

**DISEÑO Y CONSTRUCCIÓN DE UNA ESTRUCTURA ROBÓTICA TIPO DELTA
INVERTIDO PARA EL LABORATORIO LADPRER, DE LA UNIVERSIDAD DE
BRASILIA**

Autor

MIGUEL ANGEL BAREÑO BERNAL

Director

César Augusto Peña Cortés

Director (Universidad de Brasilia)

Efraín Andres Rodríguez Gasca



**UNIVERSIDAD DE PAMPLONA
FACULTAD DE INGENIERAS Y ARQUITECTURAS
DEPARTAMENTO DE MECÁNICA MECATRÓNICA E INDUSTRIAL
PAMPLONA-NORTE DE SANTANDER
2020**

Resumen

La estructura Delta invertido a diferencia de una disposición clásica, presenta un área de trabajo al interior del volumen comprendido por las piernas del robot, propuesto por el profesor Reymond Clavel en 2016 es una configuración poco estudiada a la fecha.

En el presente trabajo, se presenta el desarrollo conceptual, análisis diseño y simulación de una estructura robótica de tipo delta invertido para el laboratorio LaDPRER de la universidad de Brasilia. Se desarrolla el análisis cinemático directo e inverso al igual que la optimización dimensional utilizando algoritmos genéticos el cual calcula dimensiones de los eslabones que conforman la estructura y los modifica con el objetivo de encontrar las dimensiones optimas mínimas para su construcción, para la corroboración del mismo se implementó un código el cual genera una nube de puntos comprendidos dentro del cilindro de trabajo, el código ejecutara la cinemática inversa del robot en cada uno de dichos puntos evaluando restricciones de diseño. Si los valores obtenidos por el algoritmo no generan errores dimensionales se puede corroborar la veracidad de los valores arrojados por el ga.

Como resultado final se fabricó un prototipo físico y una interfaz gráfica en el entorno matemático, los cuales se conectan mediante conexión serial enviando comandos Gcode para la realización de los desplazamientos.

Palabras claves: Cinemática Paralela, Robot Delta Invertido, Desarrollo de Producto, Optimización Dimensional.

Abstrac

The inverted Delta structure, unlike a classic arrangement, presents a work area within the volume comprised by the robot's legs, proposed by Professor Reymond Clavel in 2016, it is a configuration little studied to date.

In this work, the conceptual development, analysis, design and simulation of an inverted delta-type robotic structure for the LaDPRER laboratory of the University of Brasilia is presented. The direct and inverse kinematic analysis is developed as well as the dimensional optimization using genetic algorithms which calculates dimensions of the links that make up the structure and modifies them in order to find the minimum optimal dimensions for its construction, for the corroboration of the same implemented a code which generates a cloud of points within the working cylinder, the code will execute the inverse kinematics of the robot in each of these points evaluating design constraints. If the values obtained by the algorithm do not generate dimensional errors, the veracity of the values produced by the ga can be corroborated.

As a final result, a physical prototype and a graphical interface were manufactured in the mathematical environment, which are connected through a serial connection sending Gcode commands to carry out the movements.

Keywords: Parallel Kinematics, Inverted Delta Robot, Product Development, Dimensional Optimization.

AGRADECIMIENTOS

Agradecimiento especial a la universidad de pamplona y a la oficina de vicerrectoría de investigaciones por aportar los rubros financieros para la movilidad de esta investigación, a la universidad de Brasilia y el grupo de automatição e controle por prestar los equipos, materiales. Al profesor Cesar Augusto Peña por a verme acogido en el semillero SIARC y incentivar siempre el desarrollo investigativo, al profesor Alberto Alvares y Efrain Rodríguez por haberme abierto las puertas al laboratorio LaDPRER y brindarme la asesoría técnica necesaria para la realización de este proyecto. Sin ellos no hubiera sido posible el desarrollo de este trabajo.

CONTENIDO

1	Capítulo 1	7
1.1	Introducción	7
1.2	Planteamiento del problema	8
1.3	Justificación	9
1.4	Objetivos	9
1.4.1	Objetivo General	9
1.4.2	Objetivos específicos	9
2	Capítulo 2 Marco teórico y estado del arte	10
2.1	Robots paralelos	10
2.1.1	Historia	10
2.1.2	Robots paralelos en la industria	12
2.1.3	Algunas aplicaciones de los robots paralelos	14
2.2	Manufactura aditiva	15
2.2.1	Procesos comunes de Manufactura aditiva	15
2.3	Trabajos relacionados	26
2.3.1	Robot delta en la manufactura aditiva FFF	30
3	Capítulo 3 Metodología	33
3.1	Descripción del robot delta invertido para manufactura aditiva FDM (ID-AM)	33
3.2	Implementación de QFD	34
3.2.1	Paso 1: Identificación de los usuarios	35
3.2.2	Paso 2: Definición de las RU	35
3.2.3	Paso 3: Priorización de los requerimientos	36
3.2.4	Paso 4: Selección de los parámetros de diseño (DP)	38
3.2.5	Paso 5: Relación entre las RU y as DP	39
3.2.6	Paso 6: Definición de los objetivos específicos	41
3.3	Cinemática inversa	41
3.4	Cinemática Directa	45
3.5	Análisis de singularidades y restricciones	51
3.5.1	Singularidades	51
3.5.2	Restricciones	52
3.6	Algoritmo genético	53
3.6.1	Optimización dimensional del robot ID-AM utilizando algoritmos genéticos	55
3.6.2	Código de implementación	57
3.6.3	Apreciaciones y resultados	70
3.6.4	Comprobación de resultados	72
4	Capítulo 4 Desarrollo del robot	73
4.1	Diseño electrónico	73
4.1.1	Hardware	73
4.1.2	Firmware	85
4.1.3	Software	86
4.2	Diseño mecánico	87
4.2.1	Base fija	88
4.2.2	Brazos	89

4.2.3	Base móvil	91
4.2.4	Ensamble Completo.....	93
4.3	Construcción del prototipo	94
4.3.1	Primera versión.....	94
4.3.2	Segunda versión	96
4.3.3	Versión final	97
4.4	Costos.....	98
5	Conclusiones.....	100
6	Lista de referencias	102
7	Apéndice	¡Error! Marcador no definido.

Lista de tablas

<i>Tabla 1 Ventajas y desventajas de una impresora cartesiana</i>	<i>24</i>
<i>Tabla 2 Ventajas y desventajas de una impresora delta</i>	<i>24</i>
<i>Tabla 3 Parámetros de interés para la construcción del robot DI-AM</i>	<i>36</i>
<i>Tabla 4 Posibles configuraciones de la función rng.....</i>	<i>62</i>
<i>Tabla 5 Resultados obtenidos de la implementación del ga con 50 individuos.....</i>	<i>71</i>
<i>Tabla 6 Resultados obtenidos de la implementación del ga con 50 sujetos y alteración de la aleatoriedad.....</i>	<i>72</i>
<i>Tabla 7 Valores de diseño escogidos (conclusión del ga).....</i>	<i>72</i>
<i>Tabla 8 Comparativa de precios tarjetas de control</i>	<i>78</i>
<i>Tabla 9 Comparativa de precios sensores de temperatura.....</i>	<i>82</i>
<i>Tabla 10 Comparativa de precios sensores final de carrera</i>	<i>84</i>
<i>Tabla 11 Lista de materiales ensamble 'base fija'</i>	<i>88</i>
<i>Tabla 12 Lista de materiales ensamble 'brazo'.....</i>	<i>90</i>
<i>Tabla 13 Lista de materiales ensamble 'plataforma móvil'</i>	<i>92</i>
<i>Tabla 14 Tabla de costos de construcción del prototipo en pesos colombianos</i>	<i>99</i>

Lista de figuras

<i>Figura 1 Configuración básica de un robot delta, con 3 actuadores lineales [6]</i>	10
<i>Figura 2 Posiblemente el primer mecanismo espacial paralelo, patentado en 1931 (Patente de Estados Unidos N° 1.789.680) [7]</i>	11
<i>Figura 3 El primer hexápodo octaédrico o plataforma de Gough 1945 [8].</i>	11
<i>Figura 4 Esquema de una plataforma Stewart- Gough [4].</i>	12
<i>Figura 5 Planos anexos a la patente Position controlling apparatus 22 de abril de 1938 [5].</i>	13
<i>Figura 6 Diseño planteado por Reymon Clavel en la patente EPO250470B1 [7].</i>	14
<i>Figura 7 Modelo en proceso de impresión por método de estereolitografía [18].</i>	16
<i>Figura 8 Diagrama de impresión 3D por método SLS [16].</i>	16
<i>Figura 9 (a) Diagrama de una maquina Binder jetting [18] (b) Cabezal de inyección del material aglutinante[17][19]</i>	17
<i>Figura 10 Diferentes acabados después del proceso de impresión Binder jetting [17].</i> .	18
<i>Figura 11 (a) Impresiones Poliyet con materia de soporte (b) Impresión Poliyet con material transparente luego de ser pulido [20]</i> .	18
<i>Figura 12 Modelos a todo color de impresos por the Mcor IRIS [25].</i>	19
<i>Figura 13 (a) Cabezal de proceso BED (b)Proceso de impresión BED [27].</i>	19
<i>Figura 14 Modelos impresos en 3D FDM (Fuente autor)</i>	20
<i>Figura 15 Pieza hibrida fabricada por combinación de embutición profunda y LBM [29].</i>	21
<i>Figura 16 Cabezal de revestimiento AMBIT en el nuevo husillo INTEGRIX i-400AM de Mazak con características depositadas con láser presentado en JIMOF 2014 [30].</i>	22
<i>Figura 17 Gráficos de usuarios de impresión 3D [26].</i>	22
<i>Figura 18 Prusa i3 y minirostock RepRap proyeto [26]</i>	23
<i>Figura 19 Esquema de impresoras 3D [36].</i>	24
<i>Figura 20 “New University of western Australia Robot- NUWAR” (a) diseño (b) Prototipod [8].</i>	26
<i>Figura 21 Plano de ensamble con contrapesos [9].</i>	27
<i>Figura 22 Malla final para simulación en ANSYS [29].</i>	27
<i>Figura 23 Resultados de la optimización a través de algoritmos genéticos [10].</i>	28
<i>Figura 24 (a)Estructura del robot KEOPS (b)Prototipó fabricado del robot KEOPS [12].</i>	28
<i>Figura 25 (a)Diseño conceptual del robot "Delta Wired 3D Printer" (b)Pruebas iniciales del prototipo físico [37].</i>	29
<i>Figura 26(a) Interfaz GUI (b) Simulación ejecutada en Matlab [32].</i>	29
<i>Figura 27 Resultado comparativo de los modelos impresos por 3D cloner y Rostock Max [33].</i>	30
<i>Figura 28 en la figura de la izquierda el diseño virtual, en la figura de la derecha el modelo terminado real [34].</i>	31
<i>Figura 29 Impresión de figura compleja con inclinación de la cama de impresión [35].</i> .	31
<i>Figura 30 (a) Detalle de las juntas rotacionales implementadas (b) juntas universales convencionales (c) Propuesta desarrollada en este proyecto [37].</i>	32

Figura 31 Prototipo GUS Simpson ensamblado, tomado de la librería Rep Rap [38]. ...	32
Figura 32. Distribución de la base fija de la plataforma DI-AM en vista normal al plano XY (Fuente autor).....	33
Figura 33. Distribución de la plataforma móvil del DI-AM en vista normal al plano XY (Fuente autor).....	34
Figura 34. Vista normal de la primera pierna del robot en el plano XZ (Fuente autor) ...	34
Figura 35 Graficas relevantes de los requerimientos de los usuarios (Fuente autor).....	37
Figura 36 Desarrollo de matriz QFD.....	40
Figura 37 Representación grafica de los ejes de rotación de las juntas de la plataforma ID-AM (Fuente autor)	42
Figura 38. Resalto del triángulo isósceles generado entre los puntos MB, FB, K (Fuente autor).....	43
Figura 39. Vista normal a una de las piernas del robot DI-AM utilizando como referencia el plano XZ (Fuente autor).....	44
Figura 40. Figura con punto hipotético P en el cual se muestra la vista superior sobre el plano xy evidenciando el ángulo Alpha generado por la sombra ($MB_i - P_{vi}$) de la pierna del robot DI-AM (Fuente autor).....	44
Figura 41. Vista superior normal al plano xy en la cual se evidencian las componentes cartesianas de la sombra S formada por la pierna (i) del robot.....	44
Figura 42 Posicionamiento esférico [47]	46
Figura 43 Representación tridimensional para el modelo de trilateración[47]	46
Figura 44 traslación de los puntos de la base (Fuente autor)	48
Figura 45 Base del robot antes y después de realizar la rotación de 150° (Fuente autor)	48
Figura 46 Simulación de movimiento articular $q(i)$ mayor de 180° (Fuente autor).....	51
Figura 47 Proceso de optimización de algoritmo genético (Fuente autor)	54
Figura 48 Emulación de la estructura simple del robot ID-AM con visualización del volumen cilíndrico de trabajo (Fuente autor)	56
Figura 49 Emulación de la estructura simple del robot ID-AM con visualización del volumen del cono truncado generado a partir de los parámetros geométricos del robot (Fuente autor).....	57
Figura 50 Función Run_optimize.m porción variables globales (Fuente autor).....	58
Figura 51 Variables para la simulación del robot (Fuente autor).....	58
Figura 52 Valores de las variables globales de simulación del robot ID-AM.....	58
Figura 53 Asignación de los parámetros para la construcción del cilindro de trabajo y valores de división para el mismo (Fuente autor).....	59
Figura 54 Configuraciones iniciales para el ga (Fuente autor)	60
Figura 55 Configuración de las opciones para la utilización del algoritmo genético.....	61
Figura 56 graficas generadas en la ejecución del ga (Fuente autor).....	62
Figura 57 Programa de generación para la población inicial de numero x de sujetos (Fuente autor).....	63
Figura 58 Extracción de los parámetros de interés y asignación de las variables utilizadas dentro de esta función (Fuente autor)	64
Figura 59 Simplificación descriptiva de la estructura condicional anidada, en donde se evidencia las 3 subcategorías de los sujetos F (Fuente autor)	65

<i>Figura 60 Caso 1 evaluación de restricciones (Fuente autor)</i>	67
<i>Figura 61 Caso 2 evaluación de restricciones (Fuente autor)</i>	67
<i>Figura 62 Caso 3 evaluación de restricciones (Fuente autor)</i>	68
<i>Figura 63 Caso 4 evaluación de restricciones (Fuente autor)</i>	68
<i>Figura 64 Caso 5 evaluación de restricciones (Fuente autor)</i>	68
<i>Figura 65 Caso 6 evaluación de restricciones (Fuente autor)</i>	69
<i>Figura 66 Caso 7 evaluación de restricciones (Fuente autor)</i>	69
<i>Figura 67 Caso 8 evaluación de restricciones (Fuente autor)</i>	69
<i>Figura 68 Caso 9 evaluación de restricciones (Fuente autor)</i>	70
<i>Figura 69 Caso 10 evaluación de restricciones (Fuente autor)</i>	70
<i>Figura 70 (a)Resultado 5 sujetos (b) Resultado 15 sujetos (c)Resultado 30 sujetos (d)Resultado 50 sujetos</i>	71
<i>Figura 71 Modelo alámbrico del robot evaluado en cada uno de los puntos generados por la función Corroboration_function (Fuente Autor).</i>	73
<i>Figura 72 Esquema simplificado de control (Fuente autor)</i>	75
<i>Figura 73 Placa SKR Mini V1.3 [53].</i>	76
<i>Figura 74 Placa A5984</i>	76
<i>Figura 75 Vista superior de tarjeta Greeetech Gt2560 [54].</i>	77
<i>Figura 76 Tarjeta MKS Gen V1.4 [55]</i>	78
<i>Figura 77 Shield Ramps 1.4 [56].</i>	78
<i>Figura 78 Termopar tipo K con modulo Max6675 [56]</i>	80
<i>Figura 79 Pt100 con modulo max31865</i>	81
<i>Figura 80 (a) NTC 3950 encapsulado metálico B (b)e NTC 3950 encapsulado libre</i>	82
<i>Figura 81 Sensor óptico TCST2103 [56]</i>	83
<i>Figura 82 Sensor inductivo LJ12A3-4-Z/BX [57].</i>	84
<i>Figura 83 Final de carrera mecánico MK2b [56]</i>	84
<i>Figura 84 Configuración del puerto serial y velocidad de comunicación (fuente Autor)</i> .	87
<i>Figura 85 Configuración de velocidades y temperaturas de precalentamiento</i>	87
<i>Figura 86 Configuración de la posición de reposo y cilindro de trabajo (Fuente autor)</i> ..	87
<i>Figura 87Ipresoras 3D disponibles en el laboratorio Ladeprer utilizadas para la fabricación de los eslabones</i>	94
<i>Figura 88 Preparación de piezas para el ensamble del prototipo N°1 (Fuente autor)</i>	95
<i>Figura 89 vista normal a una de las piernas que componen la pierna del robot (Fuente autor)</i>	95
<i>Figura 90 Imagen del prototipo N°1 ensamblado en su totalidad en posicion de reposo</i>	96
<i>Figura 91 Proceso de ensamble, conexión de los componentes electrónicos y tarjeta de control</i>	97
<i>Figura 92 Prototipo terminado</i>	97
<i>Figura 93 Mecanismo de transmisión con polea sincrónica GT2</i>	98
<i>Figura 94 Prototipo final ensamblado con rejilla superficial para pruebas</i>	98

1 CAPÍTULO 1

1.1 INTRODUCCIÓN

Históricamente el desarrollo de la robótica ha sido impulsado por la pluralidad de intereses y aplicaciones a las que puede incorporarse, acercándose cada vez más al ser humano para hacer más grata su existencia. Los dominios de aplicación en donde se incorporan robots para sus tareas son cada vez más diversos y numerosos, abarcan desde aplicaciones espaciales, industria en general, laboratorios, manipulación en entornos peligrosos, agricultura, educación, asistencia especial, seguridad y defensa entre otros.

Una clasificación general puede ser la descrita en el libro de fundamentos de robótica [1].

- Consolidados: Aplicaciones en las cuales los robots han sido implementados desde hace años y forman parte interna de los procesos automatizados. Tales como en aplicaciones industriales (soldadura, ensamblado, paletización, pintura, alimentación de máquinas, carga y descarga, entre otras).
- No consolidados: Básicamente las aplicaciones más recientes tales como el sector espacial, aplicaciones médicas (cirugía, ayuda a discapacitados entre otras).

Su proyección a futuro se encamina hacia soluciones integrales en el ámbito industrial, ligado al desarrollo de la industria 4.0, en donde el robot se plantea como una parte integral del proceso global CAD/CAM/CAE/CIM en donde también intervienen sistemas de gestión, diseño y control de calidad y fabricación [2]. Su evolución está ligada por la disponibilidad tecnológica del momento adaptando los nuevos desarrollos y descubrimientos mejorando y ampliando la gama de aplicaciones en donde pueden ser implementados.

La robótica ofrece soluciones que se ajustan a las necesidades específicas de un problema en cuestión, realizando trabajos complejos de manera automática y repetitiva, para lograr su objetivo es necesario que los mecanismos y robots sean dimensionados correctamente con base en la repetitividad de la tarea, sistema de control, velocidad, precisión, capacidad carga y volumen entre otros aspectos. Para delimitar y cumplir con los objetivos del necesarios se utiliza metodologías integrales de ingeniería, las cuales integran todos los requerimientos del robot al diseño y selección de componentes, de tal manera que el producto final cumpla con las especificaciones establecidas.

1.2 PLANTEAMIENTO DEL PROBLEMA

La inclusión masiva de robots a la industria general, plantea nuevos retos para satisfacer las necesidades de su incorporación, su creciente diversidad de aplicaciones impulsa al desarrollo de los mismos, obligando la integración de nuevas tecnologías e investigación constante en campos de la robótica. De allí la proliferación de congresos, cursos, planes de estudio de Universidades y aumento de investigaciones relacionadas, todo ello en busca de satisfacer la demanda de soluciones.

Una de las principales aplicaciones de los robots en la industria es producción en masa, en la cual se requiere la elaboración de grandes cantidades de producto con características similares, manteniendo los estándares de calidad [3]. Los robots juegan un papel fundamental en la estandarización de los procesos, realizando tareas repetitivas por largos periodos de tiempo.

Diversas y numerosas son las industrias que incorporan sistemas de automatización a sus líneas de producción, en muchos casos al intentar compararlas, sus procesos productivos no tienen nada en común, aunque es posible que compartan alguna tarea en específico. Una de las tareas más comunes que se puede encontrar en casi todas las industrias es el proceso de recoger y colocar o *pick and place* la cual consiste en elegir un objeto, sacarlo de su sitio y colocarlo en algún lugar, como por ejemplo posicionar un producto dentro de una caja de cartón. Para ello es necesario que un sistema integrado calcule la posición del objeto a mover y luego un robot se posicione en dicha posición lo tome y lo coloque en el lugar deseado [4], este movimiento debe realizarse a grandes velocidades y con gran precisión además de repetirse una gran cantidad de veces por minuto para mantener un ciclo de trabajo constante dentro de la línea de producción.

En términos generales casi cualquier robot puede realizar la tarea de *pick and place*, pero sin duda alguna los robots de cadenas cinemáticas Delta destacan sobre las demás configuraciones, ya que poseen ventajas en términos de velocidad y precisión, permitiéndoles desplazarse considerables distancias en fracciones de segundo. Por ello el estudio y desarrollo de este tipo de estructuras es de vital importancia para el desarrollo de la robótica.

En la actualidad el desarrollo de estructuras robóticas de tipo delta se ha quedado en conceptos del pasado, pocos son las mejoras desde los planteamientos de Raymon Clavel. En investigación se siguen utilizando robots comerciales e industriales los cuales son costosos y de limitado acceso para los grupos de investigación con bajo presupuesto, por ello en muchos de estos casos se utilizan impresoras de tipo delta modificadas de menor costo para realizar avances en el área.

1.3 JUSTIFICACIÓN

En la actualidad son escasos los investigadores que realmente buscan innovación con las estructuras delta, proyectos como “*Kinematically Redundant Spatial Parallel Mechanisms for Singularity Avoidance and Large Orientational Workspace*”[5] de la universidad de Laval busca estar a la vanguardia de la tecnología y el desarrollo de las plataformas de tipo delta y sirven de inspiración para centros de investigación con menos recursos.

Por ello el presente trabajo desarrolla una perspectiva diferente a los conceptos básicos, retomando una propuesta experimental de Raymon Clavel, con el objetivo de hacer a los robots delta más rápidos y precisos, manteniendo una visión económica y destinada a la aplicación de manufactura aditiva de tipo FDM, buscando generar una iniciativa llamativa que incentive el desarrollo e investigación de este tipo de plataformas.

1.4 OBJETIVOS

1.4.1 Objetivo General

- Desarrollar una estructura robótica de tipo delta invertido en el contexto del proyecto de manufactura avanzada, integrada e inteligente del laboratorio LaDPRER.

1.4.2 Objetivos específicos

- Emplear una metodología de desarrollo de productos para el diseño del robot paralelo.
- Realizar un diseño con base en las especificaciones obtenidas de la metodología de desarrollo
- Generar una ingeniería de detalle para la selección e integración de los componentes electrónicos y mecánicos
- Construir un prototipo y realizar la validación mediante la simulación en un entorno de desarrollo matemático

2 CAPÍTULO 2

MARCO TEÓRICO Y ESTADO DEL ARTE

2.1 ROBOTS PARALELOS

La cadena cinemática de bucle cerrado es aquella en la cual los eslabones y las uniones están dispuestos de tal manera que generan por lo menos un bucle cerrado distinto de la base, con un grado de conectividad mayor o igual a 3. Un robot manipulador paralelo puede definirse como un mecanismo de bucle cerrado compuesto por una plataforma donde de efector final de n grados de libertad y una base fija, los cuales son enlazados a través de por lo menos dos cadenas cinemáticas independientes véase la Figura 1.

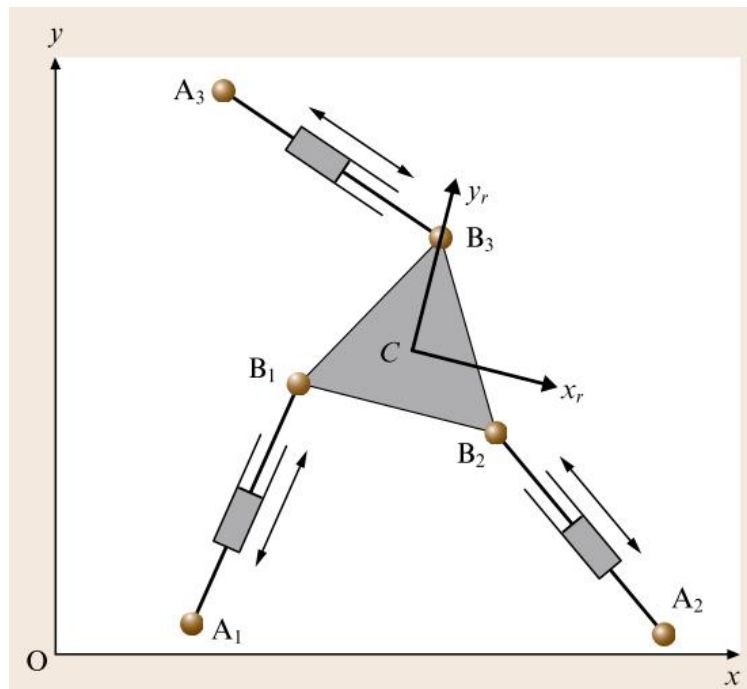


Figura 1 Configuración básica de un robot delta, con 3 actuadores lineales [6]

2.1.1 Historia

Uno de los primeros diseños de un dispositivo de un mecanismo esférico paralelo fue propuesto por el diseñador James E. Gwinnett (véase la Figura 2) quien solicitó la patente en 1928 un par de años después de la primera película a color y con audio.

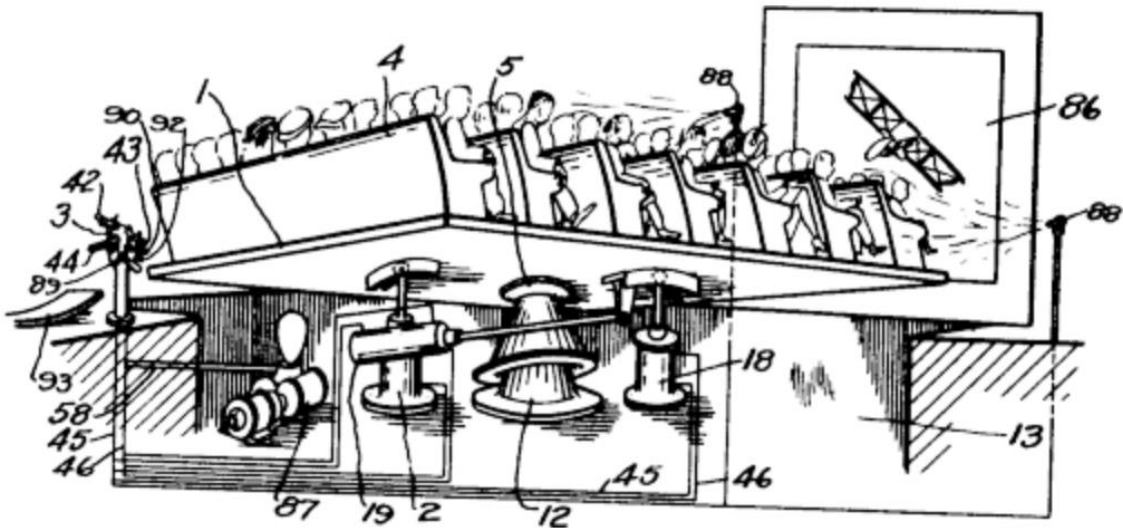


Figura 2 Posiblemente el primer mecanismo espacial paralelo, patentado en 1931 (Patente de Estados Unidos N° 1.789.680) [7].

En 1950 Gough construyó el “primer Hexápodo octaédrico” una plataforma robótica diseñada para simular el desgaste de los neumáticos por las cargas del Aero-aterrijaje. Véase la Figura 3.

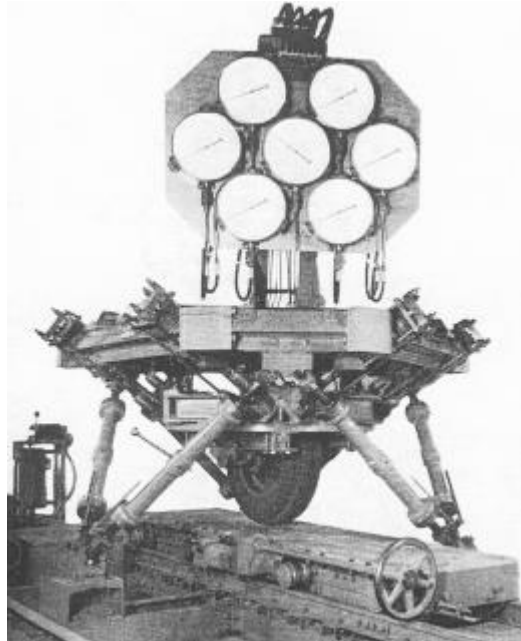


Figura 3 El primer hexápodo octaédrico o plataforma de Gough 1945 [8].

En 1965 Stewart describe un mecanismo similar al de Gough, el cual posee 6 grados de libertad (DOF) cada uno controlado por un actuador independiente de tipo lineal y con tope a tierra (véase la

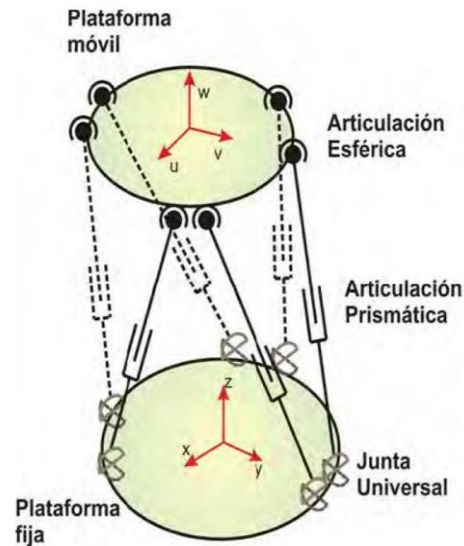


Figura 4), postulado como mecanismo para simulación de vuelo en la formación de pilotos de helicópteros [9].

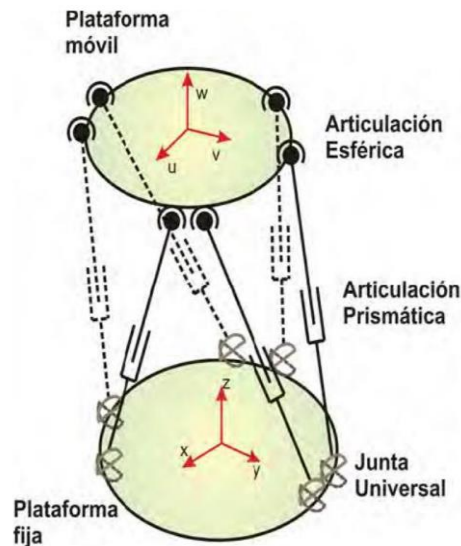


Figura 4 Esquema de una plataforma Stewart- Gough [4].

Dichos sistemas se han popularizado gracias a la facilidad de realizar pequeñas variaciones o ajustes de la junta los cuales son fáciles de interpretar. Los sistemas de este tipo o ligeramente modificados se conocen con el acrónimo MAST por sus siglas en inglés (*Multi-axis shaker table*).

2.1.2 Robots paralelos en la industria

En el desarrollo de la cinemática paralela el robot de *Pollard* es conocido como el primer robot de diseño industrial, compuesto por cinco grados de libertad (DOF), con una disposición de tres motores que controlan el mecanismo de posicionamiento y dos para la orientación, fue diseñado con el propósito de automatizar el proceso de pintura en aerosol para carrocerías de vehículos, su patente está dividida en dos partes la primera describe el sistema de control eléctrico, mientras que la segunda describe el diseño mecánico del manipulador véase la Figura 5.

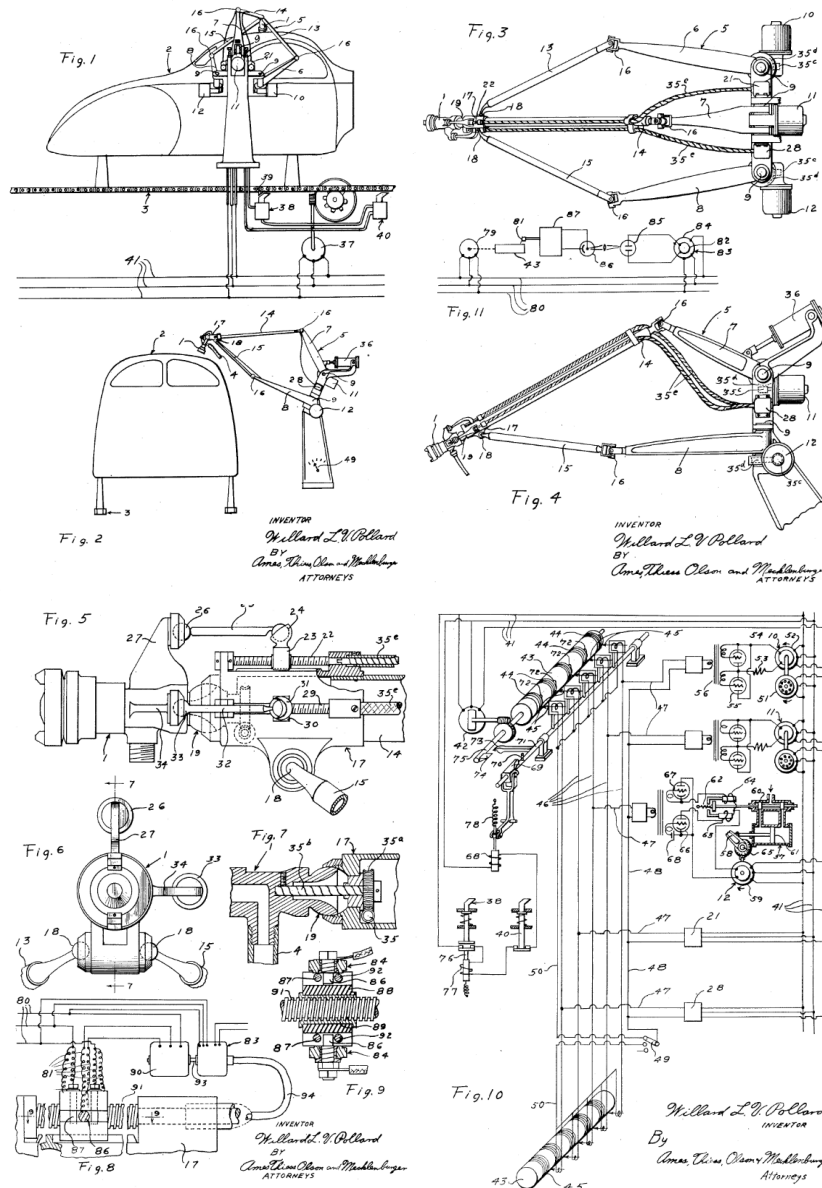


Figura 5 Planos anexos a la patente Position controlling apparatus 22 de abril de 1938 [5].

En 1980 el profesor del EPFL (*École polytechnique fédérale de lausanne*) Raymond Clavel modifico el concepto de plataforma Stewart, debido a problemas con el análisis y

costos elevados de fabricación del robot delta, modifiqué el concepto dando origen al robot manipulador delta de 3 grados de libertad (DOF), con la adición de paralelogramos en los brazos secundarios. Clavel aseguro que los grados de libertad del robot fueran solamente transnacionales véase la Figura 6.

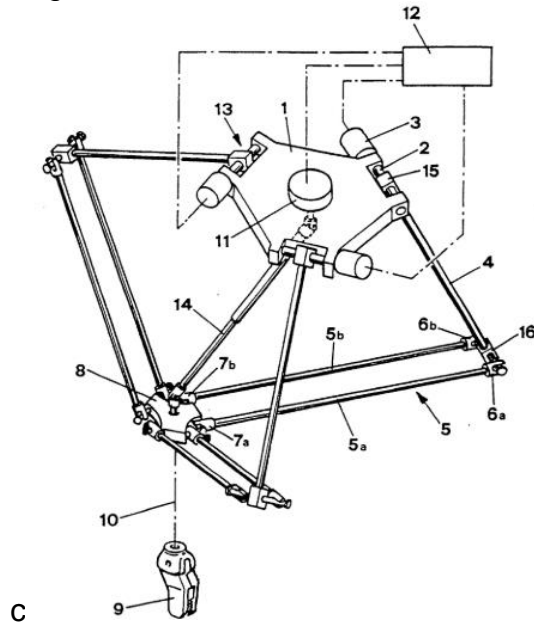


Figura 6 Diseño planteado por Reymon Clavel en la patente EPO250470B1 [7].

Gracias a estos planteamientos, los robots delta se han incorporado a gran escala en la industria, logrando generar avances tecnológicos en busca de mejorar las fortalezas de este tipo de manipuladores.

2.1.3 Algunas aplicaciones de los robots paralelos

Gracias a su velocidad de actuación y precisión al momento de realizar los desplazamientos los manipuladores tipo delta pueden ser utilizados en los siguientes campos [10] :

- Simuladores de vuelo y automóviles
- Maquinado de piezas
- Posicionamiento de precisión
- Cirugía robótica
- Robótica para rehabilitación
- Líneas de montaje del sector de la automoción
- Procesos de palatización, embalaje y empaquetado

También procesos de trabajo de alta velocidad, en los que se requiere un posicionamiento de alta precisión con espacio de trabajo limitado, tales como el montaje de PCB o fresado de alta precisión.

2.2 MANUFACTURA ADITIVA

(AM-Aditive Manufacturing), Es un proceso de unión de materiales mediante el cual se recrean objetos a partir de modelos 3D exportados en archivos STL (Stereolithography) de los cuales se extrae información sobre la geometría del objeto, generalmente recreados con la superposición de capas, es lo el proceso opuesto a las técnicas de fabricación sustractivas convencionales [11].

El desarrollo de esta tecnología está formado por una amplia lista de métodos para su aplicación, aunque casi todas las máquinas de impresión 3D existentes funcionan de manera similar. La aplicación comienza por la creación de un modelo tridimensional modelado generalmente mediante programas CAD (Computer Aided Design) [11] posteriormente se implementa un procesamiento digital de laminación o rebanado mediante el cual el modelo 3D es dividido en secciones planas bidimensionales 2D las cuales pueden ser reconstruidas en código G el cual utilizan los diferentes métodos de AM [12].

2.2.1 Procesos comunes de Manufactura aditiva

Desde sus inicios a mediados de la década de 1980, AM ha evolucionado abarcando una gran cantidad de procesos, una reciente estandarización de actividades de la ASTM (*American Society of Testing Materials*) y la ISO (*International Organization for Standardization*) reúne estos procesos en siete categorías, descritas a continuación [13].

Stereolithography: SLA o estereolitografía en español es un proceso de fabricación el cual utiliza una resina líquida de fotopolímero el cual al ser expuesto a un rayo de luz UV (radiación ultravioleta) genera una reacción que hace que el polímero se solidifique. El plato de construcción o eje z es el que se desplaza hacia abajo en pequeños incrementos en donde se crea un espacio con polímero sin curar en donde el láser UV dibuja una sección transversal por capa, repitiéndose el proceso curándose capa a capa hasta concluir el modelo [14] véase la Figura 7.

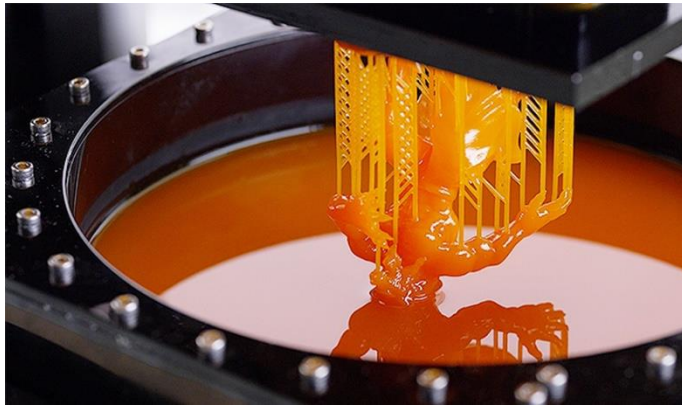


Figura 7 Modelo en proceso de impresión por método de estereolitografía [18].

Los modelos generados por este método son de baja resistencia, pero alto nivel de detalle, además por ser sensibles a los rayos UV tienden a deformarse o cambiar de color al ser expuestos a la luz solar, por ello sus aplicaciones son limitadas a prototipos no funcionales o moldes para fundición de inversión.

The selective laser sintering process: SLS o proceso de sinterización laser selectiva, similar a la SLA utiliza polímero en polvo, generalmente nailon el cual se transfiere desde el recipiente contenedor a la etapa de construcción a través de una herramienta de repintado. Posteriormente un láser de alta potencia escanea selectivamente la capa delgada de polvo, sinterizando las partículas del mismo en la forma de la sección transversal de la capa, mientras simultáneamente este proceso une la capa anterior a la capa actual, luego el plato de construcción desciende y se continúa el proceso hasta ser concluido el modelo [15] véase la Figura 8.

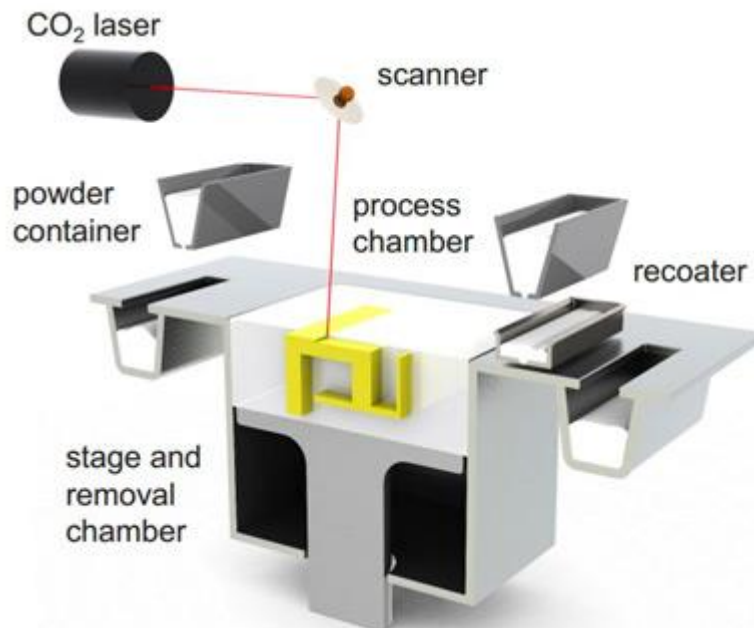


Figura 8 Diagrama de impresión 3D por método SLS [16].

A diferencia de otros tipos de manufactura aditiva, este proceso no requiere material o piezas de soporte, ya que el polvo actúa como material auto aportante, lo que permite realizar piezas de geometría compleja y totalmente completas, sin necesidad de un pos procesamiento de la misma.

Binder jetting: Inyección por aglutinante en español es un proceso el cual inyecta liquido aglutinante sobre una capa de polvo de metal, capa por capa, según sea el modelo. En cada una de las capas el polvo es curado ligeramente para la solidificación y cuando la impresión se ha completado (véase la Figura 9), la caja de ensamblaje es retirada por completo y colocada en un horno para curado. Posteriormente la pieza es retirada cuidadosamente de la caja y se elimina el polvo sobrante con cepillos o soplores de aire [17].

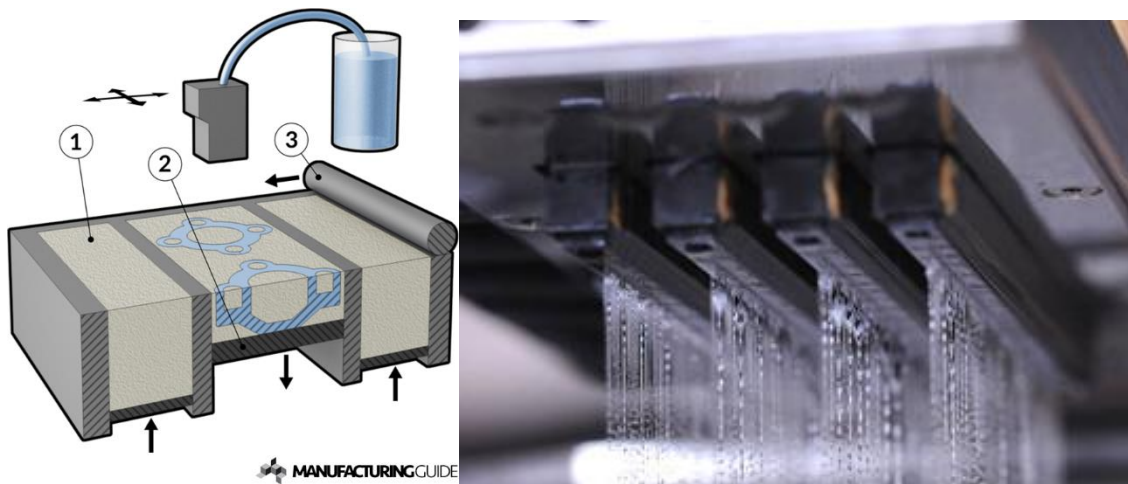


Figura 9 (a) Diagrama de una maquina Binder jetting [18] (b) Cabezal de inyección del material aglutinante[17][19] .

Este proceso es el más barato y rápido de impresión 3D con material metálico, perfecta para la creación de objetos ornamentales, decorativos y joyas, gracias a su corto tiempo de fabricación y excelentes acabados brillantes luego del pulido véase la Figura 10.



Figura 10 Diferentes acabados después del proceso de impresión Binder jetting [17].

Photopolymer jetting: También conocido como *polyjet* utiliza un método de inyección de fotopolímero en pequeñas gotas, el cual reacciona con una lámpara UV ubicada al lado del cabezal de impresión creando una solidificación instantánea, el plato de impresión se mueve hacia abajo para continuar con la siguiente capa. El grosor de una capa suele estar en el rango de 14 a 28 micras, logrando un alto nivel de detalle [20].

Este tipo de impresoras normalmente está compuesto de dos o más cabezales de inyección, uno de los cuales inyecta material de soporte para la pieza y los demás diferentes tipos de polímeros o variaciones de color véase la Figura 11.



Figura 11 (a) Impresiones Poliyet con materia de soporte (b) Impresión Poliyet con material transparente luego de ser pulido [20].

Entre sus aplicaciones se encuentran la impresión de piezas de geometrías complicadas con alto nivel de detalle, prototipos multilateral, impresiones totalmente transparentes acabado logrado luego de un proceso de pulido y lustrado, además de moldes para fundición al vacío y fundición de metales.

Selective deposition lamination: SDL Es un proceso de reconstrucción 3D el cual utiliza láminas de papel superpuestas con adhesivo entre ellas, en donde una hoja metálica muy afilada, recorta el trozo de papel pegado a la forma de la capa en proceso, luego se adiciona adhesivo la base de construcción baja y se agrega otra lamina de papel, repitiendo el proceso hasta concluir el modelo [21].

Una de sus principales ventajas con respecto a otros métodos de AM es el bajo costo de la materia prima, pues este tipo de impresión puede utilizar desde hojas de formato A4 y papel carta estándar, papel iris los cuales son mucho más baratos que las resinas y polímeros utilizados en otras técnicas. Adicional al método de fabricación, existen impresoras SDL con tecnología de inyección de tinta lo cual permite la recreación de

modelos a todo color con resoluciones de color hasta los 4800 x2400 ppp en el plano bidimensional XY y 254 en Z véase la Figura 12.



Figura 12 Modelos a todo color de impresos por the Mcor IRIS [25]

Directed Energy Deposition: BED Es un proceso de impresión complejo, el cual utiliza un sistema de boquilla montada en un brazo multi eje que deposita material fundido sobre una superficie especificada, al no estar sujeta directamente a un eje, la boquilla puede posicionarse en una superficie con diversos ángulos de inclinación sin comprometer la tarea véase la Figura 13. Utiliza un alambre de metal preformado o material metálico en polvo, el cual es fundido mediante laser, plasma o haz de electrones alcanzando temperaturas elevadas de 1.000 y 5.000 grados centígrados por segundo que generan la fundición del material dando como resultado una microestructura sólida. Los espesores típicos entre capa varían de 0.25 a 0.5 mm [22].

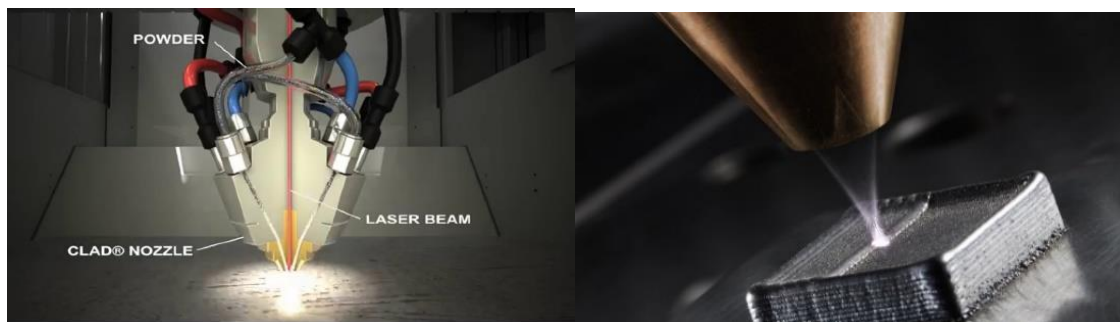


Figura 13 (a) Cabezal de proceso BED (b)Proceso de impresión BED [27].

Entre sus aplicaciones se encuentra, soldadura de objetos, aplicaciones de reparación, adición de características e impresión 3D. Adicionalmente es utilizada en el ámbito de investigación de materiales, gracias a la combinación de dos o materiales metálicos de aporte en hilo y polvo.

Fused deposition modeling: FDM, sin duda alguna el proceso más conocido y popular de la manufactura aditiva debido a la gran cantidad de impresoras de escritorio disponible en el mercado, desde kits ensamblables de impresora 3D hasta maquinas industriales de alta calidad. Siendo la tecnología más asequible en comparación al resto de tecnologías.

Funciona mediante la fusión de material plástico, el cual originalmente tiene una presentación de hilo el cual es empujado en el cabezal de extrusión hasta una boquilla precalentada a temperatura de fusión del material. El cabezal de extrusión se desplaza a lo largo del plano XY depositando el material fundido en la forma de la capa, posteriormente el plato de construcción baja y se repite el proceso hasta la culminación del modelo véase la Figura 14.

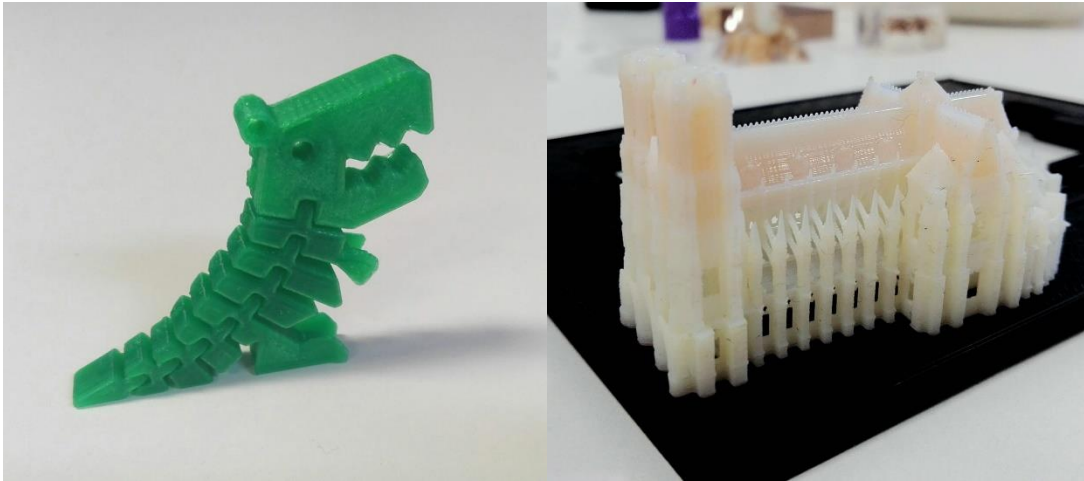


Figura 14 Modelos impresos en 3D FDM (Fuente autor)

Las resoluciones más comunes para este método de AM varían entre los 0.05 mm hasta los 0.6 mm dependiendo de la impresora utilizada, adicionalmente existen algunas máquinas equipadas con sistemas de cabezal múltiple, lo cual permite la impresión hasta en 3 colores o materiales diferentes. Siendo esta tecnología la de mayor globalización son muchas las aplicaciones que pueden realizarse por este método, a continuación, se enunciaran algunas de ellas [23].

- Arquitectura y construcción
- Industria naval
- Prototipito rápido
- Aplicaciones educativas y de investigación
- Análisis funcional de modelos mecánicos
- Robótica
- Industria química
- Bioimpresión 3D
- Impresión de instrumental quirúrgico
- Prótesis personalizadas

- Implantes óseos
- Impresión 3D de tejidos vivos
- Biomodelos
- Industria aeronáutica
- Aplicaciones aeroespaciales
- Fabricación de herramientas
- Impresión de tejidos para el sector textil
- Joyería
- Calzado
- Entre otras

Manufactura Híbrida: AMBIT puede definirse como un proceso multitarea el cual combina manufactura aditiva con métodos convencionales de mecanizado sustractivo, El cual permite realizar un modelo base mediante la adición de material que luego al cambiar el cabezal permita realizar tareas sustractivas como el acabado de superficies perforaciones, roscados y demás véase la Figura 15.

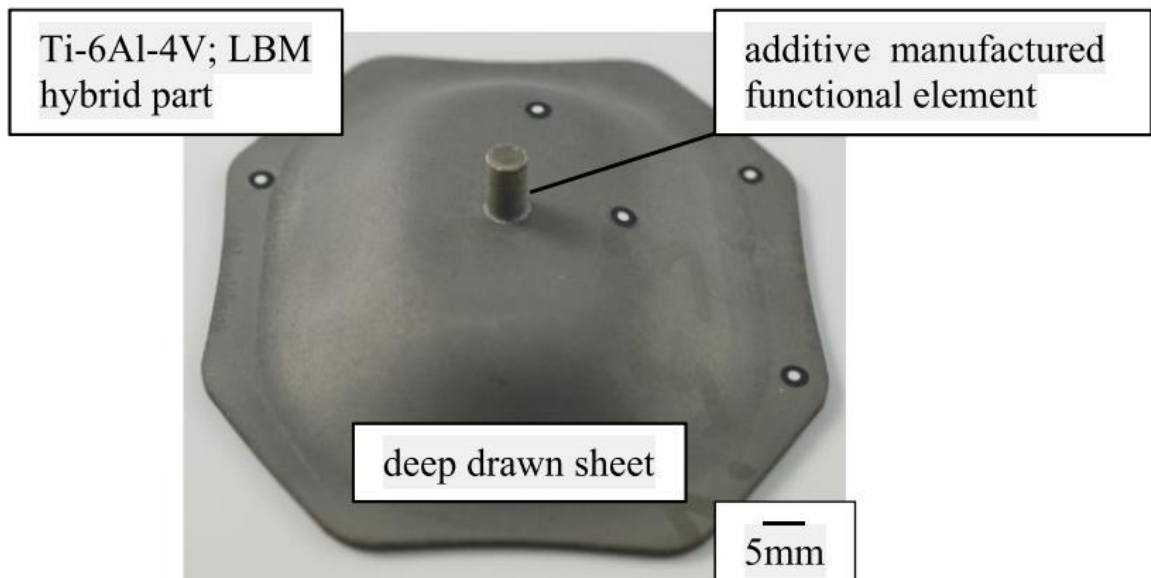


Figura 15 Pieza híbrida fabricada por combinación de embutición profunda y LBM [29].

Esta nueva apreciación permite mejorar los procesos de mecanizado reduciendo los tiempos de fabricación y residuos de fabricación [24], optimizando el uso de materia prima y energía además de permitir la creación de piezas personalizadas en materiales

metálicos manteniendo características de acabado y precisión, actualmente existen centros de mecanizado CNC que incorporan esta tecnología véase la Figura 16.

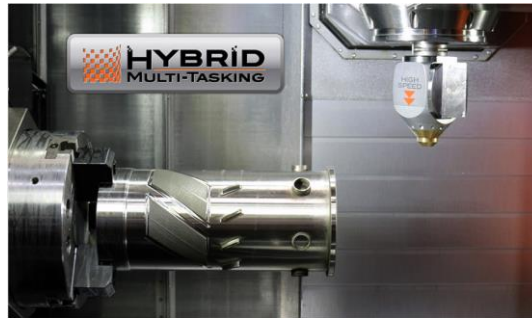


Figura 16 Cabezal de revestimiento AMBIT en el nuevo husillo INTEGRIX i-400AM de Mazak con características depositadas con láser presentado en JIMOF 2014 [30].

Proyecto RepRap

Replicating rapid prototyper, creado por Adrian Bowyer, profesor titular de ingeniería mecánica en la universidad de Bath, Reino Unido es una iniciativa cuyo objetivo es desarrollar máquinas AM en donde la mayoría de sus piezas puedan ser impresas y auto replicables. RepRap es el proyecto con la comunidad más grande en diseño (véase la Figura 17), desarrollo y construcción a nivel mundial gracias a sus modelos free CAD, bajo costo de construcción y diversidad de modelos y posibilidad de mejoras [25].

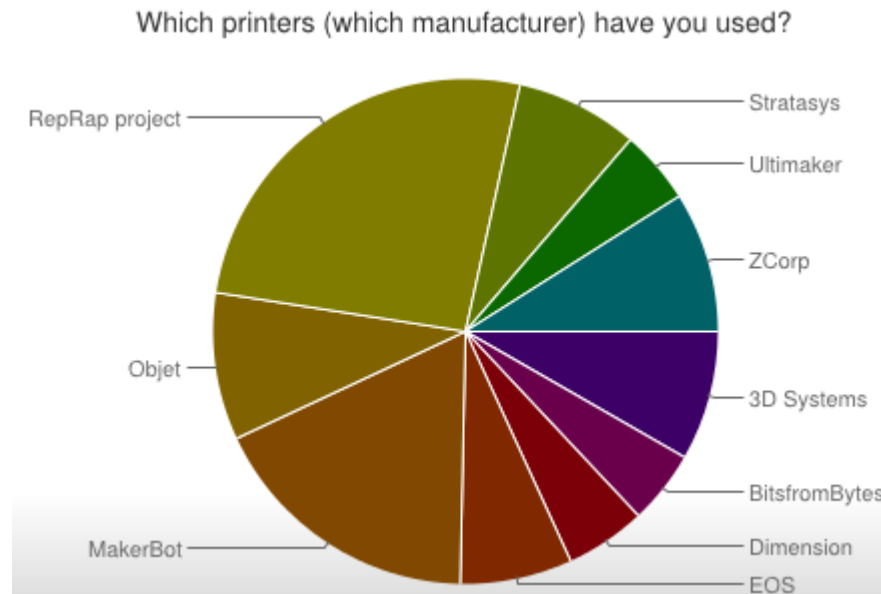


Figura 17 Gráficos de usuarios de impresión 3D [26].

Este proyecto cambio los paradigmas de la manufactura aditiva y la creación de productos de consumo personalizados, originalmente pensado para la utilización en aplicaciones de RP (*rapid prototype*) el proyecto permitió a la AM del método de FDM evolucionar rápidamente, abarcando un sin número de aplicaciones relacionadas en el desarrollo de prototipos, diversos tipos de juguetería, figuras de acción, artículos personalizados e investigación en general.

Gracias a la gran acogida de esta tecnología y su rápido crecimiento en la cantidad de usuarios, dentro del proyecto *RepRap* se mejoraron las estructuras de cinemática cartesiana existentes y se crearon nuevos modelos para este sistema de desplazamiento, también se incorporaron robots de tipo paralelos con diferentes cinemáticas, tales como escara y delta lineal siendo este último el más sobresaliente, véase la Figura 18.

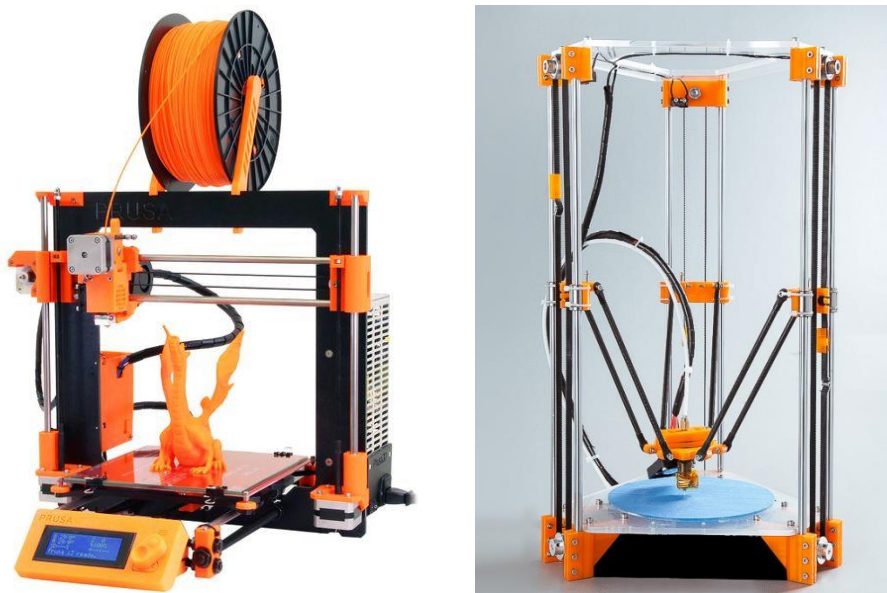


Figura 18 Prusa i3 y minirostock RepRap proyet [26]

Comparativa entre cinemática cartesiana y delta

Dentro de las implementaciones de robots para manufactura aditiva los modelos cartesianos y delta lineal son los más convencionales y fáciles de encontrar en el mercado. Ambos modelos comparten semejanza en la utilización de piezas o materiales de construcción, tales como electrónica, rodamientos, guías lineales, motores entre otros, pero la forma en que realizan los movimientos para realizar una trayectoria son totalmente diferentes (véase la Figura 19) , a manera de ejemplo mientras que en una impresora cartesiana para realizar una línea recta sobre un eje coordenado solo se requiere la acción de uno de sus motores, en un delta es necesario los 3 actuadores en conjunto para realizar la misma acción.

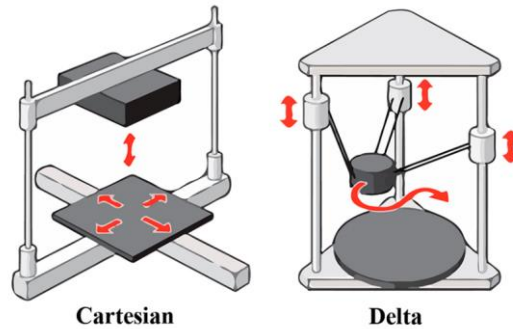


Figura 19 Esquema de impresoras 3D [36].

La aplicación de cada una de estas configuraciones trae consigo ventajas y desventajas propias de cada configuración, a continuación, se enumeran algunas de estas apreciaciones enfocadas a la implementación de impresoras 3D para manufactura aditiva FFF.

Tabla 1 Ventajas y desventajas de una impresora cartesiana

Ventajas	Desventajas
Fácil calibración	Velocidades de impresión limitadas por las inercias
Análisis y corrección de errores intuitivos	Movimiento de la cama de impresión
Gran cantidad de recursos, diseños e información disponible	Movimiento de grandes masas incluso en pequeños desplazamientos sobre todo en uno de los ejes que por lo general tiene que soportar al otro (normalmente el eje z soporta la estructura del eje x)

Tabla 2 Ventajas y desventajas de una impresora delta

Ventajas	Desventajas
Posibilidad de desarrollar grandes velocidades y aceleraciones	Calibración complicada
Mayor precisión en los desplazamientos relacionados a mejoras en la definición	Menor volumen de trabajo en relación con el volumen del equipo
Cama de impresión estática	Errores en la estructura provocan calibración errónea o insatisfactoria
Fácil redimensionamiento	Poca información disponible (modelos limitados)

Menor cantidad de motores que un sistema cartesiano tradicional	Identificación difícil de errores de construcción- calibración (no se posible separarlos con facilidad)
Posibilidad de realizar con facilidad trayectorias complejas	

Principales materiales utilizados en la impresión 3D FDM

En la implementación de impresión 3D FDM existen diversos tipos de materiales los cuales difieren tanto en colores, texturas y características especiales, lo que permite ampliar las aplicaciones para esta tecnología.

Principalmente los materiales utilizados son Termoplásticos, los cuales son provistos en hilos de diámetro 1.75 mm o 3mm, en paquetes de 1 kg principalmente, aunque en la actualidad también existes presentaciones de mayor y menor cantidad, dicho material es calentado hasta la temperatura de fusión y es forzado a pasar a través de la boquilla de extrusión. A continuación, se mencionarán los más comerciales, hacia los cuales esta propuesto el alcance del presente trabajo.

- **ABS** o *acrilonitrilo butadieno estireno* es un termoplástico de gran resistencia y flexibilidad media, posee un punto de extrusión entre los 190°C y 250°C, principalmente utilizado en piezas que requieran someterse a altas temperaturas o que requieran pos procesamiento (pintura, lijado, pulido, etc.).
- **PLA** o *ácido poliláctico*, es sin duda uno de los más usados en la aplicación de impresión 3D FDM, de características biodegradables ya que su obtención es derivada del almidón, posee un punto de extrusión entre los 180°C y los 230°C, por su gran variedad de colores incluyendo colores fluorescentes y traslucidos es ideal para todo tipo de prototipados
- **PVA** o *alcohol polivinílico* es muy utilizado principalmente como material de soporte ya que es soluble en agua tibia lo que facilita su remoción, se extruye en temperaturas entre los 170°C-195°C
- **PET** o *tereftalato de polietileno*, industrialmente utilizado en botellas plásticas y recipientes, posee propiedades cristalinas ideales para fabricar piezas cristalinas fuertes y resistentes a impactos, posee puntos de extrusión entre los 210 y 220 y punto cristalino a más de 450°C

2.3 TRABAJOS RELACIONADOS

Académicamente el desarrollo e investigación de los robots paralelos es un campo de gran actividad, los investigadores de este campo plantean el uso de nuevas estructuras que permitan aumentar los grados de libertad (GLD), también estudian la reconfiguración de las estructuras ya conocidas analizando sus impactos en la precisión, área de trabajo, velocidad y consumo energético. Otros campos de investigación están en la aplicación de este tipo de estructuras en la medicina y dispositivos hápticos, A continuación, se presentan algunas de estas investigaciones.

La universidad de Australia occidental (*University of Western Australia*) destaca en el campo del desarrollo experimental con su versión NUWAR (New UWA Robot) el cual es capaz de alcanzar aceleraciones de hasta $500 \frac{mm}{s^2}$ al modificar la configuración de distribución de sus actuadores (véase la Figura 20), y mejora de los algoritmos de control utilizando ecuaciones de Hamilton para la solución del problema dinámico cuyo resultado mejora la precisión del algoritmo de control sugiriendo una actualización de la señal del mismo cada milisegundo [27].

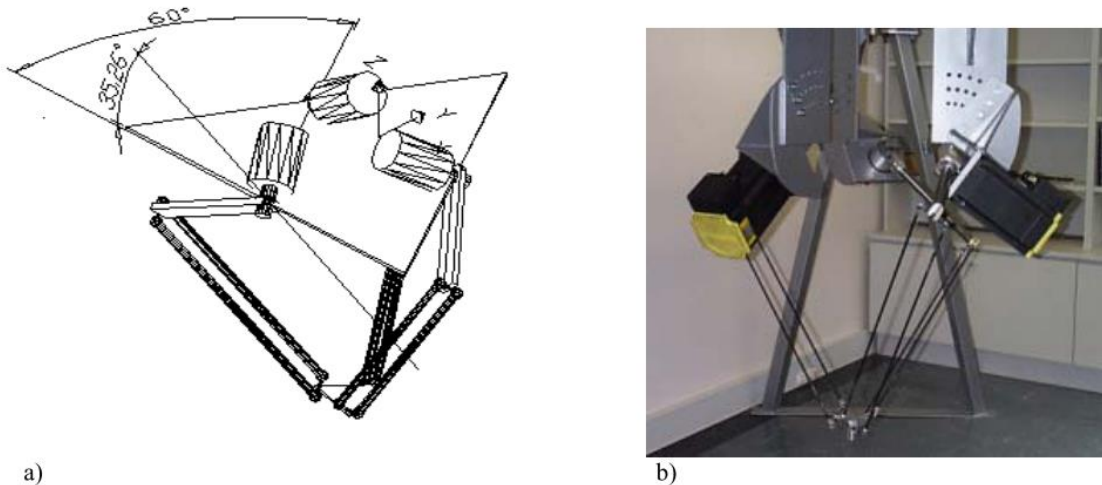


Figura 20 "New University of western Australia Robot- NUWAR" (a) diseño (b) Prototipod [8].

La universidad Autónoma de México desarrolló y evaluó el uso de contrapesos (véase la Figura 21) para reducir los torques ejercidos sobre los motores, una selección de detalle para los mismos presentando simulaciones en NX Unigraphics, en donde se muestran las distintas trayectorias y las exigencias de esfuerzo para cada uno de los motores además de presentar simulaciones sobre la integridad estructural del prototipo a través del análisis de elementos finitos [28].

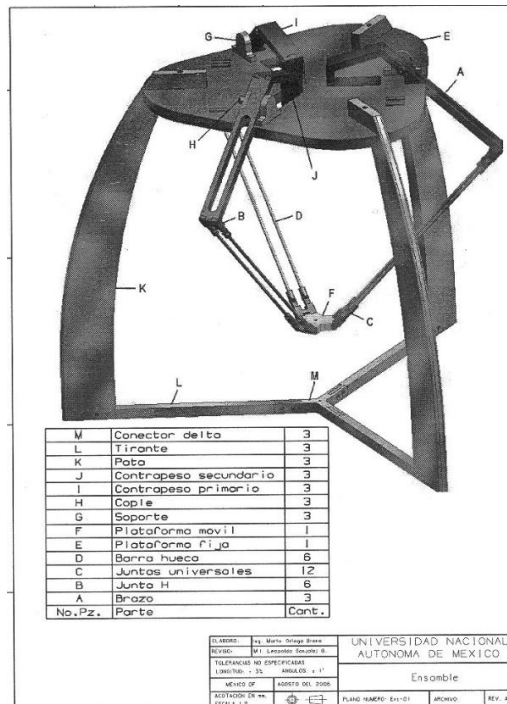


Figura 21 Plano de ensamble con contrapesos [9].

La universidad pontificia de Bucaramanga el cual abarca el desarrollo completo de un manipulador tipo delta diseñado en Solid Edge y simulación en ANSYS (véase la Figura 22) realizando una selección de material y variación del espesor de las chapas, obteniendo como resultado los esfuerzos producidos cargas, y los puntos de mayor concentración de los mismos [29].

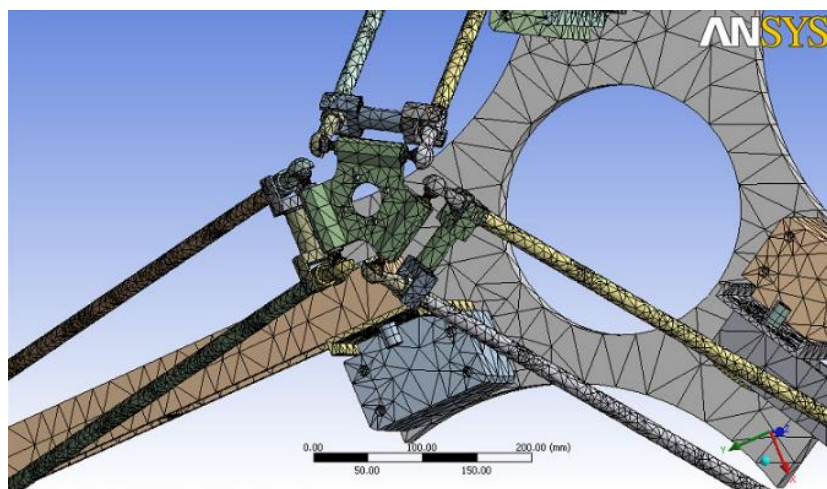


Figura 22 Malla final para simulación en ANSYS [29].

La Universidad de Pamplona ha realizado grandes aportes tales como la optimización dimensional de un robot paralelo tipo delta utilizando algoritmos genéticos para el cálculo de las longitudes óptimas de los eslabones (véase la Figura 23 donde se muestra la

convolución volumétrica resultado de la aplicación del GA) en busca de reducir el consumo energético [30].

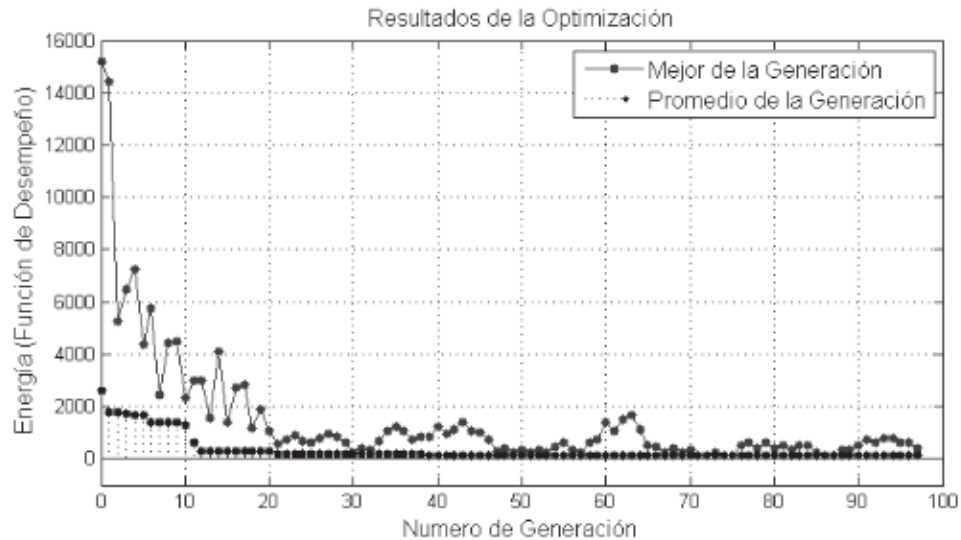


Figura 23 Resultados de la optimización a través de algoritmos genéticos [10].

A demás ha desarrollado el robot paralelo delta KEOPS (véase la Figura 24) el cual posee una estructura modificable variando los ángulos de inclinación de los actuadores, lo cual afecta directamente el área de trabajo y permite cambiar la tarea efectuada por el manipulador. Además, presenta las etapas de estudio, diseño y optimización basada en algoritmos genéticos para la optimización de las dimensiones y arquitectura del control implementado, regulando corriente velocidad y posición de los motores Brushless DC del cual está compuesto [31].

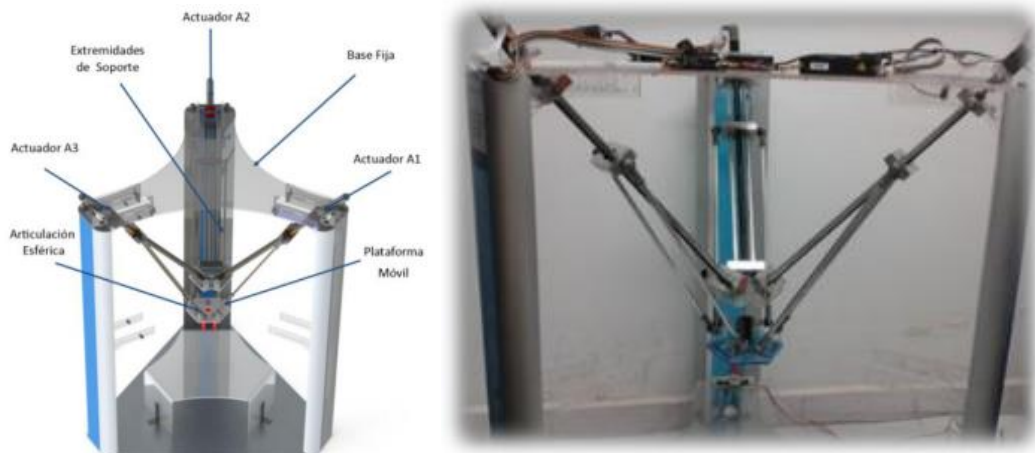


Figura 24 (a)Estructura del robot KEOPS (b)Prototipo fabricado del robot KEOPS [12].

En “Nikola Vaptsarov Naval Academy” de Varna Bulgaria, los investigadores desarrollaron un prototipo experimental en escala 1:10 de un robot paralelo de actuación

por cables para deposición de material de construcción (véase la Figura 25). El proyecto denominado “Delta Wired” plantea la construcción de una impresora 3D de gran escala la cual sea capaz de construir edificaciones utilizando concreto en deposición por capas para ello las pruebas iniciales del prototipo utilizo material de granel.

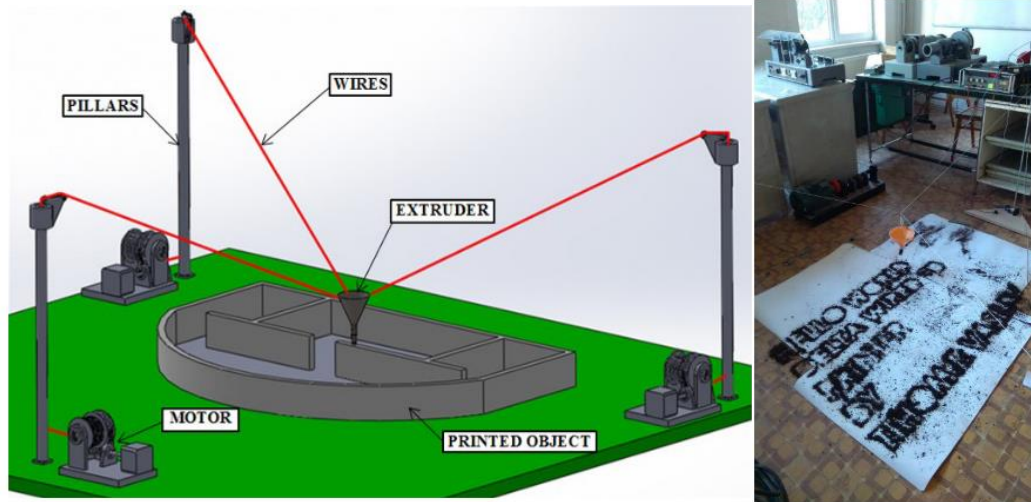


Figura 25 (a)Diseño conceptual del robot "Delta Wired 3D Printer" (b)Pruebas iniciales del prototipo físico [37].

Estudiantes de la universidad Militar de Nueva Granada en Colombia implementaron una técnica de planeación y ejecución de trayectorias para un robot delta utilizando el entorno de desarrollo matemático Matlab generaron una interfaz gráfica GUI (véase la Figura 26) sobre la cual se podría ejecutar una de dos trayectorias, variando la velocidad de ejecución y presentando la posibilidad de simular o ejecuta la trayectoria obtenida [32].

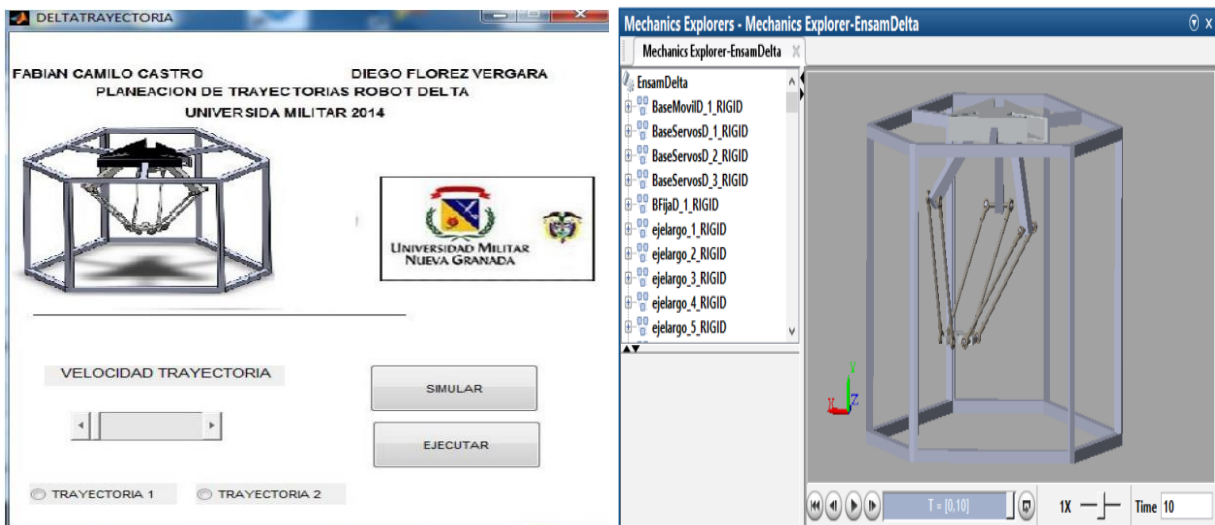


Figura 26(a) Interfaz GUI (b) Simulación ejecutada en Matlab [32].

2.3.1 Robot delta en la manufactura aditiva FFF

El método **FFF** el cual permite la aplicación de manufactura personalizada de manera asequible y económica, perteneciente a la **AM** una de las líneas transversales que soporta la industria 4.0 con gran proyección de crecimiento a futuro, una gran variedad de modelos de estructuras se ha implementado para la aplicación de esta tecnología, siendo los de cinemática cartesiana los que poseen la mayor diversidad.

Las estructuras con cinemática paralela han sido introducidas a la FFF como alternativa viable a los modelos cartesianos tradicionales gracias a permitir mayores aceleraciones y velocidades, sin perder rigidez ni precisión.

En el instituto federal de Santa Catarina, el *Grupo de Pesquisa em Processos de Fabricação e Tecnologia dos Materiais*, realizó un estudio comparativo titulado *A Comparative Study of Cartesian and Delta 3D Printers on Producing PLA Parts* [33], en el cual se compara el resultado de impresión teniendo en cuenta comparaciones dimensionales con su modelo 3D, acabados superficiales y cantidad de tiempo necesario para producir el mismo modelo.

Como resultado se fabricó un modelo de geometría compleja replicado 3 veces (véase la Figura 27) con los cuales se llegó a la conclusión que la impresora Rostock Max sobresalió en términos de acabado y tiempo de impresión,

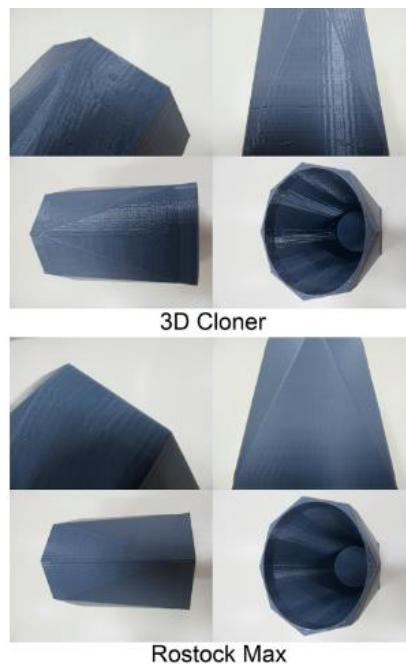


Figura 27 Resultado comparativo de los modelos impresos por 3D cloner y Rostock Max [33].

Estudiantes de ingeniería de la universidad de Galati Diseñaron una impresora tipo delta utilizando proyección 3D en entorno grafico Autodesk inventor (véase la Figura 28),

enfatisado principal mente al diseño, este proyecto planteo la utilización de una estructura robusta y un extrusor *bowden* de metal con el fin de aumentar la vida útil e intervalos de mantenimiento de la impresora [34].

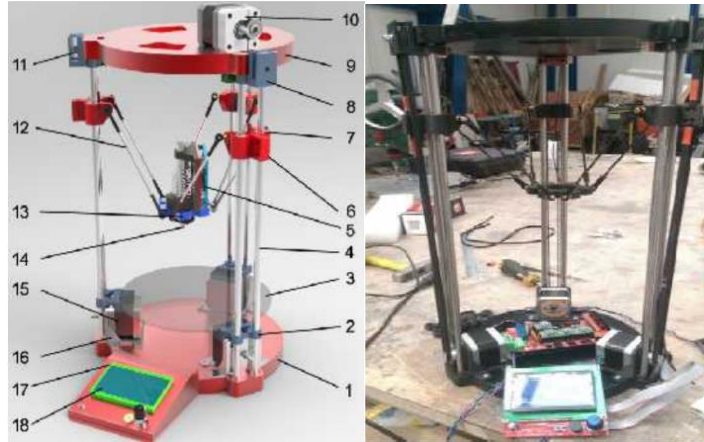


Figura 28 en la figura de la izquierda el diseño virtual, en la figura de la derecha el modelo terminado real [34].

Estudiantes de maestría de la universidad de ZHAW implementaron un prototipo funcional de impresora delta lineal con la adición de dos grados de libertad rotacionales a la base del robot, lo cual les permitió realizar impresiones con grandes ángulos de inclinación y objetos esféricos sin la necesidad de utilizar material de soporte [35] véase la Figura 29.

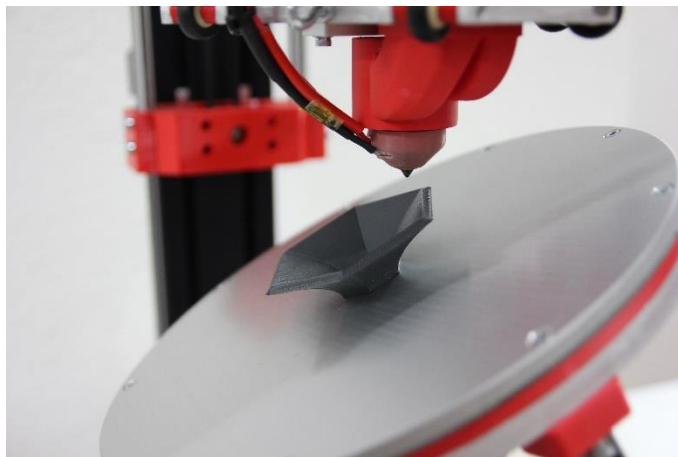


Figura 29 Impresión de figura compleja con inclinación de la cama de impresión [35].

Uno de los proyectos que realmente marca una pauta en el desarrollo de los robots delta es el desarrollado en la universidad de Brasilia, titulado *Conceptual desing and dimensional optimization of the linear delta robot with single legs for additive manufacturing* su desarrollo envuelve el proceso completo desde el planteamiento del problema hasta la implementación de un prototipo físico. Uno de los puntos más llamativos de esta propuesta es la reformulación de la construcción de los eslabones los

cuales reemplazan las juntas esféricas convencionales o juntas universales por juntas rotacionales formándose así 12 enlaces que conforman las piernas del robot [36] véase la *Figura 30*.

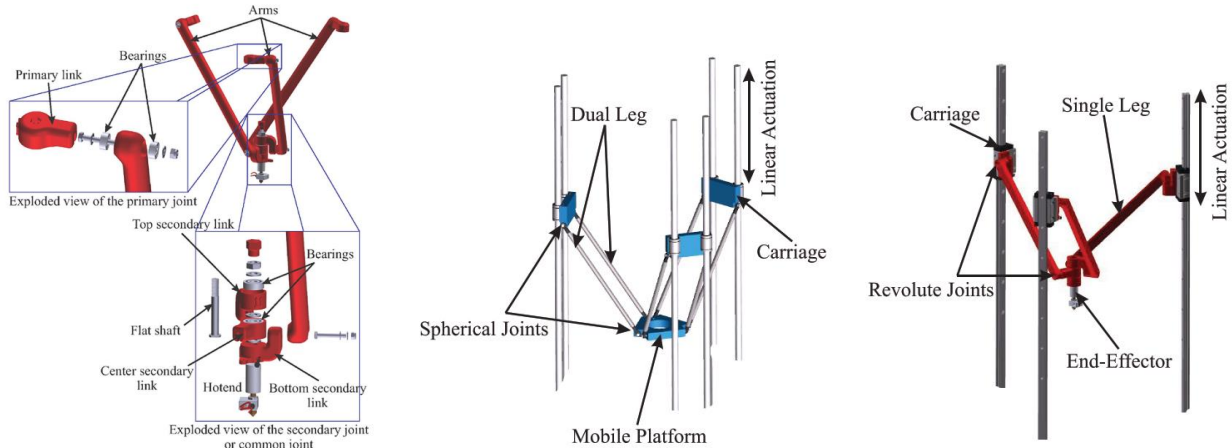


Figura 30 (a) Detalle de las juntas rotacionales implementadas (b) juntas universales convencionales (c) Propuesta desarrollada en este proyecto [37].

Uno de las pocas implementaciones de la inversión cinemática de un robot delta, es la utilizada por el proyecto GUS Simpson [38] en donde su autor Nicholas Seward a este concepto para desarrollar un prototipo (véase la *Figura 31* Prototipo GUS Simpson ensamblado, tomado de la librería Rep Rap [38]., el proyecto presenta un desarrollo complejo y poco documentado, como consecuencia su replicación es escasa y no existe una guía clara y precisa para duplicarlo.

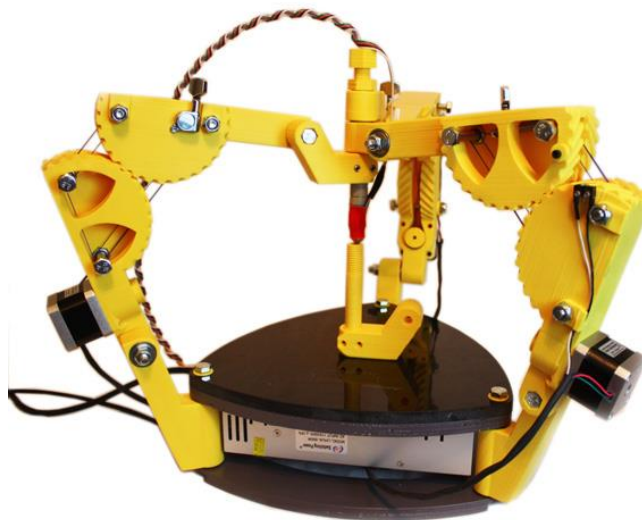


Figura 31 Prototipo GUS Simpson ensamblado, tomado de la librería Rep Rap [38].

3 CAPÍTULO 3

METODOLOGÍA

3.1 DESCRIPCIÓN DEL ROBOT DELTA INVERTIDO PARA MANUFACTURA ADITIVA FDM (ID-AM)

Diseño conceptual: el robot **ID-AM** estará compuesto por tres piernas distribuida de manera simétrica en el espacio cartesiano. Para poder definir el robot es necesario iniciar por un punto de referencia, “**O**” de coordenadas **[0 0 0]**, el cual también coincidirá con el centro del triángulo isósceles **ABC** circunscrito en el círculo de radio **FBr**. La base fija del robot o **FB** (*Fixed Base*) está compuesta por el triángulo **ABC** sus vértices están distribuidos uniformemente a 120° , los que a su vez serán el punto de anclaje de las piernas del robot, el punto **A** se encontrará en la posición **[FBr 0 0]** véase la Figura 32.

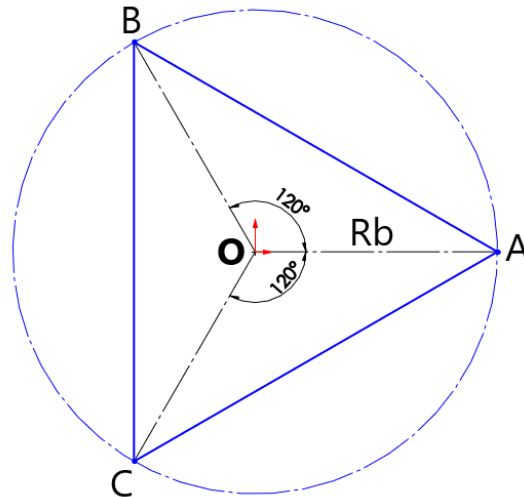


Figura 32. Distribución de la base fija de la plataforma DI-AM en vista normal al plano XY (Fuente autor)

“**P**” **[X Y Z]** se definirá como la posición deseada de la plataforma móvil o **MB** (*Mobile Base*) a la cual estará sujeta la herramienta del robot, se definirá al igual que la plataforma fija, como un triángulo “**abc**” circunscrito en un círculo de radio **MBr** cuyo centro coincidirá con el punto deseado “**P**” véase la Figura 33.

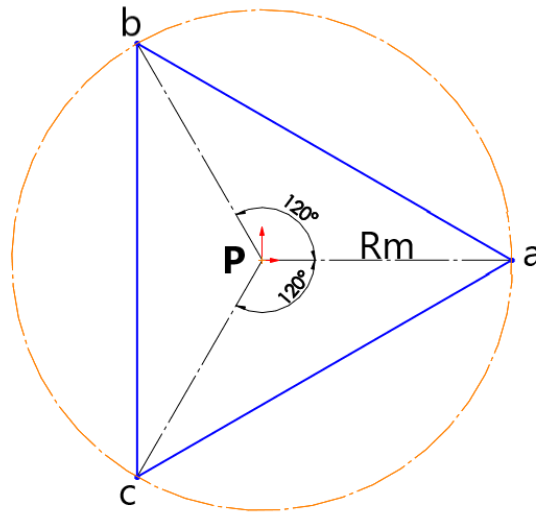


Figura 33. Distribución de la plataforma móvil del DI-AM en vista normal al plano XY (Fuente autor)

Las piernas del robot componen la cadena de bucle cerrado, uniendo la base fija del robot y la plataforma móvil, estarán compuestas de 3 estructuras principales, un pivote de distancia Pv que servirá como radio de separación para evitar la restricción de movimiento entre la base fija y las piernas del robot, y dos eslabones simétricos de longitud L entre los cuales se encontrará la variable articular q , respectivamente para cada una de las piernas que conforma el robot **DI-AM**, véase la Figura 34.

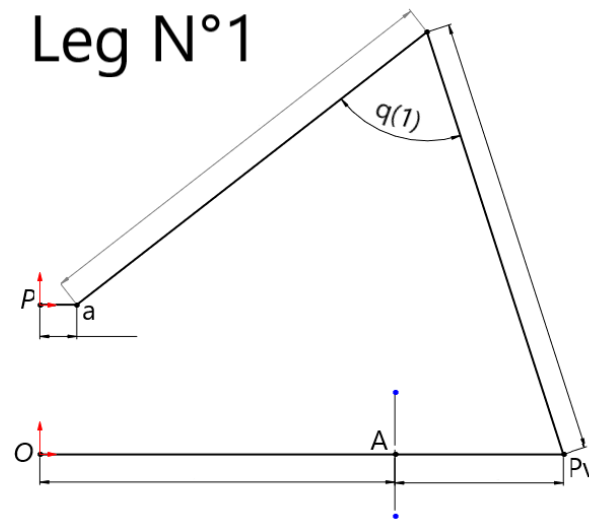


Figura 34. Vista normal de la primera pierna del robot en el plano XZ (Fuente autor)

3.2 IMPLEMENTACIÓN DE QFD

Teniendo sentadas las bases conceptuales las cuales y las variables asociadas a cada una de las dimensiones del robot, es necesario acoplar ingeniería de detalle para el modelado y construcción del mismo. Este proceso de naturaleza compleja e

interdisciplinaria, involucra áreas de diseño mecánico, electrónico y programación, por ello para el desarrollo del Robot delta invertido para manufactura aditiva (**ID-AM**) se plantea la utilización de una metodología estructurada que respalde y sintetice la toma de decisiones, correlacionando los requerimientos planteados con los requisitos de los posibles usuarios.

El método de gestión de la calidad **QFD** (*Quality Function Deployment*) es sin duda, uno de los más conocidos, utilizado y efectivo mediante el cual los diseños de un producto se pueden organizar, además de relacionar de manera positiva o negativamente los atributos correlacionados de interés [39]. **QFD** ha sido exitosamente utilizada en diversos productos, prototipos y trabajos referentes a la ingeniería mecatrónica tales como robots agrícolas [40], robots de servicio [41], robots submarinos [42], desarrollo de impresoras 3D [43], entre muchas otras aplicaciones. Para información más detallada sobre la metodología QFD visite las siguientes referencias [44], [45] .

En el presente desarrollo se aplicará la metodología **QFD** para el desarrollo del robot **ID-AM** en el cual se tendrán en cuenta los requerimientos de los posibles usuarios (**RU**) y los requerimientos de diseño dentro de la casa de la calidad, el cual tendrá como resultado los requerimientos más importantes a tener en cuenta en la construcción del prototipo.

3.2.1 Paso 1: Identificación de los usuarios

En un inicio es necesario clasificar las máquinas de manufactura aditiva en 3 subcategorías las cuales son Industriales, profesionales y de bajo costo.

Las impresoras de categoría “bajo costo” se caracterizan por ser de fácil adquisición o fabricación con tamaños reducidos ideales para el uso sobre escritorio, principalmente modelos *Free Cad* y *open source* de la biblioteca de RepRap, o variaciones comerciales de estos modelos, comparten entre si electrónica, componentes mecánicos, piezas de fabricación y Código de funcionamiento, “*Marlin*” principalmente, en general cualquier tipo de robot podría implementarse para la construcción de una impresora 3D, pero en la categoría de bajo costo resaltan solo dos tipos de cinemática, se encuentran las de tipo cartesiana y delta, esta última de menor acogida por la comunidad.

El desarrollo del robot DI-AM se ubica en la última categoría, en donde los usuarios serán principalmente aficionados, estudiantes, profesores, universidades, fablabs, laboratorios de prototipado, y pequeñas empresas en el sector cuyos nichos estarán en las artes y moda (joyas, calzado, piezas de ropa) arquitectura (edificios, maquetas, mobiliario a escala) e ingeniería (prototipos de piezas, mecanismos móviles, ensambles, encajes).

3.2.2 Paso 2: Definición de las RU

La selección de las principales características, se basan en el conocimiento adquirido mediante la fabricación de prototipos de impresoras 3D del proyecto RepRap, y los comentarios generados por la comunidad, los cuales resaltan las principales características y mejoras aplicadas a los diversos modelos.

Dicha información se organizó en la tabla 2 en la cual se declaran parámetros de interés los cuales tienen relación directa con las características de funcionamiento y tamaño.

Tabla 3 Parámetros de interés para la construcción del robot DI-AM

Categorías	Parámetros de interés
1. Capacidad	Calidad de impresión
	Velocidad de movimiento
	Volumen de trabajo
	Tipos de material soportados
2. Economía	Costo de fabricación
	• Materiales estructurales
	• Actuadores
	• Electrónica
	• Software libre
• Diseño	• Costo de mantenimiento
	• Tamaño de la maquina
	• Número de piezas
	• Facilidad para el ensamble
	• Estética general
	• Facilidad de modificar
	Facilidad de adicionar nuevas tecnologías

3.2.3 Paso 3: Priorización de los requerimientos

La relevancia de los RU fue recolectada mediante la encuesta online

(https://docs.google.com/forms/d/e/1FAIpQLSfOFY3pS7ZW9fPZpoz6VpebKSHBFUs_OJXEMXmI0H09xZ48Fg/viewform?fbclid=IwAR2AQNXXJUdXTFIJ-_CXkUgdzthYcmH3S04XlIsj6lcTEDh3jqI0oX6XkQ)

Uno de los impactos esperados es motivar a la comunidad investigativa a incluir y desarrollar proyectos utilizando Robots de arquitectura paralela por esta razón la encuesta presentada fue aplicada a estudiantes y profesores de la Universidad de Pamplona y Universidad de Brasilia incluidos en los programas de ingeniería mecatrónica, electrónica y mecánica principalmente. En estas carreras es común que los estudiantes desarrollen robots a pequeña escala por lo cual tienen conocimientos sobre la terminología, aplicaciones, y componentes utilizados para la implementación de estos mecanismos.

De acuerdo a lo anterior las respuestas obtenidas de la población objetivo a la cual se aplicó la encuesta, están sustentadas por conocimientos técnicos, teóricos y prácticos, haciéndolas altamente confiables para la sustracción de conclusiones.

A continuación, se presentan las gráficas resumen de las respuestas obtenidas de la encuesta Figura 35.

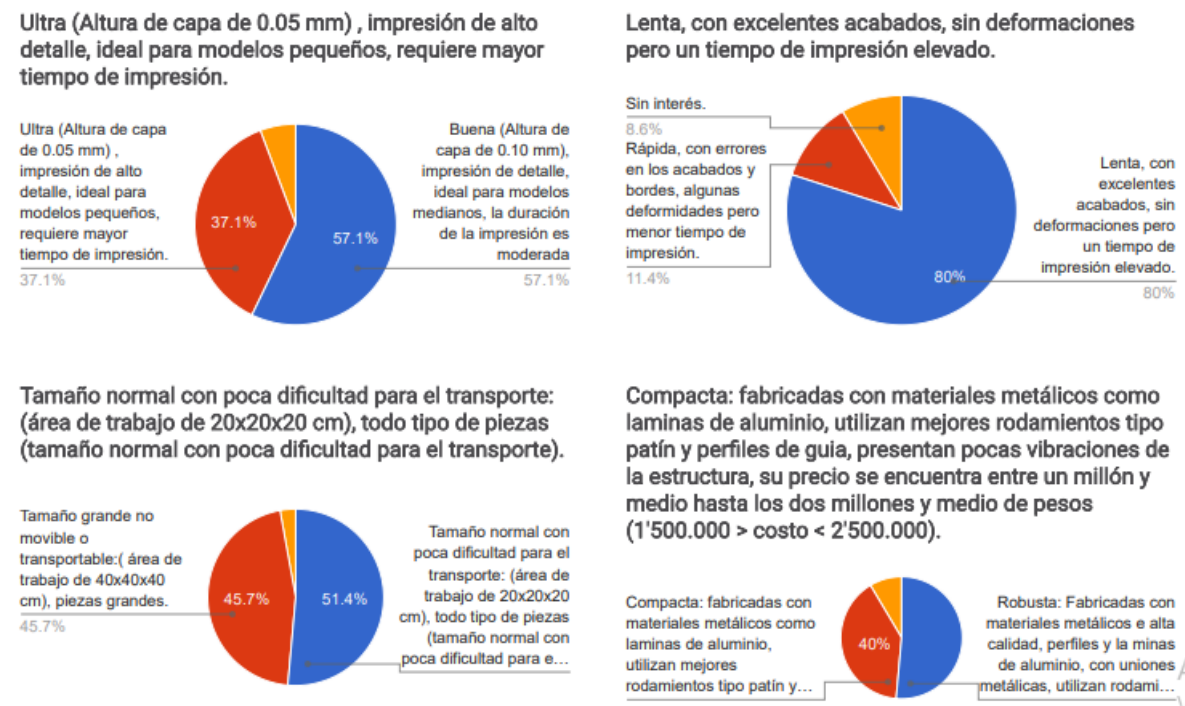


Figura 35 Graficas relevantes de los requerimientos de los usuarios (Fuente autor)

Luego de analizar los datos de las gráficas anteriores es posible sintetizar que los requerimientos más importantes para los usuarios son:

- Alta precisión dimensional
- Alta definición altura de capa (0.05)
- Tiempo de impresión elevado
- Estructura de tamaño normal
- Área de trabajo cilíndrica de 100 R *100 h
- Estructura robusta

3.2.4 Paso 4: Selección de los parámetros de diseño (DP)

Por concepto propio del diseñador, utilizando el conocimiento adquirido tras la fabricación y ensamble de múltiples impresoras, se delimitaron los siguientes parámetros de diseño

- Altura de impresión: Influencia directa en la resolución de la impresión 3D
- Configuración de los drivers de motor: influencia directa en la resolución de impresión y velocidad de impresión.
- Volumen útil de trabajo: selección del volumen máximo de impresión
- Software multiplataforma: Posibilidad de utilizar software gratuito disponible
- Rango de control de temperatura: Influencia directa en la variedad de materiales soportados
- Interfaz gráfica del usuario: Además de un elemento de seguridad y control, permite la utilización de memoria tipo micro *USB* con *licer* precargado, eliminando la necesidad del programa de impresión.
- Diseño modular: Pensado en la facilidad de ensamble y replicación del prototipo
- Numero de partes: pensado en la facilidad de replicación del prototipo también tiene influencia en el costo total del mismo.
- Estrategia de montaje: el diseño de las piezas estructurales y ensamblaje con las partes mecánicas y electrónicas, el cual permita facilidad a la hora del mantenimiento.
- Estética de diseño
- Costo total del prototipo: prototipo de bajo costo.
- *Software* y *hardware* abierto: La utilización de estos permite que el prototipo sea replicado con facilidad en cualquier parte del mundo.
- Costo de mantenimiento: bajo:
- Tipos de materiales soportados: influencia directa en el tipo de extrusor a utilizar

- Elementos de seguridad: influencia en el costo de fabricación.

3.2.5 Paso 5: Relación entre las RU y as DP

Para poder obtener resultados de la implementación de **QFD** es necesario relacionar los requerimientos de los usuarios con los parámetros de diseño de esta manera analizar cuál es la influencia que tienen entre sí, y como repercutirá en la delimitación de los objetivos de diseño del presente trabajo, para ello se realizó el siguiente procedimiento:

- En la parte superior a manera de columna se colocan los parámetros de diseño y en la fila vertical izquierda se colocan los requerimientos de usuario de esta manera los **RU** y **DP** se entrelazan en el recuadro central.
- En el recuadro central se relacionan de manera subjetiva los parámetros, utilizando las siguientes notaciones: Relación fuerte: •, Relación media: ○, Relación débil: ∇ según correspondan entre los **RU** y **DP**.
- Por último, se llena la parte superior con forma de triángulo también conocido como techo de la casa de calidad **QFD** la cual da una correspondencia positiva, negativa o nula para las relaciones existentes entre los **DP** y **RU**

A continuación, en la Figura 36, se presenta la implementación del procedimiento mencionado anteriormente, aplicando los valores correspondientes al diseño del robot DI-AM

Como resultado de la implementación de este modelo se crea una tendencia dinámica para cada **DP** del prototipo, los cuales crecen o decrecen según sean las relaciones establecidas por el diseñador entre los **RU** y **DP**, lo que permite resaltar claramente cuáles son los parámetros de mayor interés y proceder a delimitar los objetivos del desarrollo.

3.2.6 Paso 6: Definición de los objetivos específicos

Concluyendo con la aplicación del método **QFD** se obtiene un conjunto específico de normas influyentes en el desarrollo del robot **DI-AM** en la cual cada uno de estos parámetros está asociado a los objetivos a desarrollar.

Como resultado de la implementación de la matriz **QFD** se delimitan los siguientes objetivos para la realización del prototipo del robot **DI-AM**:

- Prototipo de bajo costo
- Utilización de piezas genéricas (open hardware)
- Área de impresión cilíndrica de 100mm*50mm
- Extrusor tipo *Bowden*
- Soporte de múltiples materiales de impresión
- Ensamblaje de tipo modular
- Software multiplataforma (open source)
- Resolución de 0.1 (altura de capa)
- Construcción con técnicas tradicionales de fácil acceso

3.3 CINEMÁTICA INVERSA

Puede definirse como el método por el cual se calculan los valores necesarios en cada una de las articulaciones de un robot para alcanzar una posición deseada [**X Y Z**]. Su solución este compuesto de procesos matemáticos y trigonométricos, los cuales buscan acoplar los parámetros físicos del robot de la manera más simple posible calculando las incógnitas articulares según los valores de posición dados.

El presente desarrollo presentara el análisis realizado para una articulación, cuyas ecuaciones se dejarán enunciadas genéricamente de manera que puedan ser aplicadas a las demás articulaciones, para ello se utilizara la variable “**i**” para indicar los índices de cada uno de los valores siendo el valor de **i**= [**1 2 3**].

- Como primera instancia se declara el punto de referencia “**O**” como punto de referencia en la posición **[0 0 0]** y el punto de coordenadas deseado como el vector “**P**” de coordenadas **[X Y Z]**.

NOTA: El robot DI-AM posee una distribución simétrica en las 3 piernas que constituyen su estructura, por lo cual todas las fórmulas pueden ser aplicadas con una variación de **i= [1 2 3]** que representan cada uno de los valores de interés en las 3 posiciones de distribución.

- Para las componentes articulares se plantea la utilización del vector de tres posiciones “**q**” **[q1 q2 q3]** que corresponderán a cada uno de las piernas del robot respectivamente, como simplificación se utilizara la notación **q(i)**.
- La posición de los puntos de la base fija del robot **[A B C]** se simplificará como **FB (Fixed Base)** o **FB(i)**.
- La posición de los puntos de la base móvil del robot **[a b c]** se simplificará como **MB (Mobile Base)** o **MB_i**.
- El punto de unión entre los dos eslabones que conforman las piernas del robot se denotara como **K (Knee)** o **K_i**.

Para el robot DI-AM se plantea la utilización de juntas rotacionales simples, que permitan el correcto desplazamiento dentro del espacio de trabajo, dichas juntas estarán ubicadas en cada uno de los puntos de la **FB_i** paralelas al eje **z**, en **MB_i** paralelas al plano **xy** además de una junta compartida en el centro de la base móvil paralela al eje **z** véase la Figura 37.

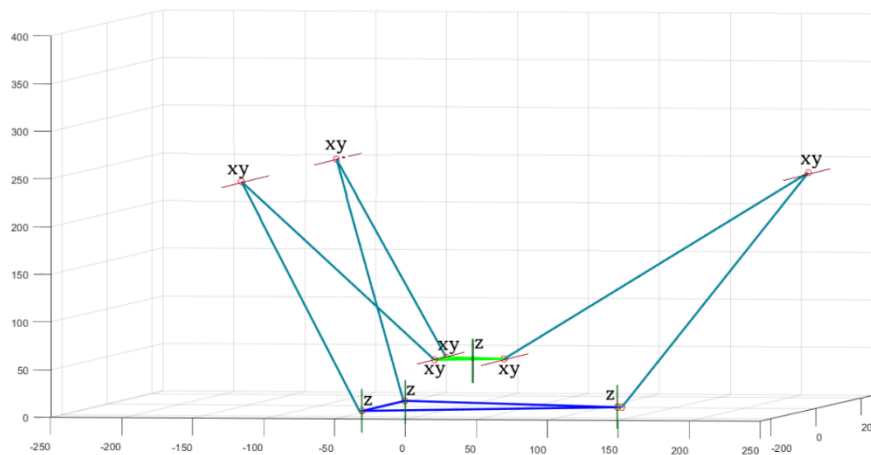


Figura 37 Representación grafica de los ejes de rotación de las juntas de la plataforma ID-AM (Fuente autor)

La configuración de las juntas planteada anteriormente permite al robot desplazarse en el espacio cartesiano $[X Y Z]$ moviendo el conjunto completo de las piernas del robot, las cuales forman un triángulo isósceles entre el conjunto FB_i , MB_i y K_i el cual mantiene las longitudes de sus lados y cambia el valor de su base dependiendo de la posición del efector final, esto también se puede expresar como la variación del ángulo de interés o valor angular del actuador q_i véase la Figura 38.

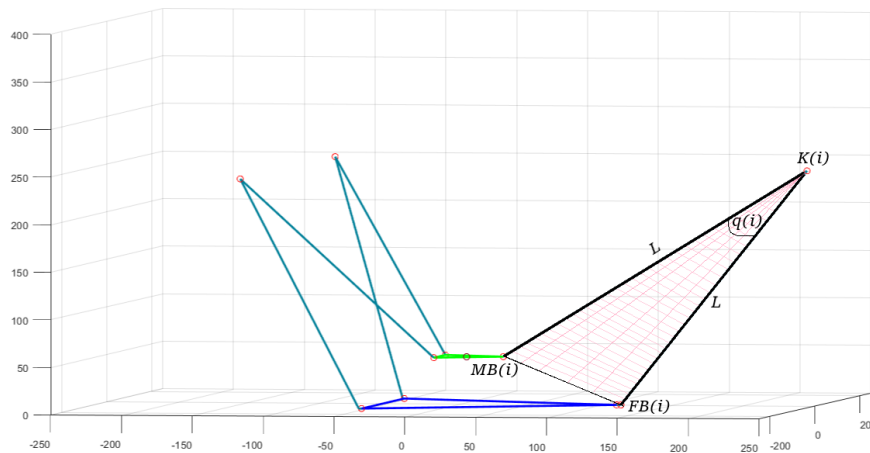


Figura 38. Resalto del triángulo isósceles generado entre los puntos MB, FB, K (Fuente autor)

Esto permite realizar un análisis trigonométrico teniendo en cuenta las siguientes condiciones

- La base del triángulo isósceles FB_i , MB_i y K_i debe calcularse como la norma del vector tridimensional $MB_i[x y z] - Pv_i[x y z]$ véase la Figura 39.

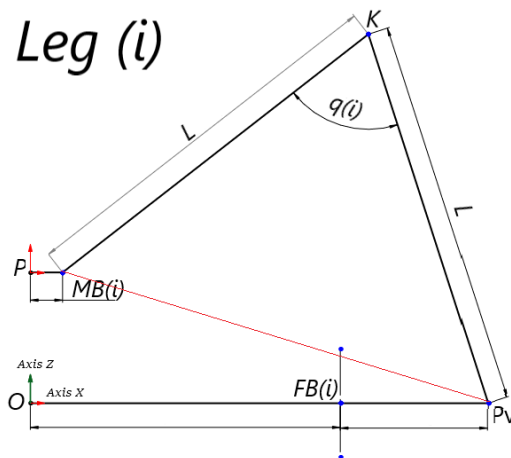


Figura 39. Vista normal a una de las piernas del robot DI-AM utilizando como referencia el plano XZ
(Fuente autor)

- Para calcular la posición del punto de pivote Pv_i es necesario tener en cuenta que este punto se mueve paralelamente a la sombra de la pierna del robot generando un ángulo dinámico **Alpha** el cual depende de la posición del efector final o punto “P” véase la Figura 40.

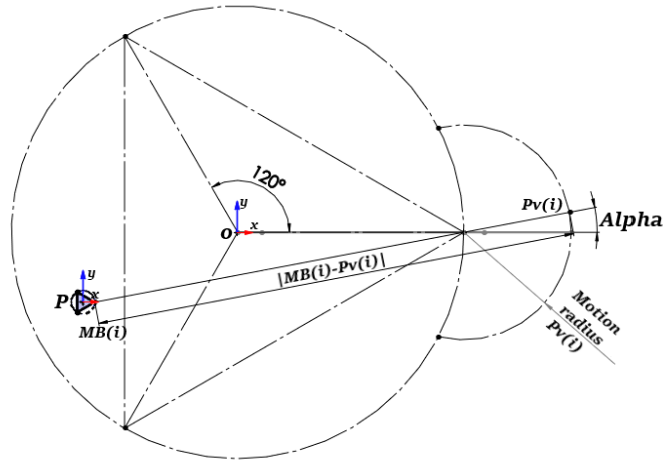


Figura 40. Figura con punto hipotético P en el cual se muestra la vista superior sobre el plano xy evidenciando el ángulo Alpha generado por la sombra ($MB_i - Pv_i$) de la pierna del robot DI-AM (Fuente autor)

- La sombra generada por la pierna del robot ($P - FB_i$) sobre el plano XY se simplificará como S (**Shadow**) o S_i véase la Figura 41.

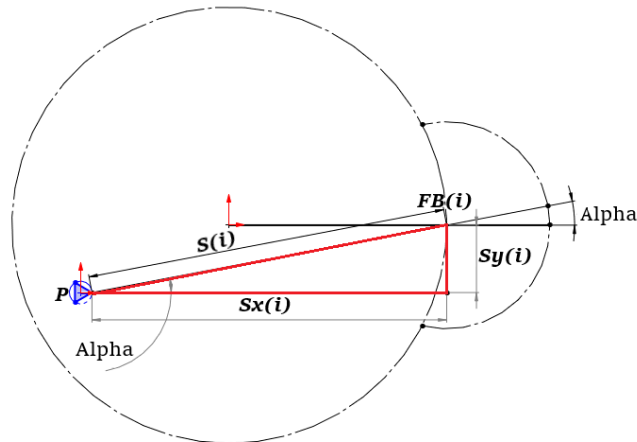


Figura 41. Vista superior normal al plano xy en la cual se evidencian las componentes cartesianas de la sombra S formada por la pierna (i) del robot

El valor del Angulo Alpha es de vital interés para conocer las coordenadas del punto Pv_i y del punto MB_i dada la posición del punto P . Para realizar el cálculo de este Angulo se utilizarán las componentes cartesianas de la sombra S_i dentro de la función **atan2** para conocer sus respectivos valores

$$Alpha = atan2(Sy_i, Sx_i)$$

El ángulo **Gamma** formado entre la sombra de la pierna y el plano **XY** es calculable utilizando el teorema de cosenos directores para vectores 3D [46], en él se utilizara la definición para el ángulo generado con el eje coordenado **Z** utilizando la siguiente ecuación.

$$Gamma = \cos^{-1}\left(\frac{Pz}{|S_i|}\right)$$

Mediante el análisis del triángulo isósceles de la Figura 38 es posible determinar el ángulo de interés utilizando la ecuación

Esta ecuación describe la solución al problema de cinemática inversa del robot DI-AM, en donde los valores cartesianos de P [x y z] se incluyen en el cálculo de la norma del vector $(MB_i - Pv_i)$ reduciendo de esta manera el costo computacional de su cálculo.

3.4 CINEMÁTICA DIRECTA

En contraparte al modelo cinemático inverso, la cinemática directa puede definirse como el procedimiento mediante el cual se calcula la posición cartesiana en **coordenadas [X Y Z]** dada las coordenadas articulares q [q_1 q_2 q_3].

Para el cálculo de este problema es posible utilizar el método de trilateración usualmente es utilizado para conocer la posición relativa de un punto en 2D utilizando tres posiciones conocidas y círculos concéntricos a dichas posiciones de tal manera que exista un punto en común entre los tres círculos véase la Figura 42.

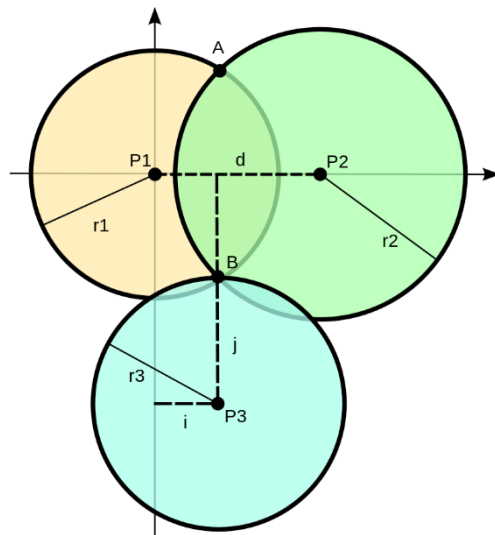


Figura 42 Posicionamiento esférico [47]

Usualmente esta metodología es usada para calcular posiciones geográficas de un dispositivo móvil utilizando como puntos de referencia balizas ultrasónicas para ello es necesario conocer la velocidad de propagación del medio y el instante de inicio de emisión de cada una de ellas. Luego se mide la distancia absoluta entre cada una de ellas y el dispositivo móvil, el cual estará situado en intersección de las circunferencias centradas en las posiciones de las balizas y cuyo radio es la distancia medida.

Aplicado al problema de cinemática directa, es posible conocer la posición del efector final aplicando una variación de la trilateración en 3D en el cual se plantea el problema en el espacio cartesiano tridimensional y se cambia la utilización de circunferencias por esferas véase la Figura 43.

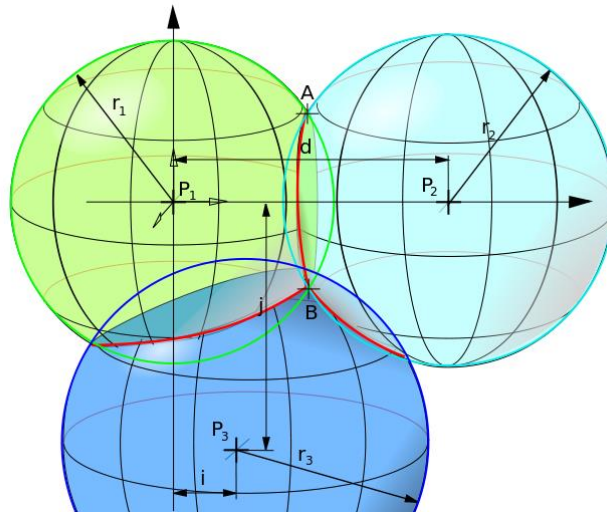
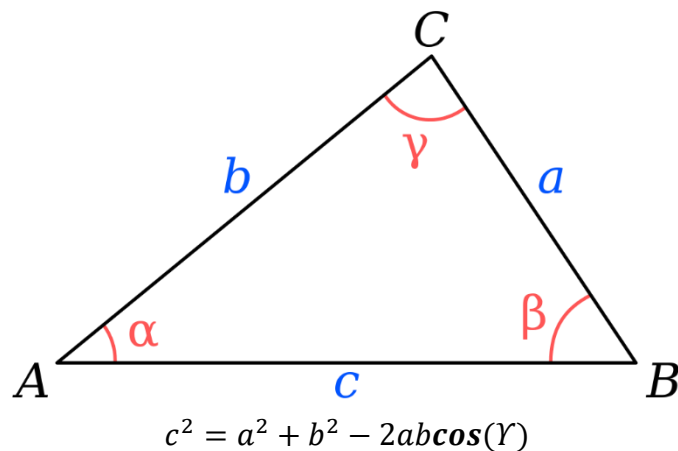


Figura 43 Representación tridimensional para el modelo de trilateración[47]

Aplicado al cálculo de la posición del efector final en el espacio cartesiano $[x\ y\ z]$ dadas las componentes articulares $[q_1\ q_2\ q_3]$ es posible aplicar el algoritmo de trilateración teniendo en cuenta las siguientes condiciones [48]:

- Se reemplazará el valor de los radios de las esferas por los valores correspondientes de la sombra (*shadow*) generada por cada una de las patas de la Plataforma **ID-AM**, para ello se utilizará la apreciación de triángulo isósceles descrita en la cinemática inversa (Capítulo 3-Cinemática inversa) analizando dicho triángulo mediante el del teorema del coseno [49] dado el triángulo **ABC**:



Intercambiando la nomenclatura del teorema del coseno por las notaciones aplicadas al triángulo conformado por los puntos MB_i, Pv_i, K_i que conforman la pierna (i) (Véase la Figura 38) del robot DI-AM obtenemos la siguiente ecuación:

$$Shadow_i^2 = L^2 + L^2 - 2LL\cos(q_i)$$

Despejando $Shadow_i$ tenemos

$$Shadow_i = \sqrt{L^2 + L^2 - 2LL\cos(q_i)}$$

Todos los puntos de referencia deben estar en el plano $z=0$ (cumple sin modificación) Uno de los puntos debe estar sobre el origen de coordenadas $[0\ 0\ 0]$. Para ello se aplicará la transformación de traslación una distancia de FBr en x negativo, de tal manera que el punto A de la base coincida con el origen "O" $[0\ 0\ 0]$ véase la Figura 44.

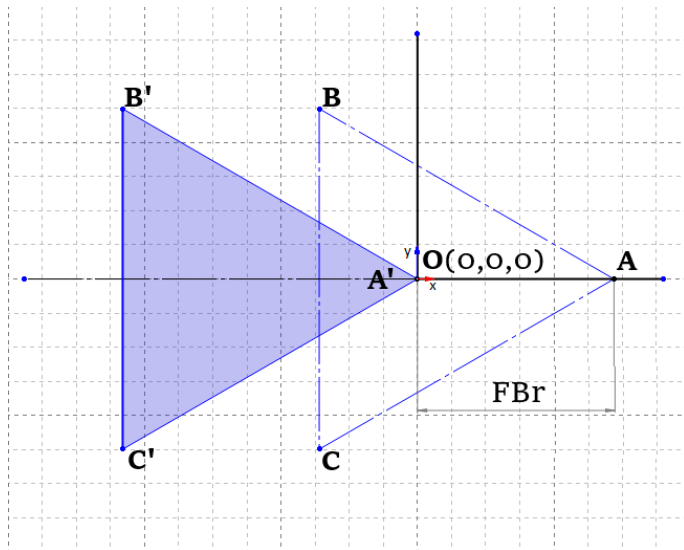


Figura 44 traslación de los puntos de la base (Fuente autor)

- Uno de los puntos debe estar sobre el eje de las **X** (diferente al punto ubicado en $[0\ 0\ 0]$). Para ello es necesario realizar una rotación a los puntos de la base 150° en sentido contrario a las manecillas del reloj véase la Figura 45.

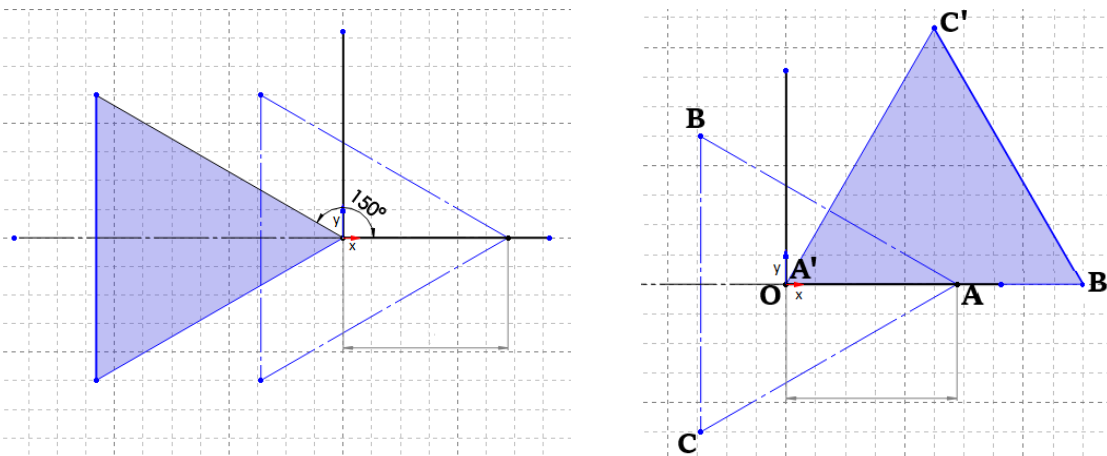


Figura 45 Base del robot antes y después de realizar la rotación de 150° (Fuente autor)

Realizando el procedimiento anterior la plataforma robótica ID-AM cumple con las restricciones para la aplicación del modelo de trilateración en 3D.

A continuación, se presenta el desarrollo matemático utilizado

- Como primer paso se realizó la generación de 3 ecuaciones partiendo de la ubicación tridimensional de los puntos de referencia de la siguiente manera

$$R1^2 = x^2 + y^2 + z^2 \quad (1)$$

$$R2^2 = (x - d)^2 + y^2 + z^2 \quad (2)$$

$$R3^2 = (x - i)^2 + (y - j)^2 + z^2 \quad (3)$$

De las tres ecuaciones descriptivas comparten en común la altura z , de esta manera se obtienen un sistema de ecuaciones con tres incógnitas y tres variables (3X3).

- Para el sistema de ecuaciones el primer paso es restar la ecuación de la esfera 1 menos la esfera 2 buscando despejar la incógnita " x ".

$$R1^2 - R2^2 = (x^2 + y^2 + z^2) - ((x - d)^2 + y^2 + z^2) \quad (4)$$

$$R1^2 - R2^2 = x^2 + y^2 + z^2 - (x - d)^2 - y^2 - z^2 \quad (5)$$

$$R1^2 - R2^2 = x^2 - (x - d)^2 \quad (6)$$

$$R1^2 - R2^2 = x^2 - x^2 - 2xd - d^2 \quad (7)$$

$$R1^2 - R2^2 = -2xd - d^2 \quad (8)$$

$$R1^2 - R2^2 + d^2 = -2xd \quad (9)$$

$$\left[x = \frac{R1^2 - R2^2 + d^2}{-2d} \right] \quad (10)$$

- Conociendo el valor de x procedemos despejar " z " en la ecuación 1 y 3, para posteriormente igualarlas y resolver despejando la incógnita " y ".

De la ecuación (1)

$$R1^2 = x^2 + y^2 + z^2 \quad \rightarrow \quad z^2 = R1^2 - x^2 - y^2 \quad (11)$$

De la ecuación (3)

$$R3^2 = (x - i)^2 + (y - j)^2 + z^2 \quad \rightarrow \quad z^2 = R3^2 - (x - i)^2 - (y - j)^2 \quad (12)$$

Igualado los (11) y (12)

$$R1^2 - x^2 - y^2 = R3^2 - (x - i)^2 - (y - j)^2 \quad (13)$$

$$-y^2 + (y - j)^2 = R3^2 - R1^2 + x^2 - (x - i)^2 \quad (14)$$

$$-y^2 + y^2 - 2yj + j^2 = R3^2 - R1^2 + x^2 - x^2 + 2xi - i^2 \quad (15)$$

$$\left[y = \frac{R3^2 - R1^2 + 2xi - i^2 - j^2}{-2j} \right] \quad (16)$$

- Conociendo el valor de “x” y “y” procedemos a despejar el valor de “z” de la ecuación número uno.

$$R1^2 = x^2 + y^2 + z^2 \quad (17)$$

$$\left[z = \sqrt{R1^2 - x^2 - y^2} \right] \quad (18)$$

Utilizando las ecuaciones (10), (16) y (18) con sus respectivos valores numéricos se obtiene la posición del efector final en **[x y z]** con coordenadas absolutas al modelo de trilateración. Para obtener las coordenadas absolutas en la posición normal del robot, es necesario realizar la rotación y traslación inversas a los resultados obtenidos de este modelo matemático.

3.5 ANÁLISIS DE SINGULARIDADES Y RESTRICCIONES

3.5.1 Singularidades

Una singularidad puede definirse como aquella configuración o posición de los actuadores en los cuales el robot pierde alguno de sus grados de libertad esto ocasiona que los actuadores requieran velocidades lineales o angulares infinitas para generar velocidades finitas del efector final o en algunos casos pérdida del control del movimiento del mismo [50] Para el robot **ID-AM** se realizó un análisis geométrico en el cual se detectaron las siguientes restricciones geométricas.

S1: Un posible caso de restricción sucede cuando los dos eslabones que componen una de las piernas del robot se encuentran paralelas generando un ángulo de 180° , pues si por alguna condición en particular el robot supera este valor angular, dicha pierna se volcara deformando la geometría (triángulo isósceles) propuesta para el robot véase la Figura 46.

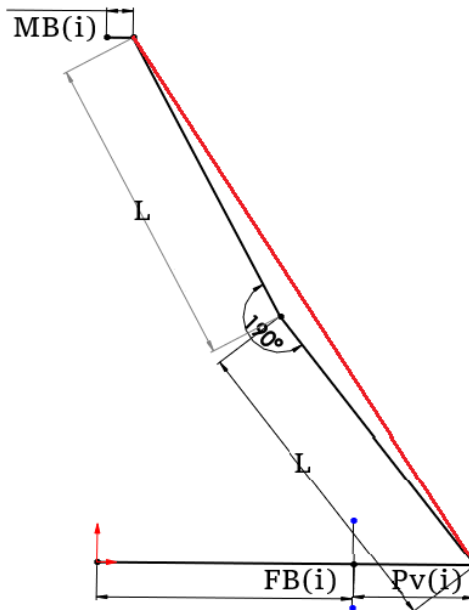


Figura 46 Simulación de movimiento articular $q(i)$ mayor de 180° (Fuente autor)

S2: El segundo punto singular se genera cuando el extremo de la pierna que conecta con la base móvil MB se encuentra exactamente en el punto de pivote (Punto de unión entre la base fija y el eslabón de pivote **Pv**), en esta posición en particular se alcanza una singularidad rotacional, permitiendo que todo el conjunto comprendido por el pivote y los dos eslabones que conforman la pierna roten sin control a pesar de que las posiciones angulares estén fijas.

3.5.2 Restricciones

En la implementación del **GA** de la plataforma **ID-AM** las restricciones están asociadas a los límites o cotas de los valores de las variables de interés. Dichos rangos están planteados según mediciones físicas de los componentes a utilizar que limitan los tamaños mínimos de las magnitudes, así también como valores articulares o relaciones de tamaños pensadas en excluir el espacio cilíndrico de trabajo de las singularidades anteriormente mencionadas, con el fin de mantener el control cinemático a lo largo de las trayectorias comprendidas dentro de dicho volumen. A continuación, se describirán las restricciones planteadas para la implementación el código del **GA** para la plataforma **ID-AM**.

- El área circular de cilindro de trabajo debe estar inscrito dentro del triángulo equilátero que compone la base.
- La longitud de los eslabones que componen las piernas del robot debe ser mayor a la suma del radio de la base fija del robot (**FBr**) más la longitud del pivote (**Pv**).
- La longitud máxima de los eslabones que componen las piernas del robot debe ser menor o igual a la suma del diámetro del círculo en el cual se circunscribe el triángulo de la base fija del robot más 2 veces la longitud del pivote (**Pv**).
- El diámetro mínimo de la base móvil del robot debe ser de 30 mm.
- El valor de la sombra (**Shadow**) generada por la base del triángulo isósceles para cada una de las piernas del robot debe ser mayor a 2 veces la longitud de uno de los eslabones (L) de manera que siempre se conserve la geometría de triángulo isósceles.
- La longitud del pivote (**Pv**) debe ser menor o igual a un cuarto de la longitud de uno de los eslabones que componen las piernas del robot (L) y no mayor a un tercio de dicha longitud.
- El ángulo máximo generado para cualquier desplazamiento debe ser menor o igual de 100° y mayor o igual a 20°
- El volumen del cono truncado debe contener el volumen del cilindro de trabajo

3.6 ALGORITMO GENÉTICO

Un algoritmo genético, podría describirse como un método adaptativo, usado en problemas de búsqueda y optimización de parámetros, el cual funciona con procesos matemáticos que buscan la imitación de la mecánica de selección, genética natural y evolución biológica, adaptados a un problema específico.

Su implementación empieza con la creación de una población inicial de individuos (**Padres**) generalmente creados de manera aleatoria, en donde cada uno de ellos representa una posible solución al problema, donde posteriormente se evaluará el porcentaje de acierto con el objetivo u objetivos planteados, en donde en caso de no cumplirse se procederá a crear una siguiente generación compuesta por los descendientes (**Hijos**) de los padres de la generación anterior, este proceso de reproducción de generaciones se llevara a cabo hasta llegar a la mejor solución del problema planteado o hasta cumplir con alguno de los criterios de parada fijados.

La generación de poblaciones es creada a partir uno de dos tipos de operadores de reproducción:

Operador Cruce: En la naturaleza el equivalente sería una reproducción de tipo sexual. En el cual su descendencia generalmente se genera a partir de dos individuos de la generación anterior [51].

Operador Copia: Su equivalente natural trata de una reproducción de tipo asexual, en donde un determinado número de individuos pasa sin cambios a la siguiente generación [51].

Una vez generados los nuevos individuos se procede con el cruce o mutación de los mismos para lo cual se define una probabilidad de mutación **PM** (Probabilidad de mutación) la cual generalmente se encuentra ente el 0.5% y 2 %.

Los criterios de parada más utilizados suelen ser:

- Los mejores individuos de la población cumplen satisfactoriamente el objetivo u objetivos planteados para la solución del problema
- La población converge. Caso tal que un gen converge cuando el 95% de la población tiene el mismo valor o se encuentran dentro de un rango determinado.
- Se alcanza un número máximo de generaciones establecido.

A continuación, se presenta un diagrama de flujo resumido sobre el proceso descriptivo del GA.

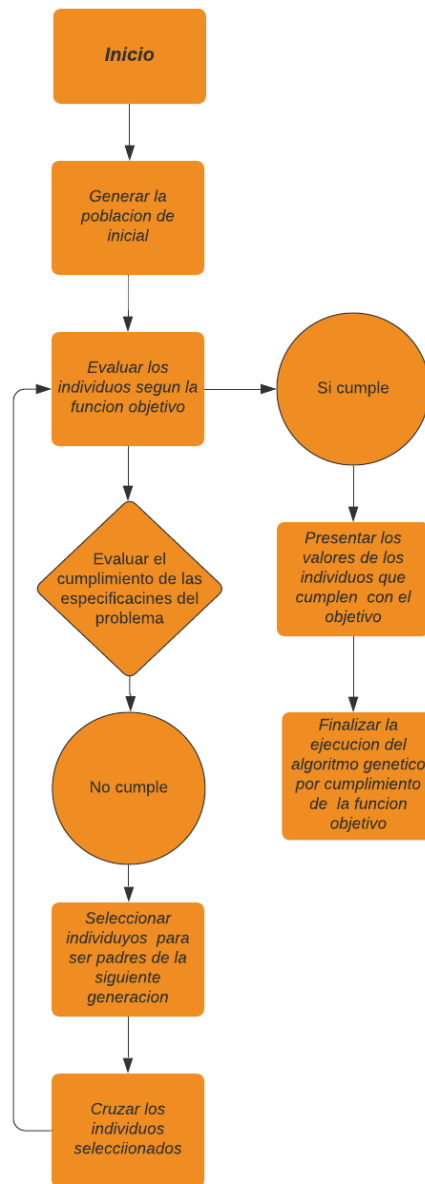


Figura 47 Proceso de optimización de algoritmo genético (Fuente autor)

Para resolver un problema con **GA** es necesario definir [52]

- La representación cromosomal de la población
- La forma de crear una población inicial
- Definir la función objetivo sobre la cual se interprete la acción de los individuos sobre el problema planteado

- Seleccionar el tipo de operador genético (cruce, mutación, etc.) que alteren la composición de los hijos
- Evaluar los parámetros de control (tamaño de la población, probabilidad de reproducción, etc.)
- Definir los criterios de parada del algoritmo

3.6.1 Optimización dimensional del robot ID-AM utilizando algoritmos genéticos

En busca del mejor aprovechamiento de los materiales dispuestos para este proyecto se propuso la utilización de un método de optimización dimensional utilizando algoritmos genéticos (**GA**) los cuales permitieran encontrar las dimensiones ideales para diferentes puntos estructurales del robot, de esta manera encontrar los valores óptimos que le permitieran a la plataforma robótica alcanzar todos los puntos dentro del área de trabajo pre establecida , para ello se incorporó los parámetros de diseño (**DP**) al algoritmo los cuales están relacionados directamente con las dimensiones físicas del robot.

Los parámetros que definen las dimensiones físicas del robot son los siguientes: Diámetro de la base fija (**FMr**), Diámetro de la base móvil (**MBr**), Longitud de los eslabones del robot (**L**) y la longitud del pivote del robot (**Pv**). Para lograr la optimización de los resultados se utilizará el siguiente procedimiento:

En cada interacción se procederá a generar una población de individuos con una solución característica, estos individuos estarán sujetos a dos tipos de verificación.

- La primera verificación corresponde al análisis del modelo geométrico del robot, es decir, si los valores obtenidos en la iteración del algoritmo genético, permiten construir el modelo alámbrico del robot, para ello se evaluarán las restricciones geométricas, así como algunas condiciones o relaciones entre las medidas.
- La segunda verificación corresponde a la corroboración del alcance en el espacio de trabajo del robot, es decir si los valores obtenidos en la iteración del algoritmo genético son capaces de realizar desplazamiento dentro del volumen de trabajo propuesto. Para ello se genera una nube de puntos correspondientes al volumen de trabajo propuesto, y dichos valores serán evaluados utilizando la cinemática inversa del robot, corroborando si la estructura es capaz de llegar a dichos puntos. Véase la Figura 48.

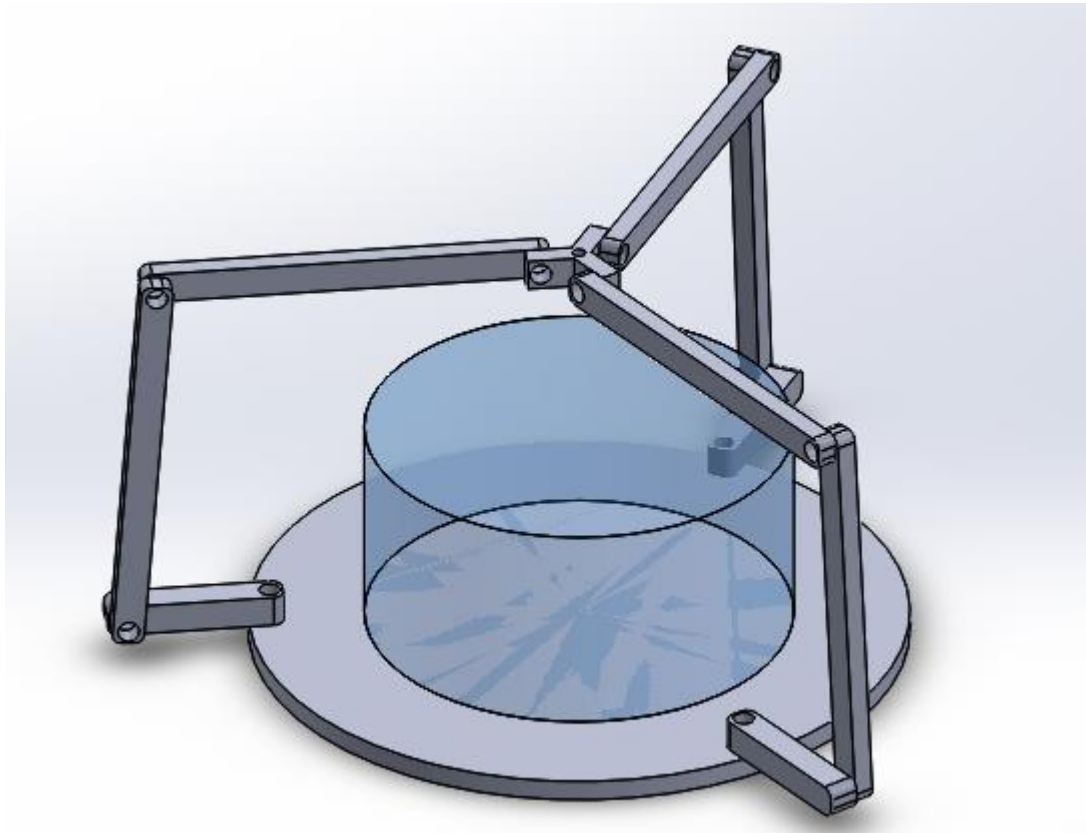


Figura 48 Emulación de la estructura simple del robot ID-AM con visualización del volumen cilíndrico de trabajo (Fuente autor)

Por último, se pasan los valores de los parámetros que más se acerquen o satisfagan las condiciones anteriormente mencionadas. Estos valores se utilizarán como individuos solución y serán utilizados por el algoritmo genético para generar la siguiente población hasta cumplir con la cantidad máxima propuesta de generaciones o converger en los valores mínimos de interés.

Para que el **GA** pueda converger minimizando los parámetros dimensionales es necesario que la evaluación del conjunto de parámetros generado en cada una de las interacciones del **GA** tenga un peso o un valor de verificación y comparación, para la solución del problema del robot **ID-AM**, se utilizó como base geométrica el cono truncado, puesto que al colocar todas las piernas del robot en un ángulo de 180° puede construirse dicha figura con los parámetros del robot **ID-AM** véase la Figura 49.

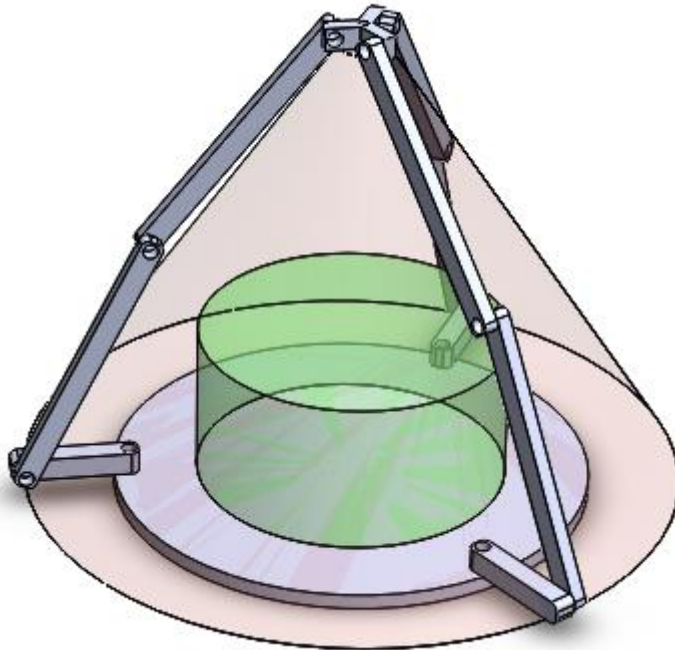


Figura 49 Emulación de la estructura simple del robot ID-AM con visualización del volumen del cono truncado generado a partir de los parámetros geométricos del robot (Fuente autor)

La implementación del **GA** modificara los valores geométricos del robot **ID-AM** en busca de minimizar el volumen del cono truncado visto en la Figura 49, cumpliendo dichos valores puedan satisfacer el desplazamiento del efector final en las posiciones comprendidas dentro del volumen cilíndrico del área de trabajo visto en color verde en la Figura 49.

La profundización matemática de evaluación de restricciones, corroboración del volumen alcanzado y demás códigos, se describirán en detalle en la siguiente sección.

3.6.2 Código de implementación

El desarrollo de la plataforma robótica **ID-AM** utiliza diversas funciones que en conjunto componen el programa principal. Dichas funciones se encargan de tareas específicas lo que simplifica su interpretación, para ello se utilizó el entorno de desarrollo matemático Matlab, sobre el cual se escribieron los distintos códigos para el funcionamiento, a continuación, se describirán a detalle el proceso de optimización.

Script principal

La ejecución del algoritmo genético sucede en el *script* principal con nombre **Run_optimize.m**, es aquí donde tienen declaración las variables globales que se usaran

en distintas funciones, en esta porción de código encontraremos las variables **H** (altura del cilindro de trabajo), **R** (radio del cilindro de trabajo), **Angle_joint_Max** (ángulo articular máximo permitido), **Resolution_cloud** (número de veces en que se dividirá la altura de trabajo), **Resolution_circle** (número de veces en que se dividirá el perímetro del círculo de trabajo), **Resolution_points** (número de veces en que se dividirá el radio de trabajo), **Penal** (Valor máximo de penalización), **FBr** (radio de la base fija), **L** (longitud de los eslabones de que componen las piernas del robot), **MBr** (radio de la base móvil) , **Pv** (valor de distancia del punto pivote) véase la *Figura 50*.

```

5      % declaration of global variables
6 -   global H;                % Work area height
7 -   global R;                % Base radius of work
8 -   global Angle_joint_Max;  % Maximun angular value
9 -   global Resolution_cloud;
10 -  global Resolution_circle;
11 -  global Resolution_points;|
12 -  global Penal;
13 -  global FBr;              % Fixed base radius
14 -  global L;                % Length of links
15 -  global MBr;              % Movile base radius
16 -  global Pv; %Pivot

```

Figura 50 Función Run_optimize.m porción variables globales (Fuente autor)

Las siguientes declaraciones corresponden a las variables utilizadas para la creación de la figura 1 en la cual se dibujará el robot, además de la variable para la creación del objeto video, el cual permitirá hacer una grabación de la evolución de las variables del robot **DI-AM** véase la *Figura 51*.

```

18     % statements to draw the model and record the evolutionary process
19 -   global writerObj;
20 -   global fl;
21 -   global record;
22 -   global R_Draw;|

```

Figura 51 Variables para la simulación del robot (Fuente autor)

A continuación, se procede con la asignación de los valores a las distintas variables globales empezando por las asignaciones para dibujar el robot y grabar el procedimiento evolutivo, véase la *Figura 52*.

```

24     % Initialization of the constans
25     |
26     %figure name
27 -   fl = figure('Name','Evolución de la plataforma ID-AM'), set(fl, 'Color', [1,1,1]), clf
28 -   writerObj = VideoWriter('Simulacion.avi');%video object creation
29 -   writerObj.FrameRate = 25;%sampling rate
30
31 -   record=0; %recording condition
32 -   R_Draw = 0;%drawing condition

```

Figura 52 Valores de las variables globales de simulación del robot ID-AM

La simulación del robot y grabación del proceso evolutivo, están condicionadas por las variables **R_Draw** y **Record** respectivamente se activan con parámetro binario alto (1) y

se desactivan con (0), activarlas aumenta el costo computacional lo que puede ocasionar una dilatación significativa en el tiempo de ejecución del programa.

La creación del volumen del trabajo se tuvo en cuenta la geometría del mismo, siendo una arquitectura tipo delta el volumen más adecuado corresponde a un cilindro, el cual esta descrito por la siguiente ecuación [35].

$$V = \pi * R^2 * H$$

Donde R corresponde al radio de la base circular y H la altura del cilindro de trabajo. Con el objetivo de mejorar los tiempos de ejecución del programa, las variables de **Resolution_cloud**, **Resolution_circle** y **Resolution_points** fueron declarados con enteros pequeños véase la Figura 53.

```
34 - load('Poblacion.mat');% Initial population
35 - R=100;
36 - H=110;
37 - Angle_joint_Max= 70;
38 - Resolution_cloud=2;
39 - Resolution_circle=2;
40 - Resolution_points=5;
41 - Penal=10000;
```

Figura 53 Asignación de los parámetros para la construcción del cilindro de trabajo y valores de división para el mismo (Fuente autor)

La aplicación de **ga** (*genetic algorithm*) en matlab contiene una amplia variedad de parámetros los cuales deben ser configurados de acuerdo al problema específico a desarrollar. Para la aplicación de la presente propuesta se realizaron las siguientes configuraciones véase la Figura 54 .

```

43 % Genetic algorit declaration
44 - FitnessFunction = @Objective_function2; % Function handle to the fitness function
45 - numberOfVariables = 4; % Number of decision variables
46 - PopulationSize_Data = 50;%number of individuals in each generation
47
48 - CrossoverFraction_Data= 0.25; %for CrossoverFraction
49 - ratio=1.2; % for crossoverintermediate
50
51 - Max_Generations=1000;%maximum number of generations
52 - Max_StallGenLimit=300;% maximum number of corroborations without change
53
54 % InitialPopulation_Data=Poblacion;
55 - lb = [R;H;30;30];
56 - ub = [2*R;2*H;60;80]; % Upper bound
57 - A = []; % No linear inequality constraints
58 - b = []; % No linear inequality constraints
59 - Aeq = []; % No linear equality constraints
60 - beq = []; % No linear equality constraints

```

Figura 54 Configuraciones iniciales para el ga (Fuente autor)

En la figura anterior se pueden apreciar las declaraciones fundamentales para la utilización del **ga** en la cual se asigna **@Objective_function2** como función *fitness* o función de optimización, se declaran el número de variables que intervienen en el algoritmo de optimización (4 para esta aplicación **FBr**, **L**, **MBr** y **Pv**), el número de individuos por cada generación (50 individuos), el número máximo de generaciones y la cantidad máxima de generaciones que se aceptaran sin presentar un cambio o mejora del valor fitness (300).

La lectura de la población Inicial **InitialPopulation_Data** (si no se utiliza la declaración de la población inicial, esta será generada por el algoritmo genético de forma automática utilizando los límites **lb** y **ub**).

Los valores de **lb** y **ub** delimitan el rango entre los cuales cada una de las variables podrá tomar un valor, siendo un vector de una fila y 4 columnas sus asignaciones correspondientes en orden son [**FBr**; **L**; **MBr**; **Pv**]. La delimitación de estos valores fue tomada con base en las dimensiones del cilindro de trabajo, lo cual permite que los límites puedan moverse en función de la asignación del mismo.

A continuación, se configuran las opciones del algoritmo genético véase la Figura 55.


```

62 %Start with the default options
63 - options = gaoptimset;
64 - options = gaoptimset(options, 'PopulationSize' ,PopulationSize_Data);
65 - options = gaoptimset(options, 'CrossoverFraction' ,CrossoverFraction_Data);
66 - options = gaoptimset(options, 'Generations',Max_Generations);
67 - options = gaoptimset(options, 'StallGenLimit',Max_StallGenLimit);
68 - options = gaoptimset(options, 'Display', 'iter');
69 - options = gaoptimset(options, 'MutationFcn', {@mutationadaptfeasible});
70 - options = gaoptimset(options, 'PlotFcns',{ @gaplotbestindiv @gaplotbestf ...
71 |@gaplotselection @gaplotstopping });
72 % options = gaoptimset(options, 'InitialPopulation' ,InitialPopulation_Data);
73 - rng(1, 'philox') % for reproducibility

```

Figura 55 Configuración de las opciones para la utilización del algoritmo genético

En la figura anterior apreciar la asignación de la variable **options= gaoptimset**, esto permitirá la configuración de las opciones de manera individual sin escribir los cambios a la función **ga** directamente. También podemos encontrar la asignación del tamaño de población (**PopulationSize**), máximo número de generaciones (**Generations**), máximo número de generaciones sin cambio (**StallGenLimit**), cuyos valores numéricos fueron asignados en el apartado anterior.

La asignación del tipo de cruce **CrossoverFraction** y el tipo de mutación **MutationFcn** están correlacionadas a la utilización de límites **lb** y **ub**.

La función **CrossoverFraction** especifica la fracción de cada población, a parte de los hijos elite que se cruzaran. Una fracción cruzada de 1 expresa que todos los hijos que no sean de elite serán cruzados, mientras que una fracción de 0 expresa que todos los hijos son niños con mutación, en resumen, controla la fracción de la población que será cruzada. El valor de fracción está asignado como **CrossoverFraction_Data** y puede ser modificado entre valores de 0 y 1 en la declaración de la Figura 54.

El tipo de mutación utilizado **mutationadaptfeasible** es el tipo de mutación predeterminada cuando hay restricciones, genera aleatoriamente direcciones que son adaptativas con respecto a la última generación exitosa o no exitosa. La mutación elige una dirección y longitud de paso que satisfaga los límites y sus restricciones lineales.

Debido a que el algoritmo genético es estocástico, es decir toma decisiones aleatorias, se obtiene un resultado diferente cada vez que se ejecuta el algoritmo, para ello utiliza un flujo numérico pseudoaleatorio el cual está asociado al criterio de reproductibilidad. La configuración de este número aleatorio está dada por la función **rng** la cual admite las siguientes configuraciones

Tabla 4 Posibles configuraciones de la función *rng*

Valor	Nombre del generador
'twister'	Mersenne Twister
'simdTwister'	SIMD-oriented Fast Mersenne Twister
'combRecursive'	Combined multiple recursive
'multFibonacci'	Multiplicative Lagged Fibonacci
'philox'	Philox 4x32 generator with 10 rounds
'v4'	Legacy MATLAB version 4.0 generator
'v5uniform'	Legacy MATLAB version 5.0 uniform generator
'v5normal'	Legacy MATLAB version 5.0 normal generator

La configuración de la función **PlotFcns** permite seleccionar que tipo de graficas serán mostradas en la ejecución del ga. Para esta aplicación se escogieron las gráficas más relevantes que permitan corroborar la correcta evolución de las variables es decir aquellas configuraciones que satisfagan las restricciones establecidas disminuyendo el volumen del robot para ello se generaron las siguientes gráficas, véase la Figura 56.

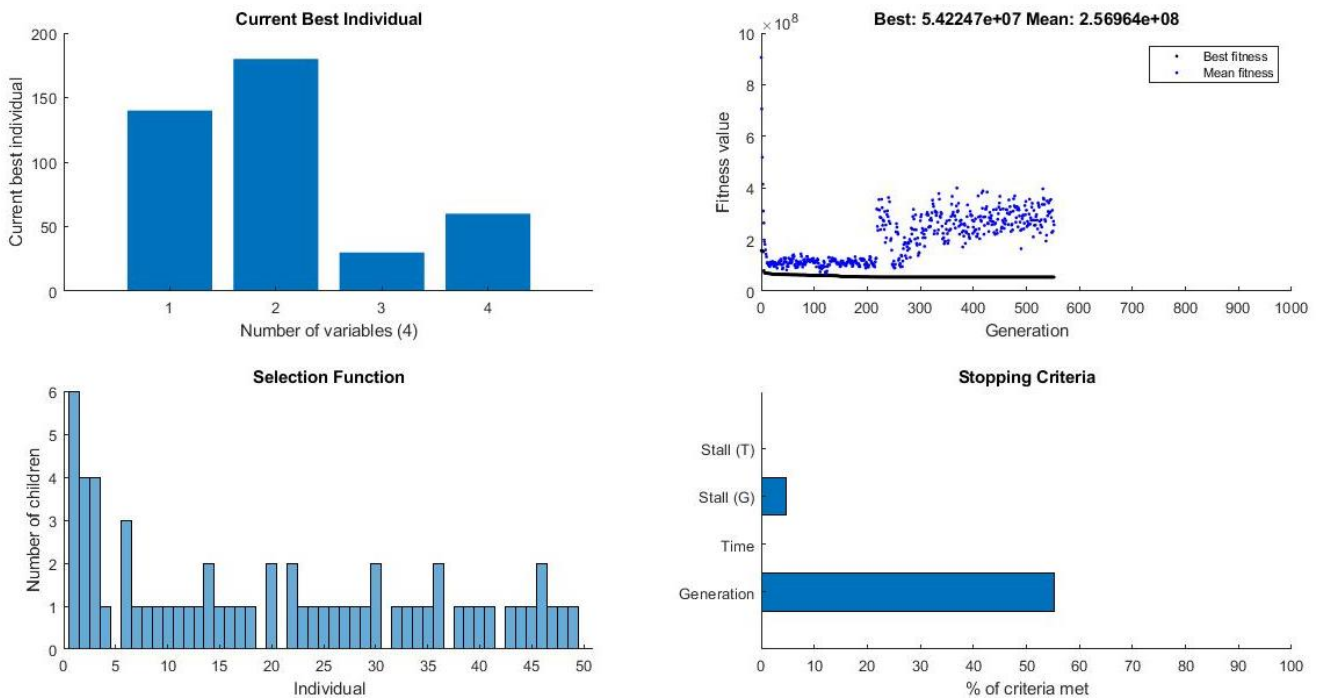


Figura 56 graficas generadas en la ejecución del ga (Fuente autor)

En la figura anterior se muestran las 4 graficas más relevantes, en la parte superior izquierda se encuentran los valores individuales del mejor sujeto hasta el momento, es decir la combinación de variables que obtiene el menor valor fitness, dicha grafica solo

cambia si cuando exista una mejor solución para el problema, de lo contrario permanece estática.

En la parte superior derecha se encuentra la evolución del valor fitness con respecto a cada una de las generaciones (puntos de color azul) y la comparación de menor valor que satisface las condiciones (puntos de color negro).

En la parte inferior izquierda se encuentra de manera gráfica la cantidad de niños o hijos que se generaron de cada sujeto en la generación actual, la variación de esta grafica está ligada al criterio de cruce y reproductividad.

Por último, en la parte inferior derecha se encuentra el criterio de parada, aquí se observa cuáles de las posibles condiciones de parada se está cumpliendo en la generación actual. En la Figura 56 se observa claramente los dos criterios de parada establecidos para la solución del presente desarrollo, los cuales son cantidad máxima de generaciones y cantidad mínima de comprobación de cambios.

Generador de población inicial

La asignación correspondiente a **InitialPopulation** e **InitialPopulation_Data** en la línea 54 de la Figura 54 corresponde a la declaración de una población inicial previamente especificada, para utilizar esta función es necesario correr el programa **Generador_poblacion_ID_AM** contenida dentro de la carpeta de optimización, de la siguiente manera:

Generador_polaion_ID_AM(x, lb, ub)

Este programa genera una matriz de x filas por 4 columnas, los cuales contienen valores aleatorios comprendidos dentro de los límites **lb** y **ub**, estos límites deben estar declarado como un vector horizontal de 4 filas. Como resultado se guarda la matriz solución como un archivo “**Poblacion.mat**” dentro de la carpeta de optimización, disponible para ser cargado posteriormente al *script* principal, véase el código en la Figura 57.

```
1 function Generador_poblacion_ID_AM(x, lb, ub)
2 % lb= [100,100,30,30];
3 % ub= [200,200,60,80];
4 for i=1:1:x;
5     for j=1:1:4
6         Poblacion(i,j)=randi([l(1,j) u(1,j)]);
7     end
8 end
9 save('Poblacion.mat','Poblacion');
```

Figura 57 Programa de generación para la población inicial de numero x de sujetos (Fuente autor)

Función Fitness

Es una función claramente definida la cual tiene como objetivo calcular un valor de aptitud de acuerdo a la solución del problema, para ello toma como entrada el vector solución producido por el algoritmo genético y calcula de manera cuantitativa que tan adecuada es la combinación de cromosomas para resolver el problema.

Aplicado al problema de optimización de la plataforma ID-AM, los mejores resultados deben tener un menor valor. Para ello se generó la función **Objective_fuction2.m** con la siguiente sintaxis

$$F = \text{Objective_fuction2}(\text{chr})$$

En donde la variable **chr** es el vector de 4 posiciones que contiene los valores numéricos que describen al sujeto a evaluar y el valor **F** devuelto por la función es el valor de aptitud que describe que tan apto es el sujeto.

El primer paso para calcular el valor de **F** es la extracción individual de los parámetros de interés para ello véase la Figura 58.

```
2 | % chromosomes
3 - | FBr=chr(1); % Fixed base radius
4 - | L =chr(2); % Length of links
5 - | MBr=chr(3); % Movable base radius
6 - | Pv=chr(4); %Pivot
```

Figura 58 Extracción de los parámetros de interés y asignación de las variables utilizadas dentro de esta función (Fuente autor)

En la figura anterior podemos ver nuevamente la declaración de las variables de interés, nombradas con **FBr**, **L**, **MBr** y **Pv** según lo descrito en el apartado “Descripción del robot delta invertido para manufactura aditiva FDM (ID-AM)”.

Teniendo en cuenta la dinámica del valor **F** es necesario dividir los sujetos en tres subcategorías

- Sujetos que no cumplen con las restricciones establecidas.
- Sujetos que cumplen con las restricciones establecidas solo en una porción parcial del volumen de trabajo.
- Sujetos que cumplen con las restricciones en la totalidad del volumen de trabajo.

Para ello se desarrolló una estructura condicional **for** anidada que dividiera el volumen de trabajo para comprobar el desempeño del sujeto en cada punto de la nube de la siguiente manera, véase la

```

1  for h=0:H/Resolution_cloud:H
2  for r=10:(R/Resolution_circle):R
3  for ang=0:(2*pi)/Resolution_points:2*pi
4      x(i)=r*cos(ang);
5      y(i)=r*sin(ang);
6      z(i)=h;
7  %      Cordinate=[x(i),y(i),z(i)]
8  %Check Restrictions
9  Penalty=Evaluate_restrictions(Robot,x(i),y(i),z(i));
10
11     if Penalty>1
12         %No cumple con las restricciones establecidas
13         F= No optimo;
14     else
15         %cumple con las restricciones establecidas en las
16         %coordenadas [x y z] descritas por el radio r y altura h
17         F= Parcialmente optimo;
18     end
19 end
20 end
21 end
22 %Cumple con las restricciones en la totalidad del volumen de trabajo
23 F= Totalmente optimo;

```

Figura 59 Simplificación descriptiva de la estructura condicional anidada, en donde se evidencia las 3 subcategorías de los sujetos F (Fuente autor)

Con esta división, es posible evaluar hasta qué punto la combinación de cromosomas generados por el algoritmo genético, cumple con las restricciones establecidas. Para evaluar el valor de aptitud F se utilizaron los siguientes volúmenes

- Volumen cilíndrico total del área de trabajo

$$\pi * (R^2 * H)$$

Donde **R** y **H** son las longitudes que describen el cilindro de trabajo

- Volumen cilíndrico faltante para completa el área de trabajo

$$\pi * (R - r)^2 * (H - h)$$

Donde **r** y **h** son los valores alcanzados dentro de la función *fitness* antes de incumplir alguna restricción

- Volumen compuesto por la multiplicación de todas las variables de interés

$$Robot.FBr * Robot.L * Robot.MBr * Robot.Pv$$

- Volumen del cono truncado

$$\frac{\pi * h}{3} * (FBr^2 + MBr^2 + (FBr * MBr))$$

Representado por un cono de punta cortada donde la base mayor está compuesta por la variable **FBr** y la base menor es **MBr**.

Usando los volúmenes anteriores se calculó **F** de la siguiente manera

$$F = (Volumen\ total - volumen\ alcanzado) + (volumen\ del\ cono\ truncado * Penalización) + volumen\ compuesto$$

Esta declaración permite cancelar la primera parte de la ecuación si se cumple la totalidad del volumen de trabajo, no obstante, aumenta el volumen del cono truncado si existe algún incumplimiento de restricciones y si se cumple el tercer caso en donde la combinación de cromosomas cumple con todas las características el volumen compuesto solo disminuirá si sus dimensiones son las mínimas.

Evaluación de restricciones

Esta función se encarga de evaluar los valores de las variables de interés proporcionados por el **ga** en las Singularidades y Restricciones, dando un valor de penalización el cual se encargará de aumentar el volumen del cono truncado desarrollado en la función *fitness*. Para ello se desarrolló una estructura case de 10 condiciones, asumiendo un

peso acumulativo dependiendo la cantidad de condiciones incumplidas. A continuación, se describen en detalle.

- La primera verificación evalúa la inclusión del círculo base del área de trabajo dentro del círculo del triángulo circunscrito que compone la base fija del robot **FBr**, véase la Figura 60.

```
36 - case 1 %La base circular del area de trabajo inscrito dentro del
37 -     %triangulo de la base
38 -
39 -     R_maximo=Robot.FBr*sind(30);%radio baricentro
40 -     R_actual=norm([x,y]);
41 -
42 -     if R_maximo > R_actual
43 -         Penalty=Penalty;%sin penalizacion
44 -     else
45 -         disp('NO cumple restricción 1')
46 -         Penalty=2+Penalty;
47 -     end% end if case 1
```

Figura 60 Caso 1 evaluación de restricciones (Fuente autor)

- La segunda verificación corresponde al radio mínimo de la base móvil del robot, véase la Figura 61. Figura 61 Caso 2 evaluación de restricciones (Fuente autor)

```
50 - case 2 %Valor minimo del radio de la base movil del robot
51 -
52 -     if Robot.MBr>=30
53 -         Penalty=Penalty;%sin penalizacion
54 -     else
55 -         disp('NO cumple restricción 2')
56 -         Penalty=3+Penalty;|
57 -     end
```

Figura 61 Caso 2 evaluación de restricciones (Fuente autor)

- La tercera verificación evita el volcamiento de la estructura triangular descrita para el mecanismo de las piernas del robot, véase la Figura 62.

```

60 - case 3 %evaluacion de la condicion de triangulo isoceles
61 -     for i=1:1:3
62 -         if Points.Shadow(i,1)<(2*Robot.L)
63 -             Penalty=Penalty;%sin penalizacion
64 -         else
65 -             disp('NO cumple restriccion 3')
66 -             Penalty=5+Penalty;
67 -         end
68 -     end

```

Figura 62 Caso 3 evaluación de restricciones (Fuente autor)

- La cuarta restricción limita la longitud mínima pivote, véase la Figura 63.

```

71 - case 4 %evaluacion de la longitud del pivote
72 -     if Robot.Pv>(2*Robot.MBr)
73 -         Penalty=Penalty;%sin penalizacion
74 -     else
75 -         disp('NO cumple restriccion 4')
76 -         Penalty=8+Penalty;
77 -     end

```

Figura 63 Caso 4 evaluación de restricciones (Fuente autor)

- La quinta condición evalúa que el ángulo empleado para alcanzar la posición deseada sea menor que el ángulo máximo permitido, véase la Figura 64.

```

79 - case 5 % Comprobacion del angulo maximo y minimo
80 -     for i=1:3
81 -         if Angulo_minimo<Points.q(i,2)<Angulo_maximo;
82 -             Penalty=Penalty;%sin penalizacion
83 -         else
84 -             disp('NO cumple restriccion 5')
85 -             Penalty=13+Penalty;
86 -         end
87 -     end

```

Figura 64 Caso 5 evaluación de restricciones (Fuente autor)

- La sexta evaluación corrobora que el radio de la base fija sea mayor que el radio de la base de trabajo, véase la Figura 65.


```

90 - case 6 % Radio de la base fija mayor que la base de trabajo
91 -
92 -     if Robot.FBr>(R+(0.4*R)) ;
93 -         Penalty=Penalty;%sin penalizacion
94 -     else
95 -         disp('NO cumple restricción 6')
96 -         Penalty=21+Penalty;
97 -     end

```

Figura 65 Caso 6 evaluación de restricciones (Fuente autor)

- La séptima evaluación corresponde a la longitud mínima de los eslabones que componen las piernas de plataforma, véase la Figura 66.

```

99 - case 7 %Longitud minima de los eslabones que componen las piernas del robot
100 -
101 -     if (2*Robot.L)>(Robot.FBr+Robot.Pv+(R*0.75))
102 -         Penalty=Penalty;%sin penalizacion
103 -     else
104 -         disp('NO cumple restricción 7')
105 -         Penalty=34+Penalty;
106 -     end % end if case2

```

Figura 66 Caso 7 evaluación de restricciones (Fuente autor)

- La octava evaluación corresponde a la longitud máxima de los eslabones que componen las piernas de plataforma, véase la Figura 67.

```

108 - case 8 %Longitud maxima de los eslabones que componen las piernas del robot
109 -
110 -     if Robot.L<= ((2*Robot.Pv)+(2*Robot.FBr)+65) ;
111 -         Penalty=Penalty;%sin penalizacion
112 -     else
113 -         disp('NO cumple restricción 8')
114 -         Penalty=55+Penalty;
115 -     end

```

Figura 67 Caso 8 evaluación de restricciones (Fuente autor)

- La novena restricción declara que la longitud de los eslabones que componen las piernas de la plataforma sea mayor que la longitud del pivote, véase la Figura 68.

```

117 -         case 9 %
118 -
119 -             if Robot.L > (Robot.FBr)
120 -                 Penalty=Penalty;%sin penalizacion
121 -             else
122 -                 disp('NO cumple restricción 9')
123 -                 Penalty=89+Penalty;
124 -             end

```

Figura 68 Caso 9 evaluación de restricciones (Fuente autor)

- La décima restricción evalúa que el volumen del cilindro de trabajo este contenido dentro del volumen del cono truncado generado con las dimensiones de las variables de interés, véase la Figura 69.

```

127 -         case 10 % Volumen del cilindro de la posicion acutual dentro del volumen
128 -             % del cono truncado
129 -
130 -             h_cone=sqrt((2*Robot.L)^2-(Robot.Pv+Robot.FBr-Robot.MBr)^2);
131 -
132 -             V1=pi*(R^2)*H;           % Cylinder volume
133 -
134 -             V2=((pi*h_cone)/3)*(Robot.FBr^2+Robot.MBr^2+(Robot.FBr*Robot.MBr)); % Truncated cone volume
135 -
136 -             if V1<V2;
137 -                 % disp('Cumple restricción 8')
138 -                 Penalty=Penalty;%sin penalizacion
139 -             else
140 -                 disp('NO cumple restricción 10')
141 -                 Penalty=144+Penalty;
142 -             end

```

Figura 69 Caso 10 evaluación de restricciones (Fuente autor)

El valor de penalización va incrementando según la secuencia de Fibonacci de tal manera que las restricciones sean evaluadas de menor a mayor impacto.

3.6.3 Apreciaciones y resultados

Debido a la naturaleza variable del algoritmo genético, sus resultados varían en cada ejecución, aunque sus valores no sean modificados. Para sintetizar una solución coherente se realizaron dos experimentos.

El primero consistió en modificar la cantidad de sujetos por iteración utilizando un rango de valor de cruce comprendido desde 0.1 hasta 0.95 en intervalos de 0.05, iniciando con una población inicial de 5 sujetos se buscó que la solución generada por el algoritmo genético presentara una tendencia decreciente al igual que los sujetos generados en las últimas iteraciones, cumplieran con la totalidad de las restricciones. Utilizando el siguiente procedimiento

Se utilizo las gráficas generadas por el algoritmo genético, fijándose especialmente el comportamiento del valor de aptitud **F**

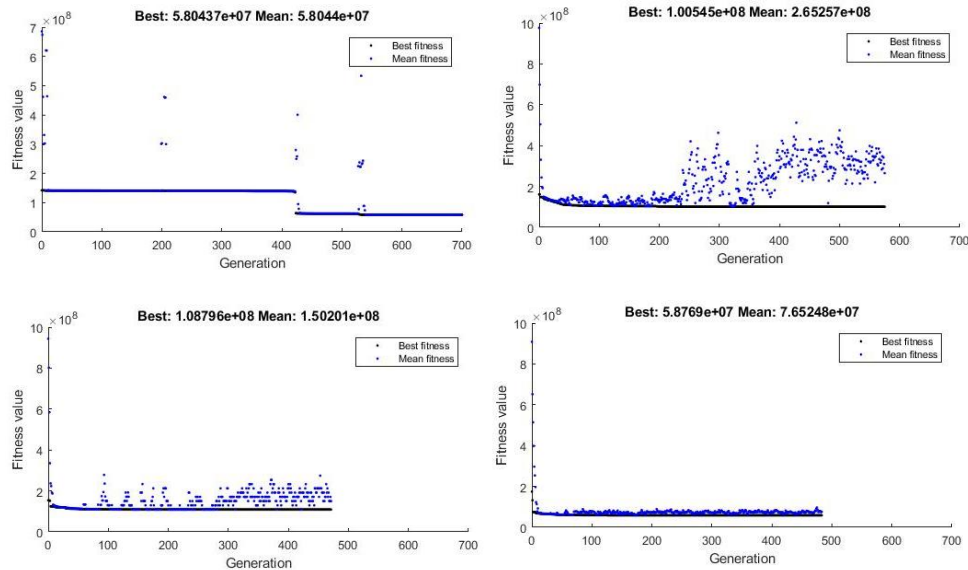


Figura 70 (a)Resultado 5 sujetos (b) Resultado 15 sujetos (c)Resultado 30 sujetos (d)Resultado 50 sujetos

De la Figura 70 (a)Resultado 5 sujetos (b) Resultado 15 sujetos (c)Resultado 30 sujetos (d)Resultado 50 sujetosFigura 70 se puede apreciar como al aumentar el número de sujetos en cada una de las generaciones, se va estabilizando el valor de aptitud, de esta manera se incrementó el número de sujetos hasta conseguir el resultado de la figura d con una cantidad de 50 sujetos obteniendo los resultados de la

Tabla 5 Resultados obtenidos de la implementación del **ga** con 50 individuos

Crossover Fraction	FBr	L	MBr	Pv
0.1	140	180	30	30
0.15	140	180	30	30
0.2	140	180	30	60
0.25	140	180.001	30	60
0.3	140	180	30	60
0.35	140	180	30	30
0.4	140	185.7946	30.0001	60.0004
0.45	140.0001	180.001	30	60.0001
0.5	140	180	30	60
0.55	140	180	30	60
0.6	140	180	30	60
0.65	140.001	193.051	30	60.001
0.7	140	196.0167	30	60.001
0.75	140	180.0059	30	30.001
0.8	140	180	30	60

0.85	140	194.656	30	60
0.9	140	180	30	60
0.95	140	185.3596	30	60

Como se observa en la tabla anterior, al llegar a este número de individuos, el algoritmo estabiliza muy claramente los valores solución.

Luego manteniendo como constante el número de individuos se varió la aleatoriedad de cruce de los padres del **ga**, para ello se modificó la función **rng** en cada valor permitido, mencionados anteriormente en Tabla 4 Posibles configuraciones de la función **rng** manteniendo como constante los valores de número de individuos, valor de cruce y máximo número de generaciones, obteniendo los siguientes resultados

Tabla 6 Resultados obtenidos de la implementación del ga con 50 sujetos y alteración de la aleatoriedad

Configuración rng	Valor de FBr	Valor de L	Valor de MBr	Valor de Pv
pilox	140	180	30	60
combRecursive	140.4766	181.0867	30.2173	60.649
multFibonacci	140.2225	180.3107	30.3125	60.984
threefry	140.2565	181.4077	30.0466	60.1046
simdTwister	140.3073	180.8618	30.0974	60.3295
twister	140.1666	180.4938	30.0268	60.1498
v4	140.101	180.333	30.0159	60.3227

Los resultados de la variación de aleatoriedad de cruce muestran que existe una desviación pequeña de los valores obtenidos. Por ello los valores de diseño que se tomaran para el desarrollo del prototipo son:

Tabla 7 Valores de diseño escogidos (conclusión del ga)

FBr	L	MBr	Pv
140	180	30	60

3.6.4 Comprobación de resultados

Utilizando la función Corroboration_function.m se prueban los resultados, utiliza la misma arquitectura que la función fitness, con la diferencia de la utilización de una nube de puntos más densa (véase la implementación en la Figura 71) , en esta función solo se evalúa si existe alguna condición que no se cumpla en cuyo caso se dará un valor de 1, de lo contrario se cumplirán todas las condiciones en la totalidad del volumen de trabajo y el resultado de la función será 0.

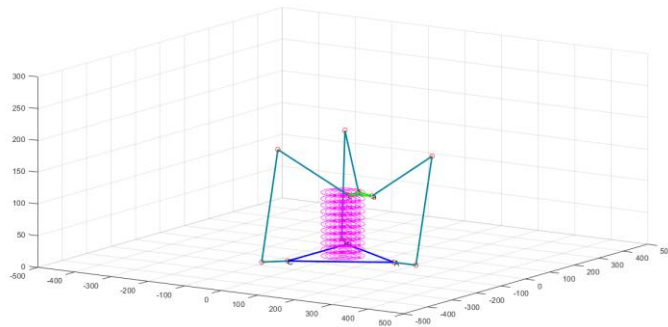


Figura 71 Modelo alámbrico del robot evaluado en cada uno de los puntos generados por la función *Corroboration_function* (Fuente Autor).

El resultado numérico de la corroboración obtuvo una puntuación de 0 corroborando el correcto desempeño del conjunto de valores.

4 CAPÍTULO 4

DESARROLLO DEL ROBOT

La incorporación de las ideas y conceptos en un diseño virtual, puede significar algunos retos si no se conoce la totalidad de las piezas que influyen en su funcionamiento y su interacción con los demás elementos del conjunto general. Para resolver este problema es necesario desglosar el diseño de manera conceptual en áreas definidas que permitan simplificar la interpretación e incorporación al prototipo.

Para el presente desarrollo se dividirá el problema de desarrollo en tres áreas fundamentales de diseño

- Diseño electrónico
- Firmware y software
- Diseño Mecánico

4.1 DISEÑO ELECTRONICO

4.1.1 Hardware

Compone la parte física del diseño electrónico. Está formado por los componentes electrónicos, electromecánicos y mecánicos que componen el prototipo. Para este desarrollo es necesario identificar y seleccionar dentro de las diversas gamas de

componentes los que mejor se acoplen a los objetivos planteados para el desarrollo descritos en Paso 6: Definición de los objetivos específicos.

El prototipo ID-AM utiliza los mismos principios móviles que una impresora DIY tradicional, variando su estructura y cinemática. Por ello los componentes electrónicos necesarios para su implementación son los mismos, los cuales pueden dividirse en:

- Tarjeta de control
- Motores de accionamiento
- Drivers o controladores de motor
- Finales de carrera
- Plataforma o cama de calefacción
- Cartucho de calefacción del *Hot end*
- Fuente de alimentación
- Ventilador de refrigeración de capa
- Sensores de temperatura
- Sensor de nivelación

La correcta interconexión de estos componentes permite le permite al prototipo efectuar las trayectorias, partiendo del punto de referencia o *home* y además de mantener el control sobre las variables de temperatura necesarias para la aplicación de FFF una simplificación de la conexión entre estos componentes puede verse en la Figura 72.

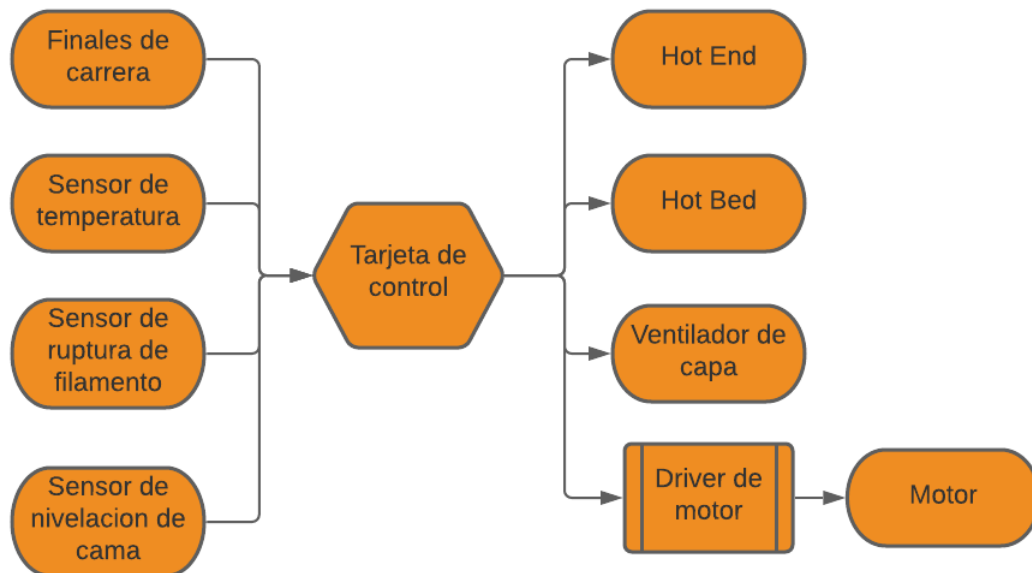


Figura 72 Esquema simplificado de control (Fuente autor)

En el presente desarrollo, se utilizó una disposición básica de los elementos de control, omitiendo sensores especiales como nivelación de cama y ruptura de filamento, a continuación, se describen a detalle, incluyendo la selección de componentes.

Tarjeta de control

Es el elemento principal del circuito electrónico, tiene como función el control de los motores y estabilidad del sistema de temperatura. Utiliza como estructura principal un sistema embebido en el cual se escribe el firmware, normalmente se utiliza conexión serial o wifi para establecer el hilo de comunicación por el cual se enviarán las trayectorias y demás comandos de control.

En el mercado es posible encontrar gran variedad de proveedores de tarjetas para impresoras 3D, tienen en común los aspectos más básicos como salidas para motores, entradas compatibles con diversos sensores de temperatura, salidas de calefacción para cama caliente y extrusor, entrada para finales de carrera, conexión USB. Algunas tarjetas cuentan con mayores prestaciones tales como conexión wifi, memoria integrada ante fallo de energía, drivers de motor integrados, a continuación, se mencionan las posibles opciones a utilizar y sus características.

SKR Mini V1.3: basada en el chip LPC1765 de 32bits soporta firmware modular y de código abierto *Marlin* y *smoothieware*, con alimentación de 12 a 24v DC y capacidad para

5 motores paso a paso con drivers fijos, soporta pantallas táctiles a color y pantallas LCD clásicas, soporte SD card incluido véase la Figura 73.

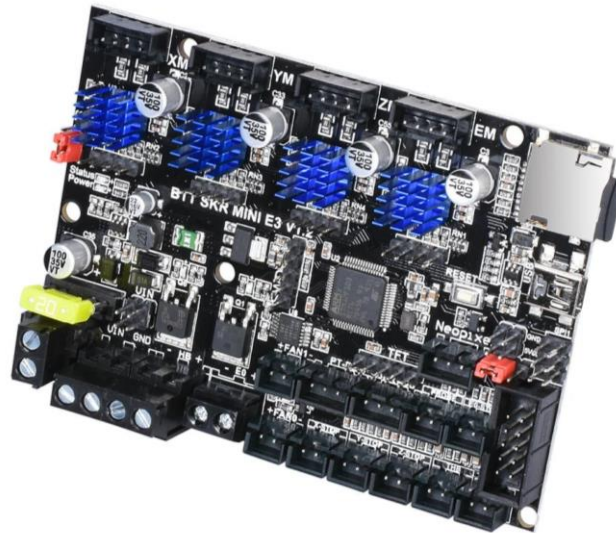


Figura 73 Placa SKR Mini V1.3 [53].

Ramps A5984: basada en el chip Atmega2560 16AU de 8bits, soporta firmware marlin, capacidad para 5 motores con drives integrados A5984 que permite el control de motores paso a paso bipolares con resolución hasta de 1/32, alimentación de 19v hasta 24 v DC.

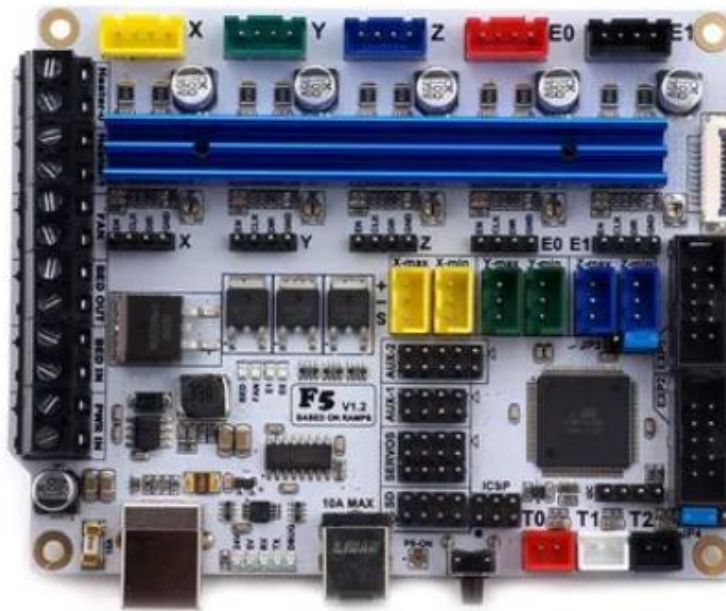


Figura 74 Placa A5984

Geeetech Gt2560: tarjeta de control compacta, basada en el chip ATmega2560 de 8bits soporta firmware de código abierto *Marlin*, con alimentación para 12 o 24v y capacidad para 5 motores con drivers intercambiables, soporta pantallas e interfaz SD card para impresión sin conexión véase la Figura 75.

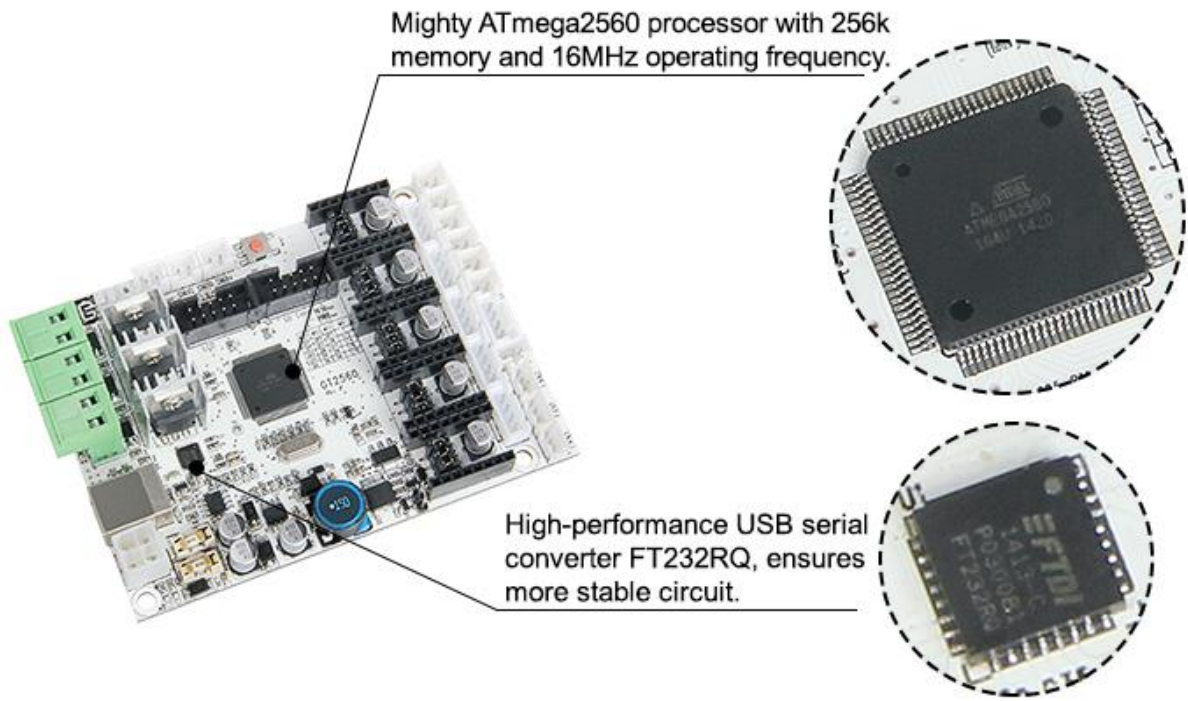


Figura 75 Vista superior de tarjeta Greeetech Gt2560 [54].

MKS Gen V1.4: Basada en el ATmega2560 cuenta con 5 salidas de motor paso a paso de drivers intercambiables, permite alimentaciones de 12 y 24 v DC, entrada para 3 sensores de temperatura, 4 salidas *mosfet* con control de pwm, conectores para 6 finales de carrera y compatibilidad con diversos tipos de pantallas como lo son *Easy display*, *RepRapDisconunt SmartController* véase la Figura 76.

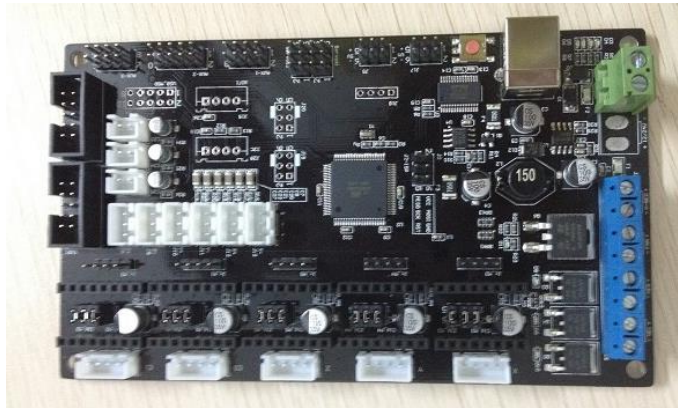


Figura 76 Tarjeta MKS Gen V1.4 [55]

Shield Ramps 1.4: Empleada en conjunto con un Arduino Mega, la ramps 1.4 ofrece capacidad para 5 motores paso a paso con drivers intercambiables, 6 finales de carrera conexión con pantallas *Easy display*, *RepRapDisconunt SmartController*, entrada para 3 sensores de temperatura y 3 mosfet con control pwm véase la Figura 77.

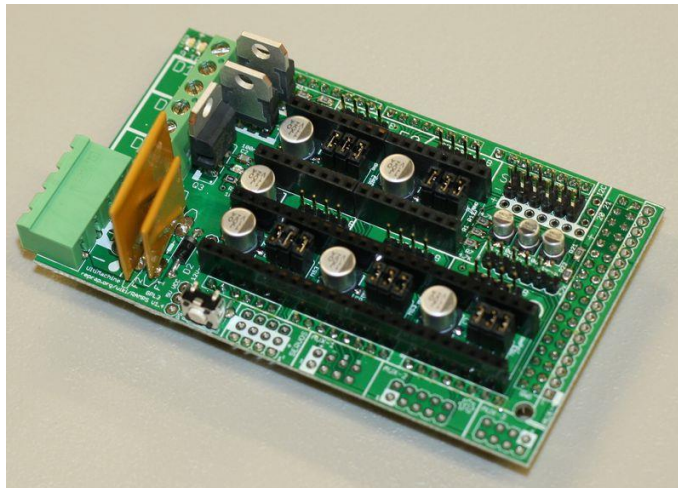


Figura 77 Shield Ramps 1.4 [56].

Tabla 8 Comparativa de precios tarjetas de control

Modelo de tarjeta	Precio en el mercado
SKR Mini V1.3	\$ 247.000
A5984	\$ 200.000
Geeetech Gt2560	\$ 140.000
MKS Gen V1.4	\$ 175.000

Shield Ramps 1.4 +Arduino Mega	\$ 88.000
-----------------------------------	-----------

La diversidad del mercado se debe a la gran variedad de fabricantes y precios, la mayoría basados en tecnología de código abierto permite la utilización de firmware libre. Sus principales diferencias técnicas se encuentran en la utilización de drivers intercambiables o soldados directamente a la placa además de la capacidad de procesamiento y posibilidad de utilización de LCD táctil, pero obtienen semejanza en los otros aspectos como la utilización de motores paso a paso, cantidad de los mismos, salidas de calefacción, cantidad de sensores, etc.

Debido a que las diferencia entre las tarjetas no tienen influencia en la implementación de una impresora 3D, la posibilidad de aumentar la resolución de control en los motores paso a paso mediante el cambio de los drivers de motor, la utilización de firmware de código abierto y su bajo costo, la opción escogida para la implementación del presente desarrollo fue la utilización de una ramps 1.4 con Arduino mega.

Sensor de temperatura

Los sensores de temperatura pueden describirse como elementos que transforman las variaciones de temperatura en señales eléctricas. Las señales eléctricas producidas por este tipo de transistores son leídas por el sistema embebido como entradas análogas que luego se procesan en una ecuación que describe el comportamiento del sensor utilizado para mostrar el resultado en unidades conocidas como grados centígrados o Fahrenheit.

Los sensores de temperatura más utilizados por los sistemas de impresión de código abierto son:

Termocupla: También conocido como termopar, es un transductor conformado por la unión de dos metales, produce una diferencia de potencial del orden de los milivoltios, lo que les permite una gran resolución en la medida. Para su implementación es necesario utilizar un módulo, su encapsulado generalmente es un tubo de acero inoxidable de diferentes longitudes, ideal para entornos industriales véase la Figura 78.

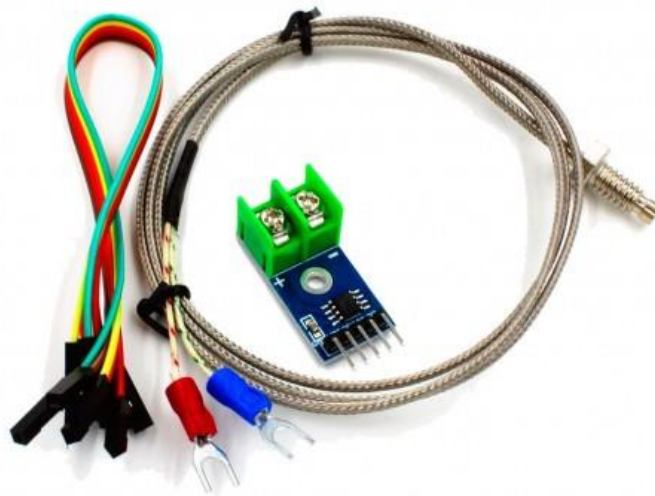


Figura 78 Termopar tipo K con modulo Max6675 [56]

PT100: Sensor de temperatura de comportamiento termo resistivo, está fabricado con un alambre de platino, con resistencia eléctrica directamente proporcional la cual aumenta con el aumento de temperatura. Su encapsulado es similar al del termopar y también requiere de un módulo para su implementación con sistemas de impresión 3D, véase la Figura 79.

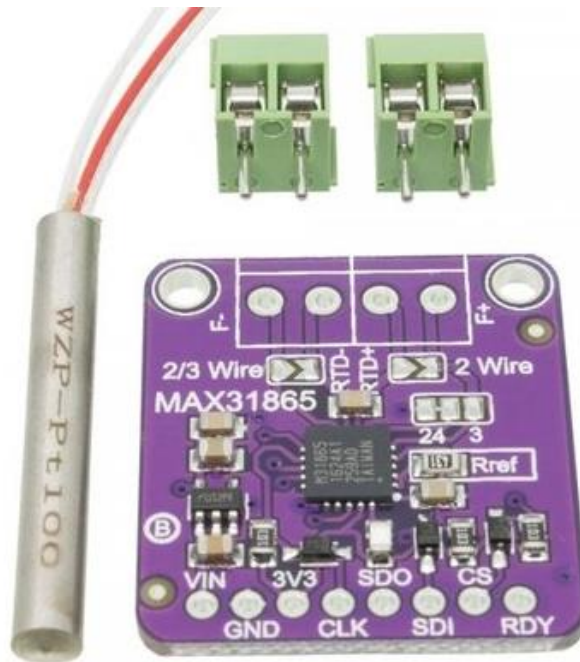


Figura 79 Pt100 con modulo max31865

Por el tamaño de su encapsulado, estos sensores son normalmente usados en aplicaciones donde se tiene una estructura encapsulada en una cabina, lo que permite la calefacción del entorno y control del mismo.

NTC B3950: Para mediciones de *hotbed* y *hotend* el sensor más utilizado es el NTC o termistor de coeficiente negativo (Negative thermal coefficient), con buena estabilidad y repetitividad, alta fiabilidad, amplia gama de resistencia: 0.1 ~ 1000K, tolerancia estrecha en los valores de resistencia y Beta. Se puede utilizar en ambientes de alta temperatura y alta humedad, pequeño, ligero.

Su presentación puede ser en encapsulado metálico de 3mm X 15 mm como mejora posee una carcasa metálica de cobre o aluminio que mejora la transferencia térmica o encapsulado libre, ideal para el bloque de calefacción por pequeño tamaño y rango de medición desde -30 °C hasta 300°C, cubriendo los rangos de temperatura de fundición de los termoplásticos más utilizados.



Figura 80 (a) NTC 3950 encapsulado metálico B (b)e NTC 3950 encapsulado libre

Tabla 9 Comparativa de precios sensores de temperatura

Modelo de sensor	Precio en el mercado
Termopar con modulo Max675	\$ 17.000
Pt100 con modulo Max31865	\$ 37.000
NTC B3950 encapsulado metálico	\$ 15.000
NTC 3950 encapsulado libre	\$ 3.600

Todos los sensores mencionados anteriormente cumplen con la inclusión del rango de temperatura de fundición de los diferentes materiales de impresión 3D (véase Principales materiales utilizados en la impresión 3D FDM), su configuración en el firmware es relativamente sencilla puesto que ya han sido previamente incluidos dentro de las librerías Marlin necesitando solo su la declarar que tipo de sensor a utilizar en los diferentes elementos de calefacción (teniendo en cuenta las entradas de los sensores generalmente son don para extrusores y una par cama caliente, en algunas versiones se utiliza una cuarta declaración para la cabina de calefacción).

Para el presente desarrollo, se eligió el sensor NTC 3950 de encapsulado libre, gracias a su pequeño tamaño ideal para el control tanto del *hotend* y *hotbed*, de fácil adquisición y bajo costo características ideales para el cumplimiento los objetivos planteados en la construcción del prototipo.

Finales de carrera

De naturaleza mecánica, óptica o inductiva, su funcionamiento consiste en la emisión de un pulso eléctrico cuando son accionados. Se utilizan en los extremos de las maquinas o articulaciones para demarcar el punto cero absoluto o home.

Sensor óptico: Basado en TCST2103, es un sensor de carácter fotoeléctrico bastante preciso, cuenta con tres terminales V, S y G (voltaje, señal y tierra) las cuales le permiten la conexión directa a cualquier tarjeta RepRap (véase la Figura 81). Posee dos estados NC (normalmente cerrado) y NO (normalmente abierto), y para su utilización es necesario diseñar una pieza de geometría plana y fina que ingrese dentro de la ranura del sensor. Su correcto funcionamiento puede ser interrumpido por la acumulación de polvo del ambiente.



Figura 81 Sensor óptico TCST2103 [56]

Sensor inductivo: equipado un interruptor transistor NPN normalmente abierto, es un sensor de corto alcance posee un alcance de 4mm con una alimentación de 6 a 36v DC, con unas dimensiones de 6.2 x 2 cm (véase la Figura 82) es ideal como sensor de nivelación cuando se usa una base metálica ferrosa.



Figura 82 Sensor inductivo LJ12A3-4-Z/BX [57].

Sensor mecánico: De referencia MK2b este final de carrera funciona como un pulsador de 2 estados NC y NO, gracias a su funcionamiento simple soporta largos ciclos de trabajo y es de fácil adquisición. Para su implementación solo es necesario fijarlo de tal manera que al mover el eje de interés una de las piezas del mecanismo toque directamente el sensor, también es recomendado la utilización de piezas a medida que sirvan de topes para su función.



Figura 83 Final de carrera mecánico MK2b [56]

Tabla 10 Comparativa de precios sensores final de carrera

Modelo Sensor	Precio en el mercado
TCST2103	\$ 6.300
LJ12A3-4-Z/BX	\$ 19.500
MK2b	\$ 5.000

Por su facilidad de adquisición y robustez ante trabajo constante el sensor de tipo mecánico ofrece mejores prestaciones para la implementación en este proyecto.

4.1.2 Firmware

Es la configuración o programación de más bajo nivel en un sistema electrónico, controla los circuitos electrónicos de un dispositivo, encargándose del control calculo y ejecución correcta de las instrucciones externas manteniendo para ello una relación directa con el hardware.

En el desarrollo actual, esta programación fue escrita directamente en el chip de la tarjeta de control, utilizando para ello el código Marlin, en el cual se realizaron las siguientes modificaciones de alto y bajo nivel

Las modificaciones de alto nivel se llevaron a cabo en el archivo *configuration.h* en el cual se llevaron a cabo las siguientes declaraciones

Velocidad de comunicación del puerto serial **#define BAUDRATE 11520**

Declaración del tipo de placa **#define MOTHERBOARD BOARD_RAMPS_13_EFB**

Configuración de los sensores a utilizar **#define TEMP_SENSOR_0 1**

Definición del robot delta **#define DELTA**

Declaración de las variables de interés que componen el robot delta **#define DELTA_RADIUS 140, #define DELTA_PRINTABLE_RADIUS 95, #define DELTA_RADIUS_MOVILE 30, #define DELTA_LINKS 180, #define DELTA_PIVOT 60**

Declaración de los pasos por rotación **#define XYZ_FULL_STEPS_PER_ROTATION 200**

Las configuraciones de bajo nivel se realizaron en el archivo **marlin_main.cpp** en el cual se modificó la cinemática inversa, declarada en la función **void calculate_delta(float cartesian[3])**, en la cual se convierte las coordenadas cartesianas en coordenadas lineales para cada uno de los actuadores de las piernas del robot. A continuación, se presenta las modificaciones realizadas.

```
float Alpha_1 = pi-atan2(cartesian[Y_AXIS]-delta_tower1_y, cartesian[X_AXIS]-delta_tower1_x); // Slope Y, Slope X
float Alpha_2 = pi+atan2(cartesian[Y_AXIS]-delta_tower2_y, cartesian[X_AXIS]-delta_tower2_x); // Slope Y, Slope X
```

```

float Alpha_3 = -pi+atan2(cartesian[Y_AXIS]-delta_tower3_y, cartesian[X_AXIS]-
delta_tower3_x); // Slope Y, Slope X

float delta_MB1_x= cartesian[X_AXIS]+(30*cos(Alpha_1));
float delta_MB1_y= cartesian[Y_AXIS]+(30*sin(Alpha_1));
float delta_MB2_x= cartesian[X_AXIS]+(30*cos(Alpha_2));
float delta_MB2_y= cartesian[Y_AXIS]+(30*sin(Alpha_2));
float delta_MB3_x= cartesian[X_AXIS]+(30*cos(Alpha_3));
float delta_MB3_y= cartesian[Y_AXIS]+(30*sin(Alpha_3));

float delta_Pv1_x= delta_tower1_x+(68.36*cos(Alpha_1));
float delta_Pv1_y= delta_tower1_y+(68.36*sin(Alpha_1));
float delta_Pv2_x= delta_tower2_x+(68.36*cos(Alpha_2));
float delta_Pv2_y= delta_tower2_y+(68.36*sin(Alpha_2));
float delta_Pv3_x= delta_tower3_x+(68.36*cos(Alpha_3));
float delta_Pv3_y= delta_tower3_y+(68.36*sin(Alpha_3));

float delta_Shadow1= sqrt(pow((delta_MB1_x-delta_Pv1_x),2)+pow((delta_MB1_y-
delta_Pv1_y),2)+pow(cartesian[Z_AXIS],2));
float delta_Shadow2= sqrt(pow((delta_MB2_x-delta_Pv2_x),2)+pow((delta_MB2_y-
delta_Pv2_y),2)+pow(cartesian[Z_AXIS],2));
float delta_Shadow3= sqrt(pow((delta_MB3_x-delta_Pv3_x),2)+pow((delta_MB3_y-
delta_Pv3_y),2)+pow(cartesian[Z_AXIS],2));

float q_complemento1= acos((delta_Shadow1/2)/Robot_L);
float q_complemento2= acos((delta_Shadow2/2)/Robot_L);
float q_complemento3= acos((delta_Shadow3/2)/Robot_L);

float q1= ((pi/2)-q_complemento1)*2;
float q2= ((pi/2)-q_complemento2)*2;
float q3= ((pi/2)-q_complemento3)*2;

// Conversión a distancia lineal y relación de poleas
float q_1= (100*q1)*8;
float q_2= (100*q2)*8;
float q_3= (100*q3)*8;

delta[X_AXIS] = q_1;
delta[Y_AXIS] = q_2;
delta[Z_AXIS] = q_3;

```

4.1.3 Software

El software escogido para este desarrollo, fue Repetier, por ser de uso libre interfaz intuitiva y fácil configuración. Su configuración se basó en la declaración de una

impresora de tipo paralela, para la cual se configuraron los valores de velocidad de comunicación (véase la Figura 84), velocidad de desplazamiento (Figura 85) y configuración del área cilíndrica de trabajo (Figura 86).

Puerto:	COM3	▼
Baudrate:	115200	▼

Figura 84 Configuración del puerto serial y velocidad de comunicación (fuente Autor)

Firmware Type:	Autodetect	▼
Velocidad Desplazamiento sin extruir:	1000	[mm/min]
Velocidad Avance del Eje Z:	500	[mm/min]
Manual Extrusion Speed:	2	20 [mm/s]
Manual Retraction Speed:	30	[mm/s]
Temperatura precalentamiento Extrusor:	200	°C
Temperatura precalentamiento Plataforma:	55	°C

Figura 85 Configuración de velocidades y temperaturas de precalentamiento

Reposo X:	0	Reposo Y:	0	Reposo Z:	100
Radio de impresión:	50	mm			
Altura de impresión:	100	mm			

Figura 86 Configuración de la posición de reposo y cilindro de trabajo (Fuente autor)

4.2 DISEÑO MECÁNICO

El diseño mecánico incorpora la selección de materiales, geometrías, dimensiones y tecnologías de fabricación que intervienen en la creación de un producto. Para el presente desarrollo se utilizó software CAD para la generación de las piezas estructurales las cuales en conjunto con el Firmware proporcionarían al robot de la sinergia necesaria para su funcionamiento, para ello se tuvieron en cuenta las dimensiones obtenidas en el proceso de optimización incorporándolas al diseño como valores base.

Desarrollando el diseño para una metodología de ensamble tipo modular se dividió el robot **DI-AM** en 3 partes principales

- Base fija
- Base Móvil
- Brazos

Cada una de estas estructuras se diseñó con la incorporación de los elementos mecánicos y electrónicos seleccionados, manteniendo una ideología de mantenimiento práctico que permita el cambio o revisión de algún componente de manera rápida y sencilla sin requerir el desmantelamiento completo o parcial del prototipo general.

A continuación, se presentan el diseño a detalle

4.2.1 Base fija

Es el componente estructural principal, el cual como su nombre lo indica es la parte inmóvil del robot, sobre la cual se soportan las demás estructuras, también puede denotarse como eslabón 0 o de anclaje del robot.

Consta de dos bases simétricas de estructura básica triangular con perforaciones para los ejes 3 ejes de rotación verticales del primer eslabón del brazo distribuidos uniformemente cada 120° el cual a su vez también sirve de elemento separador. En su estructura interior soporta la electrónica (9) y fuente de alimentación (10), en la parte superior soporta la base caliente (12) y en uno de sus costados utiliza la pieza de soporte (7) para la LCD (8).

Su centro coincide con el 0 absoluto del robot **DI-AM** Véase la **Fig.49** para la notación numérica de los párrafos anteriores utilice la **tabla 2** y **Fig.49** presentadas a continuación como referencia.

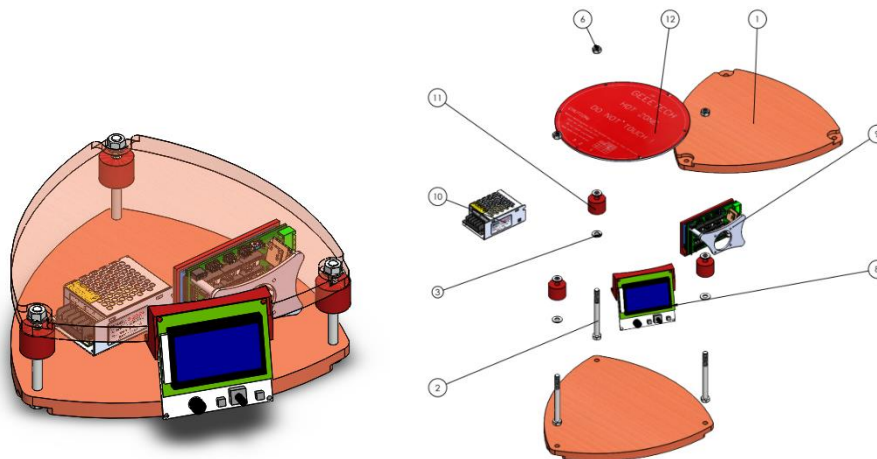


Fig. 1 Vista isométrica 'Base Fija' y modelo explosionado (Fuente autor)

Tabla 11 Lista de materiales ensamble 'base fija'

N.º DE ELEMENTO	N.º DE PIEZA	CANTIDAD
1	Base fija	2
2	M8X90	3
3	Arandela 8mm	6
4	M4*30	1
5	M4 Nut	1
6	M8 Nut	3
7	Soporte LCD	1
8	regrap LCD	1
9	Electrónica ensamblada	1
10	Fuente de alimentación 12v	1
11	Separador base	3
12	headbed	1

4.2.2 Brazos

Este ensamble está compuesto de tres estructuras simétricas distribuidas a 120° en la circunferencia del radio de la base, contiene elementos tanto electrónicos como mecánicos, los cuales en conjunto permiten la acción de control del Angulo de interés $q(i)$ desde $i=1:3$ para cada una de las articulaciones.

Este ensamble, este compuesto por 4 partes estructurales (**1,4,5,6**) mecanizadas en impresión 3D, con rodamientos incrustados 608z (**7**), unidas por tornillos M8 (**16**) con su respectiva tuerca (**11**) los cuales a su vez sirven de ejes de rotación para los rodamientos. La transmisión y control del movimiento inicia en el motor nema 17 (**10**) a su eje se le acopla la polea principal (**19**) sobre la cual se enrolla el hilo de transmisión hasta llegar a las poleas de acción (**2,3**), pasando por las poleas (**9,12**) que guían al hilo a lo largo de la estructura del robot.

Por último, esta estructura cuenta con un final de carrera tipo switch (**15**) que es accionado por la pieza de tope (**13**) incrustada a la polea B (**3**) por dos agujeros, existen dos pares de agujeros en cada una de las poleas A y B, esto permite configurar el tope a 90° y 180° el final de carrera.

Para la notación numérica de los párrafos anteriores utilice la **tabla 3** y **Fig.50** presentadas a continuación como referencia.

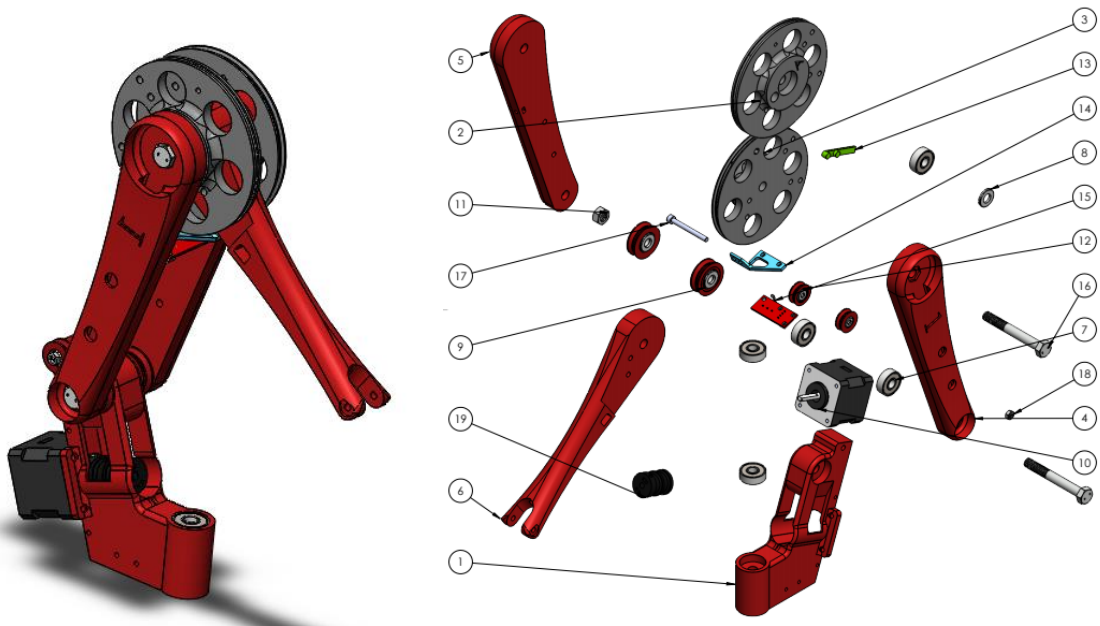


Fig. 2 Vista isométrica del ensamble 'brazo' y modelo explosionado (Fuente autor)

Tabla 12 Lista de materiales ensamble 'brazo'

N.º DE ELEMENTO	N.º DE PIEZA	DE	CANTIDAD
1	Pivote		1
2	polea 100mm_A		1
3	polea 100mm_B		1
4	Brazo_1A		1
5	Brazo_1B		1
6	Brazo_3		1
7	Rodamiento 608z		6
8	Arandela 8mm		2
9	polea guía 608z		2
10	nema 17 stepper motor		1
11	M8 Nut		2

12	polea guía 624z	2
13	tope	1
14	Soporte Endstop	1
15	Switch	1
16	M8X65	2
17	M4x40	1
18	M4 Nut	1
19	Polea principal	1

4.2.3 Base móvil

Conforma el elemento final de la cadena cinemática y soporta el denominado hotend en el cual se funde el material plástico para realizar la impresión en 3D.

Este ensamble está compuesto principalmente por 3 piezas impresas (**1,2,7**) con rodamientos incrustados 608z (**6**) unidas por el eje central (**9**), el eje tiene un final roscado en el cual se acopla el soporte de racor (**10**) el cual permite tanto el ajuste del ensamble descrito anteriormente como el soporte del racor (**11**) donde acopla con la manguera del extrusor indirecto y por la cual se alimenta el hotend.

Las piezas (**1,2,7**) poseen agujeros horizontales paralelos al plano XY del robot **DI-AM** de diámetro 4mm que permite incrustar rodamiento 624z (**5**) y utilizar como eje un tornillo **M4 X 40** con su respectiva tuerca.

Para la notación numérica de los párrafos anteriores utilice la **tabla 4** y **Fig.51** presentadas a continuación como referencia.

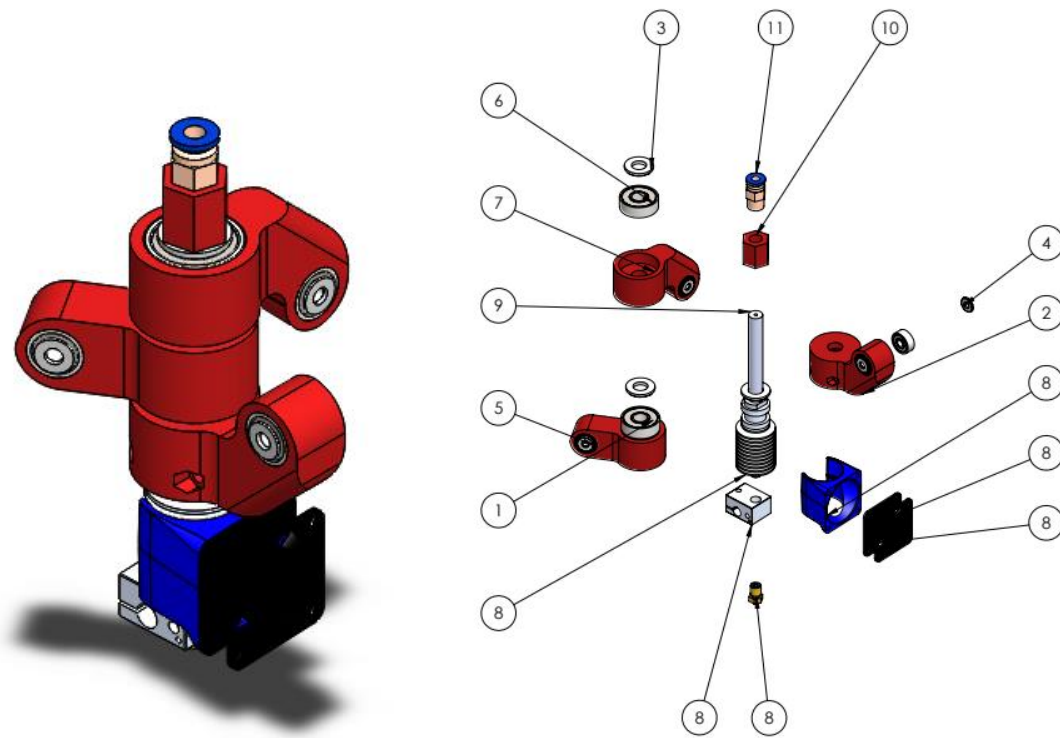


Fig. 3 Vista isométrica del ensamble 'plataforma móvil' y modelo explosionado (Fuente autor)

Tabla 13 Lista de materiales ensamble 'plataforma móvil'

N.º DE ELEMENTO	N.º DE PIEZA	CANTIDAD
1	Centro	1
2	Extremo Inferior	1
3	Arandela 8mm	3
4	Arandela 4mm	6
5	Rodamiento 624z	6
6	Rodamiento 608z	4
7	Extremo superior	1

8	Extrusor e3d v6	1
9	Eje central	1
10	Soporte racor	1
11	Racor	1

4.2.4 Ensamble Completo

Para realizar el ensamble completo del robot **DI-AM** se debe realizar el ensamble de la base móvil, siguiendo los brazos del robot utilizando una de las ranuras de internas de los mismos para acomodar los cables correspondientes al ventilador, calentador y sensor de temperatura. Por último, ensambla la base fija, teniendo en cuenta que la longitud de los cables de los motores y elementos del hotend no interfieran con la libre movilidad de los eslabones del robot.

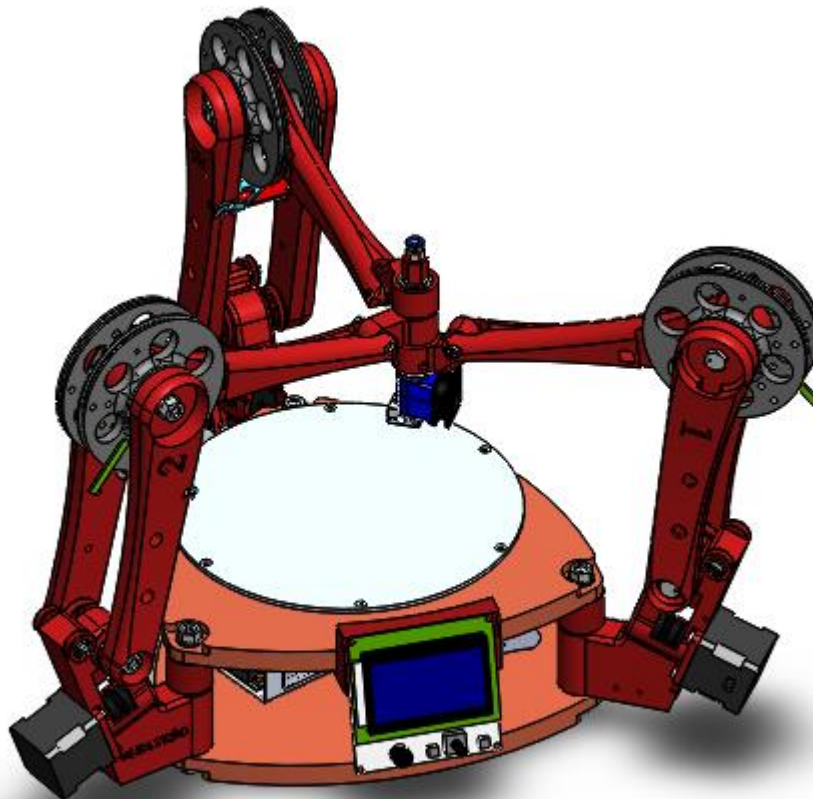


Fig. 4 Vista isométrica del ensamble total (Fuente autor)

Todos los archivos disponibles en <https://grabcad.com/library/inverted-delta-robot-1>

4.3 CONSTRUCCIÓN DEL PROTOTIPO

Para la construcción del prototipo se utilizó principalmente impresión 3D FDM utilizando las impresoras disponibles en el laboratorio Ladeprer y material PLA, con una configuración de altura de capa de 0.2 mm, se fabricaron todos los eslabones que componen la estructura del robot, dichos eslabones están diseñados para contener por encaje elementos como rodamientos y tornillos, intentando reducir al máximo las holguras consecuencia de los procesos de fabricación.

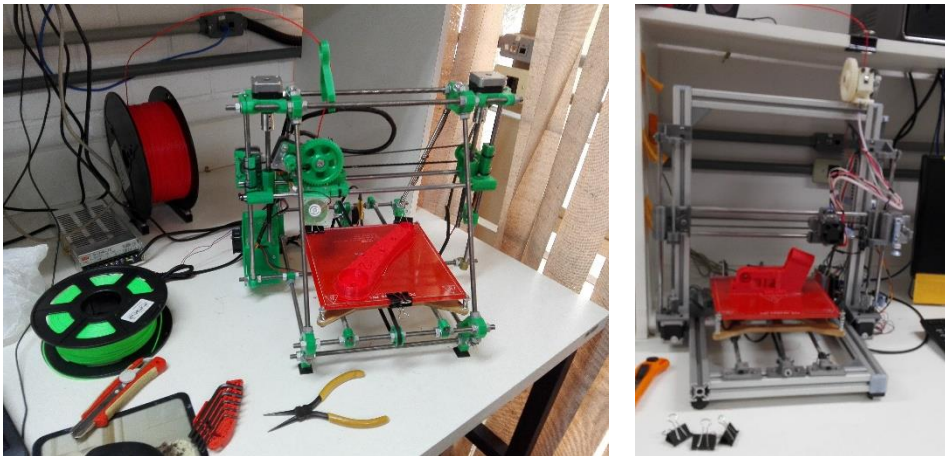


Figura 87 Impresoras 3D disponibles en el laboratorio Ladeprer utilizadas para la fabricación de los eslabones

Elementos como tornillos, tuercas, rodamientos, motores y elementos electrónicos, fueron proporcionados por el laboratorio Ladeprer para la implementación de este proyecto.

4.3.1 Primera versión

El prototipo de la plataforma robótica ID-AM está construido con piezas de fácil adquisición y electrónica de uso general para impresoras 3D, elementos que pueden ser adquiridos sin ningún inconveniente. Véase la lista final de componentes en Apéndice-Costos, a continuación se presentan todos los elementos necesarios para el ensamble del prototipo N°1 Figura 88.



Figura 88 Preparación de piezas para el ensamble del prototipo N°1 (Fuente autor)

La primera versión de la plataforma esta impresa en PLA de color amarillo, en la cual resalta la utilización de barras sencillas para las piernas del robot, el motor de actuación este fijado al segundo eslabón de la pierna, conectado a través de una correa sincrónica de tipo GT2 a la articulación de actuación, la plataforma esta mecanizada en un *router* CNC

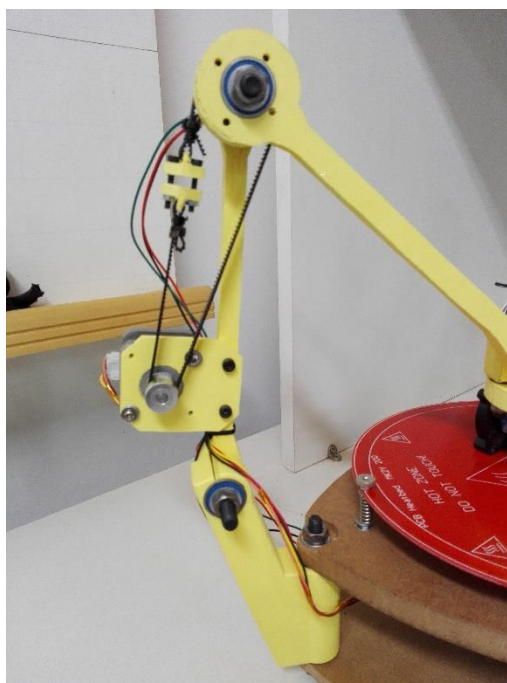


Figura 89 vista normal a una de las piernas que componen la pierna del robot (Fuente autor)

Debido al poco grosor de los eslabones, estos tendían a presentar deformaciones cuando se realizaba algún movimiento perdiendo pasos en la polea conductora, además por la ubicación de los motores de actuación, el mecanismo tendía a volcarse hacia el centro de la base fija si no se encontraban en estado de reposo (Energizados y bloqueados) véase la Figura 90.

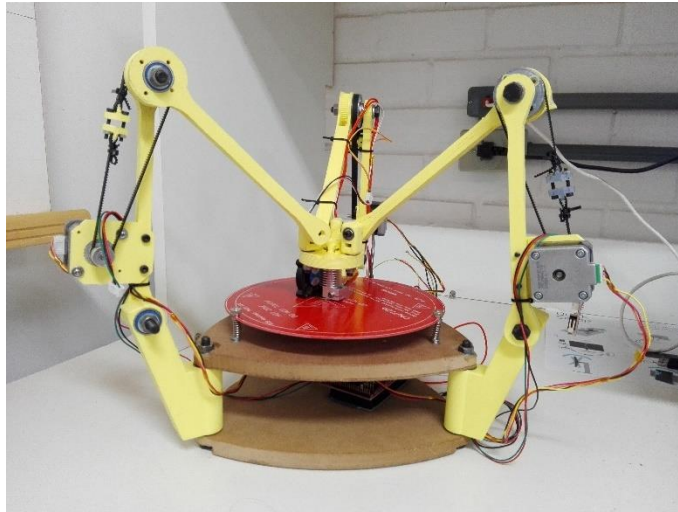


Figura 90 Imagen del prototipo N°1 ensamblado en su totalidad en posición de reposo

4.3.2 Segunda versión

Como corrección al diseño del prototipo N°1 se rediseñaron los eslabones aumentando su grosor y dividiendo uno de ellos en dos piezas simétricas que permitieran centrar el eslabón que une la base móvil del robot, de manera tal que no presentara deformaciones al momento de realizar un movimiento. Además se diseñaron las piezas de los eslabones de manera tal que los cables de conexión de los motores, finales de carrera, calentador y ventilador se organizaran en el interior de la estructura, evitando así posibles tramas que restringieran el libre movimiento de las articulaciones del robot.

Se cambió la posición de anclaje del motor de actuación al eslabón denominado Pivote, para lo cual se cambió el tipo de elemento de transmisión de movimiento, por hilos y poleas lisas, que conectaban la articulación de control, véase la Figura 91.

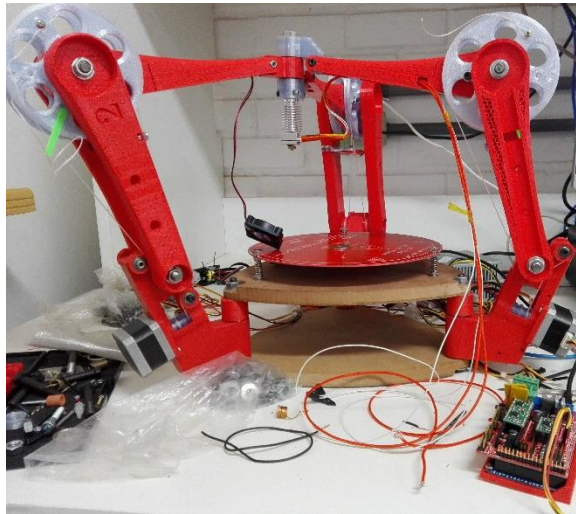


Figura 91 Proceso de ensamble, conexión de los componentes electrónicos y tarjeta de control

En este prototipo se encontró con la dificultad del estiramiento de los hilos de transmisión, presentando irregularidades en el control del mecanismo, además dicho sistema de transmisión presenta un desgaste acelerado a causa del rozamiento acelerando la necesidad de mantenimiento y recambio véase la Figura 92.



Figura 92 Prototipo terminado

4.3.3 Versión final

En la última versión se cambió el método de transmisión por correa sincrónica de tipo GT2 manteniendo la disposición del motor de actuación, como resultado se logró mejorar la fluidez de los desplazamientos y mantener el control de los mismos véase la Figura 93.



Figura 93 Mecanismo de transmisión con polea sincrónica GT2

La tercera versión logro reproducir un prototipo funcional, el cual permite realizar el control articular mediante código G, de esta manera la plataforma construida puede comunicarse con la interfaz en Matlab diseñada para corroboración de la cinemática, y en trabajos futuros podrá funcionar de manera independiente o con un programa de impresión 3D gratuito véase la Figura 94.

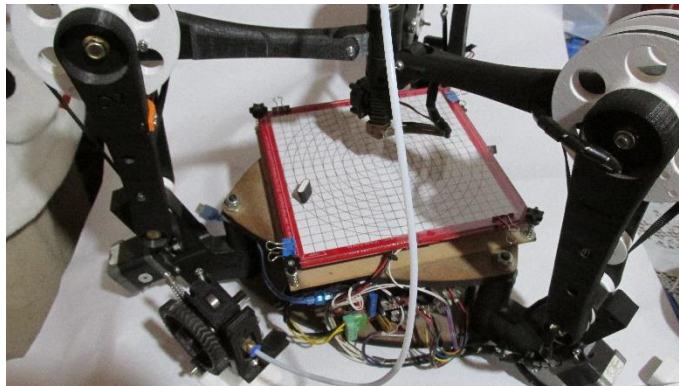


Figura 94 Prototipo final ensamblado con rejilla superficial para pruebas

4.4 COSTOS

Los costos presentados a continuación, corresponden a la inversión realizada en valor de pesos colombianos, se presenta el nombre de cada pieza en concreto, la cantidad

total utilizada para la construcción del prototipo, su precio unitario y el valor total, véase la tabla

Tabla 14 Tabla de costos de construcción del prototipo en pesos colombianos

Sub ensamble	Descripción	Cantidad	Valor unitario	Subtotal
-----------------	-------------	----------	-------------------	----------

Base fija	Base fija	2	\$ 6,500	\$ 13,000
Base fija	Tornillo M8X90	3	\$ 2,000	\$ 6,000
Base fija	Arandela 8mm	6	\$ 50	\$ 300
Base fija	Tuerca 8mm	3	\$ 400	\$ 1,200
Base fija	Soporte lcd	1	\$ 0	\$ 0
Base fija	Lcd RepRap	1	\$ 60,000	\$ 60,000
Base fija	Tornillo M4x30	2	\$ 150	\$ 300
Base fija	Tuerca M4	2	\$ 150	\$ 300
Base fija	Arduino mega	1	\$ 56,361	\$ 56,361
Base fija	Ramps 1.4	1	\$ 90,000	\$ 90,000
Base fija	Fuente de alimentación	1	\$ 50,000	\$ 50,000
Base fija	Separador Base	3	\$ 1,710	\$ 5,131
Base fija	Heatbed	1	\$ 52,021	\$ 52,021

Piernas	Pivote	3	\$ 16,174	\$ 48,523
Piernas	Poleas 100mm_A	3	\$ 7,300	\$ 21,899
Piernas	Poleas 100mm_B	3	\$ 7,213	\$ 21,638
Piernas	Brazo1_A	3	\$ 12,164	\$ 36,493
Piernas	Brazo1_3	3	\$ 11,786	\$ 35,359
Piernas	Brazo_3	3	\$ 10,759	\$ 32,277
Piernas	Rodamiento 608z	18	\$ 2,600	\$ 46,800
Piernas	Arandela 8mm	6	\$ 50	\$ 300
Piernas	Polea guía 608z	6	\$ 580	\$ 3,480
Piernas	Stepper motor Nema 17	3	\$ 58,800	\$ 176,400
Piernas	Tuerca M8 seguridad	6	\$ 400	\$ 2,400
Piernas	Correa dentada	3	\$ 5,667	\$ 17,001
Piernas	Polea dentada 20t	3	\$ 7,973	\$ 23,919

Piernas	Tope	6	\$ 166	\$ 995
Piernas	Soporte endstop	6	\$ 282	\$ 1,690
Piernas	Endstop	3	\$ 4,600	\$ 13,800
Piernas	Tornillo M8x65	6	\$ 2,000	\$ 12,000
Piernas	Tuerca M4	3	\$ 50	\$ 150
Piernas	Tornillo M4x40	3	\$ 150	\$ 450

Base móvil	Base móvil Centro	1	\$ 2,269	\$ 2,269
Base móvil	Base móvil superior	1	\$ 2,413	\$ 2,413
Base móvil	Base móvil inferior	1	\$ 2,915	\$ 2,915
Base móvil	Arandela 8mm	9	\$ 50	\$ 450
Base móvil	Rodamiento 624z	6	\$ 2,300	\$ 13,800
Base móvil	Rodamiento 608z	6	\$ 2,600	\$ 15,600
Base móvil	Extrusor E3d v6	1	\$ 47,000	\$ 47,000
Base móvil	Eje central	1	\$ 9,000	\$ 9,000
Base móvil	Soporte racor	1	\$ 553	\$ 553
Base móvil	Racor	1	\$ 7,140	\$ 7,140

Extrusor	Tornillo de tracción	1	\$ 5,700	\$ 5,700
Extrusor	arandelas 8mm		\$ 50	\$ 0
Extrusor	racor	1	\$ 7,140	\$ 7,140
Extrusor	piezas impresas	1		\$ 0
Extrusor	turecas 8mm		\$ 50	\$ 0
Extrusor	Rodamiento 608z		\$ 2,600	\$ 0
Extrusor	Stepper motor Nema 17	1	\$ 58,800	\$ 58,800
Extrusor	Tube Pet	1	\$ 6,000	\$ 6,000

TOTAL			\$	1,008,966
-------	--	--	----	-----------

5 CONCLUSIONES

El presente trabajo presenta el desarrollo de la plataforma ID-AM la cual se diferencia de otras estructuras por la utilización del concepto innovador de cinemática delta invertida.

- Utilizando la metodología de desarrollo QFD se logró sintetizar los objetivos principales a cumplir en el desarrollo de la plataforma ID-AM ayudando a la organización y desarrollo de actividades al igual que a la selección de componentes que cumplirían con los criterios obtenidos.
- Con la ayuda de la metodología de desarrollo QFD se integraron los requerimientos de los posibles usuarios recolectados mediante encuestas a las características de diseño
- Utilizando el entorno de desarrollo matemático de Matlab se consiguió la descripción matemática del robot, permitiendo la construcción de un modelo alámbrico que facilita la comprensión de la estructura.
- Gracias a la implementación del problema en el entorno matemático de Matlab, se logró la utilización de algoritmos genéticos que solucionaron de manera correcta el problema de optimización dimensional.
- Empleando diseño 3D se logró diseñar e integrar los diferentes componentes mecánicos y electrónicos utilizando sus dimensiones y puntos de anclaje como referencias para el diseño de las diferentes piezas estructurales.
- Utilizando el diseño 3D se generaron los archivos necesarios para la fabricación física de las piezas estructurales, utilizando tecnología de manufactura aditiva, de fácil adquisición para la elaboración de las mismas
- Se genero un prototipo funcional, el cual concuerda con la estructura diseñada y cuyas técnicas de fabricación y materiales open hardware permiten una fácil replicación para cualquier persona interesada en el presente desarrollo.
- Para finalizar el costo de fabricación y adquisición de las diferentes piezas mantiene un valor reducido permitiendo la colocación del presente desarrollo en la categoría de bajo costo.

6 LISTA DE REFERENCIAS

- [1] Barrientos Peñin Balaguer Arcin, *Fundamentos de Robotica. (Barrientos Peñin Balaguer Arcin)*. Madrid, 2007.
- [2] P. J. Sanz, "Introducción a La Robótica Inteligente," *Sanchez Pedro*, vol. 1, p. 80, 2006.
- [3] "Unidad 4 • Tipos de sistemas de produccion."
- [4] J. Doria García, F. Melo Cachay, and J. Chavez Chavez, "Implementación De Proceso De Pick And Place Mediante Grúa-Robot Aplicado A Celda De Manufactura Flexible," *Sci. Tech.*, vol. 18, no. 4, pp. 633–639, 2013, doi: 10.22517/23447214.8665.
- [5] C. Gosselin and L. T. Schreiber, "Kinematically Redundant Spatial Parallel Mechanisms for Singularity Avoidance and Large Orientational Workspace," *IEEE Trans. Robot.*, vol. 32, no. 2, pp. 286–300, 2016, doi: 10.1109/TRO.2016.2516025.
- [6] J. Merlet and C. Sophia-antipolis, "Springer handbook of robotics," *Choice Rev. Online*, vol. 46, no. 06, pp. 46-3272-46–3272, 2013, doi: 10.5860/choice.46-3272.
- [7] S. Bonev, Llian, Willard L.G, Pollard, Cappel, Klaus, Beeson, Mike, Evans, "The True Origins of Parallel Robots." <http://www.parallemic.org/Reviews/Review007p.html> (accessed May 17, 2019).
- [8] P. Grosch and F. Thomas, "Parallel Robots With Unconventional Joints," 2019. <http://link.springer.com/10.1007/978-3-030-11304-9> (accessed May 21, 2019).
- [9] H. A. Castillo Hernández, "Desarrollo de una plataforma stewart basada en microcontroladores, para simular el comportamiento de una aeronav en 3 grados de libertad," Instituto Politécnico Nacional, 2013.
- [10] M. Díaz Rodríguez *et al.*, "Aplicación de los Robots Paralelos." Accessed: May 22, 2019. [Online]. Available: <https://hal.archives-ouvertes.fr/hal-01907282>.
- [11] ASTM, "Standard Terminology for Additive Manufacturing," *Stand. Terminol. Addit. Manuf. – Gen. Princ. – Terminol.*, vol. ISO/ASTM 5, pp. 2–4, 2015, doi: 10.1520/F2792-12A.2.
- [12] R. Bogue, "3D printing: the dawn of a new era in manufacturing?," *Assem. Autom.*, vol. 33, no. 4, pp. 307–311, Sep. 2013, doi: 10.1108/AA-06-2013-055.
- [13] Plastics Technology México, "Las 7 familias de la manufactura aditiva," Mexico D.F. Accessed: May 23, 2019. [Online]. Available: https://www.pt-mexico.com/cdn/cms/Las_7_familias_de_la_manufactura_aditiva.pdf.
- [14] Sculpteo, "Stereolithography: 3D Printing by Laser solidifying Liquid-Resin," 2016. <https://www.sculpteo.com/en/glossary/stereolithography-definition/> (accessed May 23, 2019).
- [15] Sculpteo, "SLS (Selective Laser Sintering): 3D Printing Powder-based Process." <https://www.sculpteo.com/en/glossary/selective-laser-sintering-sls-definition/> (accessed May 23, 2019).
- [16] ANIWAA, "The 5 best professional desktop SLS 3D printers (March 2019)." <https://www.aniwaa.com/best-professional-desktop-sls-3d-printers/> (accessed May 23, 2019).
- [17] Sculpteo, "Binder Jetting Tecnología de Metal para Impresión 3D."

- <https://www.sculpteo.com/es/materiales/materiales-binder-jetting/> (accessed May 23, 2019).
- [18] Manufacturing guide, “Binder jetting | Find suppliers, processes & material.” <https://www.manufacturingguide.com/en/binder-jetting> (accessed May 23, 2019).
- [19] Make Parts Fast, “Binder Jetting: What Is It? - Make Parts Fast.” <https://www.makepartsfast.com/what-is-binder-jetting/> (accessed May 23, 2019).
- [20] 3D Printing and desing, “Photopolymer jetting 3D printers - PolyJet and MultiJet.” <https://www.additive.blog/knowledge-base/3d-printers/photopolymer-jetting-3d-printers-polyjet-multijet/> (accessed May 23, 2019).
- [21] GoPrint 3D, “What is Selective Deposition Lamination (SDL)?” <https://www.goprint3d.co.uk/blog/selective-deposition-lamination-sdl/> (accessed May 23, 2019).
- [22] Additive Manufacturing Research Group Loughborough University, “Directed Energy Deposition.” <https://www.lboro.ac.uk/research/amrg/about/the7categoriesofadditivemanufacturing/directedenergydeposition/> (accessed May 23, 2019).
- [23] S. Ezeiruaku, “Review of Additive Manufacturing and Characterization of Additive Manufacturing Machine,” *Researchgate*, no. JANUARY, 2016, doi: 10.13140/RG.2.1.3630.6009.
- [24] J. Campbell, “3D printing,” *Salem Press Encyclopedia of Science*. Salem Press, 2019, [Online]. Available: <https://bibliotecavirtual.uis.edu.co/login?url=http://search.ebscohost.com/login.aspx?direct=true&db=ers&AN=87324109&lang=es&site=eds-live>.
- [25] R. Bogue, “Assembly Automation 3D printing: the dawn of a new era in manufacturing?,” *Rapid Prototyp. J.*, vol. 33, no. 4, pp. 255–258, doi: 10.1108/AA-06-2013-055.
- [26] A. Bouyer, “RepRap - RepRap,” 2017. <https://reprap.org/wiki/RepRap> (accessed Dec. 04, 2019).
- [27] K. Miller, “Dynamics of the New UWA Robot,” Australia, 2001. Accessed: May 22, 2019. [Online]. Available: <https://pdfs.semanticscholar.org/c838/8e2436e807c1d9fe49d95a1398cac7d091fc.pdf>.
- [28] M. Ortega Breña, “Diseño mecánico de un robot paralelo delta de tres grados de libertad,” Universidad Nacional Autónoma de México, 2013.
- [29] J. D. Rueda Florez, “Metodología para el diseño de un robot paralelo industrial tipo delta,” Universidad Pontificia Bolivariana, 2013.
- [30] C. A. Peña Cortés, E. Martínez Oviedo, and P. F. Cárdenas Herrera, “OPTIMIZACIÓN DIMENSIONAL DE UN ROBOT PARALELO TIPO DELTA BASADO EN EL MENOR CONSUMO DE,” 2011. Accessed: May 17, 2019. [Online]. Available: <http://www.scielo.org.co/pdf/cein/v21n1/v21n1a05.pdf>.
- [31] C. Riaño, C. Peña, and A. Pardo, “DEVELOPMENT OF A PARALLEL ROBOT DELTA KEOPS WITH MODIFIED STRUCTURE,” *Rev. Colomb. Technol. Av.*, vol. 1, pp. 99–106, 2014, [Online]. Available: http://www.unipamplona.edu.co/unipamplona/portallG/home_40/recursos/04_v19_24/revista_23/27092014/15.pdf.

- [32] D. E. Flórez-Vergara, F. C. Castro-Riveros, and R. A. Castillo-Esteba, "Planeación y ejecución de trayectorias en el espacio de trabajo para un robot Delta TT - Trajectory planning and execution in the workspace for a Delta robot TT - Planejamento e execução de trajetórias no espaço de trabalho para um robô Delta," *Fac. Ing.*, vol. 25, no. 42, pp. 75–88, 2016, [Online]. Available: http://www.scielo.org.co/scielo.php?script=sci_arttext&pid=S0121-11292016000200007&lang=pt%0Ahttp://www.scielo.org.co/pdf/rfing/v25n42/v25n42a07.pdf.
- [33] B. M. Schmitt, C. F. Zirbes, C. Bonin, D. Lohmann, D. C. Lencina, and A. da C. S. Netto, "A Comparative Study of Cartesian and Delta 3D Printers on Producing PLA Parts," *Mater. Res.*, vol. 20, no. suppl 2, pp. 883–886, 2018, doi: 10.1590/1980-5373-mr-2016-1039.
- [34] F. Sovaila, N. Baroui, and C. Sovaila, "Delta 3d Printer," in *Jurnal of industrial desing an engineering graphics*, 2016, pp. 30–34.
- [35] L. Barrios *et al.*, "Volumen de los cuerpos geométricos.," *Matemáticas 2° E.S.O.*, pp. 181–200, 2009.
- [36] E. Rodriguez, A. J. Alvarez, and R. Bonnard, "Kinematics Analysis and Dimensional Synthesis of the Linear Delta Robot for Additive Manufacturing."
- [37] E. Rodriguez, A. J. Alvarez, and C. I. R. Jaimes, "Conceptual design and dimensional optimization of the linear delta robot with single legs for additive manufacturing," *Proc. Inst. Mech. Eng. Part I J. Syst. Control Eng.*, 2019, doi: 10.1177/0959651819836915.
- [38] N. Seward, "GUS Simpson," 203AD. http://ual.dyndns.org/biblioteca/admon_de_la_produccion/pdf/unidad_04.pdf (accessed Nov. 28, 2019).
- [39] M. Espinoza, "Aplicación del método QFD para el diseño de un envase de geometría especial en el proceso de termoformado," p. 166, 2015.
- [40] C. Sørensen, R. Jørgensen, J. Maagaard, K. Kjærhus Bertelsen, L. Dalgaard, and M. Nørremark, *Conceptual and user-centric design guidelines for a plant nursing robot*, vol. 105. 2010.
- [41] C. Riaño Jaimes, C. Peña Cortés, and H. Sánchez Acevedo, *Aplicación de técnicas de desenvolvimiento de producto para el desarrollo de un robot antropomórfico*, vol. 13. 2018.
- [42] Y. Ishino and T. Naruse, *Under water robot.*, vol. 4. 1986.
- [43] E. A. Rodríguez Gasca, E. D. J. Cortés Torres, and C. A. Peña Cortés, "Aplicación De La Metodología Qfd En El Desarrollo De Una Impresora 3D," *Rev. Colomb. Tecnol. Av.*, vol. 2, no. 28, 2017, doi: 10.24054/16927257.v28.n28.2016.2463.
- [44] A. T. Bahill and W. L. Chapman, "A tutorial on quality function deployment," *EMJ - Eng. Manag. J.*, vol. 5, no. 3, pp. 24–35, 1993, doi: 10.1080/10429247.1993.11414742.
- [45] Y. Akao, *QFD: Quality Function Deployment - Integrating Customer Requirements into Product Design*. 2004.
- [46] M. A. Donso and D. E. Soto, "UNIDAD I: GEOMETÍA EN R3 Y GEOMETRÍA VECTORIAL," in *ÁLGEBRA II CM 214*, Facultad de Cs.Básicas and D. de Matematicas, Eds. Universidad de Antofagasta, 2007.

- [47] D. Ruiz, "Localización y autocalibrado simultáneo con sistemas de posicionamiento local ultrasónicos." p. 251, 2011.
- [48] D. Gómez Bertoli, "Estudio, implementación y análisis de métodos de trilateración para la localización de usuarios desde sus terminales móviles," 2012, [Online]. Available: <http://e-archivo.uc3m.es/handle/10016/15909>.
- [49] M. Barot, "22. Trigonometría, parte II," in *Matemáticas II*, 2012.
- [50] I. Zaplana, J. A. Claret, and L. Basanez, "Análisis Cinemático de Robots Manipuladores Redundantes: Aplicación a los Robots Kuka LWR 4+ y ABB Yumi," *RIAI - Rev. Iberoam. Autom. e Inform. Ind.*, vol. 15, no. 2, pp. 192–202, 2018, doi: 10.4995/riai.2017.8822.
- [51] R. Contreras, "Introducción a los Algoritmos Genéticos," no. June, 2000, [Online]. Available: <http://scholar.google.com/scholar?hl=en&btnG=Search&q=intitle:Introducci+on+a+los+Algoritmos+Gen+eticos#0>.
- [52] M. Galante, "Un algoritmo genético simple para la optimización de estructuras planas articuladas," *Rev. Int. métodos numéricos para cálculo y diseño en Ing.*, vol. 9, no. 2, pp. 179–200, 1993.
- [53] "bigtreetech/BIGTREETECH-SKR-mini-E3: BIGTREETECH SKR-mini-E3 motherboard is a ultra-quiet, low-power, high-quality 3D printing machine control board. It is launched by the 3D printing team of Shenzhen BIGTREE technology co., LTD. This board is specially tailored for Ender 3 printer, perfectly replacing the original Ender3 printer motherboard." <https://github.com/bigtreetech/BIGTREETECH-SKR-mini-E3> (accessed Dec. 11, 2020).
- [54] "GT2560 - Geeetech Wiki." <http://www.geeetech.com/wiki/index.php/GT2560> (accessed Dec. 11, 2020).
- [55] "MKS GEN - RepRap." https://reprap.org/wiki/MKS_GEN (accessed Dec. 11, 2020).
- [56] "RAMPS 1.4/es - RepRap." https://reprap.org/wiki/RAMPS_1.4/es (accessed Dec. 11, 2020).
- [57] L.--z Bx, "Lj12a3-4-z/bx," p. 140.