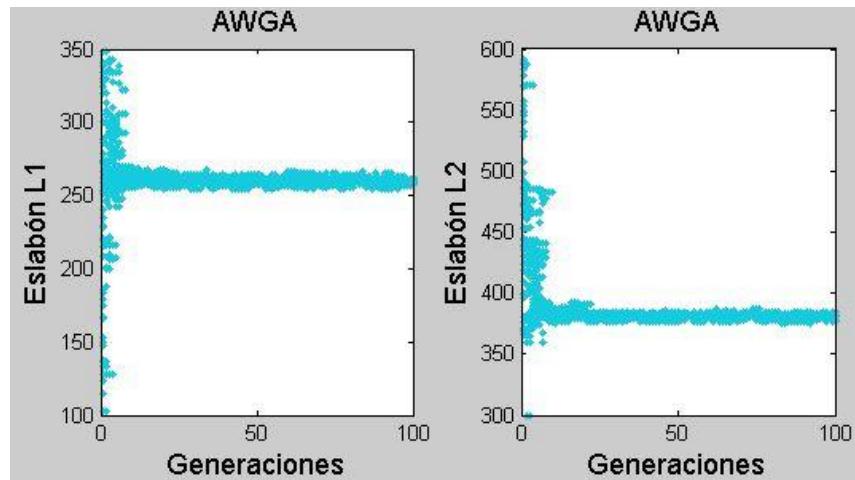


Figura 44. Evolución de las dimensiones de los eslabones del robot: algoritmo AWGA
método de truncamiento, MATLAB.

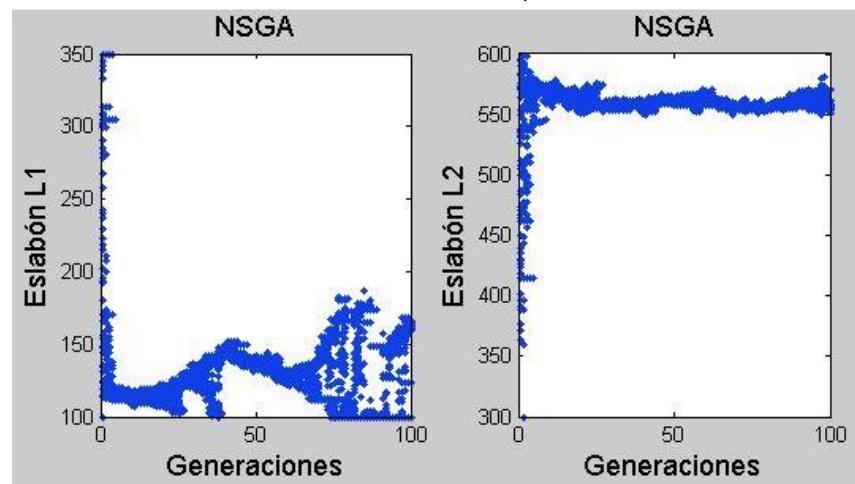


Figura 45. Evolución de las dimensiones de los eslabones del robot: algoritmo NSGA
método de truncamiento, MATLAB.

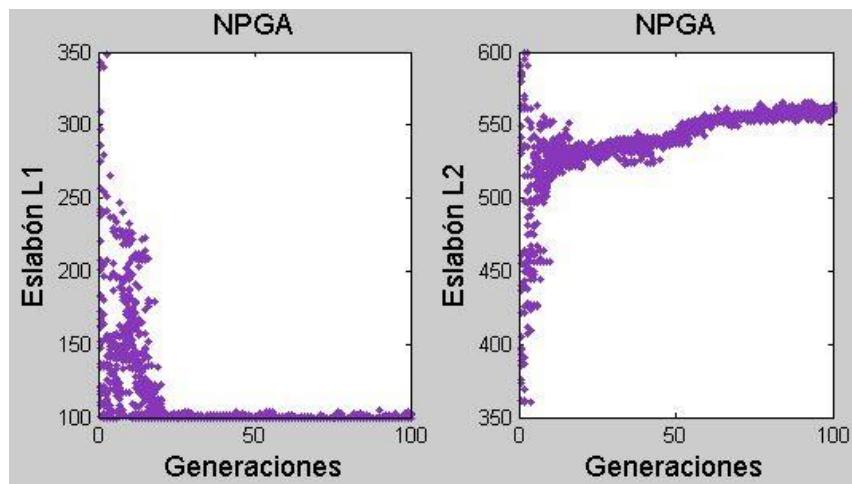


Figura 46. Evolución de las dimensiones de los eslabones del robot: algoritmo NPGA
método de truncamiento, MATLAB.

La Figura 47 muestra el conjunto de soluciones de los algoritmos empleando el método de muestreo universal estocástico como operador de selección.

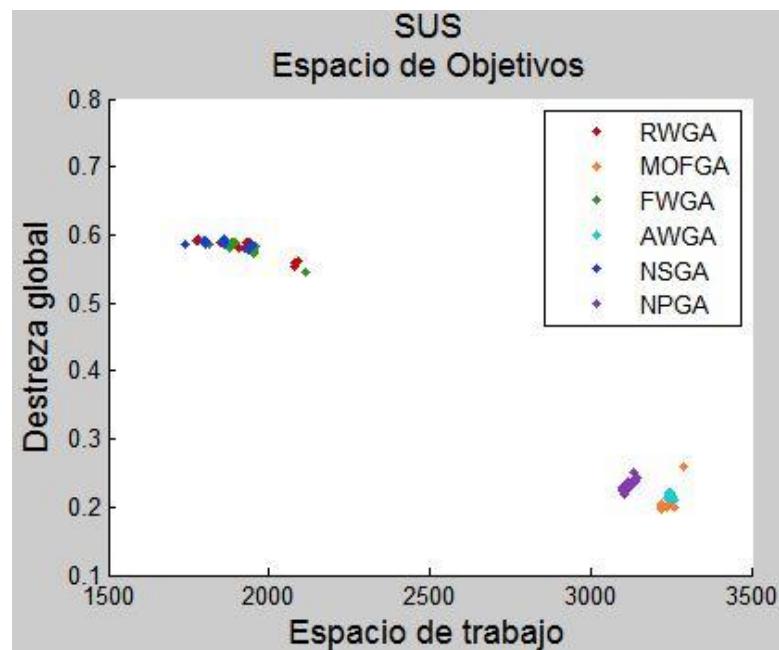
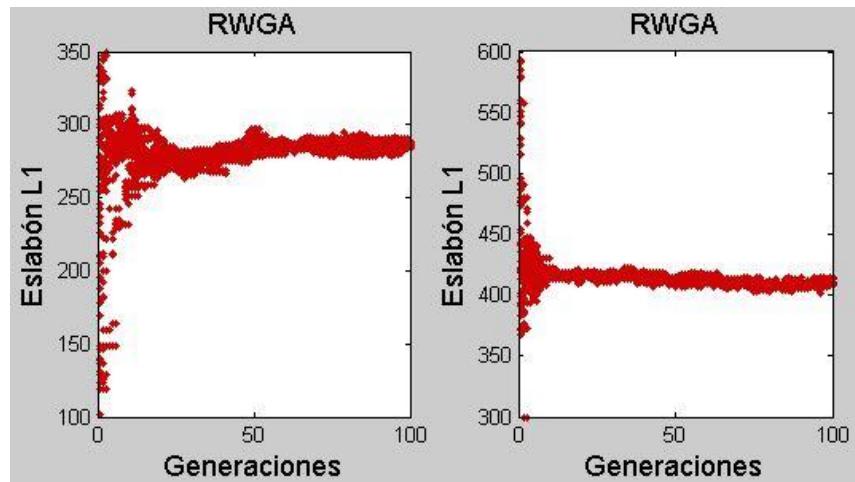
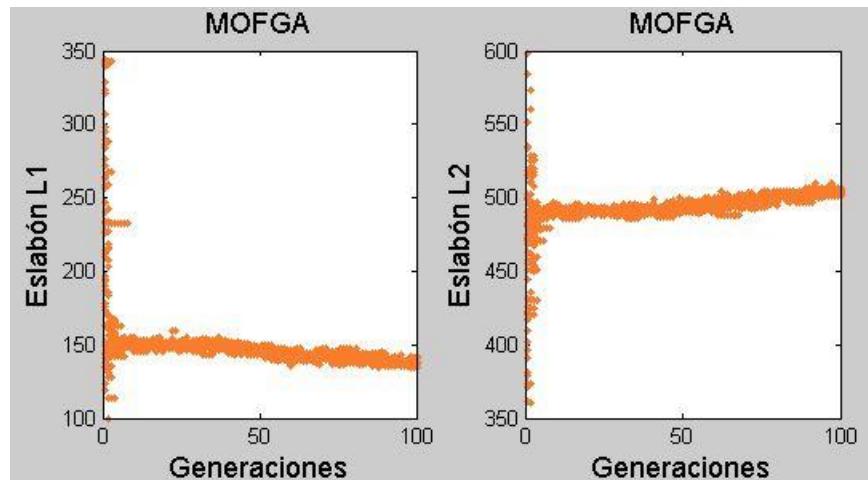


Figura 47. Conjunto de soluciones con el método SUS, MATLAB.

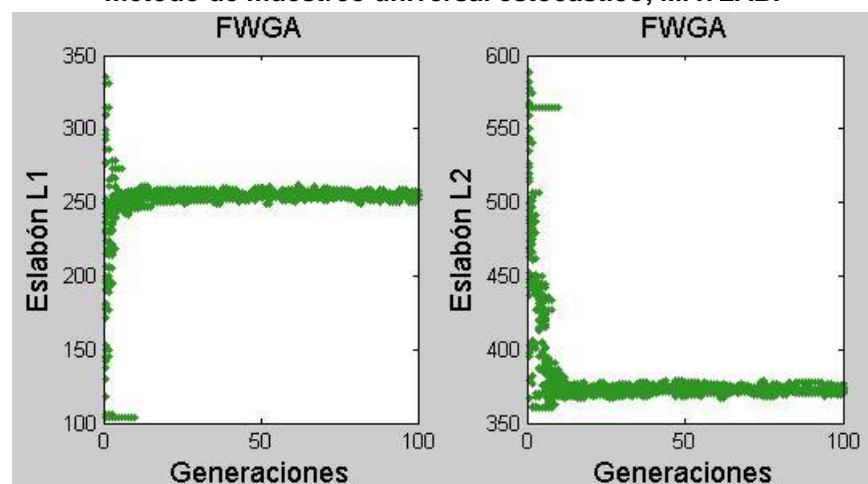
Las Figuras 48 a 53 muestran la evolución de las dimensiones de los eslabones a través de las generaciones, empleando el método de muestreo universal estocástico en cada algoritmo genético.



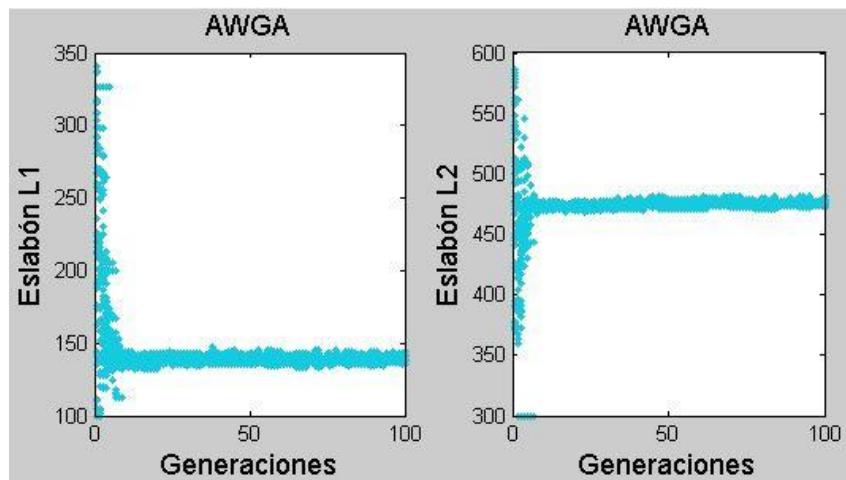
**Figura 48. Evolución de las dimensiones de los eslabones del robot: algoritmo RWGA
método de muestreo universal estocástico, MATLAB.**



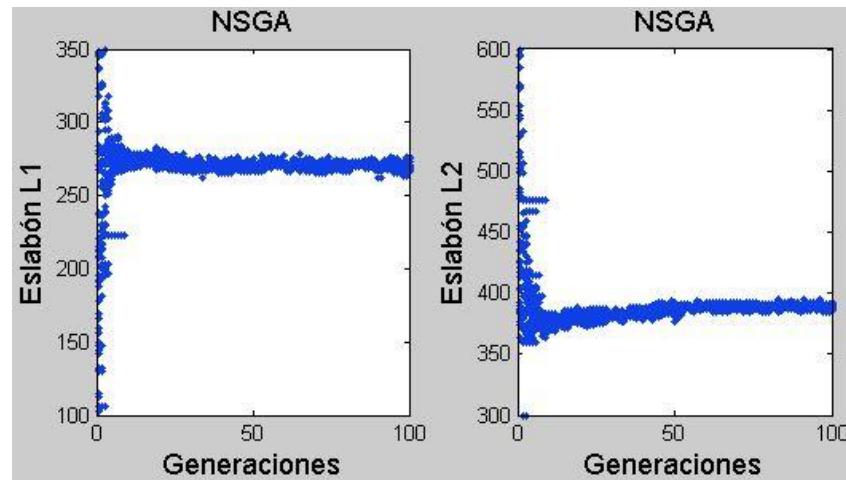
**Figura 49. Evolución de las dimensiones de los eslabones del robot: algoritmo MOFGA
método de muestreo universal estocástico, MATLAB.**



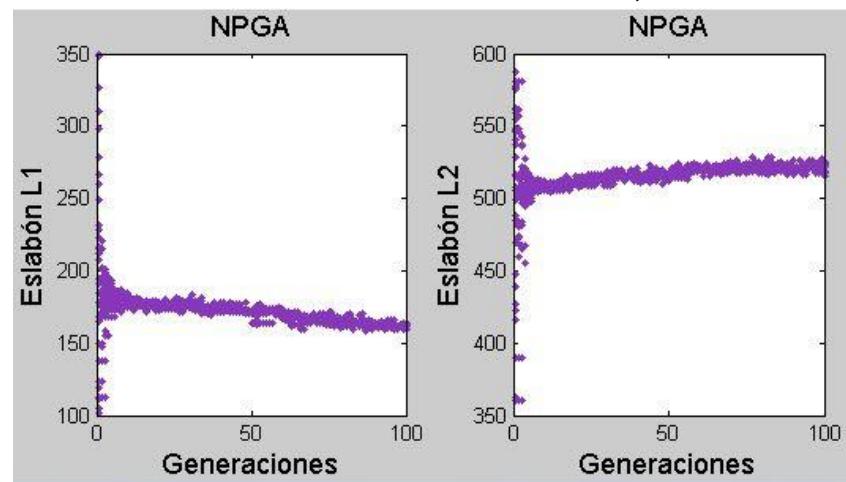
**Figura 50. Evolución de las dimensiones de los eslabones del robot: algoritmo FWGA
método de muestreo universal estocástico, MATLAB.**



**Figura 51. Evolución de las dimensiones de los eslabones del robot: algoritmo AWGA
método de muestreo universal estocástico, MATLAB.**



**Figura 52. Evolución de las dimensiones de los eslabones del robot: algoritmo NSGA
método de muestreo universal estocástico, MATLAB.**



**Figura 53. Evolución de las dimensiones de los eslabones del robot: algoritmo NPGA
método de muestreo universal estocástico, MATLAB.**

La Figura 54 muestra el conjunto de soluciones de los algoritmos empleando el método de la ruleta como operador de selección.

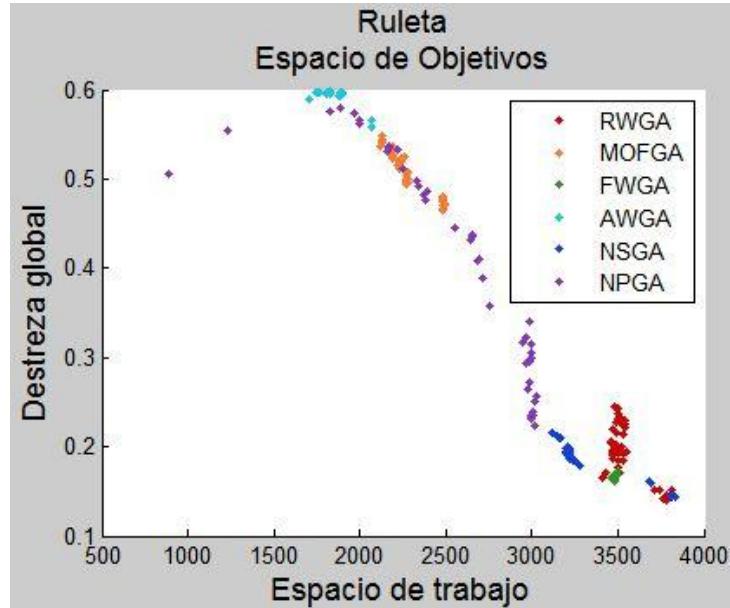


Figura 54. Conjunto de soluciones con el método de la ruleta, MATLAB.

Las Figuras 55 a 60 muestran la evolución de las dimensiones de los eslabones a través de las generaciones, empleando el método de la ruleta en cada algoritmo genético.

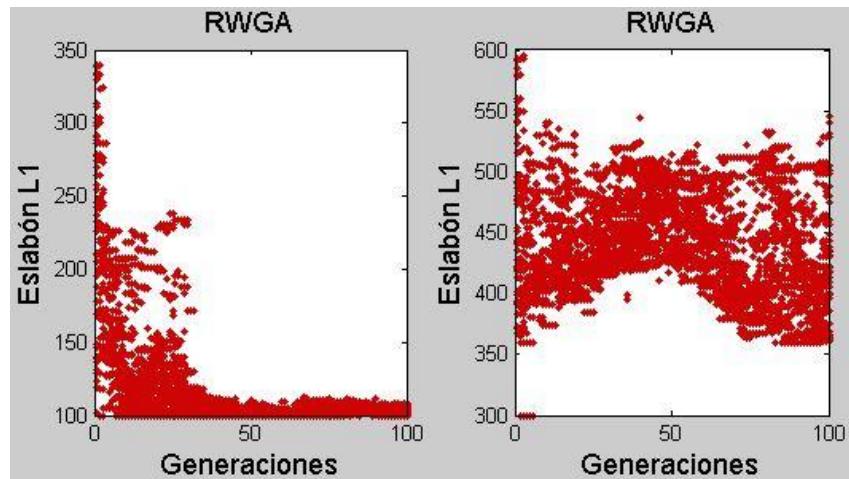


Figura 55. Evolución de las dimensiones de los eslabones del robot: algoritmo RWGA, método de la ruleta, MATLAB.

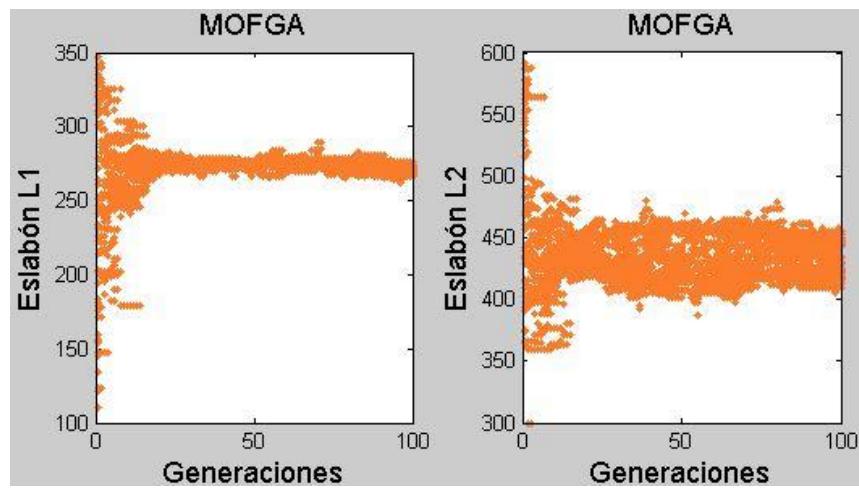


Figura 56. Evolución de las dimensiones de los eslabones del robot: algoritmo MOFGA método de la ruleta, MATLAB.

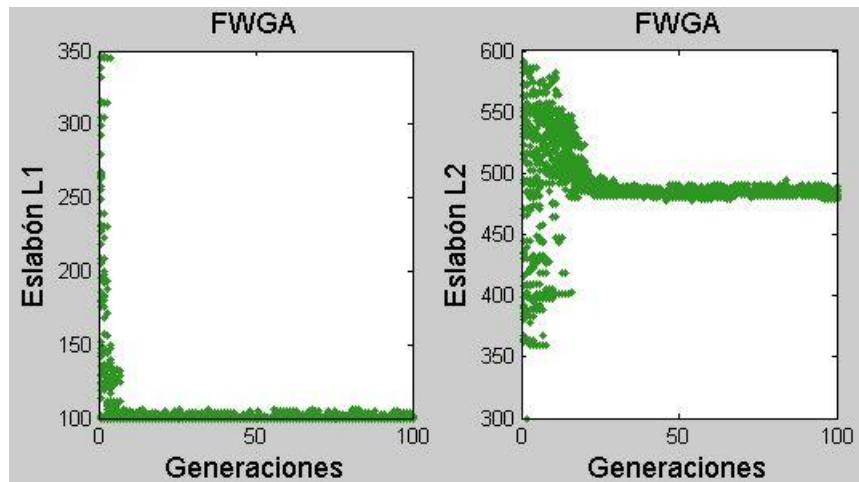


Figura 57. Evolución de las dimensiones de los eslabones del robot: algoritmo FWGA método de la ruleta, MATLAB.

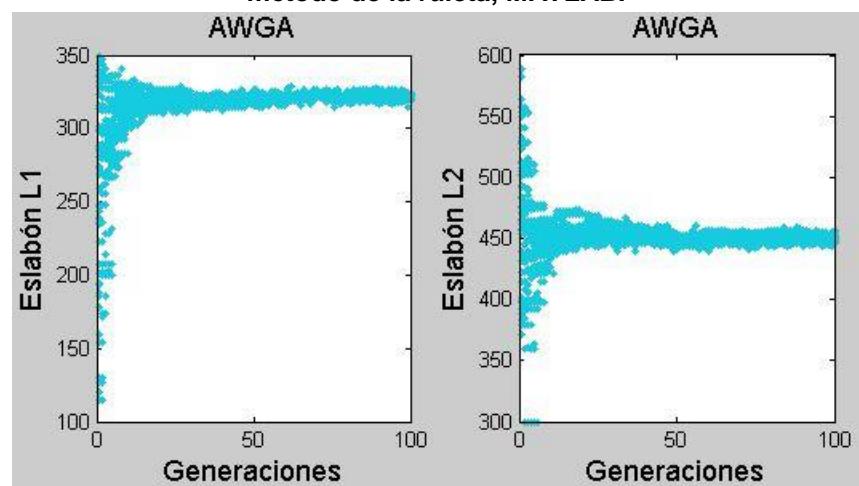


Figura 58. Evolución de las dimensiones de los eslabones del robot: algoritmo AWGA método de la ruleta, MATLAB.

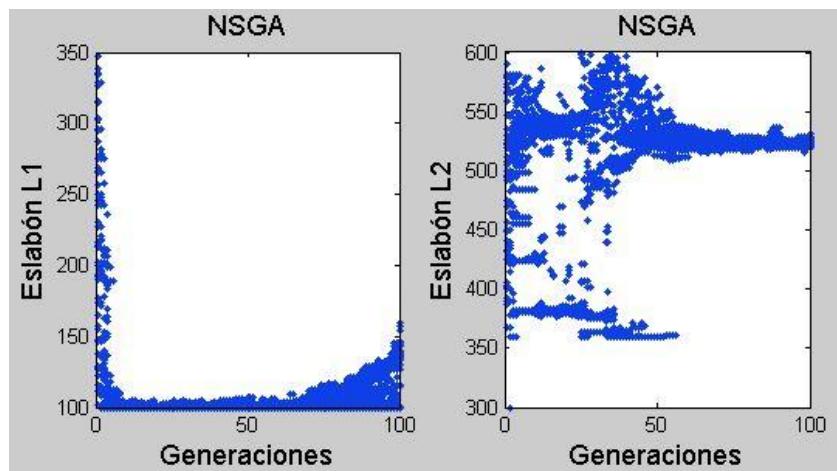


Figura 59. Evolución de las dimensiones de los eslabones del robot: algoritmo NSGA
método de la ruleta, MATLAB.

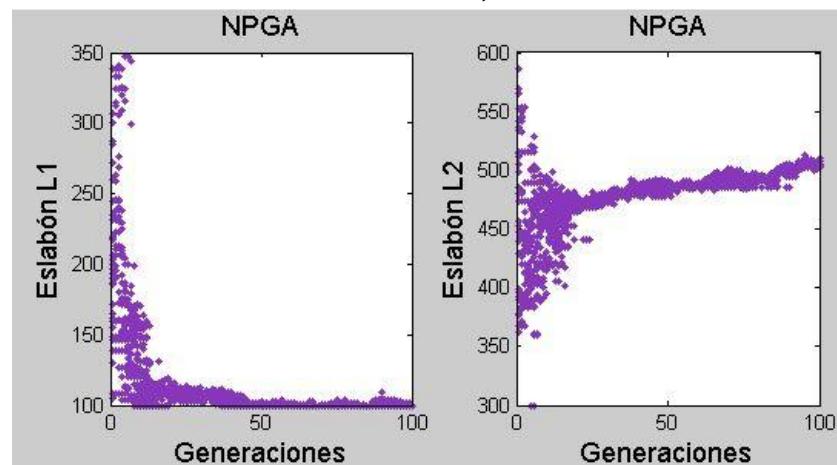


Figura 60. Evolución de las dimensiones de los eslabones del robot: algoritmo NPGA
método de la ruleta, MATLAB.

La Figura 61 muestra el tiempo que tarda cada algoritmo en encontrar un conjunto de soluciones al problema a través de las 100 generaciones.

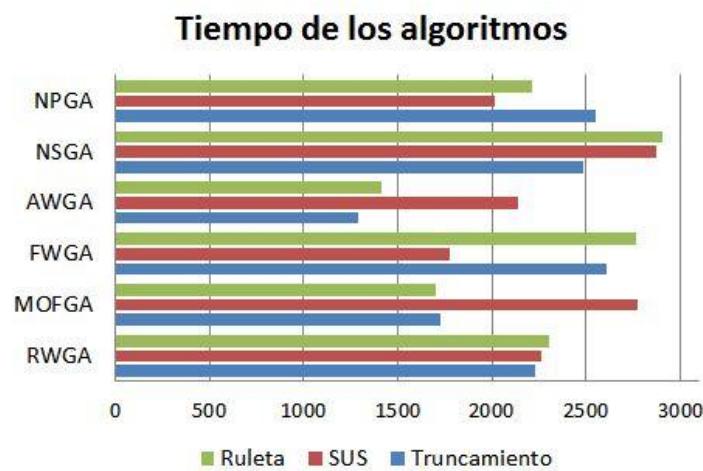


Figura 61. Tiempo en segundos la de ejecución de los algoritmos, EXCEL.

Las Figuras 40, 47 y 54 muestran un conjunto de solución del problema, según el método de selección empleado. Para encontrar el conjunto final de soluciones, se unieron estos tres conjuntos en uno solo y se identificó su frontera óptima de Pareto (soluciones no dominadas), como se aprecia en la Figura 62.

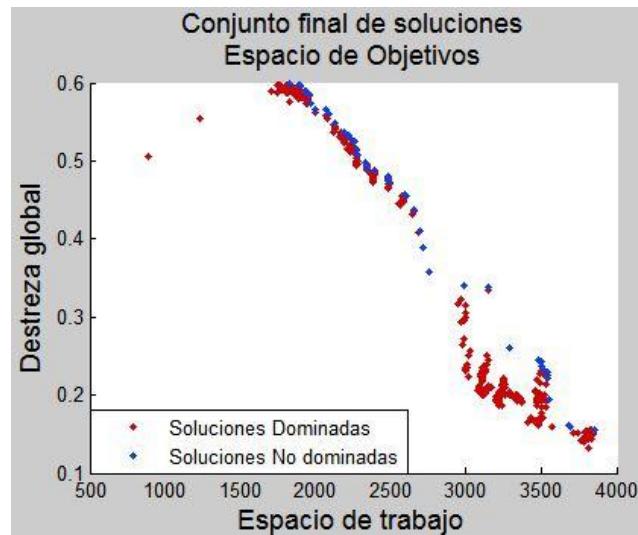


Figura 62. Frontera de Pareto del conjunto final de soluciones, MATLAB.

De esa frontera óptima de Pareto, los individuos con la mayor destreza global (mayor a 0,47) y la mayor cantidad de puntos en el espacio de trabajo (más de 2000 puntos) son los de interés para obtener las dimensiones de los eslabones del robot Delta. Para tomar una decisión, se evaluaron de nuevo las funciones objetivo de cada uno de estos individuos disminuyendo el incremento de las variables de los actuadores de 10° a 2° . Analizando cada uno de los individuos y sus nuevos valores de funciones objetivo, las dimensiones de los eslabones del robot son:

- $L_1 = 260 \text{ mm}$
- $L_2 = 420 \text{ mm}$

Las Figuras 63 a 66 muestran las diferentes vistas del espacio de trabajo real del robot Delta, con las dimensiones de los eslabones encontradas. Los datos de dicho espacio de trabajo son:

- Puntos en el espacio: 259 656
- Destreza global: 0,556
- Alcance mínimo: $X = -342,8015 \text{ mm}$, $Y = -296,9611 \text{ mm}$ y $Z = 105,7695 \text{ mm}$
- Alcance máximo: $X = 342,4004 \text{ mm}$, $Y = 296,9611 \text{ mm}$ y $Z = 677,0132 \text{ mm}$

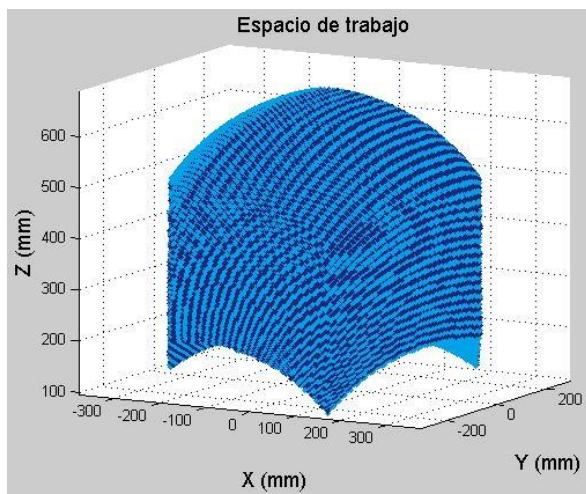


Figura 63. Espacio de trabajo del robot Delta, MATLAB.

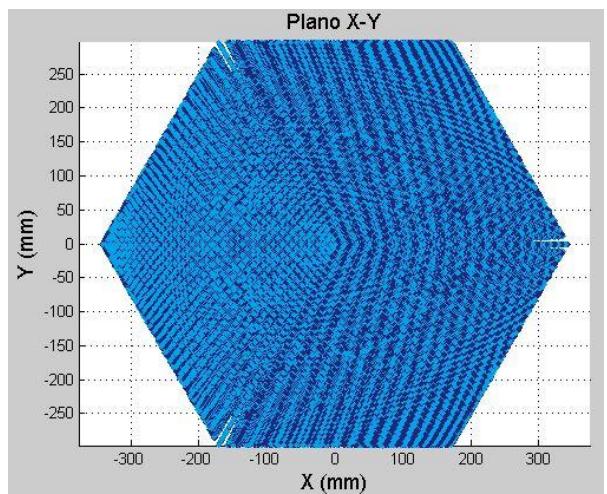


Figura 64. Vista superior del espacio de trabajo del robot, MATLAB.

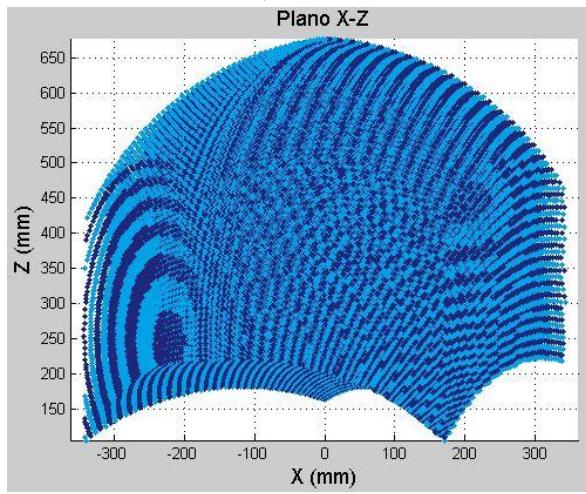


Figura 65. Vista del espacio de trabajo del robot en plano XZ, MATLAB.

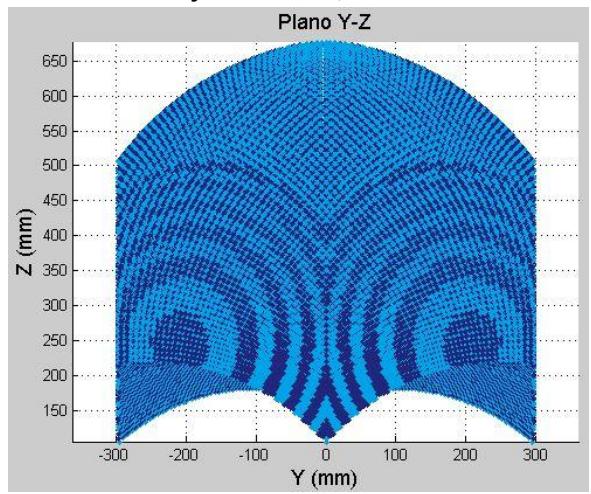


Figura 66. Vista del espacio de trabajo del robot en plano YZ, MATLAB.

CAPÍTULO 5: DISEÑO E IMPLEMENTACIÓN DEL ROBOT DELTA

La etapa de diseño e implementación del robot delta se puede resumir en las siguientes fases:

- Fase 1: Modelo matemático del robot
 - ✓ Cinemática del robot.
 - ✓ Optimización del espacio de trabajo (dimensión de los eslabones).
- Fase 2: Diseño mecánico
 - ✓ Diseño de cada componente del robot (CAD).
 - ✓ Selección de materiales.
 - ✓ Análisis del robot por elementos finitos.
- Fase 3: Diseño electrónico
 - ✓ Selección de los actuadores del robot.
 - ✓ Diseño del circuito electrónico del robot.
- Fase 4: Implementación del robot
 - ✓ Construcción del prototipo (acople mecánico y electrónico).
- Fase 5: Interfaz gráfica de usuario.

La fase 1 se llevó a cabo en el capítulo anterior. A continuación se desarrollan las demás fases de implementación del robot. En vista que este trabajo de grado es una continuación de un proyecto de iniciación científica (PIC) de la universidad, en la siguiente sección se muestra el rediseño mecánico de todos los componentes del robot.

5.1 DISEÑO MECÁNICO

5.1.1 Bases fija y móvil

La base fija (Figura 67) está compuesta por dos tapas en forma de triángulos equiláteros de 40 cm de longitud con un espesor de 4 mm, cuya función es proteger y dar estructura mecánica al robot. Adicionalmente, cuenta con tres soportes en donde se fijan los tres actuadores (Figura 68) que permiten posicionar la plataforma móvil en el espacio.

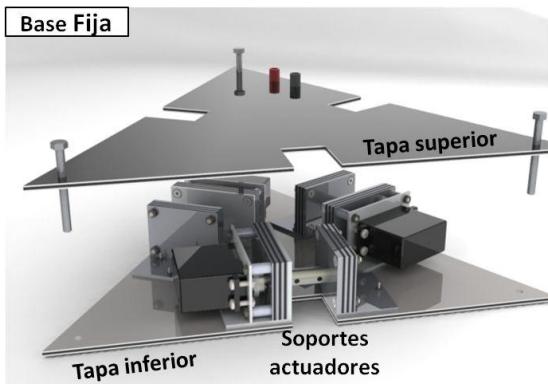


Figura 67. Base fija del robot, CAD SolidWorks.

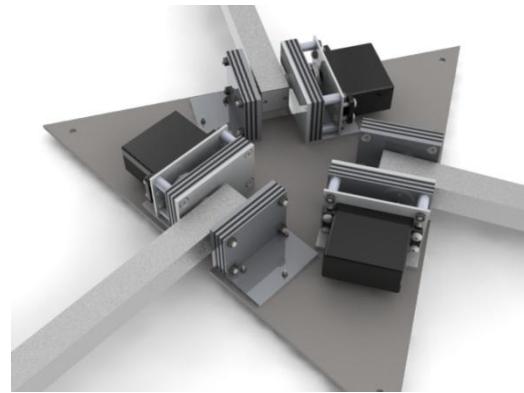


Figura 68. Vista superior de la base fija, CAD SolidWorks.

El soporte de los actuadores está compuesto por dos soportes para rodamientos, dos soportes fijos, dos rodamientos, un eje principal, un servomotor, una platina, cuatro topes y un acople para la galleta del servomotor (Figura 69). Este conjunto tiene como función soportar al eje que transmite la energía mecánica del servomotor al eslabón.

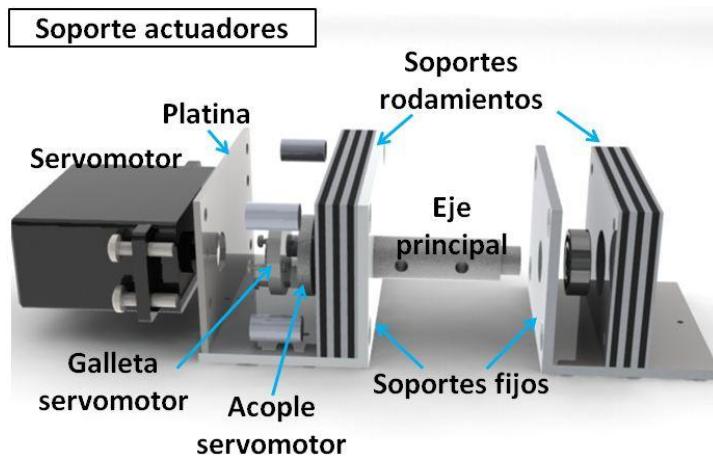


Figura 69. Explosiónado del soporte de los actuadores de la base fija, CAD SolidWorks.

Al igual que la base fija, la base móvil (Figura 70) consta de dos tapas en forma de triángulo equilátero de 11 cm de longitud que tienen como función sujetar los tres soportes que permiten unir esta base a los segundos eslabones de las tres cadenas cinemáticas (Figura 71). Adicionalmente, la tapa inferior soporta el servomotor que le da el 4 DOF al robot (orientación).

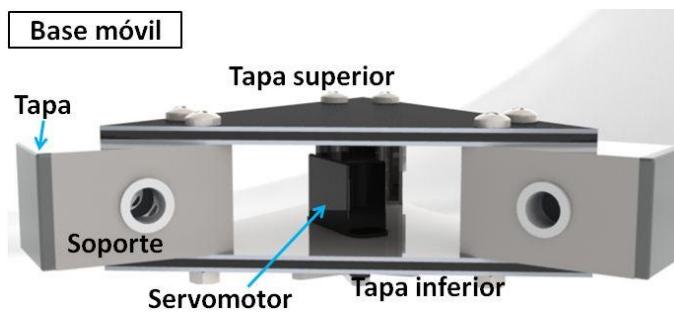


Figura 70. Base móvil, CAD SolidWorks.

Como se aprecia en la Figura 72, cada soporte consta de un eje, dos prisioneros, dos empaques, dos juntas universales y una tapa. La finalidad de los empaques es reducir la fricción del movimiento del eje. Los dos prisioneros con forma de anillo mantienen el eje centrado, es decir, evitan que el eje se tenga desplazamientos axiales.



Figura 71. Soportes de la base móvil, CAD SolidWorks.

Soporte de base la móvil

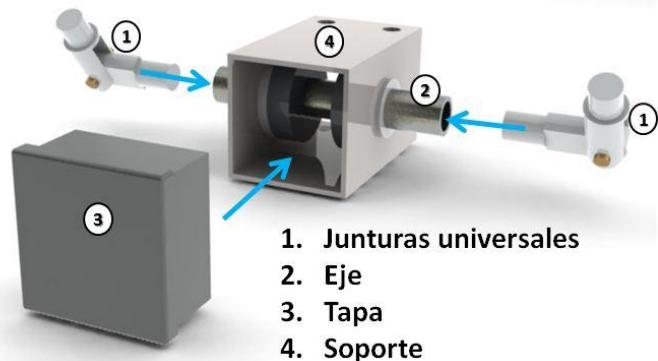


Figura 72. Explosión de un soporte de la base móvil, CAD SolidWorks.

5.1.2 Eslabones

Cada cadena cinemática está compuesta de un prisionero y dos eslabones (Figura 73) que se unen por medio de juntas universales, las cuales se tratarán en detalle en la siguiente sección.

El primer eslabón es un perfil hueco cuadrado de 1" x 1", con una longitud de 29 cm y dos perforaciones de $\frac{1}{2}$ " distanciadas 25 cm. En un extremo va un prisionero para unir el eslabón con el soporte del actuador (Figura 74) y en el otro extremo se conecta con el segundo eslabón.

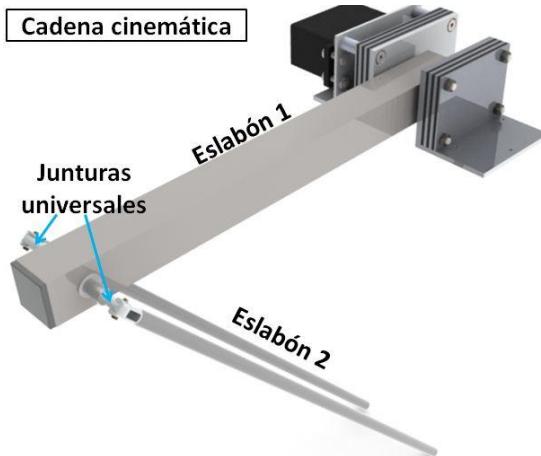


Figura 73. Una cadena cinemática del robot, CAD SolidWorks.

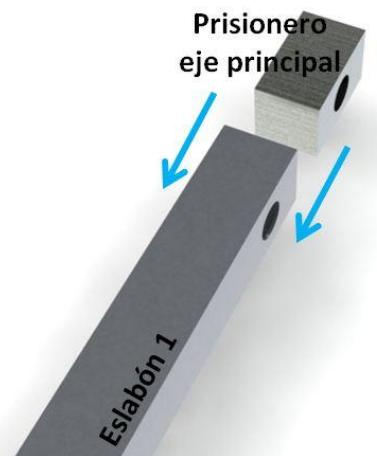


Figura 74. Primer eslabón de la cadena cinemática, CAD SolidWorks.

El segundo eslabón consta de dos tubos huecos de 8 mm de diámetro y 39,8 cm de longitud. Este eslabón, permite unir el primer eslabón con base móvil.

5.1.3 Juntas universales

La función de las juntas universales es unir dos ejes no alineados y transmitir potencia entre ellos. El robot necesita las juntas para transmitir el movimiento desde los actuadores hasta la base móvil. El sistema de juntas

consta de un eje, dos anillos prisioneros, dos empaques y dos juegos de junturas (Figura 75).

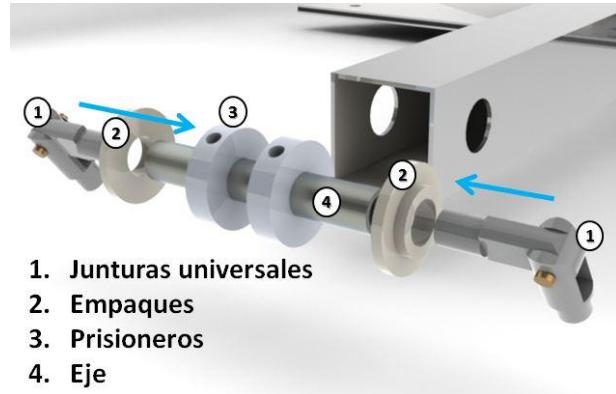


Figura 75. Explosión de las junturas en el primer eslabón, CAD SolidWorks.

Al igual que en el soporte de la base móvil, los anillos prisioneros evitan que el eje tenga desplazamientos axiales y los dos empaques reducen la fricción de movimiento del eje con el eslabón.

Como se aprecia en la Figura 76, una juntura universal consta de tres partes: un pasador y las piezas J1 y J2.



Figura 76. Explosión de las junturas universales, CAD SolidWorks.

5.1.4 Estructura soporte

Es una estructura hexagonal (Figura 77) la cual encierra todo el espacio de trabajo del robot, permitiendo que el efecto final se pueda mover sin chocar con el suelo o un objeto a su alrededor. Se compone de tres partes, el hexágono inferior el cual es la base de la estructura, el hexágono superior el cual soporta el robot, y los tubulares, encargados de distanciar estos 2 hexágonos (figura 86).

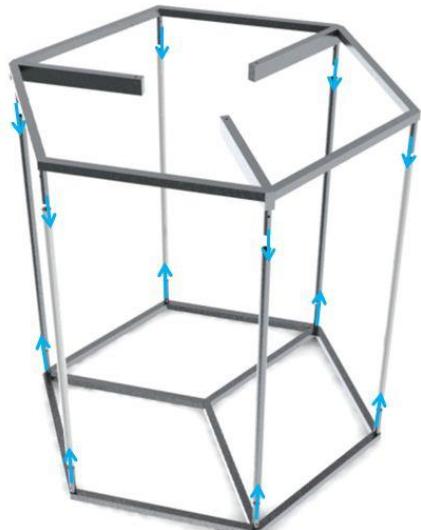


Figura 77. Estructura soporte, CAD SolidWorks.

En los anexos en el manual del usuario del robot se encuentra toda la información a cerca de como ensamblar el robot.

5.1.5 Selección de materiales

Es importante que la estructura del manipulador sea rígida y ligera a la vez. Por esta razón el material seleccionado para su construcción fue el **aluminio** ya que es el más ligero de los metales, por su bajo peso específico. Este material es el más utilizado en la industria debido a sus propiedades mecánica:

- Resistencia a la humedad, al agua, al sol, a la abrasión, a la corrosión y al desgaste.
- Peso específico bajo lo que lo hace un metal ligero.
- Dúctil y maleable.
- Se deja mecanizar con facilidad por ser un metal blando, se corta y raya con suma facilidad.
- Es un material que protege el medio ambiente.
- No es tóxico.
- En caso de incendio no ocasiona la emisión de sustancias nocivas y no es inflamable.
- Bajo mantenimiento.

Los Elementos en Aluminio son los eslabones, las juntas, los soportes de los ejes y los conectores de los eslabones con la base móvil.

El **aluminio compuesto** es un panel compuesto por dos láminas de aluminio y con un núcleo central de polietileno con un espesor total de 2 a 6 mm. Su pintura es de alta durabilidad fabricada para intemperie con garantías

superiores a los 10 años y muy resistente a los agentes químicos y atmosféricos⁶.

Los soportes de los rodamientos y las tapas de las bases fija y móvil son elementos en aluminio compuesto.

El **acero cold rolled**, por su bajo contenido de carbono, permite buena maleabilidad y soldabilidad, es apto para hacer todo tipo de estructuras metálicas livianas. Su terminado mate u opaco facilita todos los procesos de pintura, cromado y galvanizado. La estructura del robot Delta está hecha de acero cold rolled.

5.1.6 Diseño asistido por computadora (CAD)

La Figura 78 muestra el diseño asistido por computadora del robot. Como se mencionó anteriormente, el robot cuenta con una base fija, tres cadenas cinemáticas don dos eslabones cada una, unidos por medio de juntas universales, una base móvil y una estructura soporte (Figura 79).



Figura 78. Robot Delta, CAD SolidWorks.

⁶ <http://www.alpanel.com/productos/>

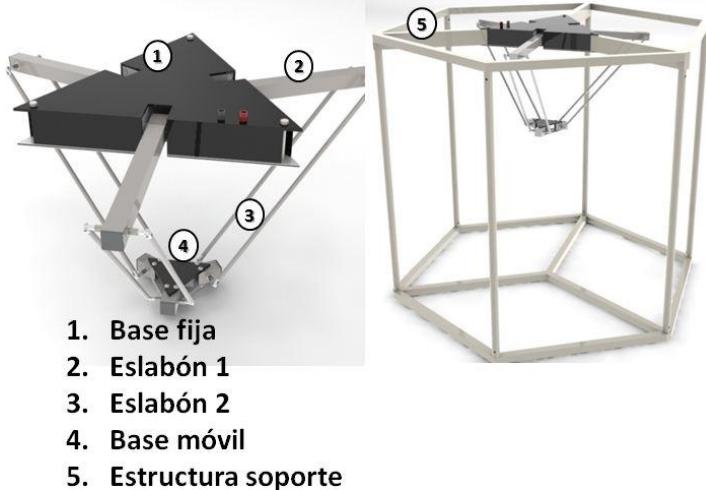


Figura 79. Componentes del Robot Delta, CAD SolidWorks.

5.1.7 Resultados de Análisis por elementos finitos

El análisis estructural del robot es un problema que se representa por medio de ecuaciones diferenciales. La solución de estas ecuaciones analíticamente es compleja y por lo que se recurre a los métodos numéricos, como el análisis de elementos finitos, resolviendo las ecuaciones diferenciales iterativamente. La solución obtenida del problema es aproximada, pero eso no indica que no sea una solución aceptable.

Para poder dar solución a un problema físico por el método de elementos finitos es necesario: definir la geometría del problema, asignar las propiedades físicas de cada componente que conforman la estructura, especificar condiciones de frontera, aplicar las cargas, realizar el mallado de la geometría, resolver el problema y analizar los resultados obtenidos.

El análisis de elementos finitos del robot se hizo por medio de *SolidWorks Simulation*, analizando cada componente del robot diseñado antes de la etapa de construcción.

El robot no está diseñado para levantar cargas, ya que el objetivo del robot es orientarse y ubicarse en el espacio, pero eso no quiere decir que no pueda hacerlo, el diseño mecánico del robot permite levantar cargas, por lo tanto el análisis de elementos finitos se hará únicamente en las piezas que soportan la mayor carga. Las cuales son los ejes principales y el primer eslabón L1.

El primer estudio estático se realizó en el eje principal que soporta el primer eslabón de cada cadena cinemática. Tiene una longitud total de 65 mm, la parte central tiene una sección de 15 mm de diámetro, y en los extremos 6 mm de diámetro. Para realizar su análisis primero se define el material, se seleccionan las sujeteciones, se aplican las diferentes cargas, se realiza el mallado y por último se ejecutan los diferentes análisis de elementos finitos.

Las sujeteciones del eje, son las caras cilíndricas de 6mm de diámetro, en las cuales el eje se soporta sobre 2 rodamientos, uno en cada extremo, se eligió

una fuerza de 15 N aplicada en el mismo sentido de la gravedad, la cual también fue incluida para obtener un resultado más preciso. Como se observa en la Figura 80, las flechas verdes representan la sujeción, la flecha roja representa la gravedad y la flecha morada representa la fuerza aplicada.

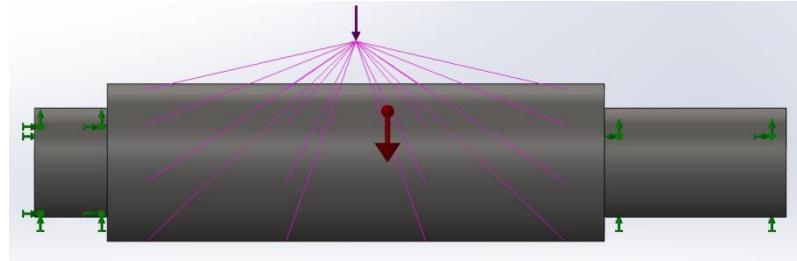


Figura 80. Fuerzas y reacciones del eje principal, SolidWorks.

La malla del eje es una malla sólida basada en curvatura, con tamaños de los elementos de 0,947 mm, con una calidad de malla de alto orden. En la figura 81 se describen todas las características de la malla y se aprecia el eje mallado.

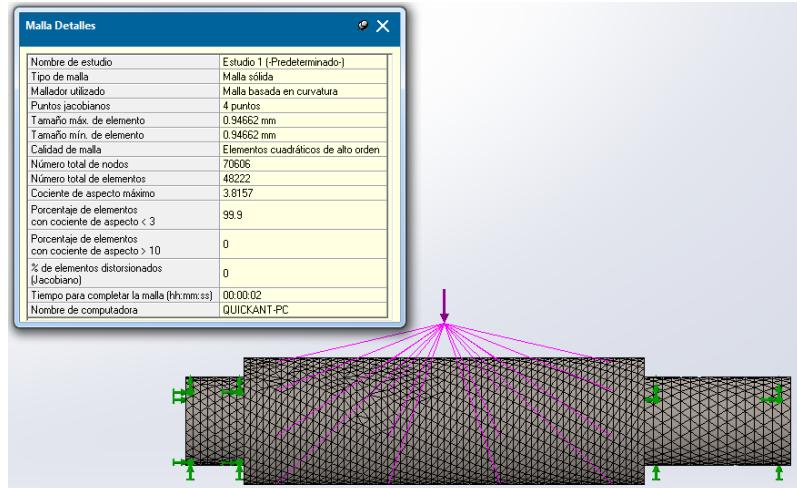


Figura 81. Mallado del eje principal, SolidWorks.

A continuación se muestran los análisis de elementos finitos: deformaciones unitarias (Figura 82), desplazamientos (Figura 83) y tensiones (Figura 84).

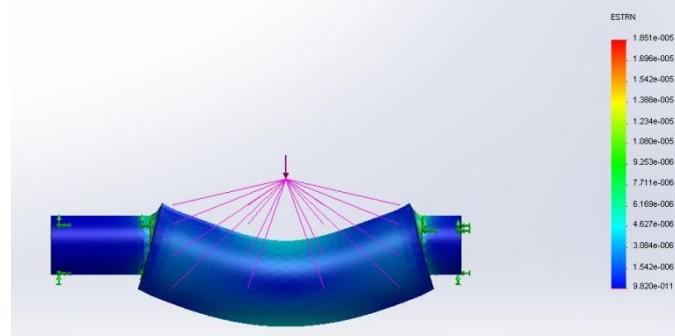


Figura 82. Deformaciones en el eje principal, SolidWorks.

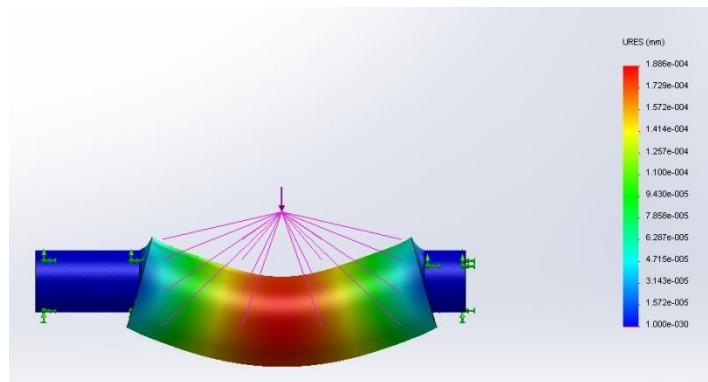


Figura 83. Desplazamientos en el eje principal, SolidWorks.

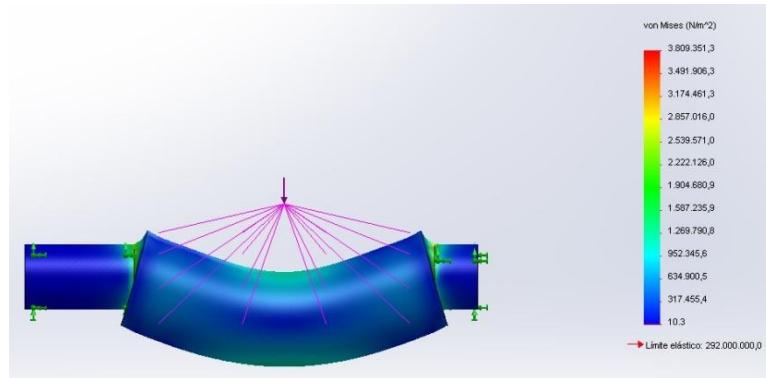


Figura 84. Tensiones en el eje principal, SolidWorks.

El segundo estudio estático se realizó en el primer eslabón que permite transmitir el movimiento del actuador al segundo eslabón y a la plataforma móvil. Este eslabón es un perfil de aluminio hueco de 1" x 1" de 288 mm de longitud. Para realizar su análisis primero se define el material, se seleccionan las sujeciones, se aplican las diferentes cargas, se realiza el mallado y por último se ejecutan los diferentes análisis de elementos finitos.

La sujeción del eslabón es el eje principal. La fuerza aplicada es de 15 N y está aplicada al otro extremo del eslabón en el mismo sentido de la gravedad, la cual también fue incluida para obtener un resultado más preciso. Como se observa en la Figura 85, las flechas verdes representan la sujeción, la flecha roja representa la gravedad y la flecha morada representa la fuerza aplicada.

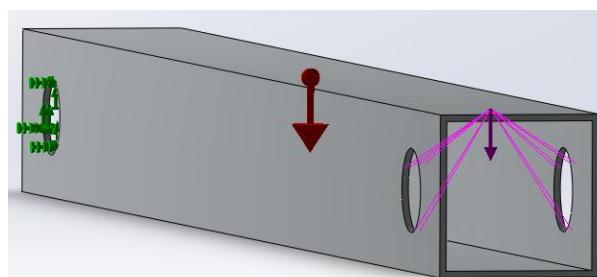


Figura 85. Fuerzas y reacciones del primer eslabón, SolidWorks.

La malla del eslabón es una malla sólida basada en curvatura, con tamaños de los elementos entre 1,2 mm y 3,7 mm, con una calidad de malla de alto orden. En la figura 86 se describen todas las características de la malla y se aprecia el eslabón mallado.

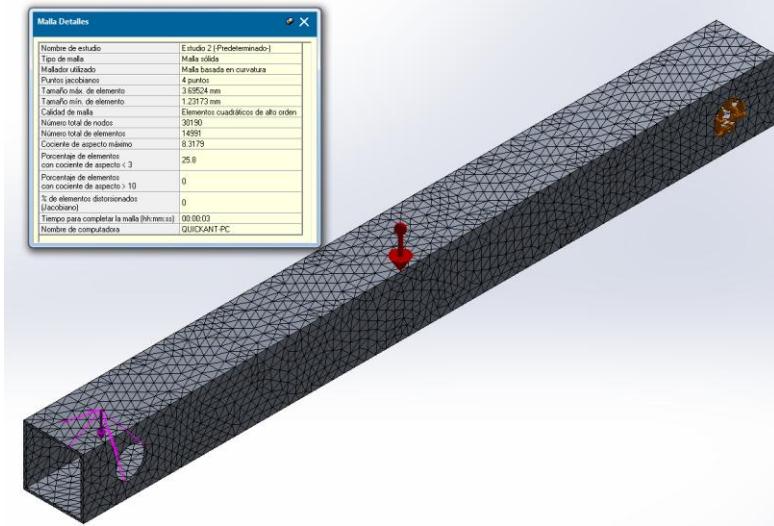


Figura 86. Mallado del primer eslabón, SolidWorks.

A continuación se muestran los análisis de elementos finitos: deformaciones unitarias (Figura 87), desplazamientos (Figura 88) y tensiones (Figura 89).

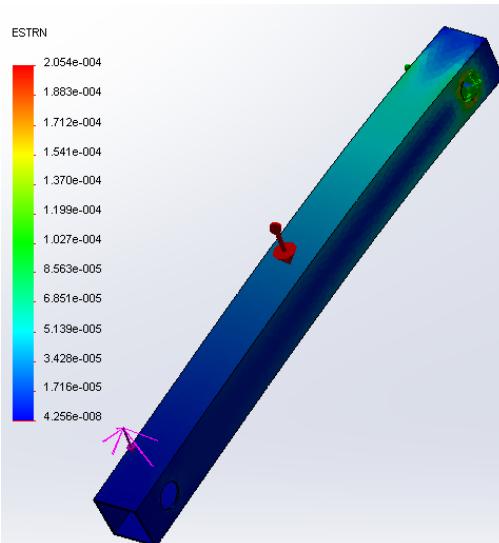


Figura 87. Deformaciones del eslabón, SolidWorks.

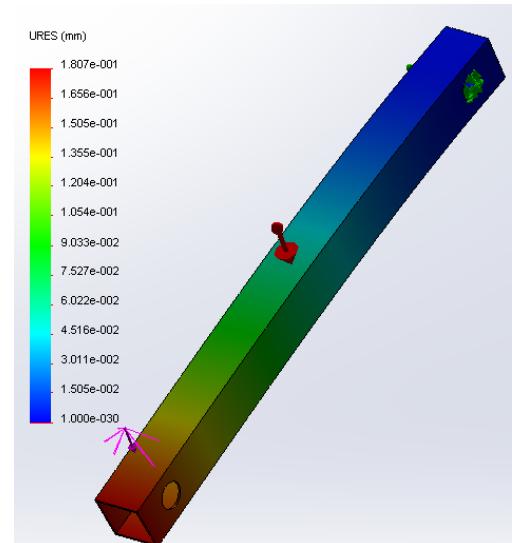


Figura 88. Desplazamientos del eslabón, SolidWorks.

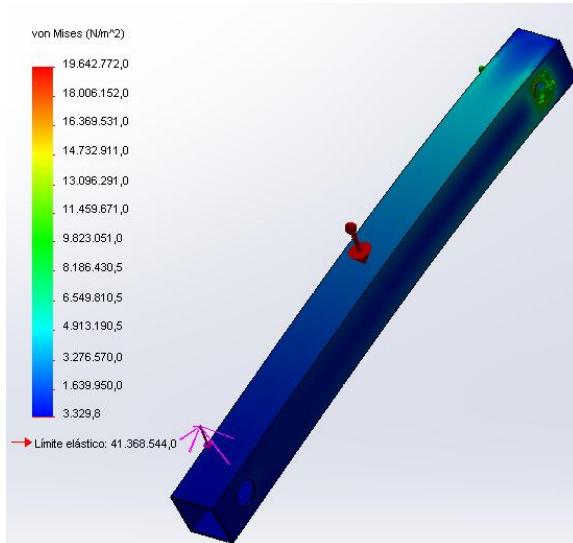


Figura 89. Tensiones del eslabón, SolidWorks.

5.1.8 Implementación de la estructura mecánica del robot

A continuación se hace una comparación entre el prototipo implementado para el PIC y el prototipo final de este trabajo de grado. En el primer prototipo, el primer eslabón estaba acoplado directamente al servomotor teniendo solo un punto de apoyo (Figura 90), lo que generaba inestabilidad y errores de posición en la plataforma móvil. Por esta razón, se diseñó el soporte de los actuadores presentado en la sección 5.1.1. Este rediseño (Figura 91), permite que el primer eslabón cuente con dos puntos de apoyo y que el movimiento del servomotor a él, sea transmitido por medio de un eje y no directamente.

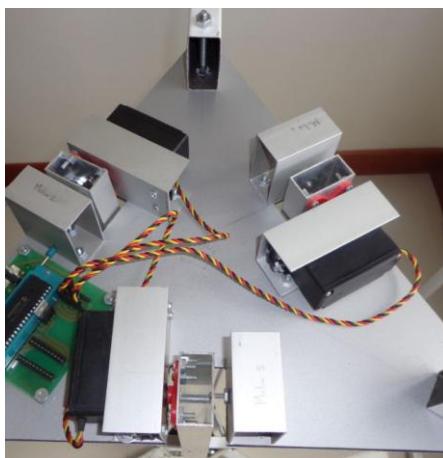


Figura 90. Base fija del primer prototipo.



Figura 91. Base fija del prototipo final.

En el primer prototipo, los movimientos rotacionales de las junturas eran restringidos por la fricción de su eje con los perfiles de aluminio (Figura 92), por

esta razón, se diseñaron un nuevo tipo de juntas permitieran disminuir la fricción en sus movimientos (Figura 93).

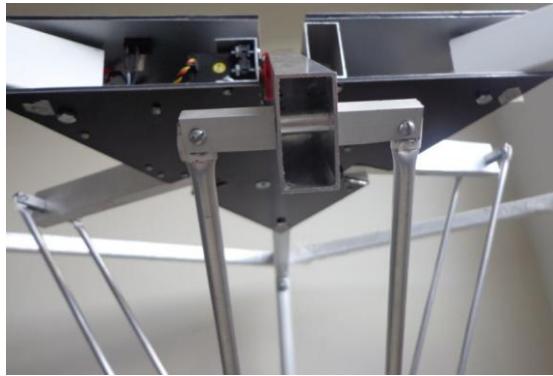


Figura 92. Juntas universales del primer prototipo.



Figura 93. Juntas universales del prototipo final.

En el primer prototipo, el robot solo contaba con tres grados de libertad, por esta razón, la base móvil fue rediseñada para permitir acoplar un cuarto actuador que permitiera la orientación del efecto final que se acoplará al robot. La Figura 94 muestra la base móvil del primer prototipo y la Figura 95 muestra la base móvil del prototipo final.



Figura 94. Base móvil del primer prototipo.



Figura 95. Base móvil del prototipo final.

Las Figuras 96 y 97 permiten comparar el prototipo implementado en la etapa del PIC y el prototipo implementado con este proyecto.



Figura 96. Robot Delta, primer prototipo.



Figura 97. Robot Delta, prototipo final.

5.2 DISEÑO ELECTRÓNICO

5.2.1 Selección de los actuadores

Los actuadores seleccionados son servomotores. Un servomotor es un motor eléctrico al que se le puede controlar su posición y/o su velocidad y se compone de un motor de corriente continua, un juego de engranajes para reducir la velocidad y aumentar el torque, un potenciómetro conectado al eje central del motor que permite sensar la posición angular y un pequeño circuito que permite el control del servomotor. Esta clase de motores generalmente cuentan con un control proporcional, es decir, la velocidad del servomotor varía según la distancia angular a recorrer. Si la distancia angular es pequeña, la velocidad angular del servo es baja y si la distancia angular es grande, la velocidad angular del servo es alta.

Como se aprecia en la Figura 98, un servomotor cuenta con un cable rojo para la alimentación positiva (V_{cc} , generalmente entre 4,8 V y 6 V), un cable

negro para la tierra y un cable amarillo para la señal de control. La señal de control es una onda cuadrada en donde su ancho de pulso es el que permite controlar la posición del servomotor.



Figura 98. Servomotor Hitec Hs755 HB.

En este proyecto se utilizaron servomotores **Hitec Hs755 HB**. Su torque máximo es de $1,2 \text{ Nm}$, el cual es un torque elevado para su tamaño ($58,93 \text{ mm} \times 28,96 \text{ mm} \times 49,78 \text{ mm}$) y peso (110 g). Su velocidad es de $43,47 \text{ rpm}$ o $4,553 \frac{\text{rad}}{\text{s}}$, su sensibilidad es de aproximadamente 2° y consumen poca corriente. La restricción de estos servomotores es que solo pueden posicionarse entre 0° y 180° .

5.2.2 Diseño del circuito electrónico

El posicionamiento de los servomotores se realiza por medio de la modulación por ancho de pulso o PWM (por sus siglas en inglés *Pulse Width Modulation*), que es una técnica de modulación en la que se modifica el ciclo de trabajo de una señal periódica (Figura 99). En este caso, el ancho de pulso de la señal determina la posición angular del servomotor. Para que el servomotor permanezca en la posición deseada es necesario mantener el pulso.

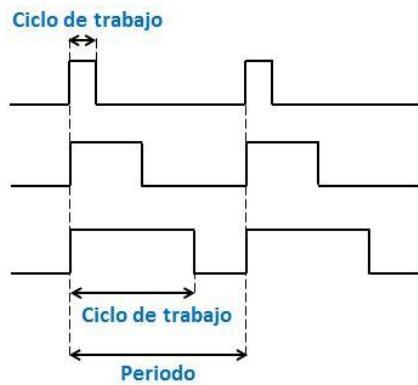


Figura 99. Descripción gráfica de un PWM.

Para diseñar el PCB (Printed Circuit Board), utilizamos un software llamado Eagle, el cual nos permite crear el PCB a partir de un diagrama esquemático de los componentes necesarios (Figura 100).

Para nuestro proyecto utilizamos un Pic 18f4550, un regulador de 5v L7005, un regulador de 3.3v lm1107, 2 LED (diodo emisor de luz), 2 condensadores de

20 nanofaradios, un cristal de 20 Mhz, un xbee (véase sección 5.3.1), 2 resistencias de 220 ohms y los sockets necesarios.

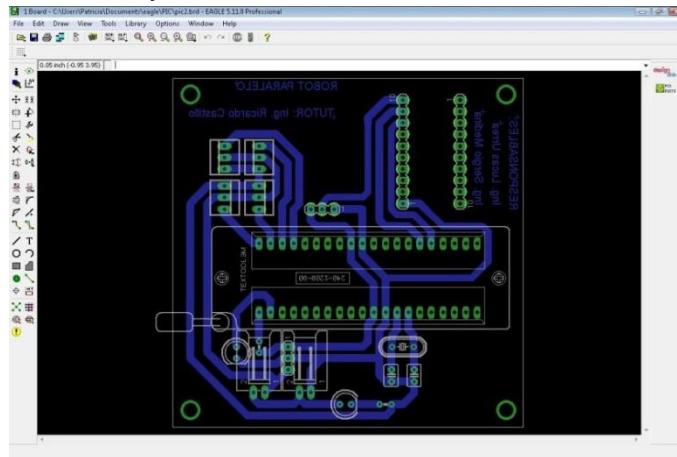


Figura 100. Diseño del PCB, EAGLE.

5.2.3 Resultados del circuito electrónico

Una vez se tenga el PCB listo (Figura 101) lo siguiente es soldar cada elemento (Figura 102) asegurando que exista continuidad electrónica, para evitar problemas en un futuro.

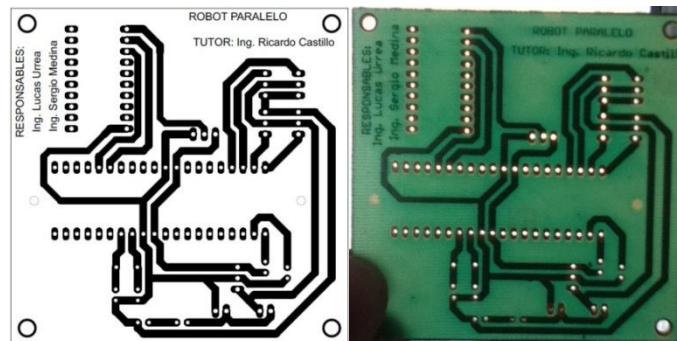


Figura 101. Diseño PCB vs Circuito implementado.



Figura 102. Circuito electrónico.

La función del circuito electrónico es la de recibir, interpretar y ejecutar la información proveniente del computador, MATLAB es el encargado de

solucionar la cinemática según las necesidades del usuario, es decir, el usuario por medio del software ingresa las coordenadas y posición que requiere para el robot (véase sección 5.3.2), MATLAB soluciona la cinemática y encuentra 4 valores que en este caso son ángulos, los codifica para facilitar la transmisión y los envía al robot por medio de los modulos xbee (sección 5.3.1), el microcontrolador es el encargado de interpretar la información codificada, y asignar a cada motor la información respectiva, la cual es una señal PWM diferente para cada posición (Figura 103), obteniendo como resultado, que el robot paralelo se ubique y se oriente según la necesidad del usuario, cabe resaltar, que el microcontrolador tiene programado unos valores iniciales los cuales son los 4 motores en la posición central, es decir 0°.

El microcontrolador 18f4550 está sobredimensionado para esta aplicación, pero toca recordar que el robot tiene un enfoque educativo, por lo tanto cualquier estudiante puede explotar los diferentes módulos adicionales que trae este microcontrolador.

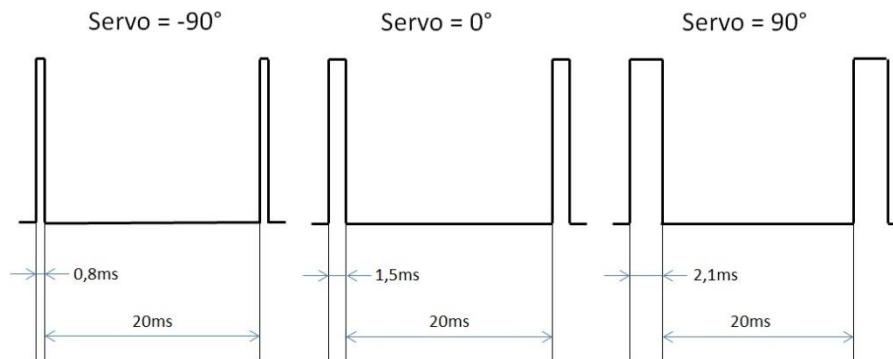


Figura 103. Señales PWM para posicionar el motor en 0°, -90° y 90°.

5.3 SOFTWARE DE CONTROL DE MOVIMIENTOS DEL ROBOT

5.3.1 Comunicación computador-robot

Para la comunicación entre el robot y el computador se usaron los módulos Xbee que son módulos de radio frecuencia que trabajan en la banda de 2,4 GHz con protocolo de comunicación IEEE 802.15.4. Su alcance en interiores es de 30 m y en exteriores de 100 m aproximadamente y sus grandes ventajas son el bajo costo y la muy baja potencia.

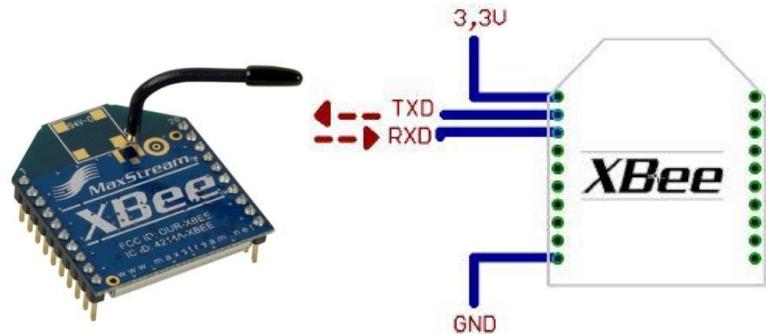


Figura 104. Módulo Xbee y sus conexiones del Xbee⁷.

La Figura muestra que el módulo Xbee requiere un voltaje de alimentación de 3,3 V, una conexión a tierra y las líneas de transmisión de datos, TXD y RXD, por medio del UART (Universal).

La tarjeta “Xbee Explorer” (Figura 105) es una mini placa que basa su funcionamiento en la conversión USB a Serial. Esta tarjeta permite conectar el Xbee al puerto USB, alimentándose por medio de este. Para que el computador pueda transmitir información al microcontrolador del robot, es necesario conectar un Xbee a la tarjeta “Exbee Explorer” y a su vez, conectar la tarjeta con el computador por medio de un conector miniUSB (Figura 106).

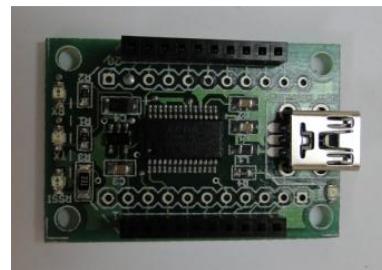


Figura 105. Tarjeta EXPLORER para el Xbee.



Figura 106. Comunicación entre el computador y el robot.

⁷ http://www.olimex.cl/pdf/Wireless/ZigBee/XBee-Guia_Usuario.pdf

5.3.2 GUI (Interfaz gráfica de usuario)

EL “Simulador Robot Delta” es una interfaz gráfica implementada en MATLAB que le permite al usuario controlar los servomotores del robot desde un computador, posicionando la plataforma móvil del robot en el espacio. Para ello, en la interfaz se implementaron las ecuaciones de los modelos cinemáticos directo e inverso, obtenidos en las secciones 4.2 y 4.3 respectivamente.

La interfaz consta de un menú principal en donde el usuario puede escoger entre tres opciones:

- Cinemática del robot: cinemática directa o inversa.
- Espacio de trabajo: permite visualizar el volumen de trabajo del robot.
- Acerca de: ventana con una breve descripción de la interfaz.

5.3.3 Resultados de la GUI

EL “Simulador Robot Delta” es una interfaz gráfica implementada en MATLAB (Figura 107) que le permite al usuario controlar los servomotores del robot desde un computador.

[Simulador Robot Delta - Ingenieria en Mecatronica - UMNG \(Bogota, Colombia\)](#)



UNIVERSIDAD MILITAR NUEVA GRANADA
FACULTAD DE INGENIERIA
INGENIERIA EN MECATRONICA

Integrantes: Lucas Mateo Urrea Mantilla
Sergio Alejandro Medina Papagayo

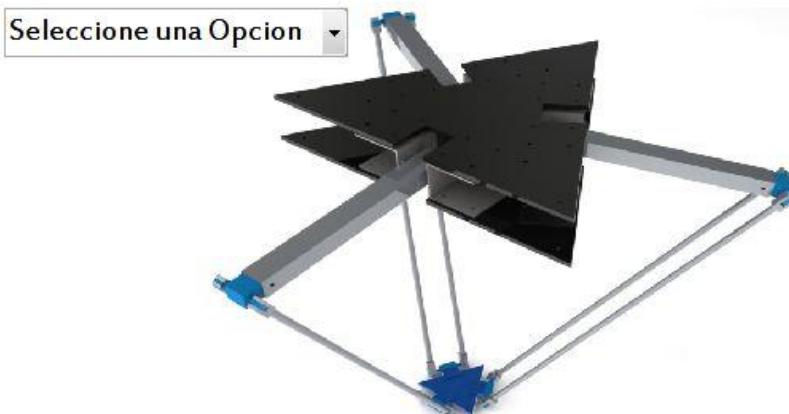


Figura 107. GUI, Matlab.

La interfaz cuenta con tres opciones:

- Cinemática del robot: opción que le permite al usuario manipular el robot desde el computador usando la cinemática directa o inversa.
- ✓ Cinemática directa: opción que le permite al usuario controlar los cuatro grados de libertad del robot directamente, por medio de sliders

(Figura 108). El objetivo de la cinemática directa es conocer la posición de la plataforma móvil si se conocen las variables de los actuadores.

Los pasos para resolver el modelo cinemático directo en la interfaz son:

1. Definir las dimensiones del robot.
 2. Definir las posiciones angulares de los servomotores (θ_{11}, θ_{12} y θ_{13}).
 3. Aplicar las ecuaciones del modelo planteado en la sección 4.3.
 4. Encontrar la posición de la plataforma móvil (P_x, P_y, P_z). La plataforma móvil puede alcanzar el punto encontrado si P_x, P_y o P_z no tienen parte imaginaria y si el ángulo entre el eslabón uno y el eslabón, de cada cadena cinemática, es mayor a 10° y menor a 180° .
- ✓ Cinemática inversa: opción que le permite al usuario controlar la posición y orientación de la plataforma móvil, por medio de sliders (Figura 109). El objetivo de la cinemática inversa es conocer la posición angular de los servomotores si se conoce la posición y orientación de la plataforma móvil.

Los pasos para resolver el modelo cinemático directo en la interfaz son:

1. Definir las dimensiones del robot.
2. Definir la posición de la plataforma móvil (P_x, P_y, P_z).
3. Aplicar las ecuaciones del modelo planteado en la sección 4.2.
4. Encontrar las posiciones angulares de los servomotores (θ_{11}, θ_{12} y θ_{13}). La plataforma móvil puede alcanzar el punto deseado si θ_{11}, θ_{12} o θ_{13} no tienen parte imaginaria y su valor esta entre -90° y 90° .

Las dos ventanas de cinemática cuentan con un modelo 3D del robot para que el usuario observe como es el movimiento del mismo.

Si alguna una posición no puede ser alcanzada por la plataforma móvil aparecerá un mensaje de error, como se aprecia en la Figura.

Para que se puedan enviar los ángulos en los que se debe posicionar cada servomotor es importante activar la comunicación (Botón “Conectar”). Debajo de la pestaña donde se elige la cinemática, aparece un mensaje que indica si el computador está o no comunicándose con el robot .

CINEMATICA ROBOT DELTA

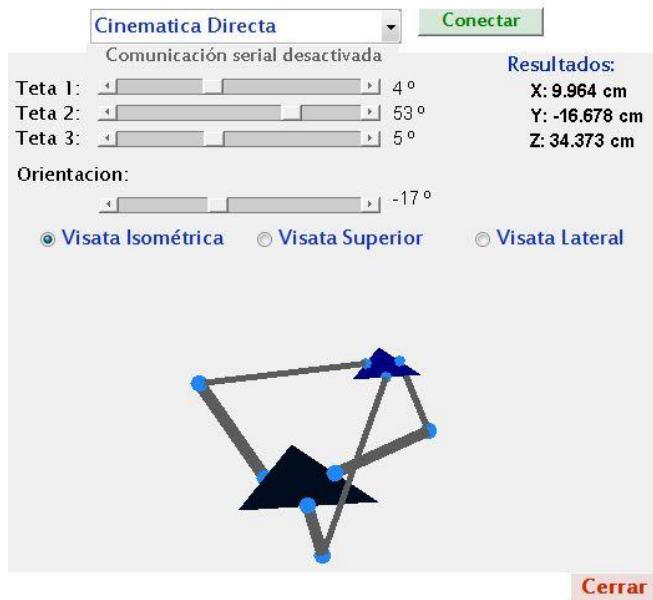


Figura 108. Ventana de la cinemática directa del robot Delta, Matlab.

CINEMATICA ROBOT DELTA

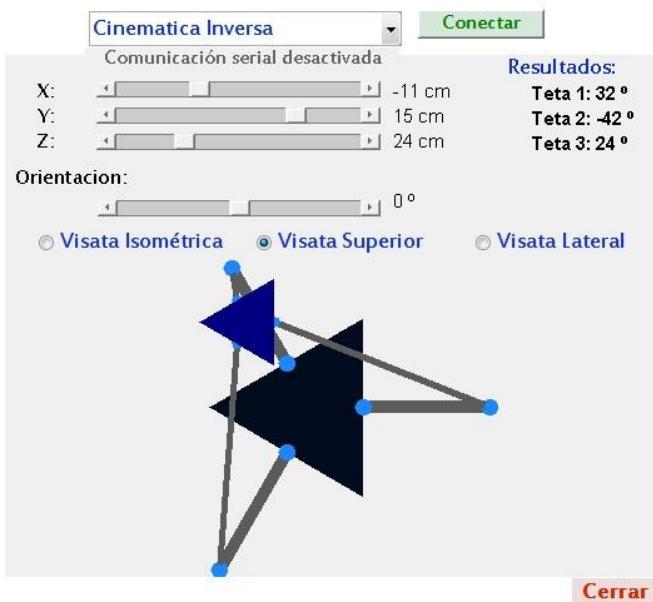


Figura 109. Ventana de la cinemática inversa del robot Delta, Matlab.

- Espacio de trabajo: opción que le permite al usuario visualizar el espacio de trabajo del robot Delta en cuatro perspectivas: plano XYZ, plano XY, plano XZ y plano YZ (Figura 110). Adicionalmente, el usuario conocerá la cantidad de puntos en el espacio en los que la plataforma móvil se puede ubicar, la cantidad de puntos en el espacio en los que la

destreza es mayor a 0,5 y el valor de la destreza global de ese espacio de trabajo.

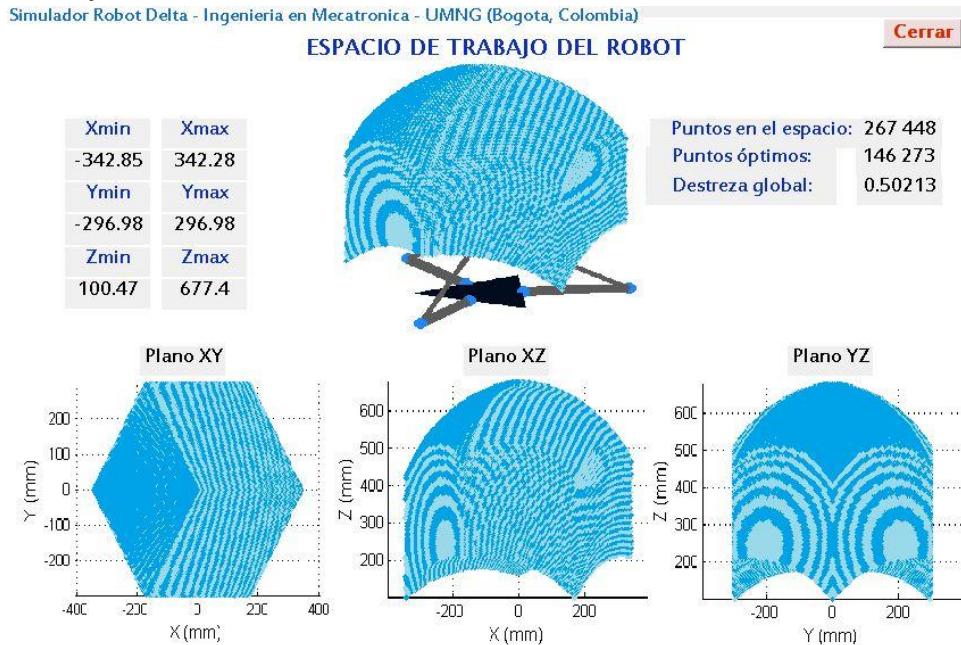


Figura 110. Ventana del espacio de trabajo del robot Delta, Matlab.

- Acerca de: opción que describe brevemente la finalidad de la GUI y las opciones con las que cuenta (Figura 111).

Simulador Robot Delta - Ingeniería en Mecatrónica - UMNG (Bogotá, Colombia)

SIMULADOR ROBOT DELTA

El "SIMULADOR ROBOT DELTA" es una interfaz grafica que le permite al usuario controlar los movimientos del robot Delta desde su computador. El programa esta desarrollado en MATLAB, permitiendo a los estudiantes hacer modificaciones.

Cinematica del Robot: El usuario puede manipular el robot por medio de la Cinematica Directa o la cinematica Inversa.

Espacio de trabajo: El usuraio puede ver el espacio de trabajo del robot Delta.

Cerrar

Figura 111. Ventana de ayuda sobre el simulador del robot Delta, Matlab.

Para mayor información acerca de la GUI y la forma de utilizarla, revisar en los anexos el manual de usuario del robot Delta.

5.4 RESUMEN

En este capítulo se ha presentado el diseño mecánico en SolidWorks de cada uno de los componentes del robot: base fija, base móvil, eslabones, juntas universales y estructura soporte. Además se seleccionaron los materiales para cada componente para luego analizar el comportamiento estático del robot por el método de elementos finitos.

Una vez se verificó que el diseño no contaba con ningún problema se procedió a la construcción del prototipo, al diseño e implementación del circuito electrónico y a la selección de los actuadores. Finalmente, se dedica un espacio al protocolo de comunicación entre el computador y el robot y se muestra la interfaz gráfica de usuario que se implementó en Matlab para poder posicionar la plataforma móvil del robot desde un computador, por medio de la cinemática directa e inversa desarrollada en el capítulo anterior.

CAPÍTULO 7: CONCLUSIONES Y TRABAJOS FUTUROS

7.1 CONCLUSIONES

En este trabajo de grado se ha presentado el diseño y la implementación de un prototipo funcional de un robot paralelo tipo Delta de 4 grados de libertad, tres para posicionar su plataforma móvil en el espacio y uno para orientar algún efecto final que se adapte al robot. Este robot fue construido con el propósito de dotar al laboratorio de robótica del programa de ingeniería en Mecatrónica con un robot paralelo espacial, que les permita a los estudiantes realizar prácticas para entender mejor la cinemática de esta clase de robots. Además, por ser una plataforma abierta permite realizar investigaciones en áreas como la teleoperación y cooperación en células manufactureras, área de la industria en la que los robots Delta han tomado mucha fuerza.

El diseño de cada componente del robot se realizó en SolidWorks y se validó por medio de un análisis estático del robot con elementos finitos. Los actuadores seleccionados fueron servomotores por su velocidad, precisión en el posicionamiento, facilidad de control y relación torque-masa del motor.

Se estudia el problema cinemático del robot diseñado, presentando las ecuaciones que describen la geometría del robot y relacionan las dimensiones del robot con la posición angular de los actuadores y la posición de la plataforma móvil. Dichas ecuaciones permiten obtener los modelos cinemáticos directo e inverso y la matriz Jacobiana, la cual es indispensable para el análisis de las singularidades del robot y de su destreza local y global. Adicionalmente, por medio de la cinemática directa se implementó un algoritmo en Matlab que permite visualizar el volumen de trabajo del robot y permite conocer cuál es la cantidad de puntos en el espacio en los que la plataforma móvil se puede posicionar y cuántos de esos puntos son óptimos, es decir, tienen un índice de destreza local mayor a 0,5.

El diseño se ha optimizado desde el punto de vista cinemático, donde se definieron las dimensiones de los eslabones del robot por medio de algoritmos genéticos multi-objetivos, utilizando el espacio de trabajo y el índice de destreza global como funciones objetivo. La optimización se realizó implementando seis algoritmos genéticos distintos en Matlab:

- RWGA: Random Weight Genetic Algorithm.
- FWGA: Fixed Weight Genetic Algorithm.
- AWGA: Adaptive Weight Genetic Algorithm.

- NSGA: Non-dominated Sorting Genetic Algorithm.
- NPGA: Niched Pareto Genetic Algorithm.
- MOFGA: Multiplied Objectives Genetic Algorithm.

Cada uno de ellos implementado por tres métodos diferentes de selección (truncamiento, muestreo universal estocástico y ruleta), para un total de 18 algoritmos implementados. El objetivo de todos ellos era encontrar un conjunto de soluciones en donde los puntos en el espacio alcanzaos por la plataforma móvil fueran más de 200 000 y el índice de destreza global fuera mayor a 0,47.

Una vez ejecutado el programa, se obtuvieron 18 conjuntos de posibles soluciones, uno por cada algoritmo genético implementado. Para encontrar la mejor solución, se agruparon los resultados según el método de selección, reduciendo los 18 conjuntos de posibles soluciones a tan solo 3 conjuntos para luego formar un solo conjunto de posibles soluciones. De este conjunto se obtuvo un subconjunto cuyos elementos fueron las soluciones óptimas de Pareto y de este subconjunto se escogieron las dimensiones de los eslabones.

El tiempo de ejecución del algoritmo de optimización fue de aproximadamente 10 horas en vista a la cantidad de operaciones realizadas para cada algoritmo genético, teniendo en cuenta que cada población tenía 50 individuos y cada algoritmo finalizaba cuando completara 100 generaciones.

Finalmente, las ecuaciones de la cinemática directa e inversa del robot diseñado se implementaron en una interfaz gráfica de usuario (GUI) en Matlab que permite posicionar la plataforma móvil del robot desde un computador. Adicionalmente, la interfaz cuenta con un modelo 3D que permite visualizar los movimientos reales del robot.

7.2 TRABAJOS FUTUROS

En vista que este proyecto es el primer prototipo de un robot Delta que se construye para la Universidad Militar Nueva Granada, se proponen los siguientes trabajos futuros, que están fuera de los objetivos planteados en este proyecto:

- Desarrollar la dinámica del robot Delta.
- Optimizar las dimensiones del robot basándose en su comportamiento cinemático y dinámico.
- Desarrollar algoritmos que permitan planear trayectorias de la plataforma móvil del robot. Al tener en cuenta la cinemática y la dinámica del robot para planear una trayectoria dada, se pueden obtener los valores de ángulos y torques de los actuadores y la velocidad, aceleración y fuerza que experimenta la plataforma móvil en cada punto en el espacio.
- Por medio de la dinámica se pueden obtener los torques de los actuadores para que el robot sea capaz de desplazarse dentro de todo su espacio de trabajo bajo fuerzas externas en su plataforma móvil con una velocidad y

una aceleración dada. Los actuadores se eligen basándose en la velocidad máxima de la plataforma móvil y torque máximo obtenidos de un conjunto de trayectorias a las que se somete el robot. Es importante que los nuevos actuadores del robot permitan una sensibilidad mayor a 2° para aumentar la exactitud de la posición de la plataforma móvil.

- Se puede mejorar la interfaz gráfica de usuario en MATLAB, incluyendo la dinámica y la planeación de trayectorias del robot. Adicionalmente, gracias a que el proyecto es una plataforma abierta, otros estudiantes pueden unir la GUI con simulink para obtener simulaciones más reales. Simulink cuenta con una herramienta (Simulink link) que permite cargar un CAD desde Solid Works (Figura 112) para poder hacer simulaciones cinemáticas y dinámicas

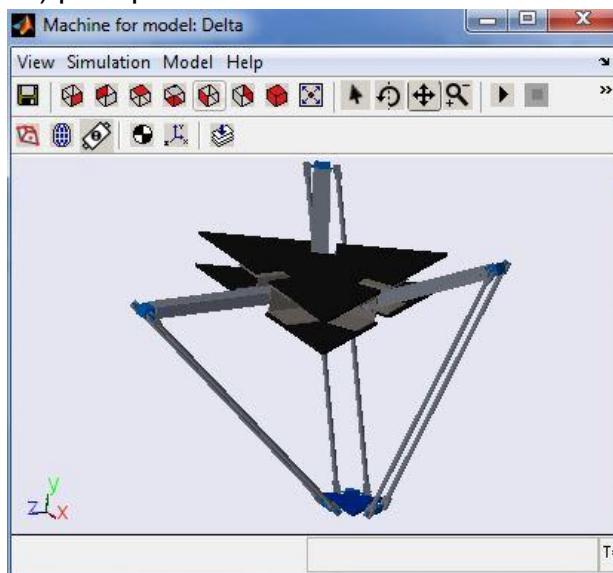


Figura 112. CAD del robot Delta importado de SolidWorks a Simulink.

- Desarrollar estrategias de control para el robot (p.e. control de posición control de velocidad) como por ejemplo: control clásico (p.e. PID), control difuso, control por redes neuronales, entre otros.
- Acoplar el efecto final para que pueda desarrollar algún tipo de trabajo, p.e. implementarle una ventosa para que pueda transportar elementos de un lugar a otro. Además, se puede implementar un sistema de visión artificial para identificar en donde están los objetos que se el robot debe transportar. Esto implicaría todo un desarrollo en procesamiento de imágenes e investigación en inteligencia artificial.
- Se pueden desarrollar trabajos en la teleoperación del robot en células manufactureras, empleando al robot como control de calidad de productos que se desplacen en una banda transportadora o simplemente darle una aplicación de pick and place al robot.

BIBLIOGRAFÍA

- Affenzeller, M., Wagner, S., Winkler, S., Beham, A., 2009. Genetic Algorithms and Genetic Programming, Modern concepts and practical applications. CRC Press.
- Angel, L., Sebastian, J.M., Saltaren, R., Aracil, R., 2005. RoboTenis System Part II: Dynamics and Control. Presented at the Fortieth IEEE Conference on Decision and Control and European Control Conference (CDC-ECC 2005), Sevilla, España, pp. 2030– 2034.
- Angel, L., Sebastian, J.M., Saltaren, R., Aracil, R., Gutierrez, R., 2005. RoboTenis: design, dynamic modeling and preliminary control. Presented at the IEEE/ASME International Conference on Advanced Intelligent Mechatronics, Moterey, California, pp. 747–752.
- Aracil, R., Saltarén J., R., Sabater, J.M., Reinoso, O., 2006. Robots paralelos: Máquinas con un pasado para una robótica del futuro. Revista Iberoamericana de Automática e Informática Industrial 3, 16–28.
- Badescu, M., Morman, J., Mavroidis, C., 2002. Workspace optimization of 3-UPU parallel platforms with joint constraints, in: IEEE International Conference on Robotics and Automatio (ICRA '02). Washington, D.C., pp. 3678 – 3683.
- Barrientos, A., 2007. Fundamentos de Robotica. McGraw-Hill.
- Bekey, G.A., 2005. Autonomous Robots: From Biological Inspiration to Implementation and Control. MIT Press.
- Bhatti, M.A., 2005. Fundamental Finite Element Analysis and Applications: with Mathematica and Matlab Computations. Wiley.
- Bonasso, R.P., Kortenkamp, D., Murphy, R.R. (Eds.), 1998. Artificial Intelligence and Mobile Robots: Case Studies of Successful Robot Systems. The MIT Press.
- Bruzzone, L., Molfino, R.M., 2003. Special-porpose parallel robot for active suspension of ambulance stretchers. International Journal of Robotics And Automation 18, 121–130.
- Cappel, K.L., 1967. Motion simulator.
- Carbone, G., Ceccarelli, M., 2005. A Serial-Parallel Robotic Architecture for Surgical Tasks. Robotica 23, 345–354.
- Carvajal, J., 2009. Modelamiento y diseño de robots industriales. Universidad de la Salle, Facultad de ingeniería de diseño y automatización electrónica, Bogotá, Colombia.

- Castro, R., Arangure, A., 2010. Desarrollo de un entorno de realidad virtual para interactuar con un prototipo de robot paralelo.
- Ceccarelli, M., Lanni, C., 2004. A multi-objective optimum design of general 3R manipulators for prescribed workspace limits. *Mechanism and Machine Theory* 39, 119–132.
- Chapra, S.C., Canale, R., 2001. Numerical Methods for Engineers: With Software and Programming Applications, 4th ed. McGraw-Hill.
- Clavel, R., 1990. Device for the movement and positioning of an element in space.
- Coello, C.A.C., Lamont, G.B., Veldhuizen, D.A.V., 2007. Evolutionary algorithms for solving multi-objective problems. Springer.
- Dwarakanath, T.A., Bhaumick, T.K., Venkatesh, D., 1999. Implementation of Stewart platform based force-torque sensor. Presented at the 1999 International Conference on Multisensor Fusion and Integration for Intelligent Systems (MFI '99), Taipei, Taiwan, pp. 32–37.
- Fish, J., Belytschko, T., 2007. A First Course in Finite Elements. Wiley.
- Gen, M., Cheng, R., 1999. Genetic Algorithms and Engineering Optimization. Wiley-Interscience.
- Gen, M., Cheng, R., Lin, L., 2010. Network Models and Optimization: Multiobjective Genetic Algorithm Approach, Softcover reprint of hardcover 1st ed. 2008. ed. Springer.
- Gómez, J.A., Guacaneme, J.V., 2011. Diseño e implementación de una plataforma de Stewart.
- Gosselin, C.M., 1990. Stiffness mapping for parallel manipulators. *Robotics and Automation, IEEE Transactions on* 6, 377 –382.
- Gosselin, C.M., Angeles, J., 1990. Singularity analysis of closed-loop kinematic chains. *IEEE Transactions on Robotics and Automation* 6, 281–290.
- Gough, V.E., Whitehall, S.G., 1962. Universal tire test machine. Presented at the Nineth Int. Technical Congress F.I.S.I.T.A., pp. 117–135.
- Groover, M.P., 1989. Robótica industrial: tecnología, programación y aplicaciones. McGraw-Hill, Madrid, España.
- Gwinnett, J., 1931. Amusement devices.
- Haupt, R.L., Haupt, S.E., 2004. Practical Genetic Algorithms, 2nd ed. Wiley-Interscience.
- Holland, J.H., 1992. Adaptation in Natural and Artificial Systems: An Introductory Analysis with Applications to Biology, Control, and Artificial Intelligence. A Bradford Book.

- Hong, J., Yamamoto, M., 2009. A calculation method of the reaction force and moment for a Delta-type parallel link robot fixed with a frame. *Robotica* 27, 579–587.
- Hongguang, W., Mingyang, Z., Lijin, F., Bo, Z., Zhigang, X., 2004. Identification of Parameters for a Stewart Platform-based Force/Torque Sensor. Presented at the 2004 IEEE International Conference on Robotics and Biomimetics (ROBIO '04), Shenyang, China, pp. 46–50.
- Hopgood, A.A., 2011. Intelligent Systems for Engineers and Scientists. CRC Press.
- Horn, J., Nafpliotis, N., Goldberg, D.E., 1994. A niched Pareto genetic algorithm for multiobjective optimization, in: , Proceedings of the First IEEE Conference on Evolutionary Computation, 1994. IEEE World Congress on Computational Intelligence. Presented at the First IEEE Conference on Evolutionary Computation, 1994. IEEE World Congress on Computational Intelligence, Orlando, FL, pp. 82 –87.
- Hunt, K.H., 1983. Structural Kinematics of In-Parallel-Actuated Robot-Arms. *Journal of Mechanisms, Transmissions, and Automation in Design* 105, 705–712.
- Hunt, K.H., 1990. Kinematic Geometry of Mechanisms. Oxford University Press, United States.
- Jazar, R.N., 2010. Theory of Applied Robotics: Kinematics, Dynamics, and Control, 2nd ed. Springer.
- Jiantao, Y., Yulei, H., Ling, L., Yongsheng, Z., 2006. Analysis of a Pre-stressed Six-component Force/Torque Sensor Based on Stewart Platform. Presented at the 2006 IEEE International Conference on Robotics and Biomimetics (ROBIO'06), IEEE, Kunming, China, pp. 346–350.
- Jingjun, Z., Lihong, S., Ruizhen, G., Chaoyang, L., 2009. The mathematical model and direct kinematics solution analysis of Delta parallel robot. Presented at the Second IEEE International Conference on Computer Science and Information Technology (ICCSIT), Beijing, China, pp. 450–454.
- Kong, X., Gosselin, C.M., 2007. Type Synthesis of Parallel Mechanisms. Springer.
- Kwon, Y.W., Bang, H., 2000. The Finite Element Method Using MATLAB, 2nd ed. CRC Press.
- Laribi, M.A., Romdhane, L., Zeghloul, S., 2007. Analysis and dimensional synthesis of the DELTA robot for a prescribed workspace. *Mechanism and Machine Theory* 42, 859–870.
- Lewis, F.L., Dawson, D.M., Abdallah, C.T., 2003. Robot Manipulator Control: Theory and Practice, 2nd ed. CRC Press.
- Liu, X.-J., Kim, J., 2005. A new spatial three-DoF parallel manipulator with high rotational capability. *IEEE/ASME Transactions on Mechatronics* 10, 502– 512.

- Liu, X.-J., Wang, J., Oh, K.-K., Kim, J., 2004. A new approach to the design of a DELTA robot with a desired workspace. *Journal of Intelligent and Robotic Systems: Theory and Applications* 39, 209–225.
- Lou, Y., Liu, G., Xu, J., Li, Z., 2004. A general approach for optimal kinematic design of parallel manipulators. Presented at the 2004 IEEE International Conference on Robotics and Automation (ICRA'04), IEEE, New Orleans, LA, pp. 3659– 3664.
- Martínez Angulo, I., Angulo Usategui, J.M., Romero Yesa, S., 2005. Introducción a la robótica: principios teóricos, construcción y programación de un robot educativo. Thomson, Madrid, España.
- Martínez, E., Peña, C., Yime, E., 2010. Diseño óptimo de un robot paralelo con configuración Detla para aplicaciones educativas. *Revista Educación en Ingeniería* 110–119.
- Maulik, U., Bandyopadhyay, S., Mukhopadhyay, A., 2011. Multiobjective Genetic Algorithms for Clustering: Applications in Data Mining and Bioinformatics. Springer.
- Merlet, J.P., 1992. On the infinitesimal motion of a parallel manipulator in singular configurations, in: 1992 IEEE International Conference on Robotics and Automation. Nice, France, pp. 320 –325.
- Merlet, J.P., 1993. Direct kinematics of parallel manipulators. *IEEE Transactions on Robotics and Automation* 9, 842–846.
- Merlet, J.P., 1996. Workspace-oriented methodology for designing a parallel manipulator. Presented at the IEEE International Conference on Robotics and Automation, Minneapolis, Minnesota, pp. 3726–3731.
- Merlet, J.P., 2003. Determination of the optimal geometry of modular parallel robots. Presented at the 2003 IEEE International Conference on Robotics and Automation (ICRA '03), IEEE, Taipei, Taiwan, pp. 1197– 1202.
- Merlet, J.P., 2007. A Formal-Numerical Approach for Robust In-Workspace Singularity Detection. *IEEE Transactions on Robotics* 23, 393–402.
- Merlet, J.P., 2010. Parallel Robots, 2nd ed. Springer.
- Moaveni, S., 2007. Finite Element Analysis Theory and Application with ANSYS, 3rd ed. Prentice Hall.
- Mohanty, S., 2006. Multiobjective optimization of synthesis gas production using non-dominated sorting genetic algorithm. *Computers & Chemical Engineering* 30, 1019–1025.
- Mühlenbein, H., Schlierkamp-Voosen, D., 1993. Predictivie Models for the Breeder Genetic Algorithm I. Continuous Parameter Optimization. *Evolutionary computation* 1, 25–49.

- Murata, T., Ishibuchi, H., 1995. MOGA: multi-objective genetic algorithms, in: , IEEE International Conference on Evolutionary Computation, 1995. Presented at the , IEEE International Conference on Evolutionary Computation, 1995, p. 289.
- Murata, T., Ishibuchi, H., Tanaka, H., 1996. Multi-objective genetic algorithm and its applications to flowshop scheduling. *Computers & Industrial Engineering* 30, 957–968.
- Niku, S.B., 2010. *Introduction to Robotics: Analysis, Control, Applications*, 2nd ed. Wiley.
- Ollero B, A., 2007. Robótica: Manipuladores y robots móviles. Alfaomega.
- Ollero Baturone, A.O., 2001. Robotica, Manipuladores Y Robots Moviles. Alfaomega - Marcombo.
- Ottaviano, E., Ceccarelli, M., 2000. Optimal Design of CaPaMan (Cassino Parallel Manipulator) With Prescribed Position and Orientation Workspace. *Robótica* 20, 159–166.
- Peña Cortés, C.A., Martínez O., E., Cárdenas H., P.F., 2011. Optimización dimensional de un robot paralelo tipo Delta basado en el menor consumo de energía. *Ciencia e Ingeniería Neogranadina* 21, 73–88.
- Pierrot, F., Nabat, V., Company, O., Krut, S., Poignet, P., 2009. Optimal Design of a 4-DOF Parallel Manipulator: From Academia to Industry. *Robotics, IEEE Transactions on* 25, 213–224.
- Pollard, W.L.V., 1942. Position controlling apparatus.
- RuiJun, Z., DingFang, C., Yong, W., ZhongHua, Y., XinXin, W., 2007. Study on Line Balancing Problem Based on Improved Genetic Algorithms, in: International Conference on Wireless Communications, Networking and Mobile Computing, 2007. WiCom 2007. Presented at the International Conference on Wireless Communications, Networking and Mobile Computing, 2007. WiCom 2007, Shanghai, China, pp. 2033 –2036.
- Sciavicco, L., Siciliano, B., 2000. *Modelling and Control of Robot Manipulators*, 2nd ed. Springer.
- Serafin, J.L., 1953. Simulated aircraft.
- Siciliano, B., Khatib, O., 2008. *Springer handbook of robotics*. Springer.
- Siciliano, B., Sciavicco, L., Villani, L., Oriolo, G., 2011. *Robotics: Modelling, Planning and Control*, 2nd ed. Springer.
- Silva, L.A., Sebastian, J.M., Saltaren, R., Aracil, R., Sanpedro, J., 2005. RoboTenis: optimal design of a parallel robot with high performance. Presented at the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), pp. 2134– 2139.

- Spong, M.W., Hutchinson, S., Vidyasagar, M., 2005. Robot Modeling and Control, 1st ed. Wiley.
- Srinivas, N., Deb, K., 1994. Multiobjective Optimization Using Nondominated Sorting in Genetic Algorithms. *Evolutionary Computation* 2, 221–248.
- Stamper, R.E., 1997. A Three Degree of Freedom Parallel Manipulator with Only Translational Degrees of Freedom.
- Stamper, R.E., Tsai, L.-W., Walsh, G.C., 1997. Optimization of a three DOF translational platform for well-conditioned workspace, in: Proceedings of the 1997 IEEE International Conference on Robotics and Automation. Albuquerque, New Mexico, pp. 3250 – 3255.
- Stan, S.-D., Manic, M., Maties, V., Balan, R., 2008. Kinematics analysis, design, and control of an Isoglide3 Parallel Robot (IG3PR). Presented at the Thirtieth Annual Conference of IEEE Industrial Electronics (IECON 2008), Orlando, FL, pp. 2636–2641.
- Stan, S.-D., Maties, V., Balan, R., 2007. Multi-objective genetic algorithms applied for optimal design of 2 DOF micro parallel robots, in: 2007 IEEE Workshop on Advanced Robotics and Its Social Impacts (ARSO '07). Presented at the 2007 IEEE Workshop on Advanced Robotics and Its Social Impacts (ARSO '07), IEEE, Hsinchu, Taiwan, pp. 1–6.
- Stan, S.-D., Maties, V., Balan, R., Lapusan, C., 2008. Genetic algorithms to optimal design of a 3 DOF parallel robot. Presented at the 2008 IEEE International Conference on Automation, Quality and Testing, Robotics (AQTR '08), IEEE, pp. 365–370.
- Stewart, D., 1965. A platform with six degrees of freedom. *Proceedings of the Institution of mechanical engineers* 180, 371–386.
- Thierens, D., Goldberg, D., 1994. Convergence Models of genetic Algorithm Selection Scheme. Presented at the PPSN III: International Conference on Evolutionary Computation, IEEE, Jerusalén, Israel.
- Tsai, L.-W., 1999. Robot Analysis: The Mechanics of Serial and Parallel Manipulators, 1st ed. Wiley-Interscience.
- Tsai, L.-W., Walsh, G.C., Stamper, R.E., 1996. Kinematics of a novel three DOF translational platform. Presented at the 1996 IEEE International Conference on Robotics and Automation, Minneapolis, Minnesota, pp. 3446–3451.
- Vivas, A., 2007. Control predictivo funcional de un robot paralelo. *Revista Colombiana de Tecnologías de Avanzada* 1, 48–56.
- Vivas, A., Poignet, P., 2006. Control predictivo de un robot paralelo 3, 46–53.
- Xin-Jun Liu, Jongwon Kim, 2002. A new three-degree-of-freedom parallel manipulator. Presented at the 2002 IEEE International Conference on

- Robotics and Automation (ICRA '02), IEEE, Washington, D.C., pp. 1155–1160.
- Yunjiang, L., Guanfeng, L., Ni, C., Zexiang, L., 2005. Optimal design of parallel manipulators for maximum effective regular workspace, in: 2005 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS 2005). Presented at the 2005 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS 2005), IEEE, pp. 795– 800.
- Zabala, I., Ros, J., 2007. Aplicaciones actuales de los robots paralelos. Presented at the En: 8º Congreso Iberoamericano de Ingeniería Mecánica (CIBIM8), Cusco, México.
- Zhang, D., 2009. Parallel Robotic Machine Tools. Springer.