

ROBOT PARALELO PARA ASISTENCIA EN CIRUGÍA LAPAROSCÓPICA



EMILIO JOSÉ HURTADO ARROYO
DIEGO ALEJANDRO PAZ BUSTAMANTE

Universidad del Cauca
Facultad de Ingeniería Electrónica y Telecomunicaciones
Departamento de Electrónica, Instrumentación y Control
Línea de I+D en Robótica y Control
Ingeniería en Automática Industrial
Popayán, Noviembre de 2012

EMILIO JOSÉ HURTADO ARROYO
DIEGO ALEJANDRO PAZ BUSTAMANTE

ROBOT PARALELO PARA ASISTENCIA EN CIRUGÍA
LAPAROSCÓPICA

Tesis presentada a la Facultad de Ingeniería
Electrónica y Telecomunicaciones de la
Universidad del Cauca para la obtención del
Título de

Ingeniero en Automática Industrial

Director:
Mag. Víctor Hugo Mosquera Leyton

Popayán, Noviembre de 2012

Hoja de Aprobación

Director

Mag. Víctor Hugo Mosquera

Jurado

Jurado

Fecha de sustentación: Popayán, de Diciembre de 2012

A Dios por permitirme alcanzar un logro más en mi vida, A ti Madre por ser la fuerza que me permite afrontar las adversidades, A Claudia y Camila por ser el motor de mi Corazón, a Manuel José por ser la luz de mi ser, A Pere por su apoyo incondicional, A Diego por ser mi compañero en la consecución de este triunfo y a todas las personas que de una u otra forma hicieron parte de este gran proceso. Mil y Mil Gracias...

Emilio H.

A Dios por guiarme, escucharme, Apoyarme y sobre todo comprenderme, A mi Padre por ser mi amigo incondicional y enseñarme el valor de la vida, el respeto, la honestidad y mucho más, A mi Hermano Andrés Felipe Por ser mi ejemplo, fortaleza, y Compañero de aventuras, A mi hermano Jorge Enrique y Sobrinas por depositar su confianza en mí y enseñarme que desde donde se vea la vida es hermosa, A mi madre por darme la vida, A mis abuelos por enseñarme que el trabajo es el mejor regalo que Dios nos ha dado, A Emilio por ser un excelente compañero, un gran amigo, que me permitió enriquecer mi conocimiento gracias a sus enseñanzas y grandes virtudes, A mis demás familiares y amigos, por ser testigos de la consecución de este logro.

Diego Paz

Agradecimientos

Los autores expresan sus agradecimientos a:

El Ingeniero Víctor Mosquera Leyton por su colaboración incondicional y por sus aportes intelectuales y personales como director de este proyecto.

Los miembros del Departamento de Electrónica, Instrumentación y Control por sus enseñanzas

A los compañeros de Universidad por toda su ayuda y alegría.

A los evaluadores de este proyecto que con su análisis apoyan el aporte científico del mismo.

A la Universidad del Cauca por su contribución administrativa y económica.

Resumen

El proyecto “Robot Paralelo para asistencia en cirugía laparoscópica” es desarrollado en la Universidad del Cauca (Colombia) por el grupo de Investigación en Automática Industrial, representado por los autores y el director.

Cuando se habla de robótica médica es muy común encontrar asistentes robóticos tipo serie, como: el *AESOP*, el *ENDOASSIST*, el *LAPBOT* o el *HIBOU*, estos dos últimos desarrollados en la Universidad del Cauca, dichos robots, por su gran tamaño, ocupan un espacio considerable en el quirófano, limitando los movimientos del cirujano y además son muy difíciles de transportar entre quirófanos, razones por las cuales se decidió estudiar el comportamiento de un robot paralelo que cumpla con las especificaciones del espacio de trabajo en una cirugía laparoscópica.

El robot diseñado en este proyecto, se basa en una estructura paralela. Se modela matemáticamente a nivel geométrico, cinemático y dinámico para comprobar que pueda desempeñarse como asistente porta endoscopio, pasando a través de un punto fijo en el espacio que representa la incisión realizada al paciente en el abdomen.

Se necesita demostrar que la estructura definida y su dinámica son las esperadas, es decir, que cumpla con las restricciones del área de trabajo en una cirugía abdominal mínimamente invasiva, es así como se analiza por medio de los modelos, su área de trabajo y las singularidades, además se realiza una prueba mediante simulación del seguimiento de trayectorias.

Las piezas que componen el robot son desarrolladas en el software CAD SolidEdge® para obtener los parámetros inerciales de cada una y así tener una aproximación muy cercana al robot real.

Se realizará una simulación en tres dimensiones que permitirá observar los movimientos del robot siguiendo unas trayectorias predefinidas y manteniendo un

punto fijo, que representa la incisión del paciente en el abdomen, sin importar la posición que adopte el endoscopio, dentro del área de trabajo.

Esta simulación tridimensional es de suma importancia ya que permite verificar el comportamiento del robot ante situaciones a las que tendrá que enfrentarse en la realidad, para observar posibles fallas y así implementar las respectivas mejoras en la construcción de las piezas del prototipo físico.

Palabras Clave: robots paralelos, modelado de robots paralelos, control de robots paralelos, robot asistencial.

Abstract

The "Parallel Robot for assistance in laparoscopic surgery" is developed at the University of Cauca (Colombia) by the Research group in Industrial Automation, represented by the authors and the director.

When it comes to medical robotics is very common to find both series robotic assistants, such as: the AESOP, the ENDOASSIST, the LAPBOT or HIBOU latter two developed at the University of Cauca, these robots, for its large size, occupy a considerable space in the operation room, limiting the surgeon's movements and these are very difficult to transport between operating rooms, reasons why we decided to study the behavior of a parallel robot that meets the specifications of the working space in laparoscopic surgery.

The robot designed in this project, is based on a parallel structure. This is modeled mathematically geometric, kinematic and dynamic level to see who can perform surgeries in the above, passing through a fixed point in space representing the patient's incision in the abdomen.

Is needed to show that the defined structure and dynamics are expected, i.e., that meets the restrictions of the work area in a minimally invasive abdominal surgery, so as analyzed by the models, its workspace and singularities, also is tested by simulation the trajectory tracking.

The component parts of the robot are developed in the CAD software Solid Edge ® before to obtain the inertial parameters of each one and so have a very close approximation to the real robot.

There will be a three-dimensional simulation that can observe the movements of the robot along a predefined path and maintaining a fixed point, which represents the patient's incision in the abdomen, regardless of the position taken by the camera, within the work area.

This three-dimensional simulation is very important because it allows verifying the robot's behavior to situations they will face in reality, to identify any faults and so implement the corresponding improvements in building the physical prototype parts.

Keywords: parallel robots, modeling of parallel robots, control of parallel robots, assistive robot.

Contenido

Lista de figuras.....	XIII
Lista de tablas	XVII
Lista de abreviaturas	XVIII
Lista de símbolos	XIX
INTRODUCCIÓN	1
1. Conceptos básicos sobre robótica y cirugía laparoscópica.	3
1.1.Laparoscopia	3
1.2.Robots porta endoscopio en cirugía laparoscópica.....	5
1.2.1. Avances Tecnológicos en Colombia y la Universidad del Cauca acerca de Cirugía Laparoscópica.....	7
1.3.Robótica paralela.....	9
1.3.1. Definición de robot paralelo.	9
1.3.2. Clasificación de los robots paralelos según su movilidad.	10
1.3.2.1. Robots paralelos planares	10
1.3.2.2. Robot paralelos espaciales	11
1.4.Manipuladores paralelos en cirugía.....	13
1.4.1. Robot miniatura para los procedimientos quirúrgicos en Neurocirugía.....	13
1.4.2. Sistema robótico para la intra-medularidad de huesos largos	13
1.4.3. CTBOT	14
1.4.4. SPINEASSIST.....	14
1.4.5. Robot paralelo exacto de 6-DOF con arquitectura HTRT	15
1.5.Robots empleados en cirugía.....	15
2. Modelado del Robot PREH.....	19
2.1.Requerimientos de diseño	19
2.2.Descripción del sistema mecánico	21
2.3.Modelo geométrico.....	23
2.3.1. Modelo geométrico Inverso	23

2.3.2.	Cálculo de los ángulos de rotación de cada brazo	27
2.3.3.	Modelo Geométrico Directo	42
2.4.	Modelo cinemático.	45
2.5.	Modelo dinámico	48
2.5.1.	Modelo Dinámico Inverso	48
2.5.1.1.	Modelo Geométrico de los 3 brazos que unen la plataforma a la base.	49
2.5.1.2.	Modelo cinemático de los 3 brazos que unen la plataforma a la base.	54
2.5.1.3.	Cálculo del modelo dinámico de cada brazo.....	59
2.5.1.4.	Modelo dinámico cartesiano de cada brazo <i>HXi</i>	64
2.5.1.5.	Análisis dinámico de la plataforma.	65
2.5.1.6.	Fuerzas deseadas y/o momentos deseados.....	65
2.5.2.	Modelo dinámico directo del robot	67
3.	Pruebas de los modelos	69
3.1.	Prueba de los modelos geométricos.	69
3.2.	Área de trabajo del robot	73
3.3.	Análisis de singularidades	75
3.4.	Diseño de las piezas del robot en Solid Edge	77
4.	CONTROL CTC DEL ROBOT	83
4.1.	Sintonización Control CTC.	85
4.2.	SEGUIMIENTO DE TRAYECTORIAS	86
4.3.	Conexión de una interfaz (Joystick) como generador de trayectorias.	88
4.3.1.	Recomendaciones de manejo de PREH con joystick.....	92
	Bibliografía	95
	ANEXOS 105	
	Anexo A. Programas implementados en Matlab.	105
A.1.	Función MGI.m	105
A.2.	Función MGD.m	105
A.3.	Función visualización.m	106
A.4.	Programa circular.m.	110
A.5.	Programa lineal.m.	111

A.6.	Programa paso_incisión.m.	112
A.7.	Programa espaciort.m.....	114
A.8.	Funciones para los modelos dinámicos inversos.	115
	A.8.1. Función mdi_f	115
	A.8.2. Función mdi_f2	116
	A.8.3. Función mdi_f3	117
A.9.	Funciones para los modelos dinámicos directos.	118
	A.9.1. Función mdd_f	118
	A.9.2. Función mdd_f2	119
	A.9.3. Función mdd_f3.	120
A.10.	Programa p_iner.	121
A.11.	Programa Inicio.m.....	122
Anexo B.	Simulación 3D.....	123
Anexo C.	Pazos para trabajar en Virtual Reality, con piezas hechas en Solid Edge. 125	
Anexo D.	Trabajo realizado en OGRE.	130
Anexo E.	Código en C++ empleado para realizar el armado y movimientos del robot. 136	
Anexo F.	Obtención de los modelos dinámicos del brazo 1 del robot.	147
	F.1.Modelo Dinámico Inverso del brazo 1	147
	F.2.Modelo Dinámico Directo del brazo 1	155
Anexo G.	Obtención de los parámetros inerciales en Solid Edge.	156

Lista de figuras

Figura 1-1: Aguja de Veress (izquierda), Trocar (derecha) [23].	4
Figura 1-2: Configuración de elementos básicos para laparoscopia [23].	4
Figura 1-3: Distribución del quirófano para una laparoscopia [24].	5
Figura 1-4: Robot ENDOASSIST [27].	6
Figura 1-5: Robot AESOP [28].	7
Figura 1-6: Ambiente 3D con dos LapBot y la interfaz de usuario [13].	8
Figura 1-7: Robot KIRUBOT [29].	8
Figura 1-8: Robot Hibou en Ogre3D [30].	9
Figura 1-9: Estructura básica de un robot paralelo [31].	9
Figura 1-10: Mecanismos de 2 y 3 GDL [32].	10
Figura 1-11: Robot paralelo planar 3RRR [33].	11
Figura 1-12: Plataforma Stewart [3].	11
Figura 1-13: Robot esférico 3 GDL [34].	12
Figura 1-14: Robot Delta [31].	12
Figura 1-15: Robot MARS colocado directamente en el cráneo [5].	13
Figura 1-16: Robot miniatura montado en el hueso [35].	14
Figura 1-17: Robot CTBOT [7].	14
Figura 1-18: Robot SPINE ASSIST [8].	15
Figura 2-1: Inclínación de la cámara.	21
Figura 2-2: Estructura inicial del robot.	22
Figura 2-3: Diagrama del robot.	22
Figura 2-4: Coordenadas de los puntos $A1, A2, A3, P1, P2, P3$.	24
Figura 2-5: Ejes de rotación de las articulaciones de cada brazo.	28
Figura 2-6: Proyecciones de P_y sobre los ejes q_{11} y q_{21} .	29
Figura 2-7: Proyecciones de P_y sobre los ejes q_{12} y q_{22} .	29
Figura 2-8: Proyecciones de P_x sobre los ejes q_{11} y q_{21} .	30
Figura 2-9: Proyecciones de P_x sobre los ejes q_{12} y q_{22} .	31
Figura 2-10: Desplazamiento de la plataforma sobre el eje y .	32
Figura 2-11: Corte transversal plano q_{13} - z .	32
Figura 2-12: Análisis del brazo 1 con movimiento en Y .	33
Figura 2-13: Corte trasversal sobre el plano eje q_{21} – z .	34
Figura 2-14: Corte trasversal sobre el plano eje q_{21} – z .	35
Figura 2-15: Análisis del brazo 2 con movimiento en y .	36

Figura 2-16: Corte transversal sobre el plano q22-z.	36
Figura 2-17: Desplazamiento de la plataforma sobre el eje x.....	37
Figura 2-18: Desplazamiento del p3 sobre el eje X.	38
Figura 2-19: Análisis q11 con movimiento en X.	39
Figura 2-20: Análisis ángulo q21.	39
Figura 2-21: Análisis de q12 con movimiento en X.	40
Figura 2-22: Análisis ángulo q21.	40
Figura 2-23: Análisis del segmento central.	43
Figura 2-24: Cosenos directores.....	44
Figura 2-25: Fuerzas externas que actúan sobre la plataforma.....	49
Figura 2-26: Estructura serie brazo i.	50
Figura 2-27: Pares articulares del brazo <i>i</i>	66
Figura 3-1: Diagrama en Simulink prueba modelos geométricos.	69
Figura 3-2: Visualización del robot.....	70
Figura 3-3: Sobreposición de trayectorias circulares.	71
Figura 3-4: Error entre las dos trayectorias circulares.	72
Figura 3-5: Paso por el trocar con trayectoria circular.....	73
Figura 3-6: Insuflación abdomen.....	74
Figura 3-7: Espacio de trabajo.....	75
Figura 3-8: Vista plano (x,z) del área de trabajo(izquierda), plano (x,y)(derecha).....	75
Figura 3-9: Diseño en Solid Edge de la base (izquierda) y la plataforma (derecha).....	78
Figura 3-10: Diseño articulación 1(izquierda), articulación 2 (centro), articulación 3 (derecha).	79
Figura 3-11: Soporte del endoscopio.	79
Figura 3-12: Endoscopio o cámara.....	80
Figura 3-13: Ensamble total del robot.....	80
Figura 3-14: Vista de planta conjunto camilla, paciente y robot.	81
Figura 3-15: Vista de frente conjunto camilla, paciente y robot.	81
Figura 4-1: Diagrama de bloques de Simulink para el control CTC.	83
Figura 4-2: Trayectoria circular deseada y obtenida.....	86
Figura 4-3: Error de seguimiento cartesiano en estado transitorio.....	87
Figura 4-4: Vista de planta del robot.....	88
Figura 4-5: Bloque Joystick Simulink.	89
Figura 4-6: Adquisición y caracterización de las señales del joystick.....	89
Figura 4-7: Diagrama de control con trayectorias generadas por el joystick.	90
Figura 4-8: Trayectoria realizada por PREH.....	90

Figura 4-9: Error de seguimiento con la trayectoria del joystick.	91
Figura 4-10: Trayectoria seguida por PREH en el plano X, Z.	91
Figura 4-11: Error de seguimiento cartesiano plano X, Z.	92
Figura B-1: Robot paralelo en V-RealmBuilder.	123
Figura B-2: Árbol de hijos (Izquierda) y padres en V-RealmBuilder (Derecha). (Piso, Camilla, base)	124
Figura B-3: Árbol de hijos y padres en V-RealmBuilder. (Brazos, pistones).....	124
Figura C-1: Grabar piezas en CAD a VRML.	125
Figura C-2:V-Realm Builder.	126
Figura C-3: Creando piso en V-Realm Builder.	126
Figura C-4: Piso e información del mismo en V-Realm Builder.....	127
Figura C-5: Abriendo piezas guardadas en VRLM.	127
Figura C-6: Copiando el grupo.....	128
Figura C-7: Trasformación creada	128
Figura C-8: Grupo (Pieza) copiado en el proyecto.....	129
Figura C-9: Camilla, soporte, robot en V-Realm Builder 2.0.	129
Figura C-10: Quirófano en V-Realm Builder 2.0.	130
Figura D-1: Como guardar piezas en STL.....	130
Figura D-2: Importación de piezas utilizando la herramienta Blender.	131
Figura D-3: Exportando piezas a OGRE.	131
Figura D-4: Herramienta para realizar el armado del Robot.....	132
Figura D-5: Pregunta y respuesta a emulador OGRE.	133
Figura D-6: Ubicación de la plataforma y ejes.	133
Figura D-7: Acoplamiento de la articulación 2.	134
Figura D-8: Ajuste de pistones a la articulación 2.	134
Figura D-9: Unión de ejes a pistones y plataforma a los brazos.	135
Figura D-10: Robot PREH armado, vista de las paredes y piso del quirófano.....	135
Figura D-11: Toma del Quirófano.....	136
Figura G-1: Pieza eje.par.	156
Figura G-2: Pieza eje.par con eje de coordenadas.	157
Figura G-3: eje.par Primer momento de inercia.	157
Figura G-4: Segundo momento de inercia.	158
Figura G-5: Pieza articulación 2.	159
Figura G-6: Pieza articulación 2 con sistema de coordenadas	160
Figura G-7: Primer momento de inercia.	160

Figura G-8: Segundo momento de inercia	161
Figura G-9: Pieza articulación_3.....	162
Figura G-10: Pieza articulación_3 con sistema de coordenadas.....	163
Figura G-11: Primer momento de inercia.	163
Figura G-12: Segundo momento de inercia.	164

Lista de tablas

Tabla 1-1: Robots más empleados en cirugía.....	15
Tabla 2-1: Resumen de las ecuaciones del MDI del robot PREH.....	41
Tabla 2-2: Parámetros geométricos del brazo 1.....	50
Tabla 2-3: Parámetros geométricos del brazo 2.....	50
Tabla 2-4: Parámetros geométricos del brazo 3.....	50
Tabla 2-5: Parámetros del segundo momento de inercia del brazo i	59
Tabla 2-6: Parámetros del primer momento de inercia del brazo i	60
Tabla 2-7: Parámetros inerciales de base del brazo i	60
Tabla 4-1: Ganancias para el controlador CTC.....	86
Tabla F-1: Parámetros geométricos del brazo 1.....	147
Tabla F-2: Parámetros inerciales de base del brazo 1.....	149
Tabla F-3: Parámetros inerciales esenciales del brazo 1.....	149
Tabla F-4: Parámetros geométricos del brazo 2.....	155
Tabla F-5: Parámetros geométricos del brazo 3.....	155

Lista de abreviaturas

AESOP:	<i>Automated Endoscopic System for Optimal Positioning.</i>
LAPBOT:	Robot para laparoscopia.
CTC:	Control por par calculado.
HIBOU:	Robot Porta Endoscopio.
CAD:	Diseño Asistido por Computador.
PREH:	Parallel Robot for Endoscope Holder
CO ₂ :	Gas Carbónico.
UPU:	Universal – Prismático – Universal.
RTL:	Ejes de rotación o vertical, Transversal o Lateral y longitudinal.
MGI:	Modelo Geométrico Inverso.
MGD:	Modelo Geométrico Directo.
MCI:	Modelo Cinemática Inverso.
MCD:	Modelo Cinemática Directo.
MDD:	Modelo Dinámico Directo.
MDI:	Modelo Dinámico Inverso.
GDL:	Grados de libertad.
FDA:	Administración Federal de Drogas de los Estados Unidos.
MPP:	Mecanismos Paralelos Planares.

Lista de símbolos

m :	Movilidad de un mecanismo de espacio isostático.
L :	Número total de sólidos de la cadena cinemática.
n :	Número total de articulaciones de la cadena cinemática.
d :	Número de grados de libertad de la articulación.
CK :	Número de grados de libertad asociados a cada segmento.
p :	Numero de segmentos.
X, Y, Z :	Coordenadas cartesianas en el marco de referencia de la base.
U, V, W :	Coordenadas cartesianas en el marco de referencia de la plataforma.
O :	Centro de la base.
P :	Centro de la plataforma.
A_1, A_2, A_3 :	Puntos de fijación de los segmentos en la base.
P_1, P_2, P_3 :	Puntos de fijación de los segmentos en la plataforma.
r_b :	Radio de la base.
r_p :	Radio de la plataforma.
β_i :	Angulo entre el eje X y el vector OA_i (con $i = 1,2,3$)
${}^O T_P$:	Matriz de transformación.
$C\beta$:	Coseno de β
$S\beta$:	Seno de β
P_x, P_y, P_z :	Coordenadas del centro de la plataforma con respecto a la base.
${}^i A_j$:	Matriz de rotación.
q_i :	Longitud del brazo i (con $i = 1,2,3$)
L_T :	Longitud del segmento central.
L_1 :	Porción del segmento central que está antes del trocar
L_2 :	Porcion del segmento central que está dentro del paciente.
α, β, γ :	Ángulos sobre los ejes X, Y y Z , respectivamente.
J_x, J_q, J :	Matrices jacobinas del robot.
\dot{q}_i :	Velocidades articulares de la junta i .
\dot{x}_i :	Velocidades cartesianas del efector final.
\ddot{q}_i :	Aceleraciones articulares de la junta i .
\ddot{x}_i :	Aceleraciones cartesianas del efector final.
Γ_i :	Pares o fuerzas del brazo i .

$A(q)$: Matriz de Inercia
 $C(q, \dot{q})$: Vector que representa las fuerzas de coriolis y centrífugas.
 $Q(q)$: Vector de fuerzas de gravedad.
 F_v : Frotamiento viscoso de la articulación.
 F_s : Frotamiento seco de la articulación.

INTRODUCCIÓN

La robótica se ha convertido en una herramienta de gran utilidad en cuanto intervenciones quirúrgicas se trata. Las cirugías convencionales son muy extensas y agotadoras; lo que ocasiona una serie de inconvenientes, como mala posición del cirujano, imprecisión en los cortes y mayor traumatismo para el paciente. Estos, entre otros riesgos, se han eliminado gracias al empleo de Robots, realizando el trabajo con mayor precisión y destreza, por lo que se han desarrollado técnicas para mejorar este tipo de procedimientos y sus aplicaciones en varios tipos de cirugía.

En la actualidad existen, gran cantidad de robots que se utilizan en cirugía, la mayoría de ellos son de tipo serie, arborescentes o híbridos. En el caso de la cirugía laparoscópica; específicamente en una colecistectomía, se encuentran robots serie, los cuales son robots muy robustos y costosos, por lo que se decidió, incursionar en el diseño de un robot paralelo, llamado PREH (*Parallel Robot for Endoscope Holder*).

Un robot paralelo, también llamado robot de cadena cerrada o manipulador paralelo, consiste básicamente en una plataforma móvil unida a una base fija por medio de varios brazos, peculiarmente cada brazo está controlado por un actuador. En general estos robots paralelos pueden manipular mayor carga, son estructuras más rígidas y precisas que los robots serie o de cadena abierta, ya que comparten la carga entre varios brazos [1].

Los robots paralelos fueron introducidos hace ya algunas décadas por Gough (1975) [2] y Stewart (1965) [3]. Clavel (1989) propuso el robot Delta, este es utilizado en aplicaciones de alta velocidad en la industria [4]. En el campo de la medicina tenemos El robot en miniatura MARS; el cual se usa en procedimientos quirúrgicos en Neurocirugía, en especial en la intra-medularidad de huesos largos [5] y [6]; el CTBot se centra en las biopsias de los órganos abdominales como el hígado y la quema de tumores mediante micro-ondas de radiofrecuencia [7]; el SpineAssist, es un robot paralelo utilizado para realizar perforaciones en la columna vertebral [8]; el robot paralelo exacto de 6-DOF con arquitectura HTRT;

usado para corregir el contragolpe estructural [9], entre otros. Y en cirugía laparoscópica tenemos a ARTEMIS, ENDOBOT, BLACK FALCON, ZEUS y DA VINCI [10] y [11]. Los dos últimos son utilizados en un gran número de procedimientos quirúrgicos, como por ejemplo la colecistectomía (extracción de una vesícula biliar enferma) [12]; recordemos que en la Universidad del Cauca se ha diseñado, un robot de este tipo llamado LAPBOT [13].

En este trabajo se presenta en el primer capítulo, algunos conceptos básicos sobre cirugía laparoscópica y robótica paralela, en el capítulo 2 se obtienen los modelos matemáticos del robot diseñado, en el tercer capítulo se muestran las pruebas realizadas a los distintos modelos y los diseños de todas las piezas del robot, además el robot ensamblado en un software CAD. Finalmente en el capítulo 4, se implementa un control CTC, con el cual se realizan varias pruebas, de seguimiento de trayectorias.

1. Conceptos básicos sobre robótica y cirugía laparoscópica.

1.1. Laparoscopia

La exploración del interior de nuestro organismo se inicia con la presentación del cistoscopio por Maximilian Nitze en la Real e Imperial Sociedad de Medicina de Viena en 1879, instrumento mejorado en 1886 por Leiter, al adaptarle una pequeña lámpara incandescente de Edison [14]; luego de unos años fue George Kelling, en Dresde, quien utilizó el cistoscopio urológico descrito por Nitze, lo introdujo a través de un orificio abierto en la pared abdominal de un perro con la finalidad de inspeccionar el contenido intestinal. En Hamburgo en septiembre de 1901, a esta técnica de exploración la denominó “celioscopia” [15], por la misma fecha OTT, un ginecólogo de San Petersburgo, describió la “ventroscopia” con la que visualizaba el interior de la cavidad a través de una cánula iluminada por un fotóforo frontal [16]. En 1910, en Estocolmo, H. C. Jacobeus [17] emplea el cistoscopio en humanos lo introduce en el abdomen a través de un trocar tras distender la cavidad con agua o aire indistintamente, para explorar su interior y denominó al método “laparoscopia”. Un problema de la laparoscopia consistía en que la iluminación abdominal requería de elementos calientes, pero a finales de la década de 1950, con el descubrimiento de las fibras ópticas, se desarrolló el endoscopio flexible que permite el acceso a la cavidad abdominal sin generar quemaduras [18].

Eric Muhe, el 12 de septiembre de 1985 en el hospital de Boblinger en Alemania, realiza la primera colecistectomía laparoscópica en el mundo [19], y en 1987 Mouret [20] desarrolló un estudio completo sobre la misma cirugía, de tal magnitud que en 1992 se convirtió en un estándar [21].

En una cirugía laparoscopia se necesita separar la pared abdominal de los órganos, elevando el abdomen para que el cirujano tenga acceso al interior del paciente y pueda manipular los instrumentos con relativa facilidad. La elevación se realiza con insuflación de CO₂, dado que suprime la combustión y el cuerpo lo

absorbe con mayor rapidez [22]. La aguja de Veress es el instrumento que se utiliza para puncionar la cavidad abdominal e insuflar el abdomen, está constituida por una camisa metálica de punta aguda, en su interior posee un vástago retráctil y su longitud está entre 0.08 y 0.2 m. Después de obtener acceso al abdomen se realizan incisiones de 0.005 a 0.01 m (5 a 10 mm), para introducir los trocares que son instrumentos con un punzón cortante, revestidos por una funda o cánula, su diámetro interior varía de 0.005 a 0.0125 m, su longitud está entre 0.1 y 0.13 m y se elaboran con acero quirúrgico [23]. En la Figura 1-1 Figura 1-1: Aguja de Veress (izquierda), Trocar (derecha) [23] se puede observar una aguja y un trocar.



Figura 1-1: Aguja de Veress (izquierda), Trocar (derecha) [23].

Por medio de los trocares se insertan la pinzas, tijeras, cauterizadores o cualquier otro instrumento que se requiera, además del sistema de visión (endoscopio) [23]. La Figura 1-2 muestra la configuración de los elementos, el sistema de visión posee el cable de luz óptica y los trocares aparecen de color violeta.

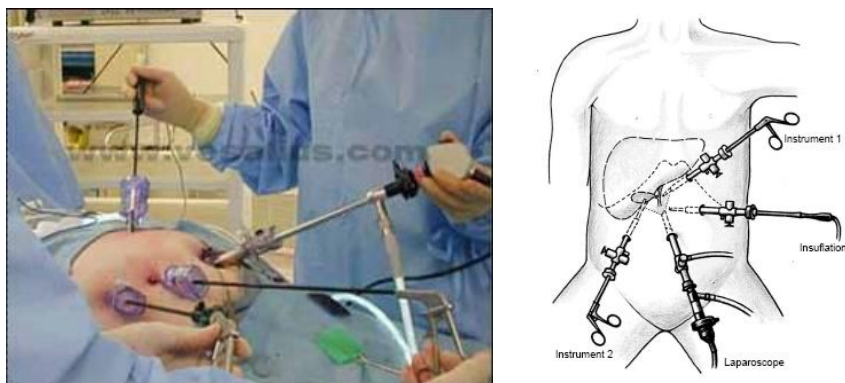


Figura 1-2: Configuración de elementos básicos para laparoscopia [23].

La distribución del equipo quirúrgico es la siguiente: el cirujano se coloca sobre el lado izquierdo del paciente y junto a él, el ayudante que manejará el laparoscopio y la cámara. Sobre el lado derecho, el primer ayudante y el instrumentista. En el sector derecho de la sala de operaciones se dispone el equipo, de tal manera que el cirujano tenga un permanente control visual del insuflador. Se utilizan dos monitores colocados lateralmente a 45 grados en relación con la mesa de operaciones, para que todos los miembros del equipo quirúrgico tengan un cómodo acceso visual a las pantallas [24]. La Figura 1-3 muestra la distribución del quirófano para una laparoscopia.

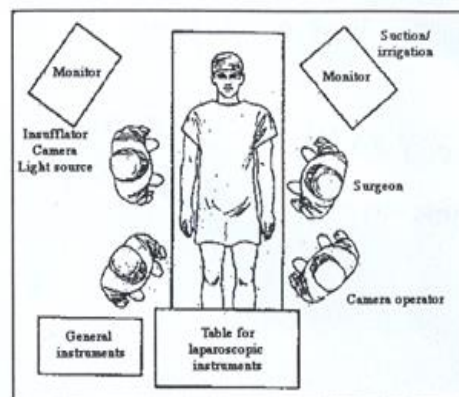


Figura 1-3: Distribución del quirófano para una laparoscopia [24].

El empleo de un robot paralelo para ser usado como asistente porta endoscopio en este tipo de cirugías, gracias a su pequeño diseño y portabilidad, permitirá una correcta ubicación sin que incomode al cirujano, mejorando la distribución del quirófano y ampliando el área de trabajo del cirujano.

1.2. Robots porta endoscopio en cirugía laparoscópica

En 1979, el instituto "Robot Institute of America", después denominado "Robot Industries Association", definía un robot como: un manipulador reprogramable y multifuncional diseñado para trasladar materiales, piezas, herramientas o aparatos específicos, a través de una serie de movimientos programados a cabo una variedad de tareas.

En el campo de la medicina y en especial en la cirugía laparoscópica son conocidos como robots clínicos y su tarea principal es auxiliar al médico cirujano, contribuyendo con la fijación y el control del laparoscopio.

Este instrumento óptico trabaja como una cámara de video permitiendo ver lo que hay en el interior del organismo. El cirujano le da instrucciones al robot para que se desplace y así ubique la zona de interés; la función del robot es sostener y posicionar el laparoscopio; estas tareas le correspondían originalmente al asistente del cirujano, sin embargo se observa que por la duración prolongada de las cirugías, que en algunos casos pueden llegar a las 8 horas, el auxiliar se fatiga, por lo que la imágenes se tornan un poco distorsionadas, lo que ocasiona una deficiencia en el proceso operatorio. Es por esta y muchas otras razones que se justifica la implementación de la robótica en este tipo de tareas; ya con esto el Robot realizará la tarea del asistente sin la preocupación del agotamiento [25].

AESOP (siglas en inglés de Sistema Óptimo de Posicionamiento Endoscópico Automatizado), es utilizado con frecuencia en centros especializados, al igual que ENDOASSIST son sistemas quirúrgicos robóticos para uso clínico durante operaciones abdominales dentro del territorio estadounidense aprobados por la Administración Federal de Drogas de los Estados Unidos (FDA), estos robots remplazan al camarógrafo y sostienen el telescopio de video durante las cirugías laparoscópicas [26], en la Figura 1-4 se puede observar este sistema.



Figura 1-4: Robot ENDOASSIST [27].

El AESOP, ver Figura 1-5, consiste en un brazo mecánico conectado a una computadora que reconoce órdenes verbales sencillas que el robot traduce en movimientos de la cámara laparoscópica. Las ventajas de utilizar este robot son la liberación de un brazo del cirujano, la disminución del número de personas que se requieren para la cirugía y la precisión de la imagen de la cirugía [28].

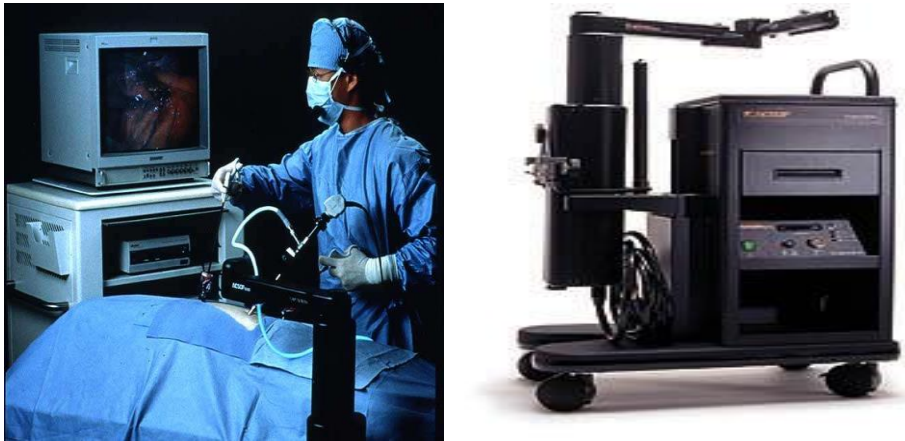


Figura 1-5: Robot AESOP [28].

1.2.1. Avances Tecnológicos en Colombia y la Universidad del Cauca acerca de Cirugía Laparoscópica

En la actualidad en Colombia los proyectos de robótica son muy pocos, la construcción y prueba real de dichos robots aún no se han llevado a cabo, sin embargo se está trabajando en el diseño y simulación de robots quirúrgicos de gran importancia como lo son:

- **LAPBOT** (Grupo de Investigación de Automática Industrial de la Universidad del Cauca, Colombia) Robot para cirugía laparoscópica, permite posicionar y orientar los instrumentos quirúrgicos dentro del abdomen de un paciente, pasando a través de un punto fijo en el espacio cartesiano que representa la incisión. Presenta un ambiente de simulación tridimensional con asistente robóticos seriales, como se puede ver en la Figura 1-6 [13].

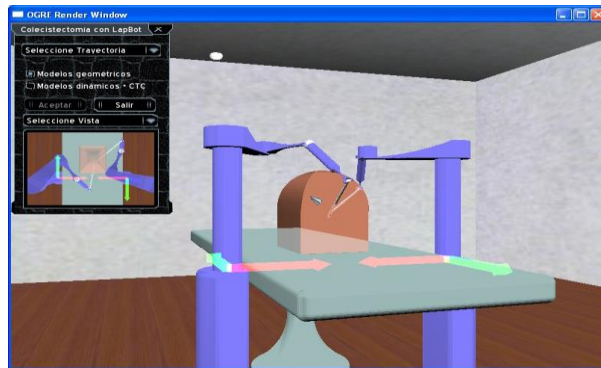


Figura 1-6: Ambiente 3D con dos LapBot y la interfaz de usuario [13].

- **KIRUBOT** (Universidad Pontificia Bolivariana, Medellín, Colombia) Es un proyecto de diseño y construcción de un prototipo de un brazo robótico controlado por instrumentación virtual para la realización de procedimientos quirúrgicos. Este brazo robótico es controlado mediante una interfaz tridimensional que permite la ubicación de las coordenadas de un punto en el espacio [29]. El KIRUBOT se puede observar en la Figura 1-7.



Figura 1-7: Robot KIRUBOT [29].

- **HIBOU** (Grupo de Investigación de Automática Industrial de la Universidad del Cauca, Colombia) Es un robot porta endoscopio serie, con siete grados de libertad. Su entorno 3D fue realizado en OGRE como se puede ver en la siguiente Figura 1-8 [30].



Figura 1-8: Robot Hibou en Ogre3D [30].

El robot diseñado en esta tesis, llamado PREH, por sus siglas en inglés: *Parallel Robot for Endoscope Holder*, busca mejorar algunos aspectos de los robots serie utilizados en cirugía laparoscópica, entre los más importantes se tendría la portabilidad entre quirófanos, la precisión y exactitud.

1.3. Robótica paralela.

1.3.1. Definición de robot paralelo.

Un robot paralelo está compuesto por una cadena cinemática cerrada, la cual consta de cadenas seriales separadas que conectan al eslabón fijo (plataforma fija) con el efector final o eslabón móvil (plataforma móvil) como se muestra en la Figura 1-9.

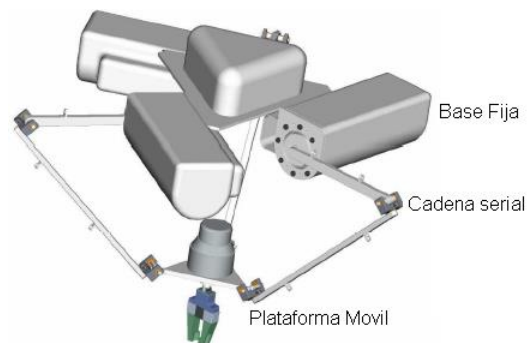


Figura 1-9: Estructura básica de un robot paralelo [31].

1.3.2. Clasificación de los robots paralelos según su movilidad.

Una manera de clasificar a los mecanismos paralelos se basa en las características del movimiento que puedan desarrollar, pueden ser planares y espaciales, dentro de los robots espaciales existen los robots paralelos esféricos y los traslacionales [31].

1.3.2.1. Robots paralelos planares

En los mecanismos paralelos planares (MPP), su movimiento está sujeto a un plano, puede tener 2 o 3 grados de libertad (GDL); traslación en el plano y una rotación sobre el eje perpendicular al mismo. La Figura 1-10 muestra algunas configuraciones estructurales con 2 y 3 grados de libertad [32].

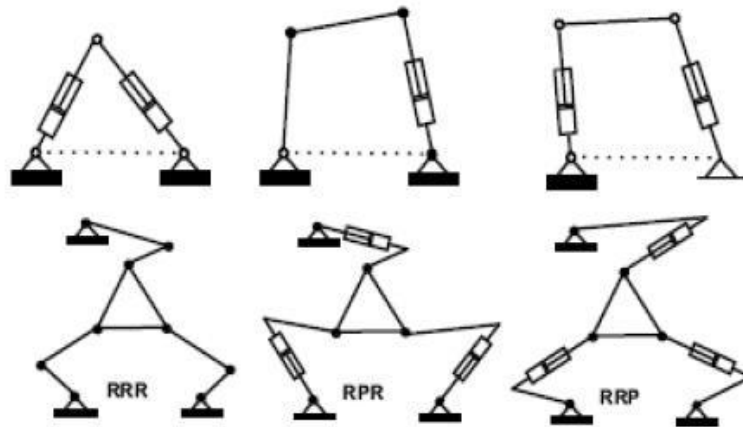


Figura 1-10: Mecanismos de 2 y 3 GDL [32].

Uno de los mecanismos paralelos más básicos es el robot paralelo planar 3RRR [33], el cual se muestra en la Figura 1-11, este robot está formado por 3 brazos con 3 articulaciones rotacionales, una articulación de cada brazo, las que están fijas a la base, deben ser actuadas. Este robot cuenta con 3 GDL, dos son traslaciones y una rotación.

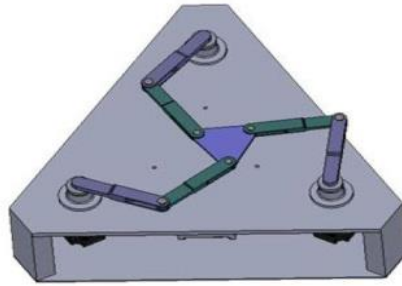


Figura 1-11: Robot paralelo planar 3RRR [33].

1.3.2.2. Robot paralelos espaciales

Un ejemplo claro de robot paralelo espacial es la “Plataforma Stewart” diseñada originalmente por Gough y Stewart [3], Figura 1-12. Inicialmente la plataforma fue pensada con 6 GDL, para ser usada como simulador de vuelo. Actualmente las variantes de este mecanismo permiten, que sea utilizado en diferentes ámbitos.

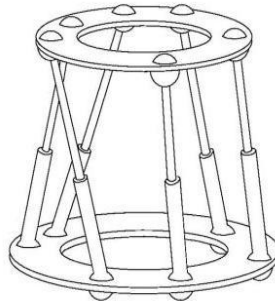


Figura 1-12: Plataforma Stewart [3].

Dentro de los robots espaciales, existentes los robots paralelos esféricos y los traslacionales.

- **Robot paralelo esférico**

En los robots paralelos esféricos el movimiento de sus eslabones describe curvas alrededor de un punto común, creando una esfera concéntrica. Se caracteriza por tener dos articulaciones de rotación y una prismática. La Figura 1-13 muestra un robot esférico de 3 GDL, con tres articulaciones rotacionales, desarrollado para obtener una rápida orientación de la cámara [34].

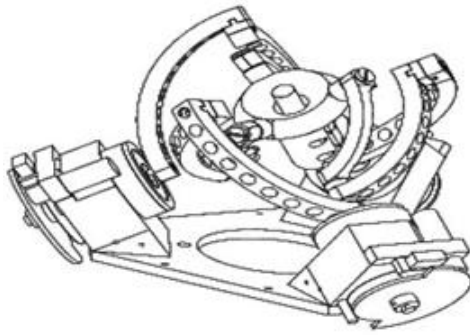


Figura 1-13: Robot esférico 3 GDL [34].

- **Robot paralelo traslacional.**

Son manipuladores de tres grados de libertad netamente traslacionales, son muy utilizados cuando se necesitan ubicaciones muy exactas y en operaciones de mecanizado [31].

El robot más famoso de este tipo es el Delta (ver figura 1-14), fue desarrollado inicialmente por Clavel, en la Escuela Politécnica de Lausanne (EPFL) [31] e implementado por la *Demaurex Company* y ABB, bajo el nombre de IRB 340 *FlexPicker*. Todas las cadenas cinemáticas de este robot son del tipo RRPaR, un motor hace que una articulación rotacional gire alrededor del eje x, a esta junta se adhiere una palanca y al final de ella se fija otra articulación rotacional con el eje paralelo al eje x. Un paralelogramo, Pa, se fija a este conjunto y permite la traslación en las direcciones paralelas a x, al final del paralelogramo se ajusta otra articulación rotoide con el eje paralelo a x, esta última está fija al efector final o plataforma móvil.



Figura 1-14: Robot Delta [31].

1.4. Manipuladores paralelos en cirugía

Entre los asistentes quirúrgicos robóticos paralelos se encuentran:

1.4.1. Robot miniatura para los procedimientos quirúrgicos en Neurocirugía

El robot en miniatura MARS, es una estructura paralela, equipado de una guía mecánica para la inserción de la aguja, catéter o la sonda. Este robot fue diseñado para la perforación y ubicación de instrumental en el cráneo (ver Figura 1-15) [5] y [6]. Entre sus ventajas se encuentran su pequeño tamaño y diseño: puede ser montado directamente sobre el hueso del paciente, y por lo tanto no requiere de seguimiento o inmovilización de la cabeza [5].



Figura 1-15: Robot MARS colocado directamente en el cráneo [5].

1.4.2. Sistema robótico para la intra-medularidad de huesos largos

Este trabajo presenta un nuevo robot basado en el sistema que ayuda a los cirujanos ortopédicos para el tratamiento de fracturas de fémur y la tibia. El sistema consta de un robot miniatura montado en el hueso (ver Figura 1-16), equipado con una guía de perforación mecánica que proporciona la orientación para la perforación manual de agujeros de los tornillos piloto distales. Por su pequeño tamaño y peso, el robot puede ser montado directamente sobre el hueso [35].

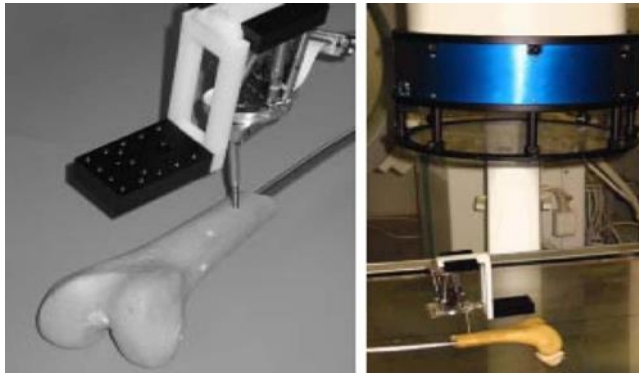


Figura 1-16: Robot miniatura montado en el hueso [35].

1.4.3. CTBOT

Este proyecto se centra en las biopsias de los órganos abdominales como el hígado y la quema de tumores mediante micro-ondas de radiofrecuencia, gracias a una aguja insertada directamente en su interior [7]. El prototipo está sobre el abdomen del paciente, de modo que los movimientos del paciente y los movimientos externos como la respiración son naturalmente compensados (ver Figura 1-17).

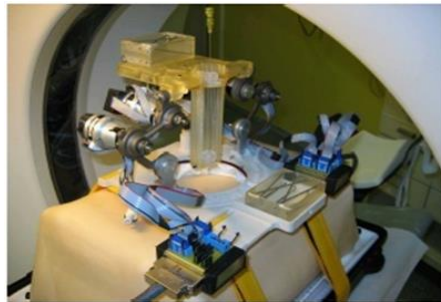


Figura 1-17: Robot CTBOT [7].

1.4.4. SPINEASSIST

Es un robot paralelo utilizado para realizar perforaciones en la columna vertebral, puede ser utilizado en otras operaciones de columna como la vertebroplastia, discografía, y varias biopsias [8]. El procedimiento se realiza en toda la columna vertebral, sobre la vértebra lumbar baja, vértebra torácica media, y en la vértebra cervical superior (ver Figura 1-18).



Figura 1-18: Robot SPINE ASSIST [8].

1.4.5. Robot paralelo exacto de 6-DOF con arquitectura HTRT

Este robot es impulsado por un servomotor de corriente alterna, conformado por una plataforma móvil, una plataforma de base y seis cadenas idénticas que conectan las dos plataformas. Una junta universal o Hooke joint, que se compone de un eje central en cruz y cuatro rodamientos de bolas específicos sin el pivote de círculo interior, es usada para corregir el contragolpe estructural [9].

1.5. Robots empleados en cirugía

En la Tabla 1-1 se presenta un resumen de los robots más empleados en cirugía [36] - [62].

Tabla 1-1: Robots más empleados en cirugía

País	Área clínica	Institución / Fabricante	Robot	Estudios
Suiza	Neurocirugía	Universidad de Lausanne	Minerva	Humanos
Francia	Neurocirugía	Integrated Surgical Systems / Hospital Universitario de Grenoble	NeuroMate	Humanos
Japón	Neurocirugía	Universidad de Tokio	MRI compatible	Muestras de tejidos

Israel	Neurocirugía	Mazor	SpineAssist	Humanos
USA	Radiocirugía	Accuracy	Cyberknife	Humanos
USA	Ortopedia	Georgetown/Hopkins	PAKY/RCM	Cadáver
USA	Ortopedia	Integrated Surgery Systems	ROBODOC	Humano
Alemania	Ortopedia	OrthoMarquet	CASPAR	Humano
USA	Ortopedia	Marconi	Kawasaki	Cerdo
Reino Unido	Urología	Imperial College	Acrobot	Humano
Usa	Urología	Hopkins	PAKY/RCM	Humano
Francia	Maxilofacial	Charite	SurgiScope	Humano
Alemania	Maxilofacial	Karlsruhe/Heidelberg	RX 90	Cerdo
USA	Oftalmología	Hopkins	Steady-Hand	En desarrollo
USA	Cardiología, Laparoscopia, Urología.	Intuitive Surgical Systems	Da vinci	Humano
USA	Cardiología, Laparoscopia, Urología.	Intuitive Surgical Systems	Zeus	Humano
Francia	Cardiología	Grenoble	PADyC	En desarrollo
USA	Cardiología	Carnegie Mellon Universidad	Heartlander	Cerdo
Francia	Dermatología	Montpellier	SCALPP	En desarrollo
USA	Laparoscopia	Kang y Wen	Endobot	Humano
USA	Laparoscopia	Madhani	Black Falcon	Humano
Alemania	Laparoscopia	Karlsruhe Nuclear Research Center and The University of Tuebingen	ARTEMIS	Humano
Francia	Neurocirugía	Armstrong Healthcare	Pathfinder	Humano
Reino Unido	Ortopedia	Acrobot Company	Acrobot	Humano
Reino Unido	Urología	Imperial Collage	Probot	Humano

Este robot paralelo PREH (*Parallel Robot for Endoscope Holder*) diseñado acorde con las limitaciones propias de un ambiente quirúrgico, involucra un proceso de análisis matemático de los modelos del robot, el cual cumple con las especificaciones de espacio abdominal en una colecistectomía, para el manejo del laparoscopia.

2. Modelado del Robot PREH

En esta sección se define la estructura del robot, se hace su descripción mecánica, se obtienen los modelos matemáticos teniendo en cuenta la restricción del paso por la incisión.

Este robot paralelo fue llamado PREH, por sus siglas en inglés: *Parallel Robot for Endoscope Holder*.

2.1. Requerimientos de diseño

El robot diseñado debe cumplir ciertos requerimientos:

- Tamaño reducido
- Portable.
- Fácil traslado entre quirófanos
- Aséptico
- Cumplir con el espacio de trabajo de una cirugía laparoscópica.
- Respetar el paso por la incisión.
- Exactitud en la ubicación

Para definir qué tipo de estructura cumple con el área de trabajo de una cirugía laparoscópica, se analiza la clasificación de los robots paralelos mostrada en el capítulo 1.

1. Los robots paralelos planares tienen 3 GDL, dos de traslación y uno de rotación sobre el eje perpendicular al plano, este tipo de robots no es útil en una cirugía laparoscópica, ya que el endoscopio no tendría profundidad, es decir desplazamiento en el eje Z.
2. Un robot paralelo espacial de 6 GDL, sería el más apropiado para una cirugía laparoscópica, ya que cuenta con 3GDL de traslación y 3GDL de rotación, dándole todos los movimientos que el endoscopio necesita dentro

del abdomen. Su gran inconveniente es que la plataforma debe estar unida a la base mediante 6 brazos o cadenas cinemáticas, lo que implica un gran tamaño, reduciendo el espacio del cirujano en el quirófano.

3. Un robot paralelo esférico, sería muy útil para garantizar el paso por la incisión en el abdomen del paciente, ya que el movimiento de sus eslabones describe curvas alrededor de un punto fijo común, creando una esfera concéntrica, pero se tendría el mismo problema que con un robot planar, el endoscopio no contaría con profundidad dentro del abdomen del paciente.
4. Por último, se analizó el robot paralelo traslacional, éste cuenta con 3GDL netamente traslacionales, dándole libertad al endoscopio para alcanzar cualquier punto en el espacio dentro del abdomen del paciente. Gracias a que la plataforma está fija a la base mediante 3 brazos, su tamaño puede ser muy reducido. Este tipo de robots son muy utilizados cuando se necesitan ubicaciones exactas.

Para los requerimientos que se necesitan, se pueden tener 3 inconvenientes utilizando este tipo de robots, los cuales son fáciles de solucionar.

Respeto del paso por el trocar: Para que el endoscopio respete el paso por la incisión abdominal, se puede adicionar un cuarto brazo, ubicado entre el centro de la base y el centro de la plataforma, esto no incrementa el tamaño del robot, ni sus grados de libertad, ya que todas las articulaciones son pasivas. Este brazo también cumplirá la función de soporte de la cámara.

El segundo inconveniente es la rotación de la cámara: se podría pensar que al no tener la plataforma rotaciones sobre sus ejes, la cámara tampoco, pero, al incluir el cuarto brazo, además de respetar el paso por la incisión, se crea un pivote para el endoscopio, el cual haría que este tenga cierto grado de inclinación, ver Figura 2-1 .

El tercer inconveniente son los movimientos rotacionales paracitos o indeseados de la plataforma: para ello se deben activar los tres actuadores al mismo tiempo, no lo podrán hacer por separado. Esta es la primera condición de diseño.

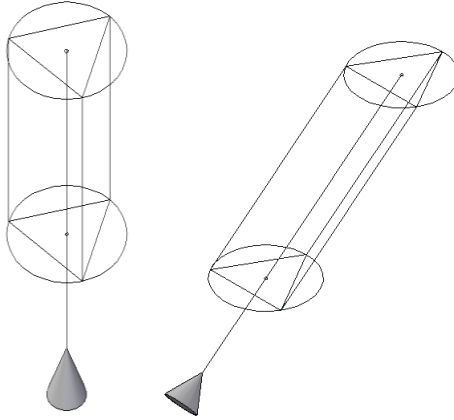


Figura 2-1: Inclinación de la cámara.

2.2. Descripción del sistema mecánico

Como se observa en la Figura 2-2, el diseño del mecanismo se describe de la siguiente manera: en la base se tienen 3 puntos A_1 , A_2 y A_3 que forman un triángulo equilátero inscrito en una circunferencia de radio r_b . El Origen del sistema de coordenadas se fija en la base sobre el punto A_1 , el plano xy coincide con el plano formado por el triángulo y el eje x apunta en la dirección del vector $\overline{A_1A_2}$. La misma situación se establece en la plataforma. Se tiene un triángulo equilátero con vértices P_1 , P_2 y P_3 inscrito en una circunferencia de radio r_p . Los parámetros r_b y r_p , son los dos primeros parámetros de diseño.

Ahora, cada par de puntos A_i, B_i están conectados por una extremidad de longitud q_{3i} con juntas universales en cada extremo. El segundo y tercer eje de cada brazo son paralelos entre sí y perpendiculares al eje de la extremidad. El primer y cuarto eje están integrados en la base y en la plataforma, respectivamente, de tal manera que cada uno de ellos apunta al centro de la circunferencia (ver Figura 2-2). Estas

condiciones garantizan que al accionar simultáneamente las articulaciones prismáticas, la plataforma realiza un movimiento de traslación pura [63].

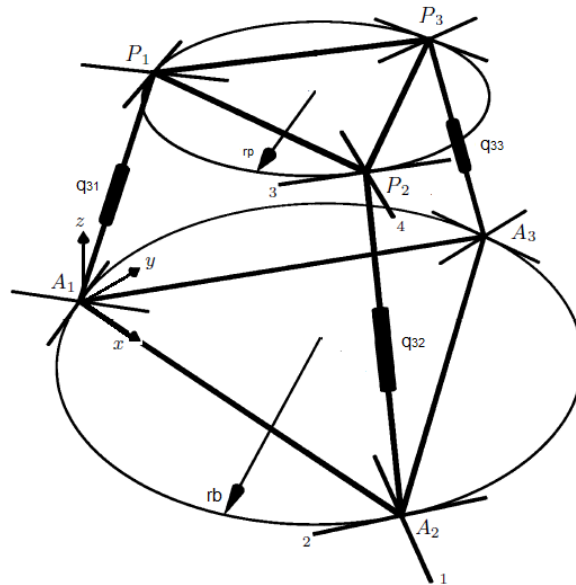


Figura 2-2: Estructura inicial del robot.

Se añadirá un segmento central, con juntas esféricas en sus extremos, para garantizar que este respete el paso por el trocar y para que sirva como soporte al laparoscopio (ver Figura 2-3).

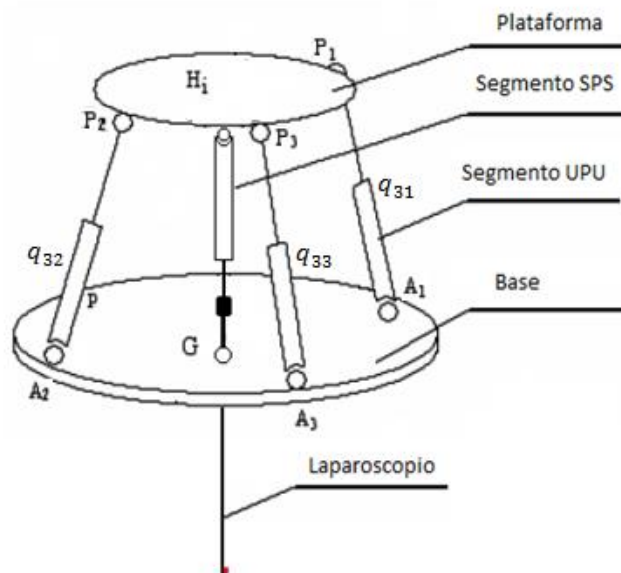


Figura 2-3: Diagrama del robot.

2.3. Modelo geométrico

El modelo geométrico representa las características físicas del robot en un sistema de coordenadas referenciado a cada articulación [64].

Existen dos modelos geométricos, el directo, que se utiliza para determinar en cualquier instante la posición exacta del órgano terminal y el inverso, que nos indica cuáles articulaciones se deben mover para llevar el órgano terminal a un punto determinado.

En otras palabras, en el modelo geométrico directo se conoce la posición de cada articulación del robot y este permite conocer la posición y la orientación del órgano terminal. En el modelo geométrico inverso se conoce la posición cartesiana del efector final y este provee todas las soluciones posibles de las coordenadas articulares.

2.3.1. Modelo geométrico Inverso

A diferencia de los robots serie, el modelo geométrico inverso de robots paralelos es en general único y fácil de obtener, con algunas relaciones geométricas básicas [65]. Este modelo proporciona las variables articulares activas (q_{3i} , para $i = 1$ a 3) en función de las coordenadas cartesianas del punto de referencia de la plataforma P_1 . Se puede notar que las expresiones de la longitud del segmento son independientes uno de otro.

Para calcular el MGI, se debe hallar la relación entre el órgano terminal y el origen del sistema de coordenadas, punto A_1 en la base, por tanto se procederá a asociar la plataforma y la base, y por último, la plataforma y el efector final (endoscopio).

En primera instancia se calculan las coordenadas de los puntos de conexión de los brazos tanto en la base como en la plataforma, ver Figura 2-4.

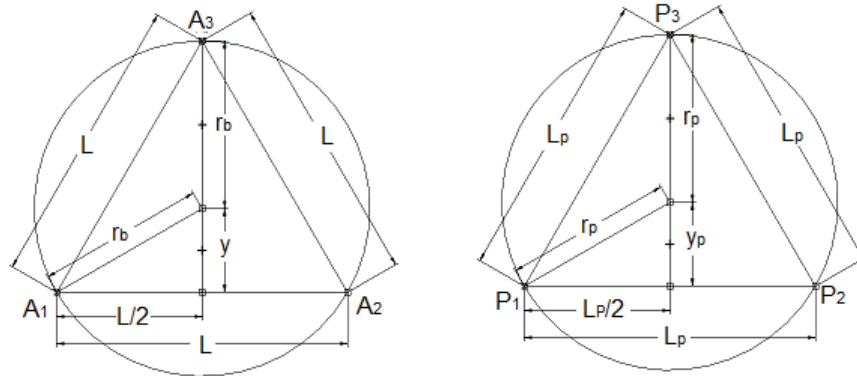


Figura 2-4: Coordenadas de los puntos $A_1, A_2, A_3, P_1, P_2, P_3$.

Dado que el origen del sistema de coordenadas se encuentra sobre el punto A_1 , se deben obtener las coordenadas de los puntos A_2 y A_3 en función de las coordenadas de este, para tal fin, se calcula la longitud, L , del lado del triángulo equilátero inscrito en la circunferencia de radio r_b y la distancia, y , medida desde el centro de la circunferencia hasta la base del triángulo, como se muestra en la Figura 2-4.

Se toma el triángulo rectángulo formado por los catetos $(L/2)$ y $(r_b + y)$ y de hipotenusa L , utilizando el teorema de Pitágoras se tiene:

$$L^2 = \frac{L^2}{4} + (r_b + y)^2$$

$$\frac{L^2}{4} = \frac{(r_b + y)^2}{3} \quad (2-1)$$

Ahora se toma el triángulo rectángulo formado por los catetos $(L/2)$ y (y) y de hipotenusa r_b , nuevamente se utiliza el teorema de Pitágoras para obtener:

$$r_b^2 = \frac{L^2}{4} + y^2$$

$$\frac{L^2}{4} = r_b^2 - y^2 = (r_b - y)(r_b + y) \quad (2-2)$$

Con esto se tiene un sistema de ecuaciones 2x2, para resolverlo se igualan las expresiones (2-1) y (2-2),

$$\frac{(r_b + y)^2}{3} = (r_b - y)(r_b + y)$$

$$r_b + y = 3r_b - 3y$$

$$y = \frac{r_b}{2} \quad (2-3)$$

Este valor se reemplaza en la ecuación (2-2),

$$\frac{L^2}{4} = r_b^2 - \frac{r_b^2}{4}$$

$$\frac{L^2}{4} = \frac{3r_b^2}{4}$$

$$L = \sqrt{3}r_b \quad (2-4)$$

Se sigue el mismo procedimiento para calcular L_p y y_p (en la plataforma), encontrando que:

$$y_p = \frac{r_p}{2} \quad (2-5)$$

$$L_p = \sqrt{3}r_p \quad (2-6)$$

Las coordenadas de los puntos O , A_1 , A_2 y A_3 están dadas por:

$$A_1 = (0, 0, 0)$$

$$A_2 = (\sqrt{3}r_b, 0, 0)$$

$$A_3 = (\sqrt{3}r_b/2, 3r_b/2, 0)$$

$$O = (\sqrt{3}r_b/2, r_b/2, 0)$$

Donde, O , es el centro de la base.

Cabe recordar que la plataforma no tiene rotaciones sobre sus ejes, por tal motivo, los puntos P_0 , P_1 , P_2 , y P_3 estarán siempre a la misma distancia P_z , medida desde la base, sobre el eje coordenado Z . Por tal motivo las coordenadas de dichos puntos se expresan como sigue:

$$P_1 = (P_{1x}, P_{1y}, P_z)$$

$$P_2 = (P_{1x} + \sqrt{3}r_p, P_{1y}, P_z)$$

$$P_3 = (P_{1x} + \sqrt{3}r_p/2, P_{1y} + 3r_b/2, P_z)$$

$$P_0 = (P_{1x} + \sqrt{3}r_p/2, P_{1y} + r_p/2, P_z)$$

Donde, P_{1x} y P_{1y} son las coordenadas, en X y Y , del punto P_1 medidas desde el origen (Punto A_1) y P_0 es el centro de la plataforma.

Se llaman q_{31} , q_{32} , q_{33} a los vectores que unen respectivamente los puntos $A_1 - P_1$, $A_2 - P_2$, $A_3 - P_3$, se puede calcular su magnitud de acuerdo con la siguiente expresión [66]:

$$\|q_{3i}\| = \sqrt{(P_{ix} - A_{ix})^2 + (P_{iy} - A_{iy})^2 + (P_{iz} - A_{iz})^2} \quad (2-7)$$

Donde i indica el número del brazo.

Para una mejor visualización se elimina el radical del lado derecho, elevando al cuadrado a ambos lados de la expresión (2-7)

$$q_{3i}^2 = (P_{iX} - A_{iX})^2 + (P_{iY} - A_{iY})^2 + (P_{iZ} - A_{iZ})^2 \quad (2-8)$$

Reemplazando los valores de las coordenadas en la ecuación (2-8), se tiene:

$$q_{31}^2 = (P_{1X} - A_{1X})^2 + (P_{1Y} - A_{1Y})^2 + (P_{1Z} - A_{1Z})^2$$

$$q_{31}^2 = (P_{1X} - 0)^2 + (P_{1Y} - 0)^2 + (P_Z - 0)^2$$

$$q_{31}^2 = P_{1X}^2 + P_{1Y}^2 + P_Z^2 \quad (2-9)$$

$$q_{32}^2 = (P_{2X} - A_{2X})^2 + (P_{2Y} - A_{2Y})^2 + (P_{2Z} - A_{2Z})^2$$

$$q_{32}^2 = (P_{1X} + r_p\sqrt{3} - r_b\sqrt{3})^2 + (P_{1Y} - 0)^2 + (P_Z - 0)^2$$

$$q_{32}^2 = (P_{1X} + \sqrt{3}(r_p - r_b))^2 + P_{1Y}^2 + P_Z^2$$

$$q_{32}^2 = P_{1X}^2 + P_{1Y}^2 + P_Z^2 + 2\sqrt{3}P_{1X}(r_p - r_b) + 3(r_p - r_b)^2 \quad (2-10)$$

$$q_{33}^2 = (P_{3X} - A_{3X})^2 + (P_{3Y} - A_{3Y})^2 + (P_{3Z} - A_{3Z})^2$$

$$q_{33}^2 = (P_{1X} + \frac{\sqrt{3}}{2}r_p - \frac{\sqrt{3}}{2}r_b)^2 + (P_{1Y} + \frac{3}{2}r_p - \frac{3}{2}r_b)^2 + (P_Z - 0)^2$$

$$q_{33}^2 = (P_{1X} + \frac{\sqrt{3}}{2}(r_p - r_b))^2 + (P_{1Y} + \frac{3}{2}(r_p - r_b))^2 + P_Z^2$$

$$q_{33}^2 = P_{1X}^2 + P_{1Y}^2 + P_Z^2 + \sqrt{3}P_{1X}(r_p - r_b) + 3P_{1Y}(r_p - r_b) + 3(r_p - r_b)^2 \quad (2-11)$$

2.3.2. Cálculo de los ángulos de rotación de cada brazo

En esta sección se calculan los ángulos de rotación de los brazos que unen la base a la plataforma, para este fin, se moverá la plataforma en cada eje coordenado y se analizará la variación de los ángulos en cada caso, se debe tener

en cuenta que el origen del sistema de coordenadas está sobre el punto A_1 , los ejes de rotación de los brazos 1 y 2 están rotados respecto a los ejes coordenados X y Y, los ejes de rotación del brazo 3 son paralelos a los ejes X y Y, en la Figura 2-5 se muestra una vista de planta del robot, indicando cada uno de los ejes:

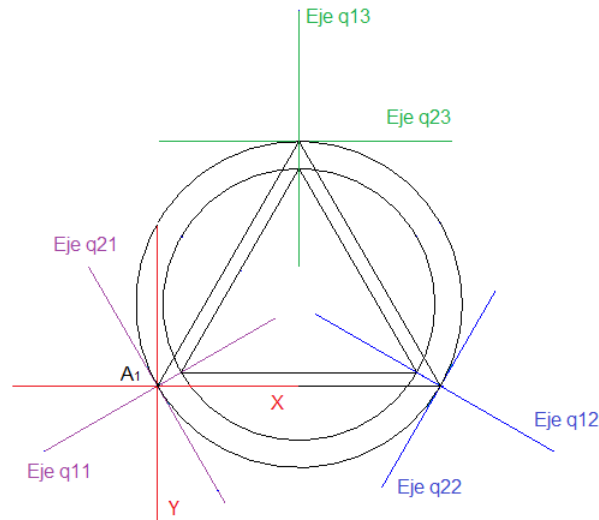


Figura 2-5: Ejes de rotación de las articulaciones de cada brazo.

Como ya se mencionó, la plataforma no tiene movimientos de rotación sobre sus ejes, además se la considera como un cuerpo rígido¹, por tal motivo al desplazar cualquier punto de ella, todos los puntos que la conforman se desplazarán a la misma distancia y en la misma dirección.

Al desplazar la plataforma una distancia P_y sobre el eje coordenado Y, se deben calcular las proyecciones lineales de este movimiento sobre los ejes q11, q21, q12 y q22.

En la Figura 2-6 se muestran las proyecciones de P_y sobre los ejes q11 y q21, que son los ejes de rotación de las articulaciones 1 y 2 del brazo 1. Se puede observar que mientras la plataforma se desplaza una distancia P_y sobre el eje Y, el desplazamiento sobre el eje q21 está dado por $-P_y \text{sen}(60)$, mientras que sobre el eje q11 está dado por $P_y \text{cos}(60)$.

¹ Un cuerpo rígido se define como aquel que no sufre deformaciones por efecto de fuerzas externas, es decir un sistema de partículas cuyas posiciones relativas no cambian.

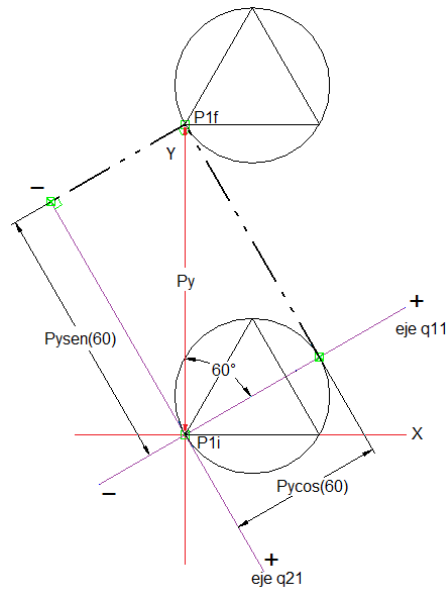


Figura 2-6: Proyecciones de P_y sobre los ejes q_{11} y q_{21} .

En la Figura 2-7 se muestran las proyecciones de P_y sobre los ejes q_{12} y q_{22} , que son los ejes de rotación de las articulaciones 1 y 2 del brazo 2. Se puede observar que mientras la plataforma se desplaza una distancia P_y sobre el eje Y, el desplazamiento sobre el eje q_{22} está dado por $P_y \sin(60)$, mientras que sobre el eje q_{12} está dado por $P_y \cos(60)$.

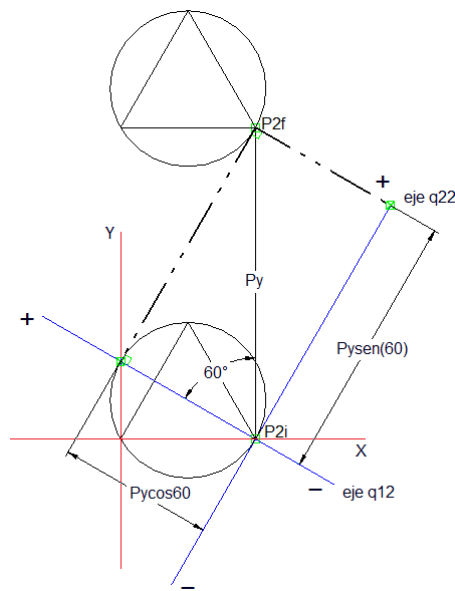


Figura 2-7: Proyecciones de P_y sobre los ejes q_{12} y q_{22} .

Ahora se realiza el mismo procedimiento pero desplazando la plataforma una distancia P_x sobre el eje coordenado X, igualmente se deben calcular las proyecciones lineales de este movimiento sobre los ejes q11, q21, q12 y q22.

En la Figura 2-8 se muestran las proyecciones de P_x sobre los ejes q11 y q21, que son los ejes de rotación de las articulaciones 1 y 2 del brazo 1. Se puede observar que mientras la plataforma se desplaza una distancia P_x sobre el eje X, el desplazamiento sobre el eje q21 está dado por $P_x \cos(60)$, mientras que sobre el eje q11 está dado por $P_x \sin(60)$.

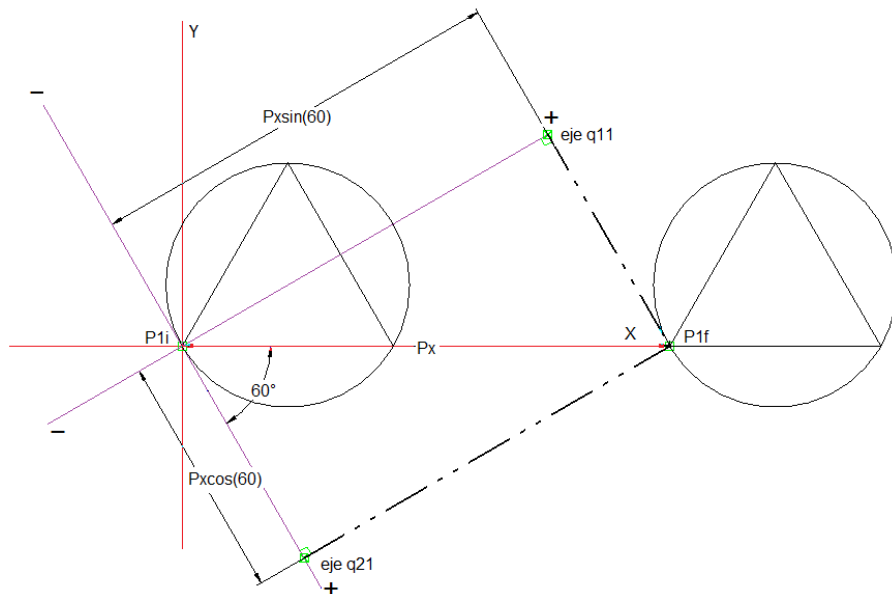


Figura 2-8: Proyecciones de P_x sobre los ejes q11 y q21.

En la Figura 2-9 se muestran las proyecciones de P_x sobre los ejes q12 y q22, que son los ejes de rotación de las articulaciones 1 y 2 del brazo 2. Se puede observar que mientras la plataforma se desplaza una distancia P_x sobre el eje X, el desplazamiento sobre el eje q22 está dado por $P_x \cos(60)$, mientras que sobre el eje q12 está dado por $-P_x \sin(60)$.

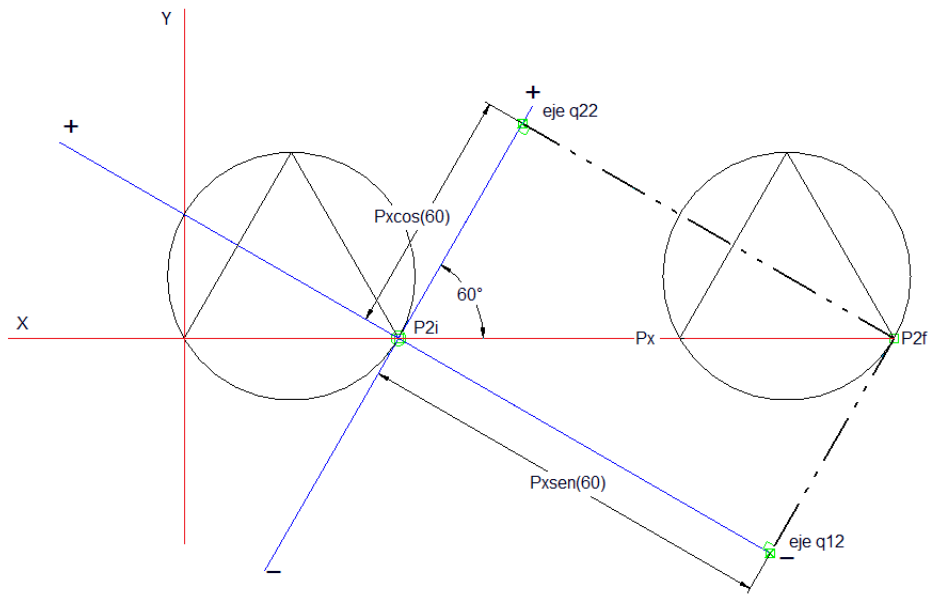


Figura 2-9: Proyecciones de P_x sobre los ejes q12 y q22.

Por último se analiza el brazo 3, en este se tiene que el eje q13 es paralelo al eje Y, por tal razón, un desplazamiento P_y sobre el eje Y ocasionará un desplazamiento P_y sobre el eje q13, de igual manera el eje q23 es paralelo al eje X, por tal razón un desplazamiento P_x sobre el eje X ocasionará un desplazamiento P_x sobre el eje q23.

Conociendo las proyecciones lineales de cada movimiento, P_x y P_y , sobre los ejes q11, q21, q12, q22, q13 y q23, se procede a calcular los valores de los ángulos de rotación de cada articulación respecto al desplazamiento de la plataforma.

Primero se moverá la plataforma una distancia P_y sobre el eje coordenado Y y se analizará la variación de los ángulos de rotación de las articulaciones que forman las juntas universales que unen cada brazo con la base del robot.

Para analizar el brazo 3, se hace un corte transversal sobre el plano q13-z, que es sobre el que variará el ángulo q13, ver Figura 2-10.

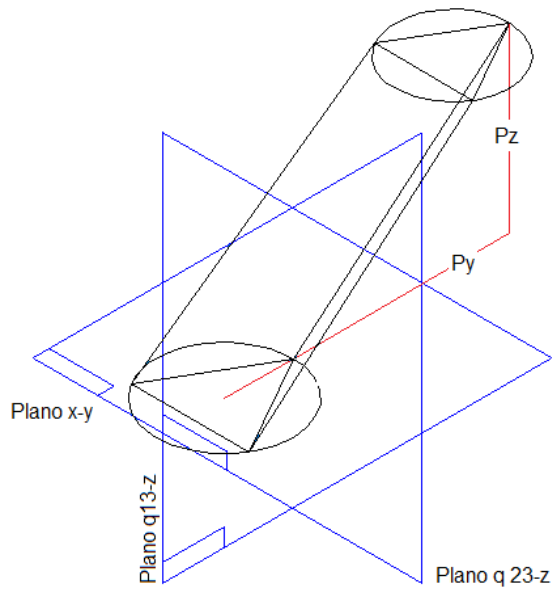


Figura 2-10: Desplazamiento de la plataforma sobre el eje y.

Como se observa en la Figura 2-11, el punto P3 se desplaza una distancia P_y sobre el eje Y, desde P_{3i} hasta P_{3f} , las posiciones en X y Z, no varían.

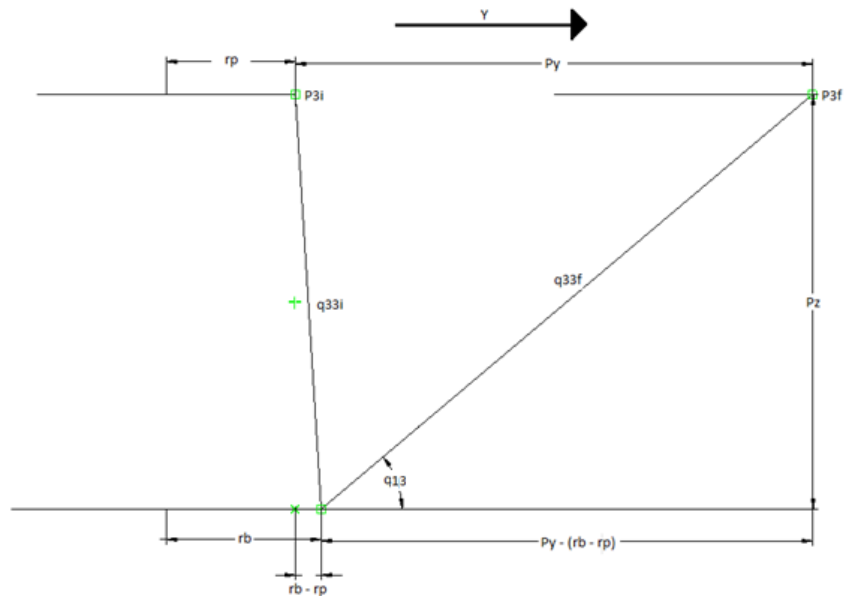


Figura 2-11: Corte transversal plano q13-z.

Utilizando las funciones trigonométricas básicas se tiene que:

$$q_{13} = \tan^{-1} \left(\frac{P_z}{P_y - (r_b - r_p)} \right)$$

Para el caso en que la plataforma se encuentra en su posición inicial, se tiene que:

$$q_{13} = \tan^{-1} \left(\frac{P_z}{-(r_b - r_p)} \right)$$

Siendo ésta la condición inicial para el ángulo q_{13} .

Ahora se analiza el brazo 1, los brazos 2 y 3 no se muestran en las figuras para que no interfieran en el análisis.

Para este caso hay que tener en cuenta que los ejes de rotación de las articulaciones q_{11} y q_{21} , están rotados 60° respecto a los ejes coordenados, ver figura 2-14.

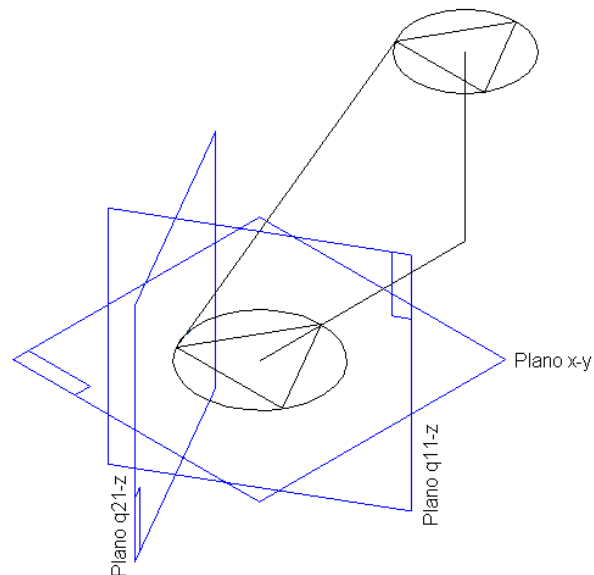


Figura 2-12: Análisis del brazo 1 con movimiento en Y.

Se hace un corte trasversal sobre el plano eje $q_{21} - z$, como se muestra en la Figura 2-13, en ella se aprecia que a pesar de que la plataforma se desplaza una

distancia P_y , el desplazamiento en el sentido del eje q_{21} está dado por $P_y \cdot \text{sen}(60)$, esto se debe a la rotación que tiene el eje de la articulación, que ya se mencionó.

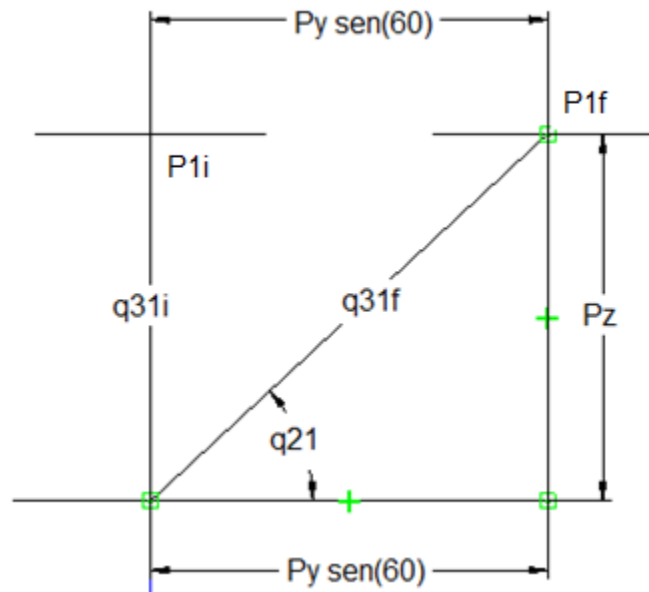


Figura 2-13: Corte transversal sobre el plano eje $q_{21} - z$.

Utilizando las funciones trigonométricas básicas se tiene que:

$$q_{21y} = \tan^{-1} \left(\frac{P_z}{P_y \cdot \text{sen}(60)} \right)$$

Para el caso en que la plataforma esté en su posición inicial, es decir, que los desplazamientos en X y Y son cero, se tiene que:

$$q_{21y} = \tan^{-1} \left(\frac{P_z}{0} \right) = 90^\circ$$

Esta es la condición inicial para el ángulo q_{21} .

Con el mismo desplazamiento de la plataforma en el eje y, se analiza la articulación q_{11} , el desplazamiento en el sentido del eje q_{11} está dado por $P_y \cdot \text{cos}(60)$, esto se debe a la rotación que tiene el eje de la articulación.

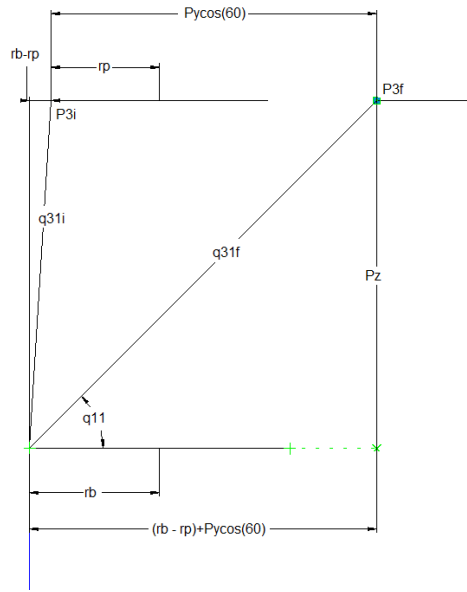


Figura 2-14: Corte transversal sobre el plano eje q21 – z.

$$q_{11y} = \tan^{-1} \left(\frac{P_z}{P_y \cos(60) + (r_b - r_p)} \right)$$

Para el caso en que la plataforma esté en su posición inicial, es decir, que los desplazamientos en X y Y son cero, se tiene que:

$$q_{11y} = \tan^{-1} \left(\frac{P_z}{(r_b - r_p)} \right)$$

Esta es la condición inicial para el ángulo q_{11} .

Finalmente se analiza el brazo 2, los brazos 1 y 3 no se muestran en las figuras para que no interfieran en el análisis.

Al igual que en el brazo 1, los ejes de rotación de las articulaciones q_{12} y q_{22} , están rotados 60° respecto a los ejes coordenados, ver figura 2-17.

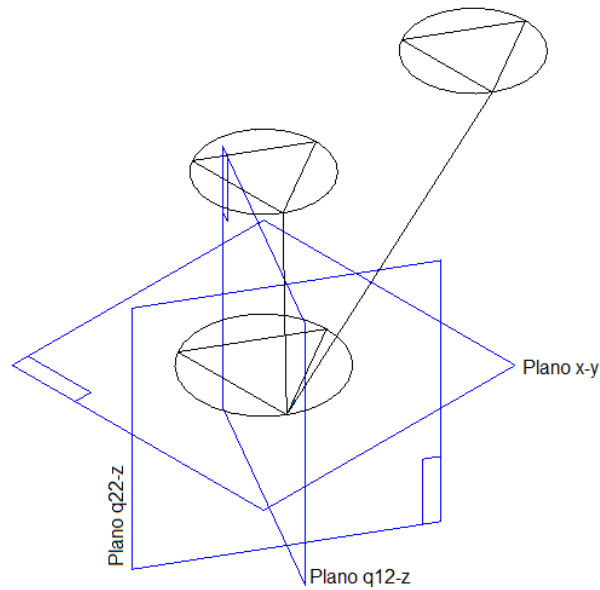


Figura 2-15: Análisis del brazo 2 con movimiento en y.

Se hace un corte transversal sobre el plano eje $q_{22} - z$, como se analizó anteriormente, Figura 2-13, a pesar de que la plataforma se desplaza una distancia P_y , el desplazamiento en el sentido del eje q_{22} está dado por $P_y \cdot \text{sen}(60)$, esto se debe a la rotación que tiene el eje de la articulación, que ya se mencionó.

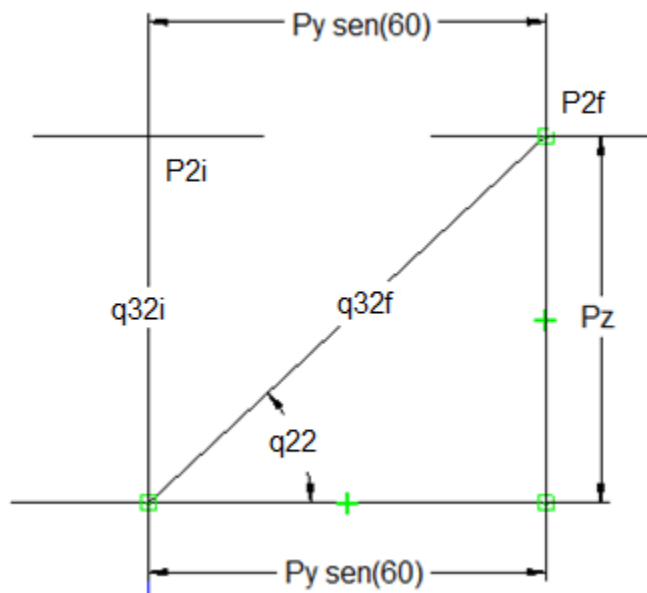


Figura 2-16: Corte transversal sobre el plano $q_{22} - z$.

Utilizando las funciones trigonométricas básicas se tiene que:

$$q_{22y} = \tan^{-1} \left(\frac{P_z}{P_y \sin(60)} \right)$$

Para el caso en que la plataforma esté en su posición inicial, es decir, que los desplazamientos en X y Y son cero, se tiene que:

$$q_{22y} = \tan^{-1} \left(\frac{P_z}{0} \right) = 90^\circ$$

Esta es la condición inicial para el ángulo q_{22} .

Continuando con el mismo procedimiento, se varía la posición de la plataforma sobre el eje x, ver Figura 2-17, se hace un corte transversal sobre el plano x-z, que es sobre el que cambia el ángulo q_{23} . En la figura 2-18 se observa que el punto P3 se desplaza una distancia P_x sobre el eje X, desde P3i hasta P3f, las posiciones en Y y Z, no varían.

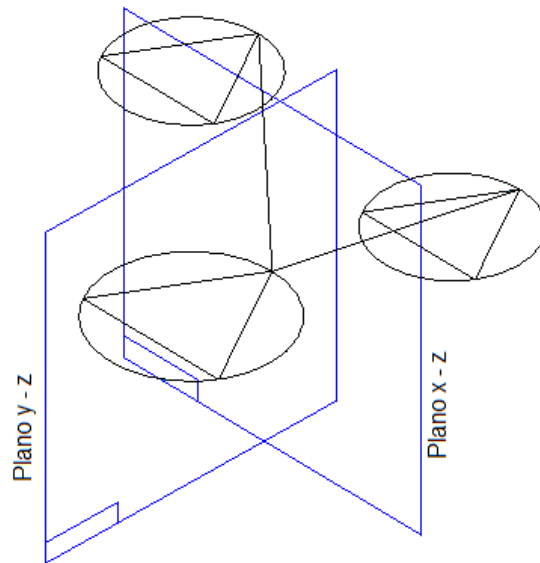


Figura 2-17: Desplazamiento de la plataforma sobre el eje x.

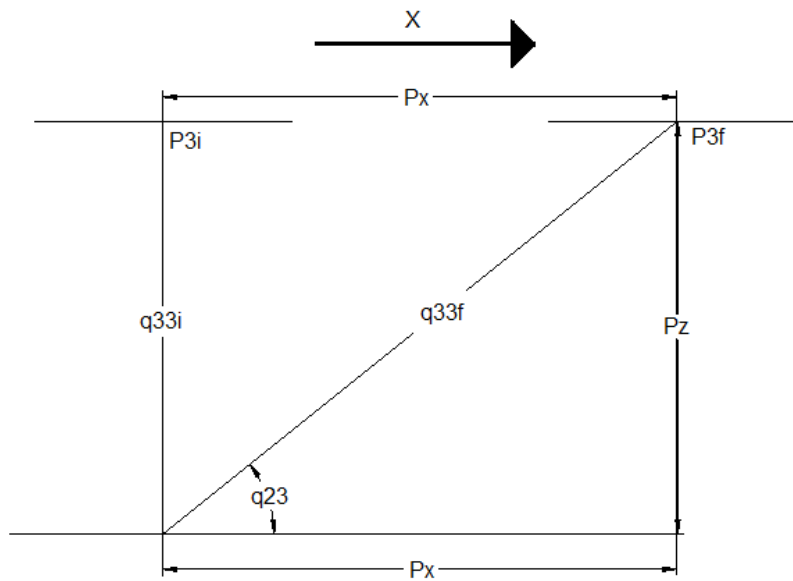


Figura 2-18: Desplazamiento del p3 sobre el eje X.

Utilizando las funciones trigonométricas básicas se tiene que:

$$q_{23} = \tan^{-1} \left(\frac{P_z}{P_x} \right)$$

Para el caso en que la plataforma esté en su posición inicial, se tiene que:

$$q_{23} = \tan^{-1} \left(\frac{P_z}{0} \right) = 90^\circ$$

Esta es la condición inicial para el ángulo q_{23} .

Analizando los brazos 1 y 2, se utiliza el mismo procedimiento que se empleó para calcular la variación de los ángulos con un desplazamiento sobre el eje Y, para calcular la variación de estos cuando el movimiento es sobre el eje X.

En las Figura 2-19 y 2-20, se muestran los ángulos q_{11} y q_{21} , respectivamente, mientras en las figuras 2-21 y 2-22 se representan los ángulos q_{12} y q_{22} , cuando la plataforma se desliza una distancia x sobre el eje X.

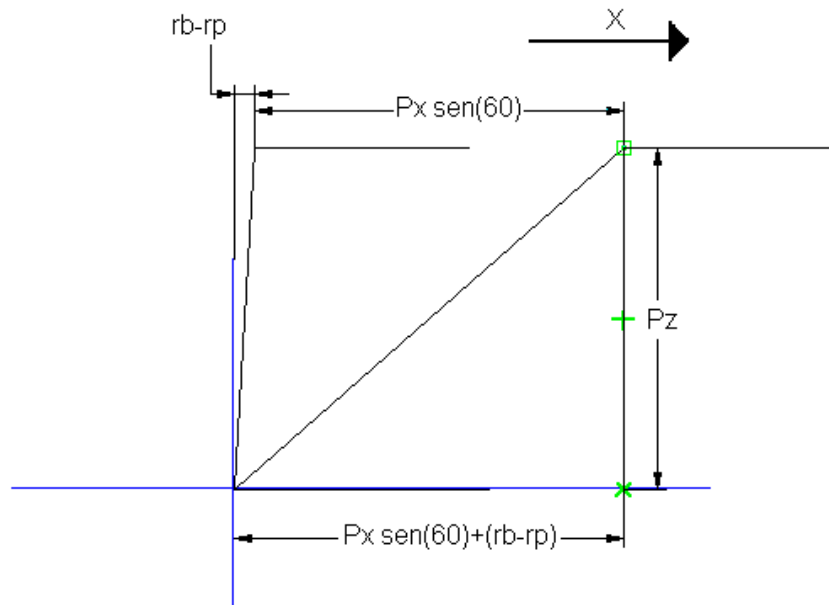


Figura 2-19: Análisis q11 con movimiento en X.

Utilizando las funciones trigonométricas básicas se tiene que:

$$q_{11x} = \tan^{-1} \left(\frac{P_z}{P_x \text{sen}(60) + (r_b - r_p)} \right)$$

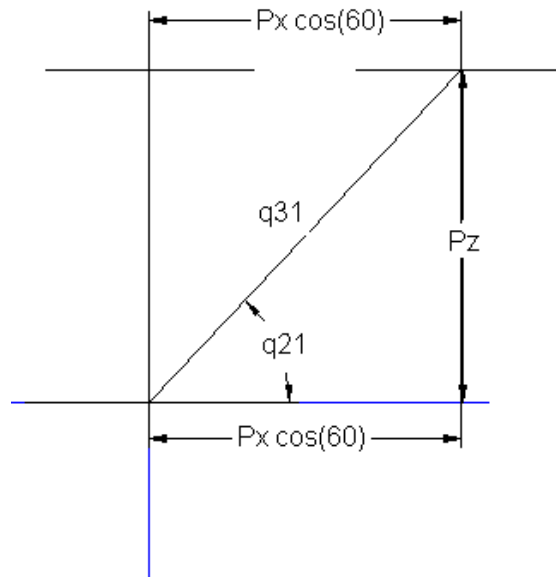


Figura 2-20: Análisis ángulo q21.

$$q_{21x} = \tan^{-1} \left(\frac{P_z}{P_x \cos(60)} \right)$$

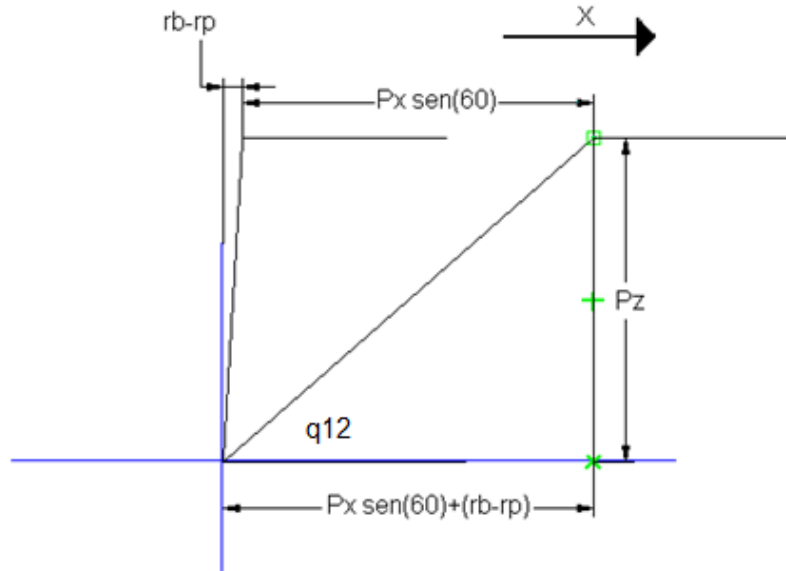


Figura 2-21: Análisis de q_{12} con movimiento en X.

Utilizando las funciones trigonométricas básicas se tiene que:

$$q_{12x} = \tan^{-1} \left(\frac{P_z}{P_x \cos(60) + (r_b - r_p)} \right)$$

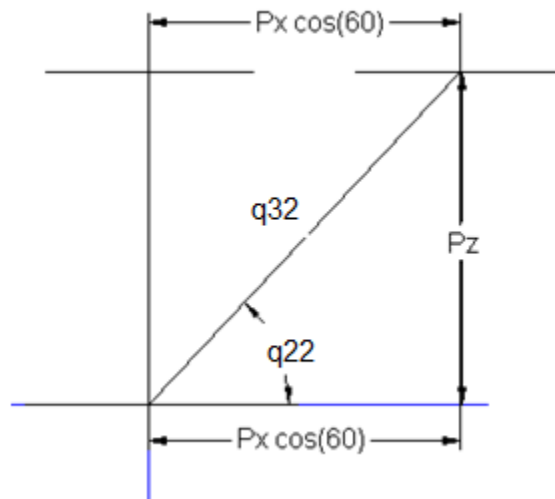


Figura 2-22: Análisis ángulo q_{21} .

$$q_{22x} = \tan^{-1}\left(\frac{P_z}{P_x \cos(60)}\right)$$

Como se puede observar, las articulaciones q_{21} , q_{11} , q_{22} y q_{12} se mueven simultáneamente cuando hay movimiento en X o Y, por tal razón las ecuaciones que describen la variación de estos dos ángulos estarán dadas por:

$$q_{11} = \tan^{-1}\left(\frac{P_z}{P_x \sin(60) + P_y \cos(60) + (r_b - r_p)}\right)$$

$$q_{21} = \tan^{-1}\left(\frac{P_z}{P_x \cos(60) - P_y \sin(60)}\right)$$

$$q_{12} = \tan^{-1}\left(\frac{P_z}{P_x \sin(60) + P_y \cos(60) + (r_b - r_p)}\right)$$

$$q_{22} = \tan^{-1}\left(\frac{P_z}{P_x \cos(60) - P_y \sin(60)}\right)$$

En la siguiente tabla se resumen las ecuaciones para el cálculo de las posiciones articulares que forman el robot.

Tabla 2-1: Resumen de las ecuaciones del MGI del robot PREH.

Articulación	Ecuación
q_{11}	$\tan^{-1}\left(\frac{P_z}{P_x \sin(60) + P_y \cos(60) + (r_b - r_p)}\right)$
q_{21}	$\tan^{-1}\left(\frac{P_z}{P_x \cos(60) - P_y \sin(60)}\right)$
q_{31}	$(P_{1X}^2 + P_{1Y}^2 + P_z^2)^{1/2}$
q_{12}	$\tan^{-1}\left(\frac{P_z}{P_x \sin(60) + P_y \cos(60) + (r_b - r_p)}\right)$
q_{22}	$\tan^{-1}\left(\frac{P_z}{P_x \cos(60) - P_y \sin(60)}\right)$

q_{32}	$\left(P_{1X}^2 + P_{1Y}^2 + P_Z^2 + 2\sqrt{3}P_{1X}(r_p - r_b) + 3(r_p - r_b)^2\right)^{1/2}$
q_{13}	$\tan^{-1}\left(\frac{P_Z}{P_Y - (r_b - r_p)}\right)$
q_{23}	$\tan^{-1}\left(\frac{P_Z}{P_X}\right)$
q_{33}	$\left(P_{1X}^2 + P_{1Y}^2 + P_Z^2 + \sqrt{3}P_{1X}(r_p - r_b) + 3P_{1Y}(r_p - r_b) + 3(r_p - r_b)^2\right)^{1/2}$

2.3.3. Modelo Geométrico Directo

El modelo geométrico directo es relativamente complejo. En general la solución no es única puesto que para algunas configuraciones de las variables articulares determinadas, la plataforma puede tener muchas situaciones diferentes. Este modelo ha sido objeto de muchos trabajos, uno de ellos, es utilizar el modelo cinemático inverso, para calcular iterativamente una solución local del MGD [67]. Otro método, es utilizar el MGI y a partir de unas relaciones matemáticas simples, encontrar el MGD [68]. Este método se utiliza con robots de 2 o 3 GDL, ya que solo se tienen 2 o 3 ecuaciones, con robots de mayor movilidad se incrementa la complejidad del sistema de ecuaciones.

Haciendo algunas operaciones matemáticas con las ecuaciones (2-9), (2-10), (2-11), se puede obtener el modelo geométrico directo.

El primer paso es calcular la diferencia entre (2-10) y (2-9):

$$q_{32}^2 - q_{31}^2 = 3(r_p - r_b) + \left(2 P_{1X}\sqrt{3}(r_p - r_b)\right)$$

$$P_{1X} = \frac{q_{32}^2 - q_{31}^2 - 3(r_p - r_b)^2}{2\sqrt{3}(r_p - r_b)} \quad (2-12)$$

Luego se encuentra la diferencia entre (2-10) y (2-11)

$$q_{32}^2 - q_{33}^2 = 2\sqrt{3}P_{1X}(r_p - r_b) - \sqrt{3}P_{1X}(r_p - r_b) - 3P_{1Y}(r_p - r_b)$$

$$P_{1Y} = \frac{q_{33}^2 - q_{32}^2 + \sqrt{3}P_{1X}(r_p - r_b)}{3(r_p - r_b)} \quad (2-13)$$

Finalmente se despeja P_Z de la ecuación (2-9), quedando de la siguiente manera:

$$P_Z^2 = q_{31}^2 - P_{1X}^2 - P_{1Y}^2$$

$$P_Z = \sqrt{q_{31}^2 - P_{1X}^2 - P_{1Y}^2} \quad (2-14)$$

Con las ecuaciones (2-12), (2-13) y (2-14), se encuentra la relación que hay entre el origen del sistema de coordenadas y la plataforma, pero para este caso es de vital importancia conocer la posición del extremo del endoscopio, por tal motivo se analiza el segmento central de la siguiente manera (ver Figura 2-23):

Se tiene que la longitud del brazo es L_T , que se puede expresar como:

$$L_T = L_1 + L_2 \quad (2-15)$$

Siendo:

L_2 : El segmento del laparoscopio que está dentro del paciente.

L_1 : El segmento del brazo que está antes del trocar.

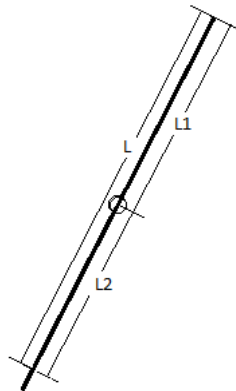


Figura 2-23: Análisis del segmento central.

Analizando a L_1 como un vector con origen en $O = (\sqrt{3}r_b/2, r_b/2, 0)$ y extremo en $P_0 = (P_{1x} + \sqrt{3}r_p/2, P_{1y} + r_p/2, P_z)$, se tiene que:

$$|L_1| = \sqrt{\left(P_{1x} + \sqrt{3}/2(r_p - r_b)\right)^2 + \left(P_{1y} + (r_p - r_b)/2\right)^2 + P_z^2} \quad (2-16)$$

Por concepto de cosenos directores se tiene que los ángulos que un vector forma con los ejes x, y, z son respectivamente α , β , γ (ver Figura 2-24) y se representan por:

$$\cos(\alpha) = \frac{P_x}{L_1}, \cos(\beta) = \frac{P_y}{L_1}, \cos(\gamma) = \frac{P_z}{L_1}$$

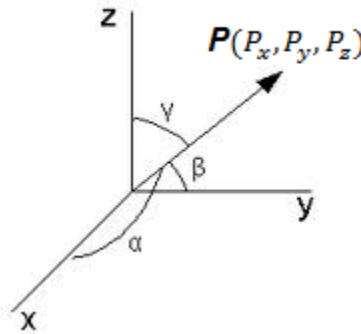


Figura 2-24: Cosenos directores.

Ahora por ángulos congruentes se tiene que:

$$\cos(\alpha) = -\cos(\alpha_1), \quad \cos(\beta) = -\cos(\beta_1), \quad \cos(\gamma) = -\cos(\gamma_1)$$

Y de (2-14):

$$L_2 = L - L_1 \quad (2-17)$$

Por último obtenemos las coordenadas del extremo del laparoscopio.

$$x = -\frac{L_2}{L_1}P_x, \quad y = -\frac{L_2}{L_1}P_y, \quad z = -\frac{L_2}{L_1}P_z$$

2.4. Modelo cinemático.

En mecanismos paralelos existen dos Jacobianos, el primero está asociado a los cambios de estado del efector final y el segundo asociado a los cambios de estado de los actuadores [65].

Es decir, se llama \mathbf{q} a la variable que define el estado de los actuadores, y \mathbf{x} describe la posición de la plataforma, entonces las restricciones cinemáticas impuestas por los brazos que componen a la plataforma analizada queda expresada mediante una función implícita de estas dos variables.

$$f(\mathbf{x}, \mathbf{q}) = 0 \tag{2-18}$$

Derivando (2-18) con respecto al tiempo y aplicando la regla de la cadena, se obtiene una relación entre las velocidades de estados de los actuadores y la velocidad del efector final.

$$\frac{\partial f}{\partial \mathbf{x}} \frac{\partial \mathbf{x}}{\partial t} + \frac{\partial f}{\partial \mathbf{q}} \frac{\partial \mathbf{q}}{\partial t} = 0 \tag{2-19}$$

Donde $\partial \mathbf{x} / \partial t$ representa las velocidades cartesianas del efector final y $\partial \mathbf{q} / \partial t$ las velocidades articulares del robot.

Si llamamos J_x a $\partial f / \partial \mathbf{x}$ y J_q a $\partial f / \partial \mathbf{q}$, la expresión (2-19) queda de la siguiente manera:

$$J_x \dot{\mathbf{x}} + J_q \dot{\mathbf{q}} = 0 \tag{2-20}$$

Esto nos da como resultado la obtención de dos jacobianos, uno correspondiente a la posición de la plataforma representado por $\partial f / \partial \mathbf{x}$, y otro correspondiente a las variables articulares expresado por $\partial f / \partial \mathbf{q}$.

Para obtener la matriz Jacobiana del mecanismo paralelo, se multiplica a la izquierda la ecuación (2-20) por J_x^{-1} (la inversa de J_x).

$$J_x^{-1} * J_x \dot{x} + J_x^{-1} J_q \dot{q} = 0 \quad (2-21)$$

$$\dot{x} = -J_x^{-1} * J_q \dot{q} \quad (2-22)$$

$$\dot{x} = J \dot{q} \quad (2-23)$$

$$\dot{q} = J^{-1} \dot{x} \quad (2-24)$$

Finalmente, la matriz Jacobiana del robot paralelo se expresa de la siguiente manera:

$$J = -J_x^{-1} * J_q \quad (2-25)$$

En conclusión, la matriz Jacobiana de un mecanismo, relaciona las velocidades articulares con las velocidades del efector final. Es por eso que el conocimiento de la misma es de vital importancia para generar las trayectorias dentro del espacio de trabajo.

El método para obtener J_x y J_q se describe en [68], para ello se reescriben las ecuaciones del MGI, ecuaciones (2-9), (2-10) y (2-11) de la forma $f(x, q) = 0$, luego se derivan con respecto a P_x, P_y, P_z para J_x y respecto a q_{31}, q_{32}, q_{33} para J_q . Como se puede observar, para el cálculo de la matriz Jacobiana J_q solo se tienen en cuenta las articulaciones activas.

$$P_{1X}^2 + P_{1Y}^2 + P_Z^2 - q_{31}^2 = 0$$

$$P_{1X}^2 + P_{1Y}^2 + P_Z^2 + 2\sqrt{3}P_{1X}(r_p - r_b) + 3(r_p - r_b)^2 - q_{32}^2 = 0$$

$$P_{1X}^2 + P_{1Y}^2 + P_Z^2 + \sqrt{3}P_{1X}(r_p - r_b) + 3P_{1Y}(r_p - r_b) + 3(r_p - r_b)^2 - q_{33}^2 = 0$$

Por lo tanto:

$$J_x = \begin{bmatrix} 2P_{1X} & 2P_{1Y} & 2P_Z \\ 2P_{1X} + 2\sqrt{3}(r_p - r_b) & 2P_{1Y} & 2P_Z \\ 2P_{1X} + \sqrt{3}(r_p - r_b) & 2P_{1Y} + 3(r_p - r_b) & 2P_Z \end{bmatrix}$$

$$J_q = -2 \begin{bmatrix} q_{31} & 0 & 0 \\ 0 & q_{32} & 0 \\ 0 & 0 & q_{33} \end{bmatrix}$$

Ahora se calcula la inversa de J_x , para finalmente calcular la matriz Jacobiana del robot *PREH* mediante la ecuación (2-25).

$$J_x^{-1} = \begin{bmatrix} \frac{\sqrt{3}}{6(r_b - r_p)} & \frac{-\sqrt{3}}{6(r_b - r_p)} & 0 \\ \frac{1}{6(r_b - r_p)} & \frac{1}{6(r_b - r_p)} & \frac{-1}{3(r_b - r_p)} \\ \frac{3(r_b - r_p) - \sqrt{3}P_{1X} - P_{1Y}}{6P_Z(r_b - r_p)} & \frac{\sqrt{3}P_{1X} - P_{1Y}}{6P_Z(r_b - r_p)} & \frac{P_{1Y}}{3P_Z(r_b - r_p)} \end{bmatrix}$$

$$J = \begin{bmatrix} \frac{\sqrt{3}q_{31}}{3(r_b - r_p)} & \frac{-\sqrt{3}q_{32}}{3(r_b - r_p)} & 0 \\ \frac{q_{31}}{3(r_b - r_p)} & \frac{q_{32}}{3(r_b - r_p)} & \frac{-2q_{33}}{3(r_b - r_p)} \\ \frac{q_{31}(3(r_b - r_p) - \sqrt{3}P_{1X} - P_{1Y})}{3P_Z(r_b - r_p)} & \frac{q_{32}(\sqrt{3}P_{1X} - P_{1Y})}{3P_Z(r_b - r_p)} & \frac{2P_{1Y}q_{33}}{3P_Z(r_b - r_p)} \end{bmatrix}$$

$$J^{-1} = \begin{bmatrix} \frac{P_{1X}}{q_{31}} & \frac{P_{1Y}}{q_{31}} & \frac{P_Z}{q_{31}} \\ \frac{P_{1X} + \sqrt{3}(r_p - r_b)}{q_{32}} & \frac{P_{1Y}}{q_{32}} & \frac{P_Z}{q_{32}} \\ \frac{2P_{1X} + \sqrt{3}(r_p - r_b)}{2q_{33}} & \frac{2P_{1Y} + 3(r_p - r_b)}{2q_{32}} & \frac{P_Z}{q_{33}} \end{bmatrix}$$

2.5. Modelo dinámico

Al igual que en los modelos anteriores, también existe dos modelos dinámicos, el modelo dinámico inverso (MDI) expresa la relación entre las fuerzas de las juntas motorizadas y las posiciones, velocidades y aceleraciones de las juntas del robot.

$$\Gamma = f(q, \dot{q}, \ddot{q})$$

El modelo dinámico directo (MDD) expresa las aceleraciones articulares de las juntas del robot en función de sus posiciones y velocidades y las fuerzas de las articulaciones motorizadas (activas).

$$\ddot{q} = f(q, \dot{q}, \Gamma)$$

Los modelos dinámicos son utilizados para la simulación de cualquier tipo de robot, calcular las fuerzas que requieren los motores para mover los cuerpos del robot, identificar los parámetros inerciales del robot y el control del robot.

2.5.1. Modelo Dinámico Inverso

Para el cálculo del modelo dinámico inverso del robot, se utiliza el procedimiento desarrollado en [69], para lo cual se analiza la base y los tres brazos como una estructura arborescente, aislando la plataforma, para finalmente y con la ayuda de la dinámica de la plataforma calcular el modelo dinámico del conjunto en general, en otras palabras, lo que se pretende es analizar al robot como un cuerpo con las mismas características físicas de la plataforma, sobre el que actúan fuerzas externas, las cuales se muestran en la Figura 2-25.

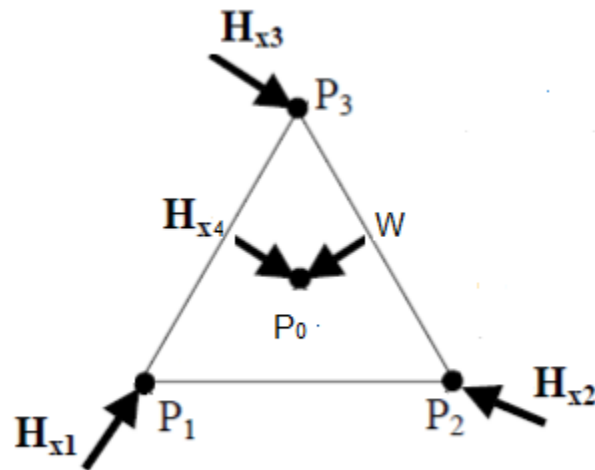


Figura 2-25: Fuerzas externas que actúan sobre la plataforma.

Las fuerzas H_{x1} , H_{x2} y H_{x3} , son aplicadas por los brazos 1, 2 y 3 sobre los puntos P_1 , P_2 y P_3 , respectivamente.

La fuerza H_{x4} , que se debe al brazo central (endoscopio y soporte) y el peso, W , se aplican sobre el centro de masa de la plataforma, punto P_0 .

Para encontrar todas las fuerzas que actúan sobre la plataforma se aplica el siguiente procedimiento, que consta de seis pasos:

1. Calcular el MG y el MC de los tres brazos que unen la base a la plataforma.
2. Calcular el modelo dinámico de cada brazo.
3. Calcular el modelo dinámico cartesiano de cada cadena cinemática.
4. Calcular las fuerzas y momentos correspondientes a cada modelo dinámico cartesiano.
5. Calcular las fuerzas y/o momentos necesarios para mover la plataforma.
6. Calcular las fuerzas y/o momentos deseados.

2.5.1.1. Modelo Geométrico de los 3 brazos que unen la plataforma a la base.

Cada brazo se puede analizar como una estructura tipo serie (RRP), ver Figura 2-26:, sin tener en cuenta la junta universal que une los brazos con la plataforma.

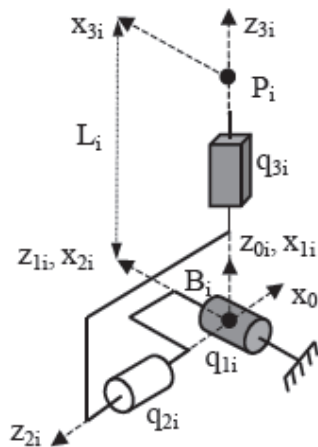


Figura 2-26: Estructura serie brazo i.

Como primera medida se deben obtener los parámetros geométricos de todas las articulaciones que forman cada brazo; basándose en el método expuesto en [67], estos se muestran en las tablas 2-1, 2-2 y 2-3.

Tabla 2-2: Parámetros geométricos del brazo 1.

J_1	a_{i1}	u_{i1}	σ_{i1}	γ_{i1}	b_{i1}	α_{i1}	d_{i1}	θ_{i1}	r_{i1}
1 ₁	0	0	0	0	0	$-\pi/2$	0	q_{11}	0
2 ₁	1 ₁	0	0	0	0	$\pi/2$	0	q_{21}	0
3 ₁	2 ₁	1	1	0	0	$\pi/2$	0	0	q_{31}

Tabla 2-3: Parámetros geométricos del brazo 2.

J_2	a_{j2}	u_{j2}	σ_{j2}	γ_{j2}	b_{j2}	α_{j2}	d_{j2}	θ_{j2}	r_{j2}
1 ₂	0	0	0	0	0	$-\pi/2$	d_{12}	q_{12}	0
2 ₂	1 ₂	0	0	0	0	$\pi/2$	0	q_{22}	0
3 ₂	2 ₂	1	1	0	0	$\pi/2$	0	0	q_{32}

Tabla 2-4: Parámetros geométricos del brazo 3.

J_3	a_{j3}	u_{j3}	σ_{j3}	γ_{j3}	b_{j3}	α_{j3}	d_{j3}	θ_{j3}	r_{j3}
1 ₃	0	0	0	$\pi/3$	0	$-\pi/2$	d_{13}	q_{13}	0
2 ₃	1 ₃	0	0	0	0	$\pi/2$	0	q_{23}	0
3 ₃	2 ₃	1	1	0	0	$\pi/2$	0	0	q_{33}

Con estos parámetros se procede a calcular las matrices de orientación, posición y transformación de acuerdo a la ecuación (2-26), (2-27) y (2-28).

La matriz de orientación se define como:

$${}^i A_j = \begin{bmatrix} C\gamma_i C\theta_i - S\gamma_i C\alpha_i S\theta_i & -C\gamma_i S\theta_i - S\gamma_i C\alpha_i C\theta_i & S\gamma_i S\alpha_i \\ S\gamma_i C\theta_i + C\gamma_i C\alpha_i S\theta_i & -S\gamma_i S\theta_i + C\gamma_i C\alpha_i C\theta_i & -C\gamma_i S\alpha_i \\ S\alpha_i S\theta_i & S\alpha_i C\theta_i & C\alpha_i \end{bmatrix} \quad (2-26)$$

Matrices de orientación del brazo 1:

$${}^0 A_{11} = \begin{bmatrix} Cq_{11} & -Sq_{11} & 0 \\ 0 & 0 & 1 \\ -Sq_{11} & -Cq_{11} & 0 \end{bmatrix} \quad {}^1 A_{21} = \begin{bmatrix} Cq_{21} & -Sq_{21} & 0 \\ 0 & 0 & -1 \\ Sq_{21} & Cq_{21} & 0 \end{bmatrix} \quad {}^2 A_{31} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 0 & -1 \\ 0 & 1 & 0 \end{bmatrix}$$

Matrices de orientación del brazo 2:

$${}^0 A_{12} = \begin{bmatrix} Cq_{12} & -Sq_{12} & 0 \\ 0 & 0 & 1 \\ -Sq_{12} & -Cq_{12} & 0 \end{bmatrix} \quad {}^1 A_{22} = \begin{bmatrix} Cq_{22} & -Sq_{22} & 0 \\ 0 & 0 & -1 \\ Sq_{22} & Cq_{22} & 0 \end{bmatrix} \quad {}^2 A_{32} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 0 & -1 \\ 0 & 1 & 0 \end{bmatrix}$$

Matrices de orientación del brazo 3:

$${}^0 A_{13} = \begin{bmatrix} Cq_{13}/2 & -Sq_{13}/2 & -\sqrt{3}/2 \\ \sqrt{3}Cq_{13}/2 & -\sqrt{3}Sq_{13}/2 & 1/2 \\ -Sq_{13} & -Cq_{13} & 0 \end{bmatrix}$$

$${}^1 A_{23} = \begin{bmatrix} Cq_{23} & -Sq_{23} & 0 \\ 0 & 0 & -1 \\ Sq_{23} & Cq_{23} & 0 \end{bmatrix} \quad {}^2 A_{33} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 0 & -1 \\ 0 & 1 & 0 \end{bmatrix}$$

El vector de posición se define como:

$${}^i P_j = \begin{bmatrix} d_i C\gamma_i + r_i S\gamma_i S\alpha_i \\ d_i S\gamma_i - r_i C\gamma_i S\alpha_i \\ r_i C\alpha_i + B_i \end{bmatrix} \quad (2-27)$$

Vectores de posición del brazo 1:

$${}^0P_{11} = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix} \quad {}^1P_{21} = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix} \quad {}^2P_{31} = \begin{bmatrix} 0 \\ -q_{31} \\ 0 \end{bmatrix}$$

Vectores de posición del brazo 2:

$${}^0P_{12} = \begin{bmatrix} d_{12} \\ 0 \\ 0 \end{bmatrix} \quad {}^1P_{22} = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix} \quad {}^2P_{32} = \begin{bmatrix} 0 \\ -q_{32} \\ 0 \end{bmatrix}$$

Vectores de posición del brazo 3:

$${}^0P_{13} = \begin{bmatrix} d_{13}/2 \\ \sqrt{3}d_{13}/2 \\ 0 \end{bmatrix} \quad {}^1P_{23} = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix} \quad {}^2P_{33} = \begin{bmatrix} 0 \\ -q_{33} \\ 0 \end{bmatrix}$$

La matriz de transformación se define por:

$${}^iT_j = \begin{bmatrix} {}^iA_j & {}^iP_j \\ 0 & 1 \end{bmatrix} \quad (2-28)$$

Matrices de transformación del brazo 1:

$${}^0T_{11} = \begin{bmatrix} Cq_{11} & -Sq_{11} & 0 & 0 \\ 0 & 0 & 1 & 0 \\ -Sq_{11} & -Cq_{11} & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad {}^1T_{21} = \begin{bmatrix} Cq_{21} & -Sq_{21} & 0 & 0 \\ 0 & 0 & -1 & 0 \\ Sq_{21} & Cq_{21} & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$${}^2T_{31} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & -1 & -q_{31} \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$${}^0T_{31} = {}^0T_{11} {}^1T_{21} {}^2T_{31}$$

$${}^0T_{31} = \begin{bmatrix} Cq_{11}Cq_{21} & Sq_{11} & Cq_{11}Sq_{21} & q_{31}Cq_{11}Sq_{21} \\ Sq_{21} & 0 & -Cq_{21} & -q_{31}Cq_{21} \\ -Cq_{21}Sq_{11} & Cq_{11} & -Sq_{11}Sq_{21} & -q_{31}Sq_{11}Sq_{21} \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Matrices de transformación del brazo 2:

$${}^0T_{12} = \begin{bmatrix} Cq_{12} & -Sq_{12} & 0 & d_{12} \\ 0 & 0 & 1 & 0 \\ -Sq_{12} & -Cq_{12} & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad {}^1T_{22} = \begin{bmatrix} Cq_{22} & -Sq_{22} & 0 & 0 \\ 0 & 0 & -1 & 0 \\ Sq_{22} & Cq_{22} & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$${}^2T_{32} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & -1 & -q_{32} \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$${}^0T_{32} = {}^0T_{12} {}^1T_{22} {}^2T_{32}$$

$${}^0T_{32} = \begin{bmatrix} Cq_{12}Cq_{22} & Sq_{12} & Cq_{12}Sq_{22} & d_{12} + q_{32}Cq_{12}Sq_{22} \\ Sq_{22} & 0 & -Cq_{22} & -q_{32}Cq_{22} \\ -Cq_{22}Sq_{12} & Cq_{12} & -Sq_{12}Sq_{22} & -q_{32}Sq_{12}Sq_{22} \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Matrices de transformación del brazo 3:

$${}^0T_{13} = \begin{bmatrix} Cq_{13}/2 & -Sq_{13}/2 & -\sqrt{3}/2 & d_{13}/2 \\ \sqrt{3}Cq_{13}/2 & -\sqrt{3}Sq_{13}/2 & 1/2 & \sqrt{3}d_{13}/2 \\ -Sq_{13} & -Cq_{13} & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$${}^1T_{23} = \begin{bmatrix} Cq_{23} & -Sq_{23} & 0 & 0 \\ 0 & 0 & -1 & 0 \\ Sq_{23} & Cq_{23} & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$${}^2T_{33} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & -1 & -q_{33} \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$${}^0T_{33} = {}^0T_{11} {}^1T_{21} {}^2T_{31}$$

$${}^0T_{33} = \frac{1}{2} \begin{bmatrix} Cq_{13}Cq_{23} - \sqrt{3}Sq_{23} & Sq_{13} & \sqrt{3}Cq_{23} + Cq_{13}Sq_{23} & d_{13} + \sqrt{3}q_{33}Cq_{23} + q_{33}Cq_{13}Sq_{23} \\ Sq_{23} + \sqrt{3}Cq_{13}Cq_{23} & \sqrt{3}Sq_{13} & \sqrt{3}Cq_{13}Sq_{23} - Cq_{23} & \sqrt{3}d_{13} - q_{33}Cq_{23} + \sqrt{3}q_{33}Cq_{13}Sq_{23} \\ -2Cq_{23}Sq_{13} & 2Cq_{13} & -2Sq_{13}Sq_{23} & -2q_{33}Sq_{13}Sq_{23} \\ 0 & 0 & 0 & 2 \end{bmatrix}$$

2.5.1.2. Modelo cinemático de los 3 brazos que unen la plataforma a la base.

El modelo cinemático directo del brazo i , representa las velocidades cartesianas del punto P_i en función de las velocidades articulares de la cadena cinemática i (q_{1i}, q_{2i}, q_{3i}) y se expresa mediante la siguiente expresión:

$$\dot{x}_i = {}^0J_{3i}\dot{q}_i \quad (2-29)$$

Donde ${}^0J_{3i}$ es la matriz Jacobiana de la cadena cinemática i .

Para simplificar el cálculo de la matriz Jacobiana del brazo i , se calcula la matriz Jacobiana de base del punto terminal de la cadena cinemática i [67], la cual define la k -ésima columna de ${}^3J_{3i}$ como se muestra en la ecuación (2-30), para finalmente calcular la matriz Jacobiana de la cadena cinemática i mediante la ecuación (2-31).

Los elementos de la columna ${}^3J_{3i:k}$ se calculan a partir de los elementos de la matriz de transformación ${}^3T_{ni}$ obtenida dentro del cálculo del modelo geométrico directo. En la ecuación (2-30) pueden distinguirse los vectores s , n , a pertenecientes a la matriz de orientación ${}^3A_{ni}$, así como los valores del vector de posición en x e y (${}^3p_{nxi}$ y ${}^3p_{nyi}$ respectivamente) [64].

$${}^3J_{3i:k} = [\sigma_k {}^3\alpha_k + \bar{\sigma}_k (-{}^k p_{3y} {}^3s_k + {}^k p_{3x} {}^3n_k)] \quad (2-30)$$

$${}^0J_{3i} = {}^0A_{3i} {}^3J_{3i} \quad (2-31)$$

Posteriormente, se calcula la matriz Jacobiana inversa del brazo i , que se describe por:

$${}^0J_{3i}^{-1} = {}^{3i}J_{3i}^{-1} {}^{3i}A_0 \quad (2-32)$$

Finalmente, se calcula la matriz traspuesta de la Jacobiana inversa del brazo i , que se define mediante la siguiente expresión.

$${}^0J_{3i}^{-T} = {}^0A_{3i} {}^{3i}J_{3i}^{-T} \quad (2-33)$$

Matrices Jacobianas del brazo 1:

$${}^{31}J_{31:1} = \begin{bmatrix} 0 \\ -q_{31}S_{q_{21}} \\ 0 \end{bmatrix} \quad {}^{31}J_{31:2} = \begin{bmatrix} q_{31} \\ 0 \\ 0 \end{bmatrix} \quad {}^{31}J_{31:3} = \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}$$

$${}^{31}J_{31} = \begin{bmatrix} 0 & q_{31} & 0 \\ -q_{31}S_{q_{21}} & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

$${}^{31}J_{31}^{-1} = \begin{bmatrix} 0 & \frac{1}{q_{31}S_{q_{21}}} & 0 \\ 1 & 0 & 0 \\ \frac{q_{31}}{0} & 0 & 1 \end{bmatrix}$$

$${}^0J_{31} = \begin{bmatrix} C_{q_{11}}C_{q_{21}} & S_{q_{11}} & C_{q_{11}}S_{q_{21}} \\ S_{q_{21}} & 0 & -C_{q_{21}} \\ -C_{q_{21}}S_{q_{11}} & C_{q_{11}} & -S_{q_{11}}S_{q_{21}} \end{bmatrix} \begin{bmatrix} 0 & q_{31} & 0 \\ -q_{31}S_{q_{21}} & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

$${}^0J_{31} = \begin{bmatrix} -q_{31}S_{q_{11}}S_{q_{21}} & -q_{31}C_{q_{11}}C_{q_{21}} & C_{q_{11}}S_{q_{21}} \\ 0 & q_{31}S_{q_{21}} & -C_{q_{21}} \\ -q_{31}C_{q_{11}}S_{q_{21}} & -q_{31}S_{q_{11}}C_{q_{21}} & -S_{q_{11}}S_{q_{21}} \end{bmatrix}$$

$${}^0J_{31}^{-1} = \begin{bmatrix} 0 & \frac{1}{q_{31}S_{q_{21}}} & 0 \\ 1 & 0 & 0 \\ \frac{q_{31}}{0} & 0 & 1 \end{bmatrix} \begin{bmatrix} C_{q_{11}}C_{q_{21}} & S_{q_{21}} & -C_{q_{21}}S_{q_{11}} \\ S_{q_{11}} & 0 & C_{q_{11}} \\ C_{q_{11}}S_{q_{21}} & -C_{q_{21}} & -S_{q_{11}}S_{q_{21}} \end{bmatrix}$$

$${}^0J_{31}^{-1} = \begin{bmatrix} \frac{Sq_{11}}{q_{31}Sq_{21}} & 0 & \frac{Cq_{11}}{q_{31}Sq_{21}} \\ \frac{Cq_{11}Cq_{21}}{q_{31}} & \frac{Sq_{21}}{q_{31}} & \frac{-Cq_{21}Sq_{11}}{q_{31}} \\ Cq_{11}Sq_{21} & -Cq_{21} & -Sq_{11}Sq_{21} \end{bmatrix}$$

$${}^0J_{31}^{-T} = \begin{bmatrix} Cq_{11}Cq_{21} & Sq_{11} & Cq_{11}Sq_{21} \\ Sq_{21} & 0 & -Cq_{21} \\ -Cq_{21}Sq_{11} & Cq_{11} & -Sq_{11}Sq_{21} \end{bmatrix} \begin{bmatrix} 0 & \frac{1}{q_{31}} & 0 \\ \frac{1}{q_{31}Sq_{21}} & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

$${}^0J_{31}^{-T} = \begin{bmatrix} \frac{Sq_{11}}{q_{31}Sq_{21}} & \frac{Cq_{11}Cq_{21}}{q_{31}} & Cq_{11}Sq_{21} \\ 0 & \frac{Sq_{21}}{q_{31}} & -Cq_{21} \\ \frac{Cq_{11}}{q_{31}Sq_{21}} & \frac{-Cq_{21}Sq_{11}}{q_{31}} & -Sq_{11}Sq_{21} \end{bmatrix}$$

Matrices Jacobianas del brazo 2:

$${}^{32}J_{32:1} = \begin{bmatrix} 0 \\ -q_{32}Sq_{21} \\ 0 \end{bmatrix} \quad {}^{32}J_{32:2} = \begin{bmatrix} q_{32} \\ 0 \\ 0 \end{bmatrix} \quad {}^{32}J_{32:3} = \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}$$

$${}^{32}J_{32} = \begin{bmatrix} 0 & q_{32} & 0 \\ -q_{32}Sq_{22} & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

$${}^{32}J_{32}^{-1} = \begin{bmatrix} 0 & \frac{1}{q_{32}Sq_{22}} & 0 \\ \frac{1}{q_{32}} & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

$${}^0J_{32} = \begin{bmatrix} Cq_{12}Cq_{22} & Sq_{12} & Cq_{12}Sq_{22} \\ Sq_{22} & 0 & -Cq_{22} \\ -Cq_{22}Sq_{12} & cq_{12} & -Sq_{12}Sq_{22} \end{bmatrix} \begin{bmatrix} 0 & q_{32} & 0 \\ -q_{32}Sq_{22} & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

$${}^0J_{32} = \begin{bmatrix} -q_{32}S_{q_{12}}S_{q_{22}} & -q_{32}C_{q_{12}}C_{q_{22}} & C_{q_{12}}S_{q_{22}} \\ 0 & q_{32}S_{q_{22}} & -C_{q_{22}} \\ -q_{32}C_{q_{12}}S_{q_{22}} & -q_{32}S_{q_{12}}C_{q_{22}} & -S_{q_{12}}S_{q_{22}} \end{bmatrix}$$

$${}^0J_{32}^{-1} = \begin{bmatrix} 0 & \frac{1}{q_{32}S_{q_{22}}} & 0 \\ \frac{1}{q_{32}} & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} C_{q_{12}}C_{q_{22}} & S_{q_{22}} & -C_{q_{22}}S_{q_{12}} \\ S_{q_{12}} & 0 & C_{q_{12}} \\ C_{q_{12}}S_{q_{22}} & -C_{q_{22}} & -S_{q_{12}}S_{q_{22}} \end{bmatrix}$$

$${}^0J_{32}^{-1} = \begin{bmatrix} \frac{S_{q_{12}}}{q_{32}S_{q_{22}}} & 0 & \frac{C_{q_{12}}}{q_{32}S_{q_{22}}} \\ \frac{C_{q_{12}}C_{q_{22}}}{q_{32}} & \frac{S_{q_{22}}}{q_{32}} & \frac{-C_{q_{22}}S_{q_{12}}}{q_{32}} \\ C_{q_{12}}S_{q_{22}} & -C_{q_{22}} & -S_{q_{12}}S_{q_{22}} \end{bmatrix}$$

$${}^0J_{32}^{-T} = \begin{bmatrix} C_{q_{12}}C_{q_{22}} & S_{q_{12}} & C_{q_{12}}S_{q_{22}} \\ S_{q_{22}} & 0 & -C_{q_{22}} \\ -C_{q_{22}}S_{q_{12}} & C_{q_{12}} & -S_{q_{12}}S_{q_{22}} \end{bmatrix} \begin{bmatrix} 0 & \frac{1}{q_{32}} & 0 \\ \frac{1}{q_{32}S_{q_{22}}} & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

$${}^0J_{32}^{-T} = \begin{bmatrix} \frac{S_{q_{12}}}{q_{32}S_{q_{22}}} & \frac{C_{q_{12}}C_{q_{22}}}{q_{32}} & C_{q_{12}}S_{q_{22}} \\ 0 & \frac{S_{q_{22}}}{q_{32}} & -C_{q_{22}} \\ \frac{C_{q_{12}}}{q_{32}S_{q_{22}}} & \frac{-C_{q_{22}}S_{q_{12}}}{q_{32}} & -S_{q_{12}}S_{q_{22}} \end{bmatrix}$$

Matrices Jacobianas del brazo 3:

$${}^{33}J_{33:1} = \begin{bmatrix} 0 \\ -q_{33}S_{q_{23}} \\ 0 \end{bmatrix} \quad {}^{33}J_{33:2} = \begin{bmatrix} q_{33} \\ 0 \\ 0 \end{bmatrix} \quad {}^{33}J_{33:3} = \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}$$

$${}^{33}J_{33} = \begin{bmatrix} 0 & q_{33} & 0 \\ -q_{33}S_{q_{23}} & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

$${}^{33}J_{33}^{-1} = \begin{bmatrix} 0 & \frac{1}{q_{33}Sq_{23}} & 0 \\ \frac{1}{q_{33}} & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

$${}^0J_{33} = \frac{1}{2} \begin{bmatrix} Cq_{13}Cq_{23} - \sqrt{3}Sq_{23} & Sq_{13} & \sqrt{3}Cq_{23} + Cq_{13}Sq_{23} \\ Sq_{23} + \sqrt{3}Cq_{13}Cq_{23} & \sqrt{3}Sq_{13} & \sqrt{3}Cq_{13}Sq_{23} - Cq_{23} \\ -2Cq_{23}Sq_{13} & 2Cq_{13} & -2Sq_{13}Sq_{23} \end{bmatrix} \begin{bmatrix} 0 & q_{33} & 0 \\ -q_{33}Sq_{23} & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

$${}^0J_{33} = \frac{1}{2} \begin{bmatrix} -q_{33}Sq_{13}Sq_{23} & -\sqrt{3}q_{33}Sq_{23} - Cq_{13}Cq_{23} & -\sqrt{3}Cq_{23} + Cq_{13}Cq_{23} \\ -\sqrt{3}q_{33}Sq_{13}Sq_{23} & q_{33}Sq_{23} + \sqrt{3}Cq_{13}Cq_{23} & \sqrt{3}Cq_{13}Sq_{23} - Cq_{23} \\ -2q_{33}Cq_{13}Sq_{23} & -2q_{33}Sq_{13}Cq_{23} & -2Sq_{13}Sq_{23} \end{bmatrix}$$

$${}^0J_{33}^{-1} = \frac{1}{2} \begin{bmatrix} 0 & \frac{1}{q_{33}Sq_{23}} & 0 \\ \frac{1}{q_{33}} & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} Cq_{13}Cq_{23} - \sqrt{3}Sq_{23} & Sq_{23} + \sqrt{3}Cq_{13}Cq_{23} & -2Cq_{23}Sq_{13} \\ Sq_{13} & \sqrt{3}Sq_{13} & 2Cq_{13} \\ \sqrt{3}Cq_{23} + Cq_{13}Sq_{23} & \sqrt{3}Cq_{13}Sq_{23} - Cq_{23} & -2Sq_{13}Sq_{23} \end{bmatrix}$$

$${}^0J_{33}^{-1} = \frac{1}{2} \begin{bmatrix} \frac{Sq_{13}}{q_{33}Sq_{23}} & \frac{\sqrt{3}Sq_{13}}{q_{33}Sq_{23}} & \frac{2Cq_{13}}{q_{33}Sq_{23}} \\ \frac{Cq_{13}Cq_{23} - \sqrt{3}Sq_{23}}{q_{33}} & \frac{Sq_{23} + \sqrt{3}Cq_{13}Cq_{23}}{q_{33}} & \frac{-2Cq_{23}Sq_{13}}{q_{33}} \\ \sqrt{3}Cq_{23} + Cq_{13}Sq_{23} & \sqrt{3}Cq_{13}Sq_{23} - Cq_{23} & -2Sq_{13}Sq_{23} \end{bmatrix}$$

$${}^0J_{33}^{-T} = \frac{1}{2} \begin{bmatrix} Cq_{13}Cq_{23} - \sqrt{3}Sq_{23} & Sq_{13} & \sqrt{3}Cq_{23} + Cq_{13}Sq_{23} \\ Sq_{23} + \sqrt{3}Cq_{13}Cq_{23} & \sqrt{3}Sq_{13} & \sqrt{3}Cq_{13}Sq_{23} - Cq_{23} \\ -2Cq_{23}Sq_{13} & 2Cq_{13} & -2Sq_{13}Sq_{23} \end{bmatrix} \begin{bmatrix} 0 & \frac{1}{q_{33}} & 0 \\ \frac{1}{q_{33}Sq_{23}} & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

$${}^0J_{33}^{-T} = \frac{1}{2} \begin{bmatrix} \frac{Sq_{13}}{q_{33}Sq_{23}} & \frac{Cq_{13}Cq_{23} - \sqrt{3}Sq_{23}}{q_{33}} & \sqrt{3}Cq_{23} + Cq_{13}Sq_{23} \\ \frac{\sqrt{3}Sq_{13}}{q_{33}Sq_{23}} & \frac{Sq_{23} + \sqrt{3}Cq_{13}Cq_{23}}{q_{33}} & \sqrt{3}Cq_{13}Sq_{23} - Cq_{23} \\ \frac{2Cq_{13}}{q_{33}Sq_{23}} & \frac{-2Cq_{23}Sq_{13}}{q_{33}} & -2Sq_{13}Sq_{23} \end{bmatrix}$$

2.5.1.3. Cálculo del modelo dinámico de cada brazo.

Para el cálculo de los modelos dinámicos es necesario definir los parámetros inerciales de base que constituyen el conjunto mínimo de valores de inercias y masas a tener en cuenta, la utilización de estos parámetros permite eliminar aquellos que no tienen efecto sobre el modelo y agrupar otros con el fin de simplificar las ecuaciones finales, así como también el tiempo computacional de cálculo.

Para realizar el análisis físico de los cuerpos se utiliza el software Solid Edge®, en el cual se diseñan, construyen y se obtiene los valores inerciales y de masas necesarios de cada cuerpo, esto valores dependen de las dimensiones, la forma y el material con el cual está construido.

De esta manera se construye el cuerpo 1 que está asociado a la articulación 1 del robot, luego se calcula la masa, representada como M1 y se introduce en todas las expresiones de reagrupamiento que se requieran; luego se construye el cuerpo 2 y cuerpo 3 y se hace el mismo procedimiento, los valores numéricos de los parámetros inerciales de cada articulación se representan en las tablas 2-5 y 2-6.

Tabla 2-5: Parámetros del segundo momento de inercia del brazo *i*.

J	XXj	XYj	XZj	YYj	YZj	ZZj
1	0.08	0	0	0.04	0	0.04
2	8.19	0	0	1268.42	0	1268.21
3	1225.8	0	0	1225.59	0	1.17

Tabla 2-6: Parámetros del primer momento de inercia del brazo i , la masa y la inercia del accionador.

J	MX _j	MY _j	MZ _j	M _j	I _{aj}
1	0	0	0	0.00012	0
2	0.00023996	0	0	0.007	0
3	0	0	0.00019248	0.004	0.0000278

El procedimiento para la obtención de los valores numéricos de los parámetros inerciales se muestra en el anexo G.

Al introducir las tablas de parámetros geométricos de cada brazo en el software SYMORO, se pueden usar las opciones de éste para reducir el número de parámetros inerciales de base, obteniendo los que se representan en la tabla 2-7, las siguientes expresiones muestran las reagrupaciones de parámetros inerciales que se pueden hacer:

$$ZZ_{1R} = YY_2 + ZZ_1 + ZZ_3 ; MY_{1R} = MY_1 - MZ_2$$

$$XX_{2R} = XX_2 + XX_3 - YY_2 - ZZ_3 ; ZZ_{2R} = YY_3 + ZZ_2$$

Tabla 2-7: Parámetros inerciales de base del brazo i .

J	XX _j	XY _j	XZ _j	YY _j	YZ _j	ZZ _j	MX _j	MY _j	MZ _j	M _j	I _{aj}
1	0	0	0	0	0	ZZ _{1R}	0	MY _{1R}	0	0	0
2	XX _{2R}	0	0	0	0	ZZ _{2R}	MX ₂	0	0	0	0
3	0	0	0	0	0	0	0	0	MZ ₃	M ₃	I _{A3}

Debido a que los tres brazos están formados por estructuras idénticas, los parámetros inerciales son los mismos en los tres casos.

El modelo dinámico inverso general de un robot tipo serie, se define como sigue:

$$\Gamma = A(q)\ddot{q} + C(q, \dot{q})\dot{q} + Q(q) + F_v\dot{q} + F_s \text{sign}(\dot{q}) \quad (2-34)$$

Donde:

$A(q)$: Matriz de Inercia.

$C(q, \dot{q})$: Vector que representa las fuerzas de Coriolis y centrífugas.

$Q(q)$: Vector de fuerzas de gravedad.

F_v : Frotamiento viscoso de la articulación.

F_s : Frotamiento seco de la articulación.

Sin embargo, debido a la dificultad para identificar los frotamientos con cierta precisión, no son tenidos en cuenta, por lo tanto se considera que los cuerpos que forman al robot son rígidos y no presentan fenómenos de deformación, además el robot no se moverá a grandes velocidades, por lo que se pueden descartar las fuerzas de Coriolis y centrífugas. Con estas consideraciones se llega a la expresión más simple con la cual se puede modelar un robot serie:

$$\Gamma_i = A_i(q)\ddot{q}_i + Q_i(q) \quad (2-35)$$

Para una definición más completa de la dinámica del robot sería necesario incluir la dinámica de los motores en la ecuación del modelo dinámico inverso. En esta tesis no se considera el modelo de los actuadores con el fin de simplificar los cálculos, lo cual no influye significativamente en los resultados obtenidos al controlar un robot serie [64].

Siguiendo la metodología desarrollada en [64] para el cálculo del MDI de robots serie, se obtuvieron las siguientes matrices de inercia y vectores de gravedad:

$$A_1 = \begin{bmatrix} ZZ1R + (M3q_{31}^2 - 2MZ3q_{31} + XX2R)S^2q_{21} & 0 & 0 \\ 0 & ZZ2R - 2MZ3q_{31} + M3q_{31}^2 & 0 \\ 0 & 0 & M3 + IA3 \end{bmatrix}$$

$$A_2 = \begin{bmatrix} ZZ1R + (M3q_{32}^2 + 2MZ3q_{32} + XX2R)S^2q_{22} & 0 & 0 \\ 0 & ZZ2R + 2MZ3q_{32} + M3q_{32}^2 & 0 \\ 0 & 0 & M3 + IA3 \end{bmatrix}$$

$$A_3 = \begin{bmatrix} A_{11} & 0 & 0 \\ 0 & A_{22} & A_{23} \\ 0 & A_{23} & A_{33} \end{bmatrix}$$

$$A_{11} = MZ3q_{33}S^2q_{23} + \sqrt{3}MZ3q_{33}Sq_{23}Cq_{23} + \frac{M3q_{33}^2S^2q_{23}}{4} + \frac{\sqrt{3}M3q_{33}^2Sq_{23}Cq_{23}}{2} \\ + \frac{3M3q_{33}^2C^2q_{23}}{4} + ZZ1R + XX2RS^2q_{23}$$

$$A_{22} = \frac{3M3q_{33}^2}{4} + MZ3q_{33} + ZZ2R + \frac{M3q_{33}^2}{4}$$

$$A_{23} = \sqrt{3}M3q_{33}$$

$$A_{33} = M3 + IA3$$

$$Q_1 = 9.81 \begin{bmatrix} MX2Cq_{11}Cq_{21} + MZ3Cq_{11}Sq_{21} - M3q_{31}Cq_{11}Sq_{21} \\ MZ3Cq_{21}Sq_{11} - MX2Sq_{11}Sq_{21} - M3q_{31}Cq_{21}Sq_{11} \\ -M3Sq_{11}Sq_{21} \end{bmatrix}$$

$$Q_2 = 9.81 \begin{bmatrix} MX2Cq_{12}Cq_{22} + MZ3Cq_{12}Sq_{22} + M3q_{32}Cq_{12}Sq_{22} \\ MZ3Cq_{22}Sq_{12} - MX2Sq_{12}Sq_{22} + M3q_{32}Cq_{22}Sq_{12} \\ M3Sq_{12}Sq_{22} \end{bmatrix}$$

$$Q_3 = 9.81 \begin{bmatrix} M3 \left(\frac{q_{33}Cq_{13}Sq_{23}}{2} + \frac{\sqrt{3}q_{33}Cq_{13}Cq_{23}}{2} \right) + MX2Cq_{13}Cq_{23} + MZ3Cq_{13}Sq_{23} \\ M3 \left(\frac{q_{33}Sq_{13}Cq_{23}}{2} - \frac{\sqrt{3}q_{33}Sq_{13}Sq_{23}}{2} \right) - MX2Sq_{13}Sq_{23} + MZ3Sq_{13}Cq_{23} \\ M3 \left(\frac{Sq_{13}Sq_{23}}{2} + \frac{\sqrt{3}Sq_{13}Cq_{23}}{2} \right) \end{bmatrix}$$

De 2-35 se tiene que los MDI de los tres brazos son:

$$\Gamma_1 = A_1 \begin{bmatrix} \ddot{q}_{11} \\ \ddot{q}_{21} \\ \ddot{q}_{31} \end{bmatrix} + Q_1 \quad (2-36)$$

$$\Gamma_2 = A_2 \begin{bmatrix} \ddot{q}_{12} \\ \ddot{q}_{22} \\ \ddot{q}_{32} \end{bmatrix} + Q_2 \quad (2-37)$$

$$\Gamma_3 = A_3 \begin{bmatrix} \ddot{q}_{13} \\ \ddot{q}_{23} \\ \ddot{q}_{33} \end{bmatrix} + Q_3 \quad (2-38)$$

Para calcular el modelo dinámico directo de cada brazo, se toma la ecuación 2-35 para hallar las aceleraciones articulares en función de las posiciones, velocidades y torques de cada articulación.

$$\ddot{q}_i = A_i^{-1}(q)(\Gamma_i - Q_i(q)) \quad (2-39)$$

Donde A_i^{-1} representa la matriz de inercia inversa del brazo i.

Finalmente los MDD de cada brazo están dados por:

$$\ddot{q}_1 = A_1^{-1}(\Gamma_1 - Q_1) \quad (2-40)$$

$$\ddot{q}_2 = A_2^{-1}(\Gamma_2 - Q_2) \quad (2-41)$$

$$\ddot{q}_3 = A_3^{-1}(\Gamma_3 - Q_3) \quad (2-42)$$

El brazo 4 o brazo central no es necesario modelarlo ya que todas sus articulaciones son pasivas.

El desarrollo completo de la obtención de los modelos dinámico directo e inverso, de un brazo del robot, se muestra en el anexo B.

2.5.1.4. Modelo dinámico cartesiano de cada brazo H_{xi}

El modelo dinámico cartesiano de cada brazo H_{xi} , expresa el modelo dinámico inverso de cada cadena cinemática en el espacio cartesiano de los puntos P_i , siendo $i=1,2$ ó 3 , es decir, fija cada brazo a su respectivo punto de conexión con la plataforma. Este se define mediante la ecuación (2-43) [71].

$$H_{xi} = {}^0J_{3i}^{-T} \Gamma_i \quad (2-43)$$

Se puede observar que el efecto de las cadenas cinemáticas sobre la plataforma es equivalente a aplicar una fuerza igual a H_{xi} ($q_i, \dot{q}_i, \ddot{q}_i$) en cada punto P_i . Por tal motivo, se deduce que el Robot PREH, puede ser representado por un único cuerpo con los mismos parámetros inerciales que la plataforma, sobre el que se aplican fuerzas externas H_{xi} en cada punto P_i .

Aplicando la ecuación (2-43), se tiene que los modelos dinámicos cartesianos de los brazos del robot están dados por:

$$H_{x1} = {}^0J_{31}^{-T} \Gamma_1$$

$$H_{x1} = \begin{bmatrix} \frac{S q_{11}}{q_{31} S q_{21}} & \frac{C q_{11} C q_{21}}{q_{31}} & C q_{11} S q_{21} \\ 0 & \frac{S q_{21}}{q_{31}} & -C q_{21} \\ \frac{C q_{11}}{q_{31} S q_{21}} & \frac{-C q_{21} S q_{11}}{q_{31}} & -S q_{11} S q_{21} \end{bmatrix} \begin{bmatrix} \Gamma_{11} \\ \Gamma_{21} \\ \Gamma_{31} \end{bmatrix}$$

$$H_{x2} = {}^0J_{32}^{-T} \Gamma_2$$

$$H_{x2} = \begin{bmatrix} \frac{S q_{12}}{q_{32} S q_{22}} & \frac{C q_{12} C q_{22}}{q_{32}} & C q_{12} S q_{22} \\ 0 & \frac{S q_{22}}{q_{32}} & -C q_{22} \\ \frac{C q_{12}}{q_{32} S q_{22}} & \frac{-C q_{22} S q_{12}}{q_{32}} & -S q_{12} S q_{22} \end{bmatrix} \begin{bmatrix} \Gamma_{12} \\ \Gamma_{22} \\ \Gamma_{32} \end{bmatrix}$$

$$H_{x3} = {}^0J_{33}^{-T} \Gamma_3$$

$$H_{x3} = \frac{1}{2} \begin{bmatrix} \frac{Sq_{13}}{q_{33}Sq_{23}} & \frac{Cq_{13}Cq_{23} - \sqrt{3}Sq_{23}}{q_{33}} & \sqrt{3}Cq_{23} + Cq_{13}Sq_{23} \\ \frac{\sqrt{3}Sq_{13}}{q_{33}Sq_{23}} & \frac{Sq_{23} + \sqrt{3}Cq_{13}Cq_{23}}{q_{33}} & \sqrt{3}Cq_{13}Sq_{23} - Cq_{23} \\ \frac{2Cq_{13}}{q_{33}Sq_{23}} & \frac{-2Cq_{23}Sq_{13}}{q_{33}} & -2Sq_{13}Sq_{23} \end{bmatrix} \begin{bmatrix} \Gamma_{13} \\ \Gamma_{23} \\ \Gamma_{33} \end{bmatrix}$$

La fuerza total correspondiente es igual a $\sum_{i=1}^3 H_{xi}$

2.5.1.5. Análisis dinámico de la plataforma.

Utilizando las leyes de Newton, para cuando el robot se encuentra en reposo en una posición, la fuerza resultante es igual a:

$$F_r = (m_l + m_p)\ddot{x}_p - (m_l + m_p)g$$

Donde, m_l es la masa del laparoscopio, m_p es la masa de la plataforma y \ddot{x}_p es la aceleración cartesiana de la plataforma.

2.5.1.6. Fuerzas deseadas y/o momentos deseados.

Es importante mencionar que los pares transmitidos por cada brazo a las juntas universales que los unen a la plataforma, son pasados directamente a esta última, por tal razón estas articulaciones no se tienen en cuenta en el análisis de la dinámica. El otro extremo de cada cadena cinemática también están conectados a la base mediante juntas universales pasivas, por tal razón los pares de las dos articulaciones rotacionales son cero, ver Figura 2-27.

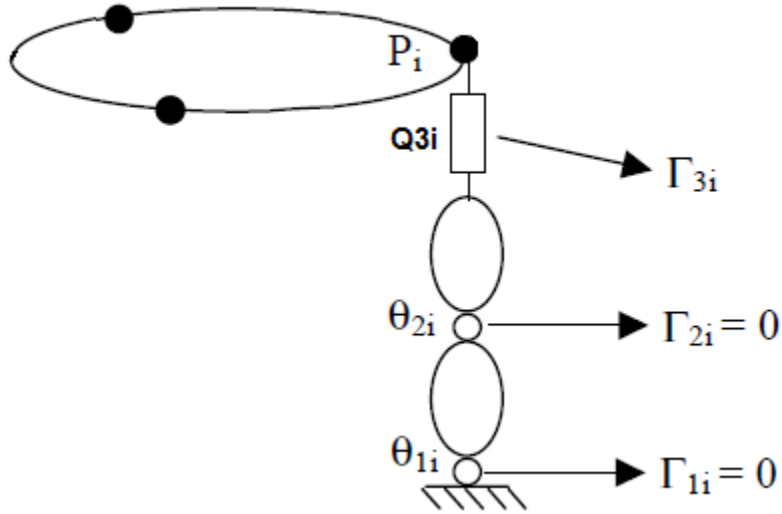


Figura 2-27: Pares articulares del brazo i .

Los pares totales de las articulaciones activas se expresan mediante la ecuación (2-43) [69] :

$$\Gamma = J^T (F_r + \sum_{i=1}^3 H_{xi}) \quad (2-44)$$

$$\Gamma = J^T \left((m_l + m_p)(\ddot{x}_p - g) + H_{x1} + H_{x2} + H_{x3} \right) \quad (2-45)$$

Del modelo cinemático del brazo i se tiene que:

$$\dot{x}_i = {}^0J_{3i}\dot{q}_i \quad (2-46)$$

Derivando la ecuación (2-45), se obtiene el modelo cinemático de segundo orden del brazo i y se representa mediante la ecuación (2-46)

$$\ddot{x}_i = {}^0J_{3i}\ddot{q}_i + \dot{{}^0J_{3i}}\dot{q}_i \quad (2-47)$$

Donde \dot{x}_i representa la velocidad cartesiana del efector final del brazo i .

Como se mencionó anteriormente, la plataforma no sufre fenómenos de deformación y no tiene movimientos de rotación sobre sus ejes, por tal razón la velocidad cartesiana de cualquiera de los puntos que la conforman es la misma.

$$\dot{x}_1 = \dot{x}_2 = \dot{x}_3 = \dot{x}_p \quad (2-48)$$

Por esta razón, la velocidad cartesiana del punto P_1 se puede reemplazar en la ecuación (2-44).

$$\ddot{x}_1 = {}^0J_{31}\ddot{q}_1 + {}^0\dot{J}_{31}\dot{q}_1 \quad (2-49)$$

$$\Gamma = J^T \left((m_l + m_p) \left(({}^0J_{31}\ddot{q}_1 + {}^0\dot{J}_{31}\dot{q}_1) - g \right) + H_{x1} + H_{x2} + H_{x3} \right) \quad (2-50)$$

Quedando así expresados los pares de las articulaciones activas en función de las posiciones, velocidades y aceleraciones articulares del robot.

$$\begin{bmatrix} \Gamma_{31} \\ \Gamma_{32} \\ \Gamma_{32} \end{bmatrix} = J^T \left((m_l + m_p) \left(({}^0J_{31}\ddot{q}_1 + {}^0\dot{J}_{31}\dot{q}_1) - g \right) + H_{x1} + H_{x2} + H_{x3} \right)$$

2.5.2. Modelo dinámico directo del robot

Para la obtención del modelo dinámico directo del robot, expresamos la aceleración cartesiana de la plataforma en términos de los pares de las articulaciones activas, las posiciones y velocidades articulares. Reescribiendo la ecuación (2-44) se tiene:

$$\ddot{x}_p = \frac{J^{-T}(\Gamma - H_{x1} - H_{x2} - H_{x3})}{m_l + m_p} + g \quad (2-51)$$

Utilizando las expresiones (2-46), (2-47) y (2-48), se pueden calcular las aceleraciones articulares del robot.

$$\begin{bmatrix} \ddot{q}_{11} \\ \ddot{q}_{21} \\ \ddot{q}_{31} \end{bmatrix} = {}^0J_{31}^{-1} \left(\frac{J^{-T}(\Gamma - H_{x1} - H_{x2} - H_{x3})}{m_l + m_p} + g - {}^0j_{31} \dot{q}_1 \right) \quad (2-52)$$

$$\begin{bmatrix} \ddot{q}_{12} \\ \ddot{q}_{22} \\ \ddot{q}_{32} \end{bmatrix} = {}^0J_{32}^{-1} \left(\frac{J^{-T}(\Gamma - H_{x1} - H_{x2} - H_{x3})}{m_l + m_p} + g - {}^0j_{32} \dot{q}_2 \right) \quad (2-53)$$

$$\begin{bmatrix} \ddot{q}_{13} \\ \ddot{q}_{23} \\ \ddot{q}_{33} \end{bmatrix} = {}^0J_{33}^{-1} \left(\frac{J^{-T}(\Gamma - H_{x1} - H_{x2} - H_{x3})}{m_l + m_p} + g - {}^0j_{33} \dot{q}_3 \right) \quad (2-54)$$

3. Pruebas de los modelos.

3.1. Prueba de los modelos geométricos.

Es de vital importancia comprobar que el robot es capaz de seguir trayectorias cartesianas, pasando a través del trocar o punto de incisión, esto se hace utilizando los modelos geométricos en el diagrama de Matlab/Simulink® que se presenta en la figura 3-1.

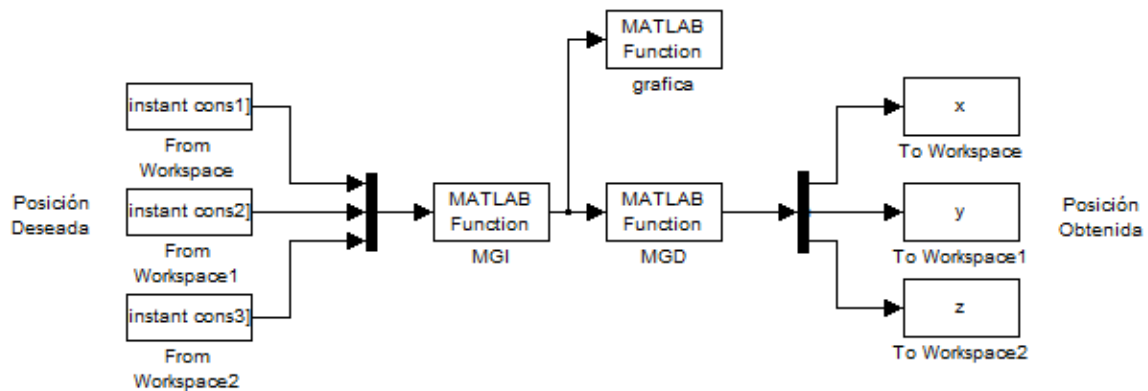


Figura 3-1: Diagrama en Simulink prueba modelos geométricos.

Donde,

Cons1, 2 y 3 representan las posiciones cartesianas deseadas.

x, y, z, representan las posiciones cartesianas alcanzadas por el robot.

La función del modelo geométrico inverso se implementa con base en las ecuaciones (2-9), (2-10), (2-11). A esta función ingresan los valores de las posiciones deseadas obteniéndose la longitud de los actuadores. El código completo se muestra en el anexo A.1.

De igual manera, la función del modelo geométrico directo se desarrolla con base en las ecuaciones (2-12), (2-13), (2-14). A ella ingresan los valores de las

longitudes de los actuadores; obteniéndose las posiciones alcanzadas por el robot. El código completo se muestra en el anexo A.2.

La visualización del robot se hace mediante el bloque llamado “grafica”, en él se introduce la función llamada “visualización”, esta función permite mostrar los movimientos del robot en un plano de tres dimensiones a partir de la longitud de sus actuadores, el código se muestra en el anexo A.3.

En la figura 3-2 se muestra el resultado de la visualización del robot, en ella se aprecia en color azul los brazos de la plataforma, incluida la parte del endoscopio que está por fuera del abdomen, mientras que la porción adentro, se muestra en color amarillo. Por simplicidad del sistema, la base y la plataforma son dibujadas en forma triangular y se muestran en color rojo.

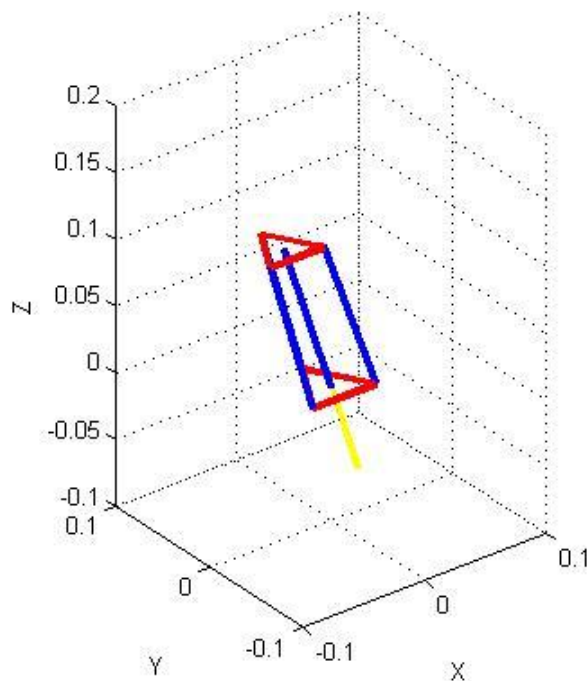


Figura 3-2: Visualización del robot.

La simulación se realizó para que el robot siga una trayectoria circular en el plano (x, y) de radio 0.05 m y a una distancia de 0.08 m, sobre el eje z de la base de la plataforma. El código para la obtención de las coordenadas de la trayectoria circular se muestra en el anexo A.4.

En la figura 3-3 se puede observar la posición de la trayectoria deseada (en color azul) y la seguida por el robot (color rojo). Calculando el error de seguimiento cartesiano por el método de mínimos cuadrados, se obtuvo un máximo valor de 14.5×10^{-17} m, este se representa en la figura 3.4, debido a que el error es tan pequeño no se nota diferencia entre las mismas.

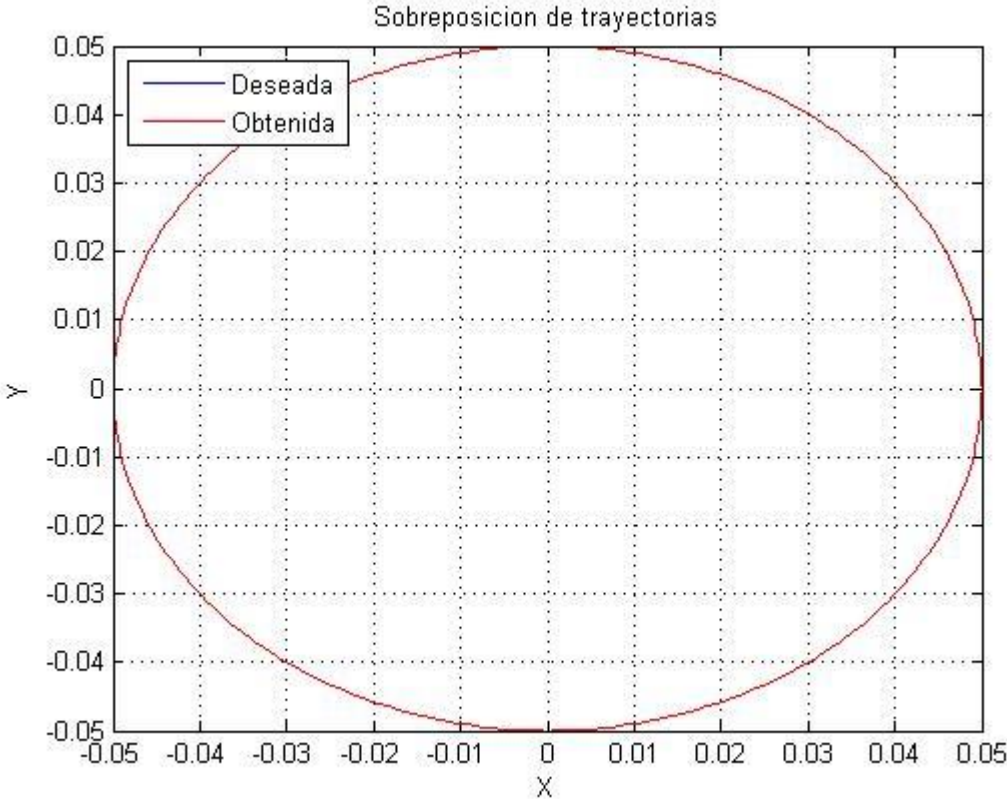


Figura 3-3: Sobreposición de trayectorias circulares.

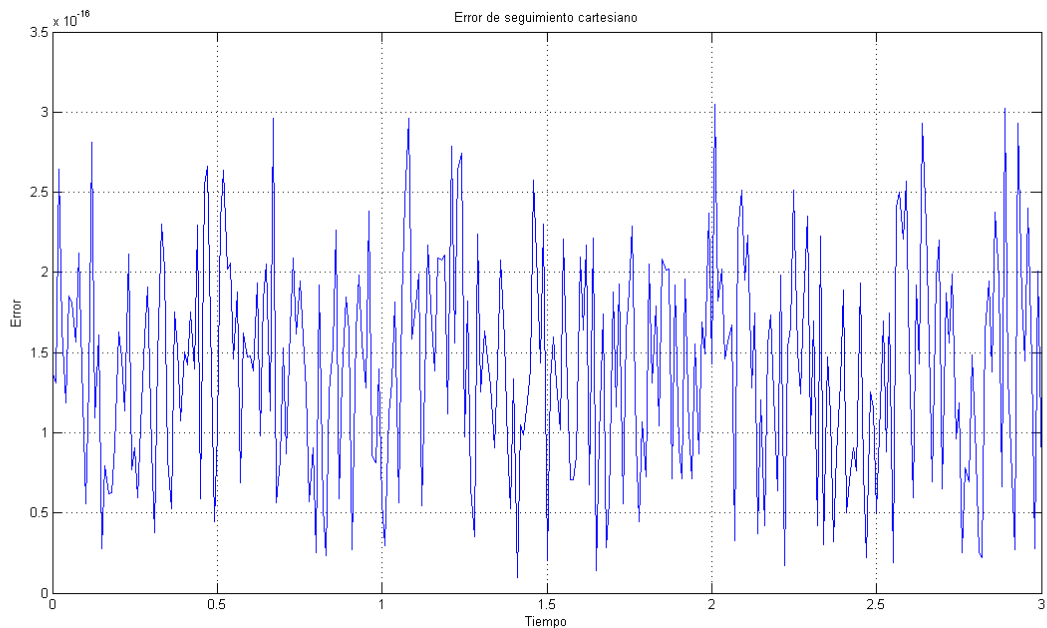


Figura 3-4: Error entre las dos trayectorias circulares.

Por medio de los modelos geométricos se verifica que el robot respeta el paso por la incisión, para ello se utiliza la misma trayectoria circular que se empleó en el caso anterior, y con la ayuda del archivo de Matlab llamado “paso_incisión” se grafican las posiciones del brazo central para todos los puntos de las trayectorias. El código necesario se muestra en el anexo A.6.

Como se ve en la figura 3-5, el brazo central respeta en todo momento el paso por la incisión, el endoscopio se muestra en color verde, mientras que los extremos son marcados en color azul, el punto de incisión es representado mediante un punto rojo.

Para la correcta visualización de las gráficas es necesario ajustar los tamaños de los ejes, para la trayectoria circular, $-0.1 \leq X \leq 0.1$ y $-0.1 \leq Y \leq 0.1$.

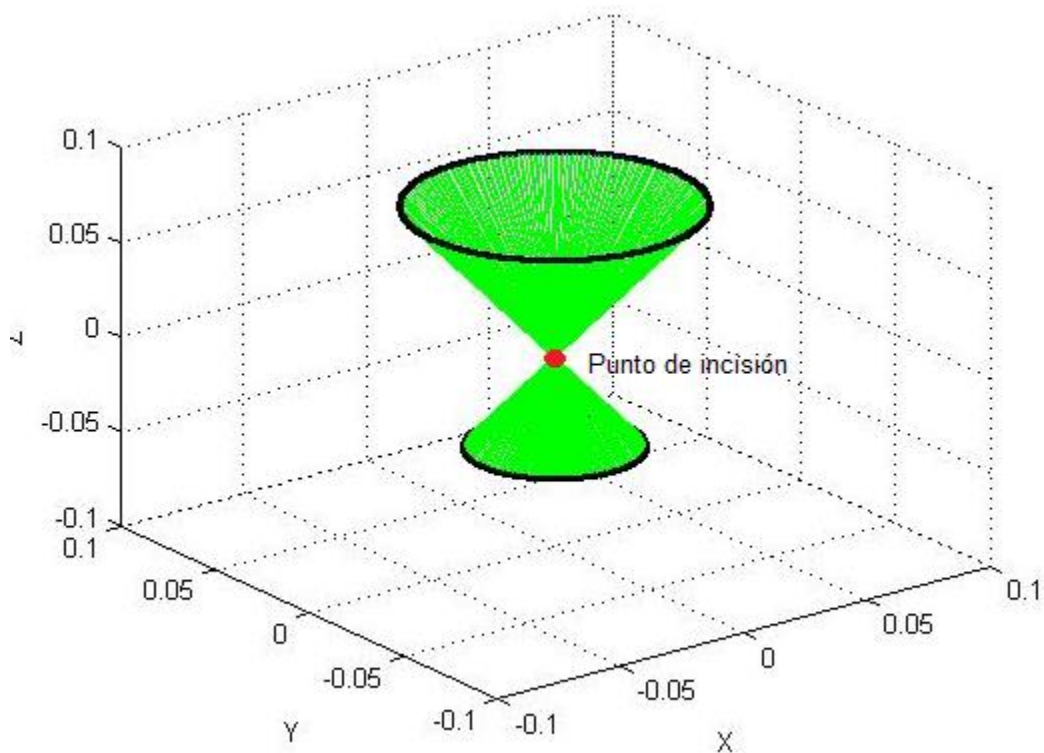


Figura 3-5: Paso por el trocar con trayectoria circular

3.2. Área de trabajo del robot

El espacio de trabajo intra-abdominal varía según la posición en la que se encuentra el paciente; y esta posición varía según el procedimiento a realizar. Para nuestro diseño la mayor limitación se encuentra en las longitudes máxima y mínima de los actuadores lineales, ya que el uso de unos motores con mayor recorrido, implicarían un mayor tamaño de la estructura total. Por tal razón se eligieron actuadores con una longitud mínima de 7 cm y máxima de 15 cm, estos cuentan con un recorrido de 8 cm. En el mercado existen diversidad de actuadores que cumplen este requerimiento entre ellos están los L12 de Fircelli [70].

Se supone que la zona abdominal después de la insuflación con CO₂ queda como una esfera con su centro en la zona superior del abdomen en posición normal y sin insuflación, ver figura 3-6. Para una colecistectomía el abdomen se insufla en promedio 2500 cm³. Por tal razón el radio será aproximadamente 8 cm, que corresponde al recorrido de los motores seleccionados.

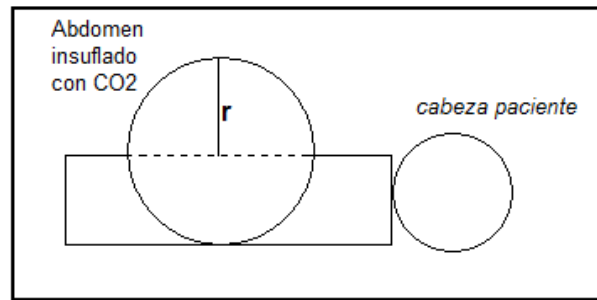


Figura 3-6: Insuflación abdomen.

Para comprobar el área de trabajo del robot, se creó el archivo de Matlab llamado "espaciot.m", en el cual se suponen una serie de coordenadas que podrá alcanzar la plataforma y con las restricciones dadas por las longitudes de los actuadores se dibujan las posiciones a las que verdaderamente llega la plataforma. Los resultados se muestran en las figuras 3-7 y 3-8, el código de este programa se describe en el anexo A.7.

En la figura 3-7 se observan dos áreas de trabajo, en rojo se muestran todos los puntos que alcanza la plataforma, mientras que en azul se indica el área que cubre la punta del laparoscopio dentro del abdomen del paciente.

En la figura 3-8 (izquierda), se hace un corte transversal del área de trabajo sobre el plano (x,z) y la derecha, es una vista de planta, plano (x,y) , en ellas se puede observar que el espacio cubierto por la plataforma es una semi - esfera de radio 14 cm aproximadamente y el de la cámara es un cono de radio 4 cm unido a una semi - esfera del mismo radio.

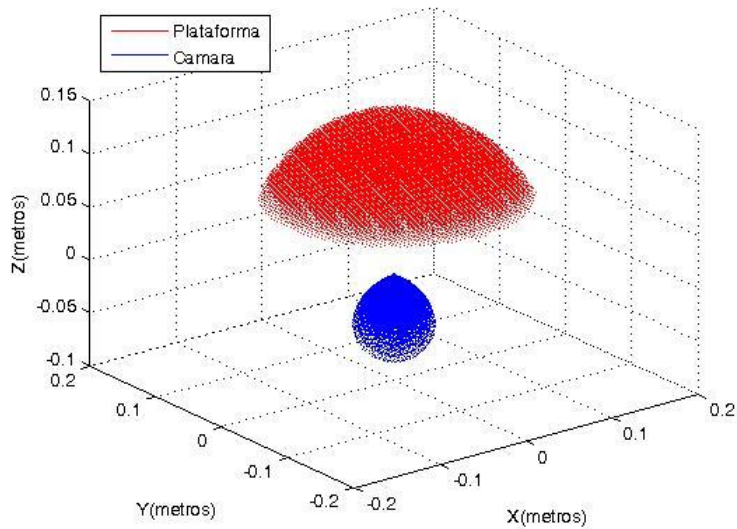


Figura 3-7: Espacio de trabajo.

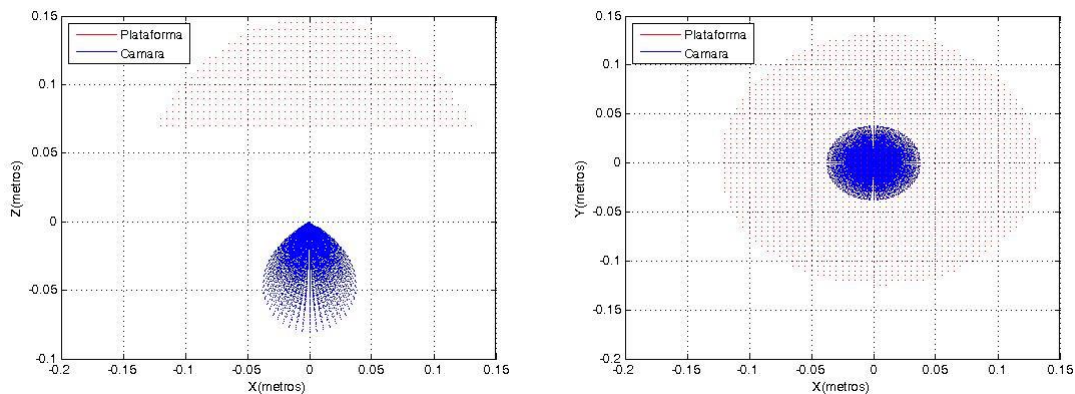


Figura 3-8: Vista plano (x,z) del área de trabajo(izquierda), plano (x,y) (derecha)

Con estos dos valores se estaría cumpliendo con el área de trabajo de una cirugía laparoscópica de extracción de vesícula biliar.

3.3. Análisis de singularidades

Las singularidades de un robot son los puntos dentro del área de trabajo a los que el robot no puede llegar.

En robots paralelos se pueden definir 2 singularidades [73].

1. Cuando el determinante de J_q es igual a cero

Si $\det(J_q) = 0$, es porque existen vectores \dot{q} diferentes de cero que resultan en vectores \dot{x} iguales a cero. Por esta razón el robot pierde uno a más grados de libertad. El mecanismo puede resistir fuerzas o momentos en determinadas direcciones aun cuando los actuadores no desarrollen ninguna fuerza o par. A este tipo de singularidades se les llama singularidades de la cinemática inversa.

Analizando J_q se tiene que:

$$J_q = -2 \begin{bmatrix} q_{31} & 0 & 0 \\ 0 & q_{32} & 0 \\ 0 & 0 & q_{33} \end{bmatrix}$$

$$\det(J_q) = -2q_{31} \begin{vmatrix} -2q_{32} & 0 \\ 0 & -2q_{33} \end{vmatrix} - 0 \begin{vmatrix} 0 & 0 \\ 0 & -2q_{33} \end{vmatrix} + 0 \begin{vmatrix} 0 & -2q_{32} \\ 0 & 0 \end{vmatrix}$$

$$\det(J_q) = -2q_{31} \times -2q_{32} \times -2q_{33}$$

$$\det(J_q) = -8q_{31}q_{32}q_{33}$$

Este determinante es cero solo cuando la longitud de alguno de los brazos es cero, esto es imposible ya que por la geometría de los actuadores, estos deben tener una longitud mínima. Por tal motivo se concluye que no existen singularidades de cinemática inversa dentro del espacio de trabajo del mecanismo. Sin embargo pueden existir singularidades de este tipo en los límites de la zona de trabajo donde uno o más de los cilindros están completamente estirados o contraídos.

2. Cuando el determinante de J_x es igual a cero

Si $\det(J_x) = 0$, es porque existen vectores \dot{x} diferentes de cero que resultan en vectores \dot{q} iguales a cero, por lo que el robot gana uno o más grados de libertad. El mecanismo puede no resistir fuerzas o momentos en determinadas direcciones. A este tipo de singularidades se les llama singularidades de la cinemática directa.

Al analizar J_x se tiene que:

$$\det(J_x) = \det \left\{ \begin{bmatrix} 2P_{1X} & 2P_{1Y} & 2P_Z \\ 2P_{1X} + 2\sqrt{3}(r_p - r_b) & 2P_{1Y} & 2P_Z \\ 2P_{1X} + \sqrt{3}(r_p - r_b) & 2P_{1Y} + 3P_{1Y}(r_p - r_b) & 2P_Z \end{bmatrix} \right\}$$

$$\det(J_x) = 12\sqrt{3}P_Z r_b^2 - 24\sqrt{3}P r_b r_p + 12\sqrt{3}P_Z r_p^2$$

$$\det(J_x) = 12\sqrt{3}P_Z (r_b - r_p)^2$$

A simple vista se observa que para que el determinante de J_x sea cero, P_Z o P_{1Y} deben ser cero, o r_b debe ser igual a r_p .

Pero por condiciones de diseño los radios de la base y la plataforma son diferentes, por tal razón no se evidencia singularidad.

Por otra parte P_Z , nunca puede llegar a ser cero, debido a que los actuadores tienen una longitud mínima, como se definió anteriormente.

3.4. Diseño de las piezas del robot en Solid Edge

Los cuerpos del robot PREH se construyeron en el software CAD SolidEdge®, el material utilizado para los cuerpos que están por fuera del paciente es aluminio, se escogió debido a su bajo peso, tiene una densidad de 2712 Kg/m^3 . Para los cuerpos que ingresan al abdomen es necesario utilizar un material que se pueda esterilizar por tal motivo se pensó en acero inoxidable quirúrgico 410, de 7750 Kg/m^3 .

En la figura 3-9 se muestran los diseños realizados en SolidEdge de la base (izquierda) y la plataforma (derecha) del robot.

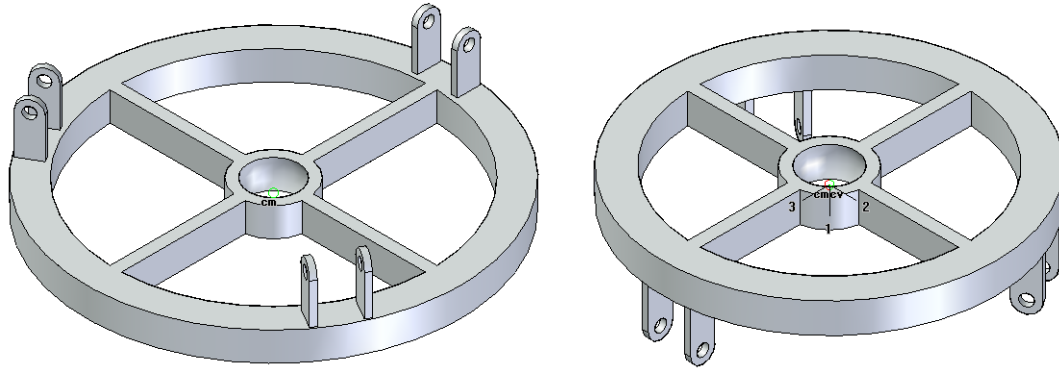


Figura 3-9: Diseño en Solid Edge de la base (izquierda) y la plataforma (derecha).

En la figura 3-10 se muestran los diseños realizados en SolidEdge de las tres articulaciones que forman los distintos brazos del robot. La articulación 1 también es usada para unir cada brazo a la plataforma.

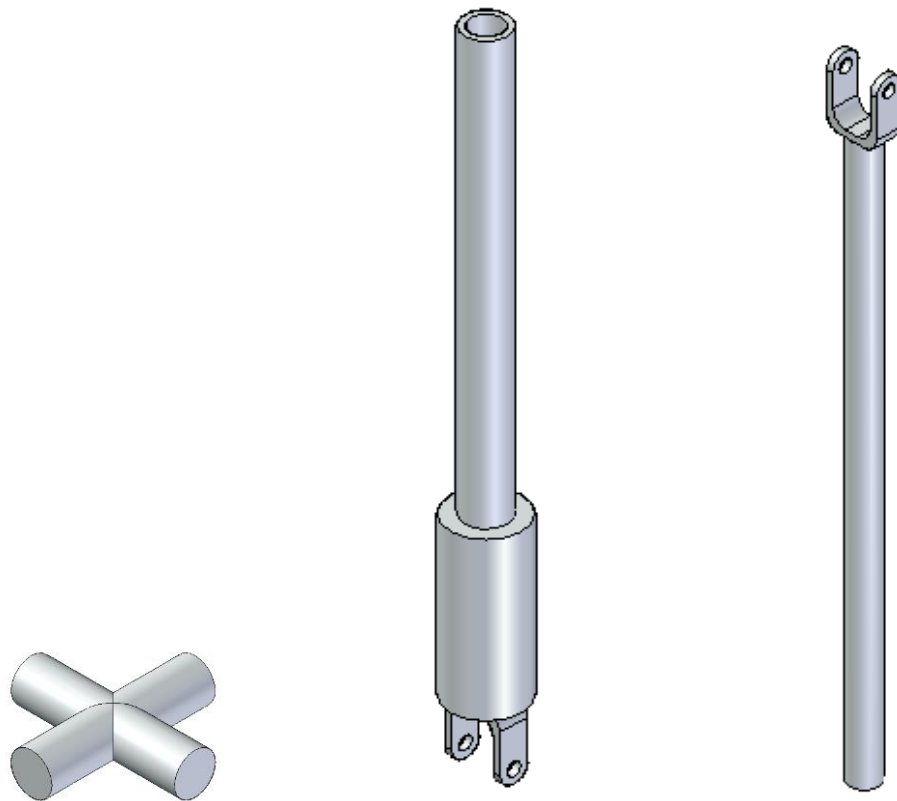


Figura 3-10: Diseño articulación 1(izquierda), articulación 2 (centro), articulación 3 (derecha).

En la figura 3-11 se muestra el diseño de los soportes que fijan el endoscopio a la base y a la plataforma. En la plataforma dicha pieza cumple la función de trocar, garantizando que el endoscopio respete el paso por la incisión.

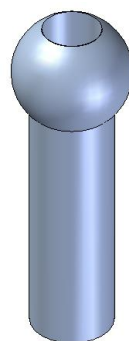


Figura 3-11: Soporte del endoscopio.

En la figura 3-12 se muestra el diseño en SolidEdge del endoscopio que servirá como sistema de visión dentro del abdomen del paciente.

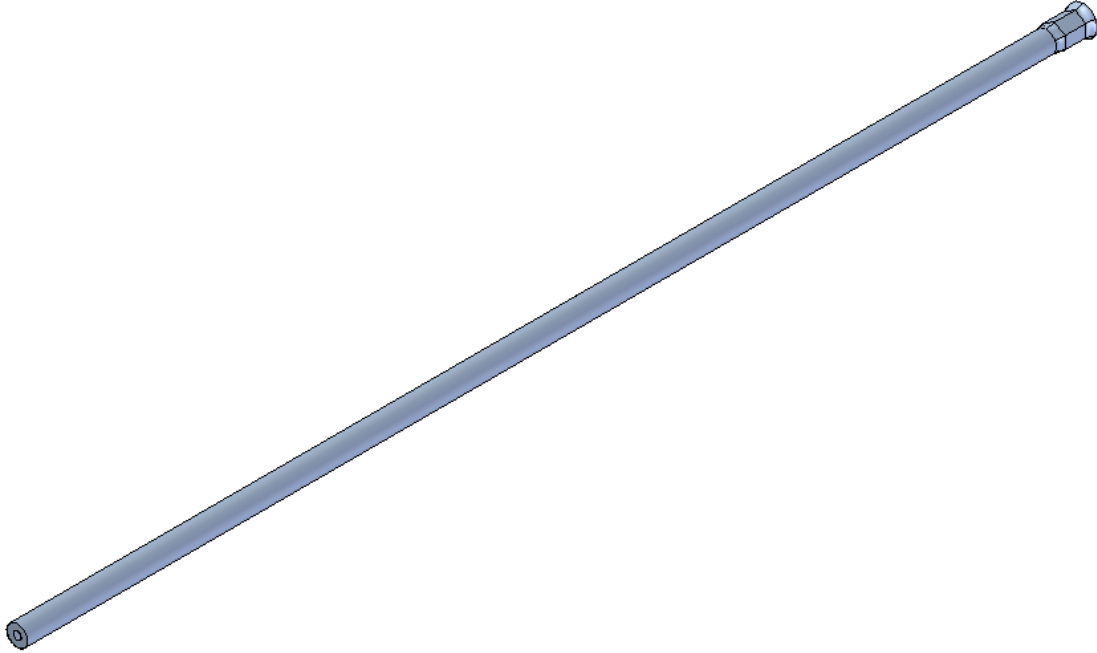


Figura 3-12: Endoscopio o cámara.

En la figura 3-13 se presenta el ensamble total del robot.

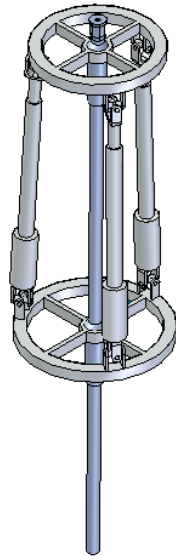


Figura 3-13: Ensamble total del robot.

En las figuras 3.14 a 3.17 se muestran diferentes vistas del montaje del robot en una cirugía laparoscópica. Por sugerencia del médico cirujano Alexei Bernardo Rojas Díaz, Jefe del departamento de Cirugía del Hospital Universitario San José de Popayán, el soporte para el robot se coloca en medio de las piernas del paciente como se puede observar en las siguientes figuras.

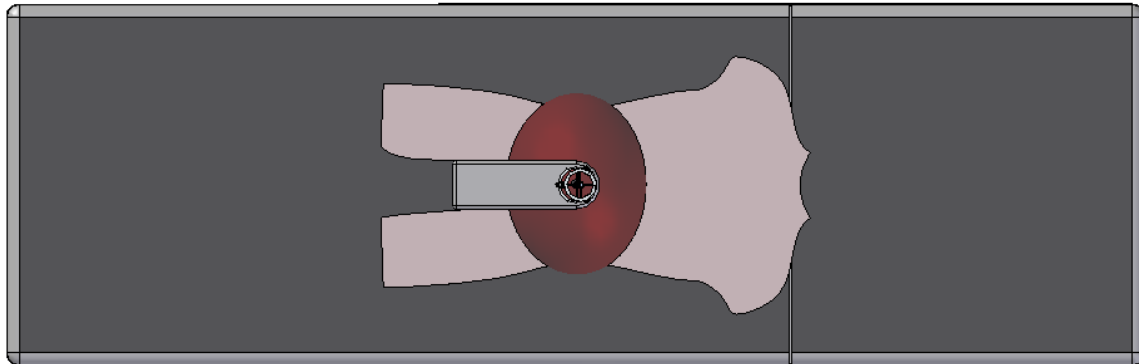


Figura 3-14: Vista de planta conjunto camilla, paciente y robot.

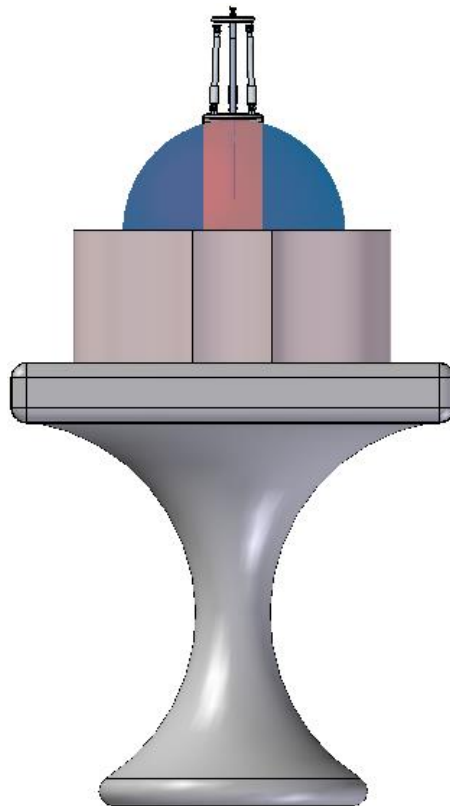


Figura 3-15: Vista de frente conjunto camilla, paciente y robot.

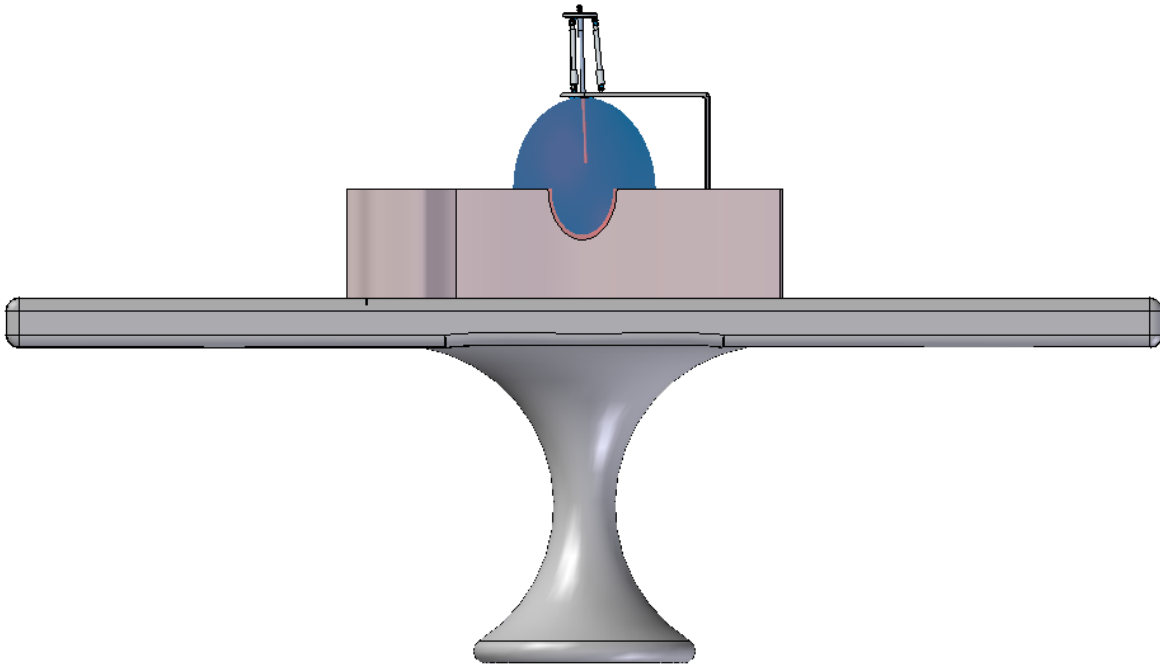


Figura 3.16: Vista de costado conjunto camilla, paciente y robot.

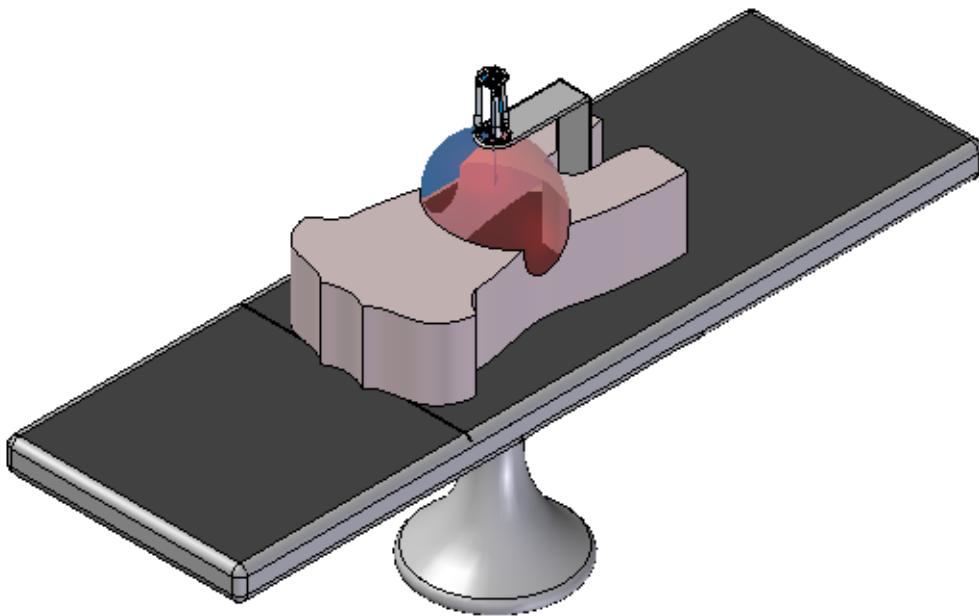


Figura 3.17: Vista de costado conjunto camilla, paciente y robot.

4. CONTROL CTC DEL ROBOT

Una vez definida el área de trabajo del robot y sus limitaciones de movimiento, se diseña un controlador de seguimiento de trayectorias teniendo en cuenta los modelos dinámicos que representan al robot real en la simulación.

El control por par calculado involucra el modelo de la planta en la misma estrategia de control, lográndose así cancelar las no-linealidades de los modelos dinámicos de los brazos del robot, lo que tiene especial importancia cuando se requiere gran velocidad de trabajo y un mejor desempeño del lazo de control [67].

En la teoría esta técnica de control asegura la linealización mediante realimentación y el desacople de las ecuaciones del modelo dinámico del sistema, proporcionando un comportamiento uniforme, cualquiera que sea la configuración del robot en un determinado momento, [74] - [76].

En la figura 4-1 se observa el diagrama de bloques en Simulink que se implementó para el control CTC del robot PREH.

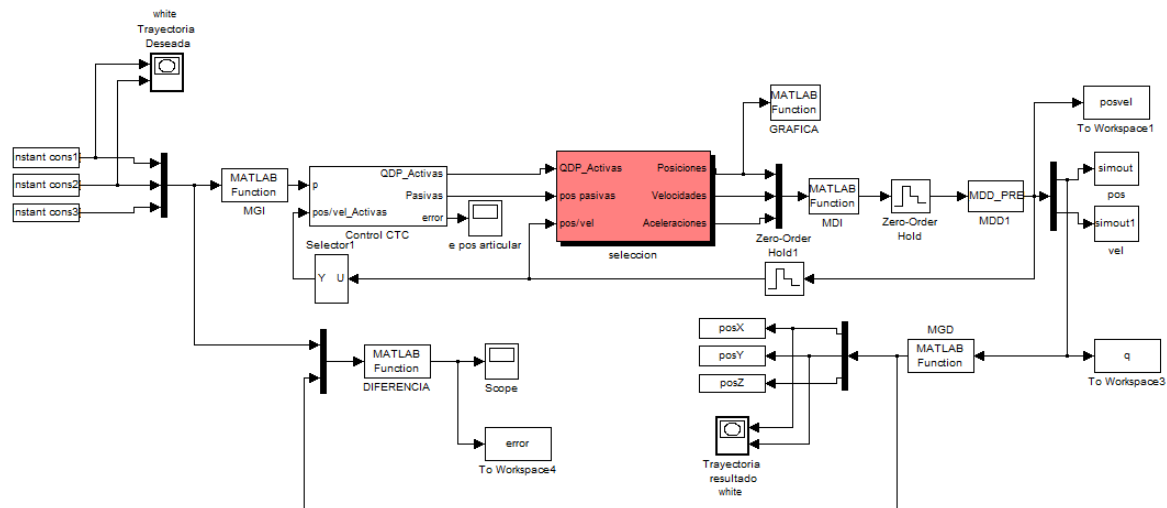


Figura 4-1: Diagrama de bloques de Simulink para el control CTC.

Las función necesaria para la obtención del modelo dinámico inverso del robot es implementada con base en las ecuacion (2-49), mientras que la del modelo

dinámico directo se basa en las ecuaciones (2-39), (2-40) y (2-41) y se implementa mediante una S-Function, donde se definen como entradas los pares de las 9 articulaciones y 18 estados, 9 posiciones y 9 velocidades, que igualmente serán las salidas del sistema. Las funciones completas se muestran en los anexos A.8 y A.9. Adicionalmente a estas funciones se implementa la función p_iner.m donde se definen los parámetros inerciales de base de todas las articulaciones del robot, el código de esta función se muestra en el anexo A.10. Para que el sistema funcione se debe ejecutar el programa inicio.m, en él se definen las ganancias del controlador, el tiempo de muestreo, las posiciones iniciales de todas las articulaciones del robot y la trayectoria que se desea seguir. El código se muestra en el anexo A.11.

En la figura 4-3, se muestra el subsistema del control CTC o control lineal, en él se puede observar que solo los valores de posición y velocidad de las juntas actuadas pasan por el controlador.

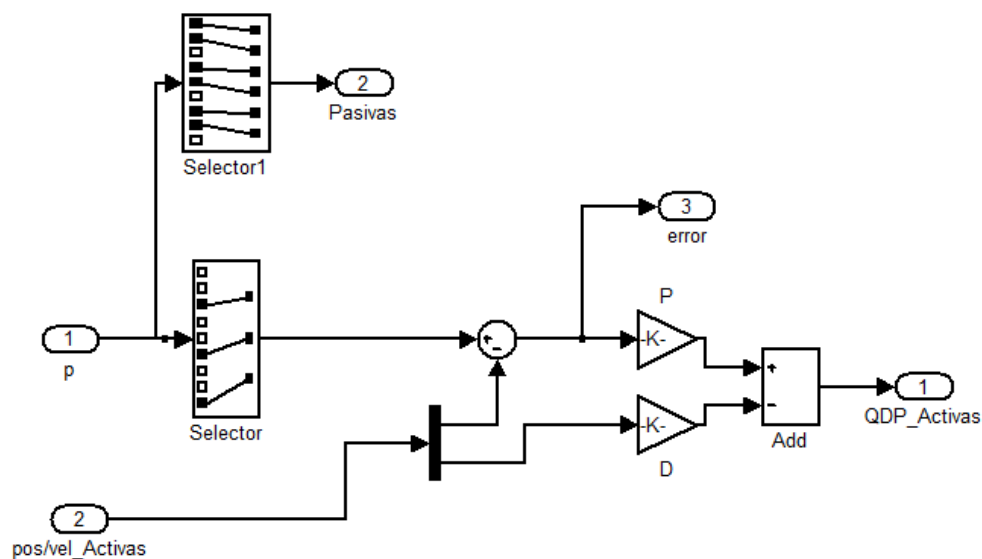


Figura 4.1: Diagrama en Simulink del controlador CTC

En el esquema de control es muy importante utilizar el modelo geométrico directo, ya que gracias a él se pueden conocer las coordenadas del robot en el espacio cartesiano X y verificar que está siguiéndose la trayectoria deseada X^d .

En el esquema se analiza cómo la posición cartesiana deseada de la plataforma (X^d) pasa por el bloque de ecuaciones del modelo geométrico inverso (MGI). Esto con el fin de obtener las variables articulares deseadas (q^d), de tal forma que el controlador CTC garantice un mínimo error entre q y q^d , así como un mínimo error entre X y X^d .

4.1. Sintonización Control CTC.

La sintonización del controlador CTC se realiza por el método manual planteado en [75][76] y es el siguiente:

- Colocar las constantes K_p en 1 y las K_v en cero. El sistema reaccionará como si no existiera controlador alguno, lo cual hará que la respuesta del robot sea inestable.
- Aumentar los valores de K_p hasta obtener estabilidad aunque aparezcan oscilaciones.
- Empezar a aumentar levemente los valores de K_v con el fin de disminuir las oscilaciones. Estos valores no pueden ser muy altos porque el sistema se hará extremadamente lento.
- Una vez disminuidas las oscilaciones se pueden incrementar de nuevo los valores proporcionales K_p con el fin de disminuir el error. Ajustar poco a poco estos dos valores (K_p y K_v): aumentar el primero para disminuir el error aunque aparezcan oscilaciones, aumentar el segundo para disminuir estas oscilaciones.

De esta manera se obtienen los valores de ganancias presentados en la Tabla 4.1. Estos valores corresponden sólo a las articulaciones activas (motorizadas), dado que las pasivas (no motorizadas) son independientes del controlador, y no interactúan en el lazo de control.

Tabla 4-1: Ganancias para el controlador CTC

j	1	2	3
K_p	25000	200000	200000
K_v	180	700	600

4.2. SEGUIMIENTO DE TRAYECTORIAS

Se llevan a cabo pruebas para verificar el correcto funcionamiento de los modelos y el controlador, se utilizan las siguientes trayectorias.

- Círculo sobre el plano XY con radio de 0.05 m, centro de la plataforma ubicado en (0, 0, 0.08), el tiempo de simulación es de 3 seg y el tiempo de muestreo de 0.01 seg. La consigna circular deseada se muestra en la figura 4-2.

En la figura 4-3 se muestra el error del seguimiento de la trayectoria, su máximo sobre impulso es aproximado a 6.5×10^{-5} metros. El error en estado estacionario, cercano a los 4×10^{-7} metros, con esto se ve que el seguimiento que hace el robot PREH es muy eficiente.

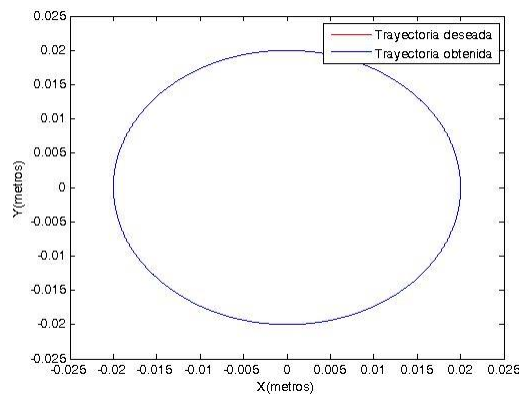


Figura 4-2: Trayectoria circular deseada y obtenida.

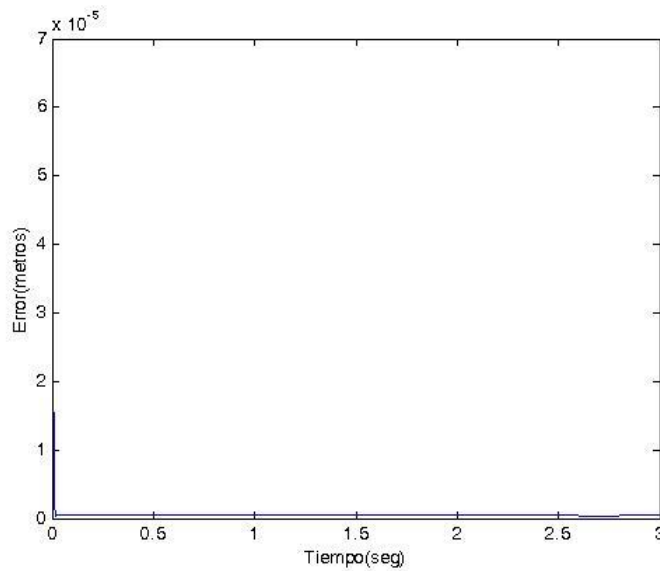


Figura 4-3: Error de seguimiento cartesiano en estado transitorio.

Al realizar las anteriores pruebas, se verificó que los modelos geométricos, dinámicos y que el controlador CTC, sirven para realizar las trayectorias cartesianas requeridas por PREH y que logra las posiciones que se pretenden en una cirugía laparoscópica. De otra parte, al aplicar el control por par calculado (CTC) a la dinámica del robot, se logran errores cartesianos de seguimiento muy pequeños, además de respetar el paso por el trocar. Es de rescatar que las oscilaciones que se presentan en las figuras del error son despreciables, comparadas con los movimientos y deficiencias de temblor de un asistente humano encargado del posicionamiento del endoscopio.

Para la correcta visualización se utiliza la función visualización.m, donde se muestra una vista en 3D del robot, además se puede modificar para obtener una vista de planta y así cerciorarse que el robot alcanza el radio de la trayectoria (ver figura 4-4)

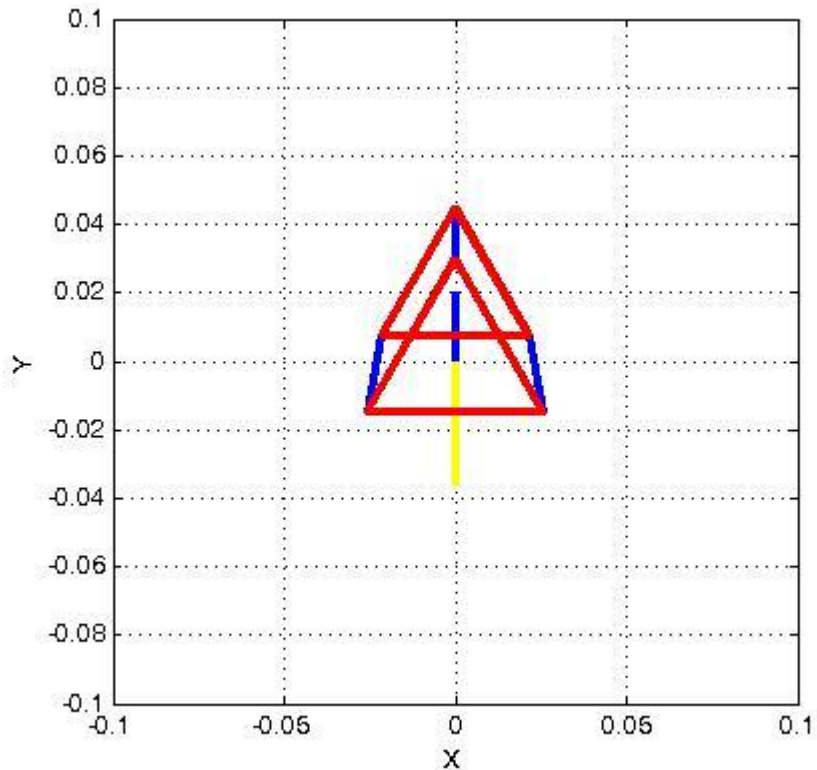


Figura 4-4: Vista de planta del robot.

4.3. Conexión de una interfaz (Joystick) como generador de trayectorias.

Como complemento a este trabajo se consideró contar con una entrada analógica para generar trayectorias libres en el plano XYZ, que puedan ser seguidas por el robot.

Con este objetivo se selecciona como entrada analógica el joystick, para facilitar la posibilidad de conectar la interfaz a un computador. Se ha decidido utilizar alguno que permita ser conectado a un puerto USB, que genere trayectorias en el plano XYZ, es decir que tenga tres salidas analógicas, una para generar movimientos en cada eje del plano.

El Joystick seleccionado para generar las trayectorias es "*Genius, Maxfire G12-U vibration*", el cual posee cuatro salidas analógicas y conexión USB. La calibración del joystick se lleva a cabo en el Panel de Control de Windows (dispositivos de juegos), por lo que no requiere de un cuidado especial.

Para la adquisición de los datos que entrega el joystick, Simulink® cuenta con un bloque que se encarga automáticamente de reconocer las entradas analógicas y digitales del joystick que se conecta (ver figura 4-5). Es muy importante llevar a cabo una caracterización de los rangos de las señales que se reciben del dispositivo, es decir, se deben llevar los valores de las señales a unos rangos que sean más apropiados para los movimientos del robot (ver figura 4-6)

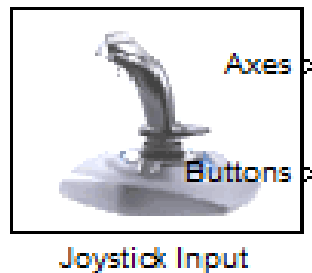


Figura 4-5: Bloque Joystick Simulink.

El nuevo diagrama de control se muestra en la figura 4-7, en él se aprecia que fue insertado un bloques más, el bloque Joystick, cuyo subsistema se muestra en la figura 4-6.

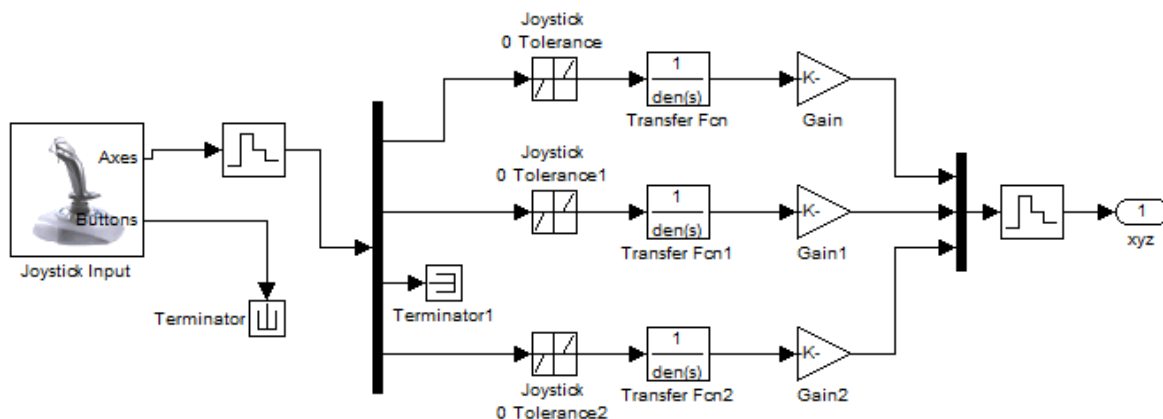


Figura 4-6: Adquisición y caracterización de las señales del joystick.

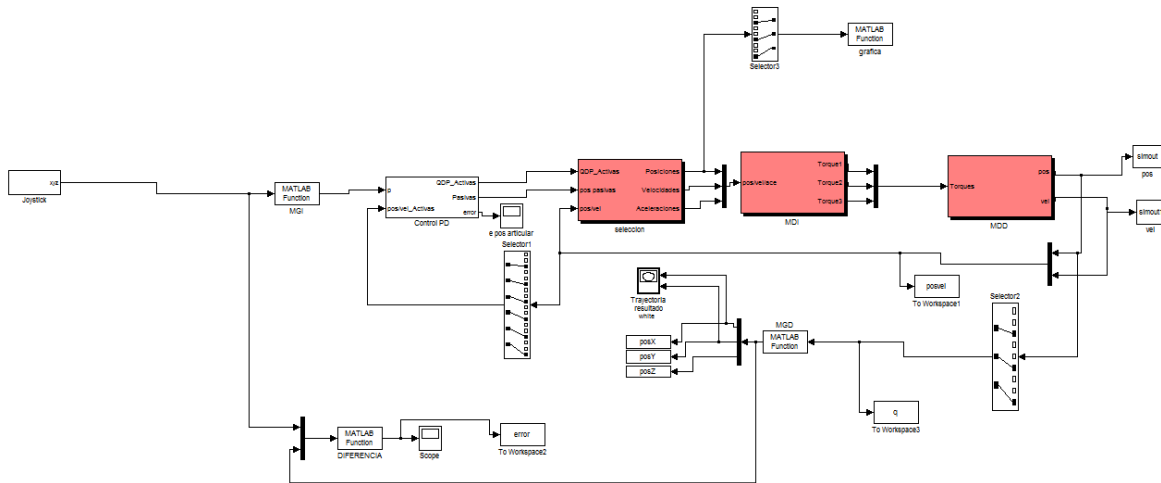


Figura 4-7: Diagrama de control con trayectorias generadas por el joystick.

Para probar que los modelos matemáticos y el controlador CTC son capaces de seguir las trayectorias que se realizan con el joystick, inicialmente se llevó el dispositivo a los puntos extremos en los ejes X e Y (generando una trayectoria de una cruz). Como resultado, el robot PREH realizó una trayectoria como la que se muestra en la Figura 4-8.

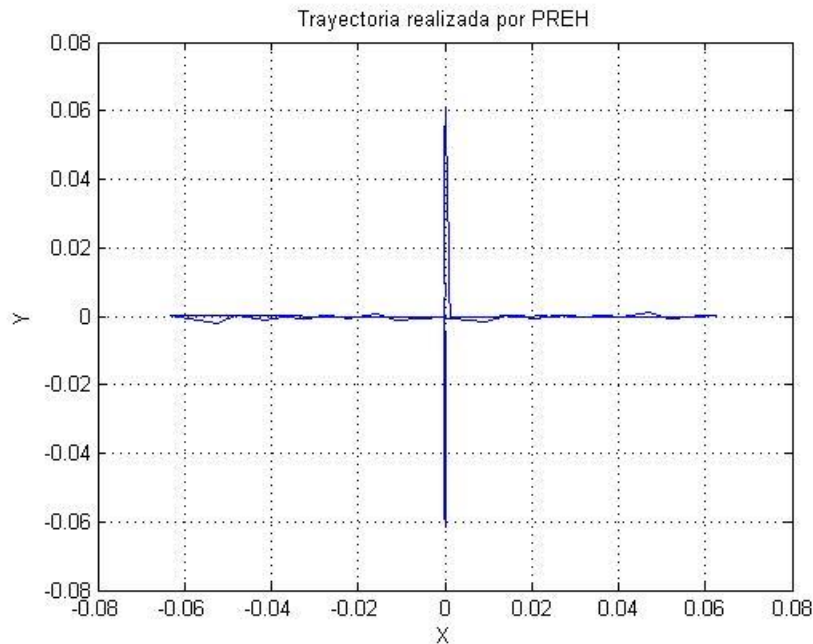


Figura 4-8: Trayectoria realizada por PREH.

En la figura 4-9 se observa el error de seguimiento cartesiano para la trayectoria generada por el joystick.

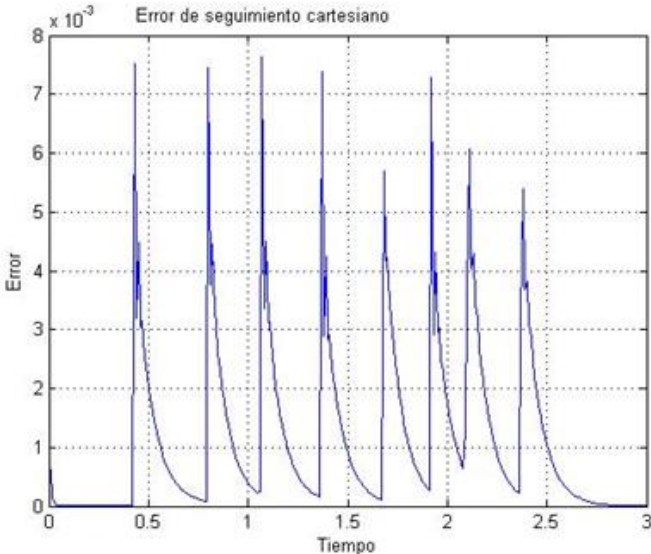


Figura 4-9: Error de seguimiento con la trayectoria del joystick.

En la figura 4-10 se muestra una trayectoria generada por el joystick en el plano X, Z y en la figura 4-11, el error de seguimiento cartesiano de dicha trayectoria.

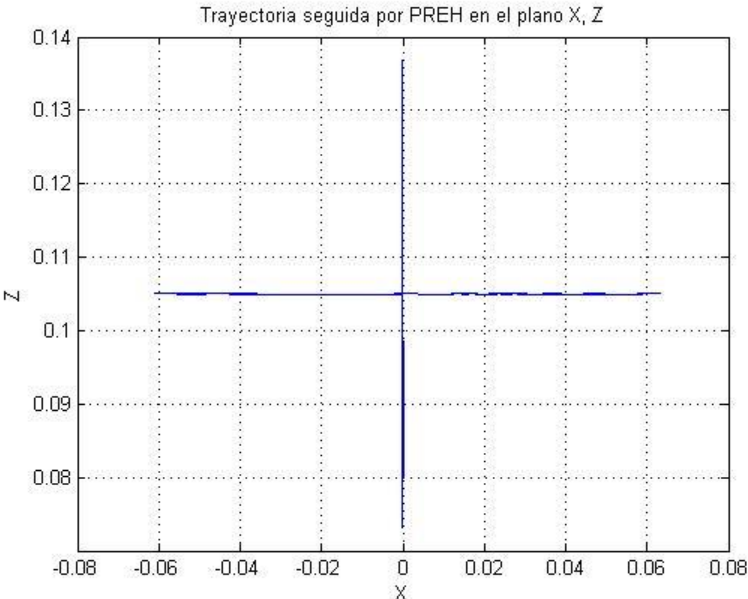


Figura 4-10: Trayectoria seguida por PREH en el plano X, Z.

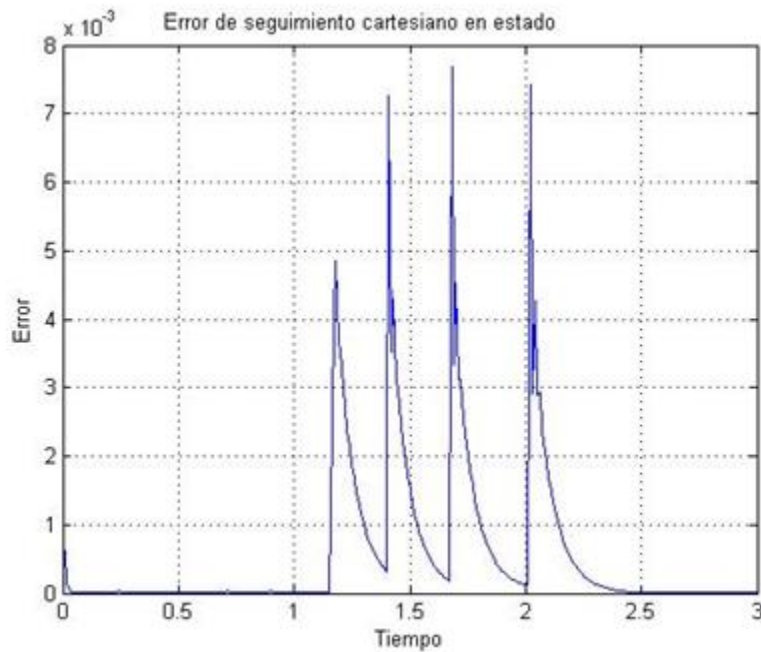


Figura 4-11: Error de seguimiento cartesiano plano X, Z.

4.3.1. Recomendaciones de manejo de PREH con joystick.

Debido a que la velocidad con la que el joystick entrega las señales al sistema es diferente a la velocidad de procesamiento del diagrama en Matlab/Simulink, siendo ésta más lenta, es necesario que los movimientos que se generan con esta entrada externa al PC sean suaves. Si los movimientos del joystick son bruscos el sistema no podrá seguir apropiadamente estas consignas, lo cual puede ocasionar que el robot no se mueva o que el sistema se bloquee, sin embargo esto es lo que sucede en el mundo real con un robot porta endoscopio, donde el cirujano tampoco debe realizar movimientos bruscos con el sistema de mando para evitar tensiones demasiado fuertes en el trocar y por ende en el abdomen del paciente.

En robots quirúrgicos se colocan intencionalmente sistemas que impiden el efectuar movimientos bruscos y es además la recomendación que se hace al personal médico. El sistema presentado en este trabajo sigue igualmente la misma recomendación.

CONCLUSIONES

Se diseñó una estructura paralela portable que cumple con los requerimientos de posicionamiento y orientación del endoscopio en una cirugía laparoscópica, esta estructura es diferente a la de los robots analizados que han sido concebidos para propósitos similares, ya que en su mayoría son robots tipo serie, de gran tamaño.

Se comprobó que dicha estructura permite al robot moverse de manera adecuada dentro del espacio de trabajo destinado para realizar las tareas propias de un robot porta endoscopio en cirugía laparoscópica. La estructura física diseñada consta de tres grados de libertad, está formada por dos plataformas unidas a través de tres brazos y un brazo central que sirve como apoyo al endoscopio, para que se respete el paso a través de un punto fijo que representa el trocar.

Para obtener el modelo geométrico inverso (MGI), se sigue un método vectorial y se calculan las longitudes de los brazos de acuerdo a la posición del extremo del brazo 1 unido a la plataforma. Realizando diferentes operaciones matemáticas con las ecuaciones del MGI, se encontraron las respectivas ecuaciones del modelo geométrico directo (MGD).

Por medio de derivaciones simples, se obtuvieron las matrices Jacobianas del robot, las cuales fueron utilizadas para encontrar los puntos dentro del área de trabajo a los cuales el robot no puede llegar.

Los modelos dinámicos directos (MDD) y los modelos dinámicos inversos (MDI) de los brazos que forman el robot PREH, fueron obtenidos siguiendo los métodos tradicionales aplicados robots seriales y utilizados para calcular los pares de las articulaciones activas del robot.

Analizando el ambiente de trabajo de una cirugía de extracción de vesícula biliar (colecistectomía), se lograron definir los parámetros geométricos del robot, entre ellos las dimensiones de la base, la plataforma y los distintos actuadores. Los parámetros dinámicos, como inercias y masas, se obtuvieron con el diseño de las piezas en el software CAD SolidEdge®.

Se verificó que el robot tiene la capacidad geométrica de pasar a través de un punto fijo en el espacio, mediante la comprobación de los modelos geométricos en MATLAB/Simulink® con trayectorias cartesianas predefinidas, como hacer un círculo con un radio y un centro determinado, una trayectoria lineal con un cambio brusco de dirección, y finalmente trayectorias libres generadas desde una interfaz. Para comprobar que PREH también tiene la capacidad dinámica de mantener un punto de inserción y rotación fijo en el espacio, se introdujo un esquema de control por par calculado (CTC), también implementado en Simulink/MATLAB®, y se introdujeron las mismas consignas cartesianas probadas anteriormente, obteniendo errores cartesianos del orden de los milímetros, ratificando entonces que PREH es un robot capaz de asistir una cirugía laparoscópica como robot porta endoscopio.

Para visualizar los resultados obtenidos se creó una función en Matlab, la cual permite observar los movimientos del robot en un plano de tres dimensiones, se verificó que el robot mantiene un punto fijo de inserción y rotación en el origen del plano, punto que representa la incisión en el abdomen del paciente.

Los principales problemas que se presentaron en el desarrollo del proyecto fueron la obtención de los modelos matemáticos, en especial los modelos dinámicos, ya que por ser estructuras cerradas, presentan un mayor grado de complejidad que las estructuras abiertas. Otra dificultad que se presentó fue la imposibilidad de comprobar los movimientos del robot en un ambiente de realidad virtual, debido al desconocimiento y la nula información que se encontró acerca de cómo manejar estructuras cerradas en el toolbox de Matlab llamado VirtualReality.

Como trabajos futuros se tendrían la inclusión de tres grados de libertad al endoscopio, un prismático, que le permitiría a PREH alcanzar mayores profundidades en el abdomen del paciente y dos rotacionales, que servirían para tener una mayor visualización. Por último se podría continuar con el proyecto construyendo un primer prototipo físico para realizar pruebas reales.

Bibliografía.

- [1] X. A. MENESES, M. C. MÉNDEZ, B. E. CORTÉS. “Diseño y control de un robot paralelo”, 6 Congreso Nacional de Mecatrónica, Noviembre 2007.
- [2] V. E. GOUGH, (1956-1957). "Contribution to discussion of papers on research in Automobile Stability, Control and Tyre performance". *Proc. Auto Div. Inst. Mech. Eng.:* 392–394.
- [3] D. STEWART, “A platform with six Degrees of Freedom”. Proc. Institute of Mechanical Engineering, Vol. 180, pp. 371-386.
- [4] R. CLAVEL, “Conception d'un robot parallèle rapide à 4 degrés de liberté”. Ph.D. Thesis, EPFL, Lausanne, Switzerland. 1991.
- [5] L. JOSKOWICZA, M. SHOHAMB, R. SHAMIRA, “Miniature robot-based precise targeting system for keyhole neurosurgery: Concept and preliminary results”. *International congress computer aided surgery*, pp. 181–193. Julio 2006.
- [6] T. TZUNG y H. YEH, “Development of a parallel surgical robot with automatic bone drilling carriage for stereotactic neurosurgery”. *Biomedical Engineering: Applications, Basis and Communications*. Vol. 19, No. 4, pp. 269–277. 2007.
- [7] L. BARBE, B. BAYLE, C. Doignon, J. GANGLOFF, M. MATHELIN y B. Maurin, “IRASIS: Insertion Robotisée d’Aiguille Sous Imagerie Scanner Robotized Needle Insertion with CT Guidance”. Diciembre de 2006.
- [8] M. SHOHAM, M. BURMAN, E. ZEHAVI, L. JOSKOWICZ, E. BATKILIN, y KUNICHER, “Bone-mounted miniature robot for surgical spinal procedures”. *IEEE transaction on robotics and automation*. Vol. 19. No. 5. Octubre 2003.

- [9] W. RONG, L. MA, L. SUN, Z. LI, "A 6-DOF Parallel Robot for Optical Precise Operation". 32nd Annual Conference on Volume, Noviembre 2006, pp. 4165–4169.
- [10] J. MARESCAUX Y F. RUBINO, "The Zeus robotic system: experimental and clinical applications". *Surgical Clinics of North America*. Vol. 83, pp. 1305-1315. Diciembre 2003.
- [11] G. GUTHARD, y K. SALISBURY, "The intuitive telesurgery system: overview and application". *Proceedings of the IEEE International Conference on Robotics and Automation*. 1999, pp. 618-621.
- [12] C. PREUSCHE, T. ORTMAIER y G. HIRZINGER, "Teleoperation concepts in minimal invasive surgery". *Control Engineering Practice*. Vol 10, pp. 1245–1250. Noviembre 2002.
- [13] S. SALINAS, "Modelado, Simulación en 3D y Control de un robot para cirugía Laparoscópica", Tesis de Maestría Universidad del Cauca. 2009.
- [14] M. NITZE, J. LEITER. "Electro endoskopische instrumente", Wilhelm Branmüller and son, 1880. Wien.
- [15] G. KELLING, "Verhandlungen II Teil; 2 Hälfte", Versammlung deutscher Naturforscher und Aerzte. Hamburg 23 sep.1901: 119.
- [16] M. PEREZ, "Historia de la laparoscopia urológica", Secla Endosurgery, Octubre-Diciembre de 2004.
- [17] H. C. JACOBBAUS, "Über die Möglichkeit die Zysten kopie bei Untersuchung seröser Höhlungen an zuwenden," *Münch. Med. Wschr.* 1910; 57: 2.090-2.092.
- [18] A. KATZIR, *Lasers and Optical Fibers in Medicine*. New York: Academic Press, 1993.

- [19] E. CASSONE: Cirugía Endo Laparoscópica 1: 16-20. Secretaria de posgrado de la Facultad de ciencias Medicas de la Universidad Nacional de Cuyo, 1994.
- [20] P. MOURET, "How I Developed Laparoscopic Cholecystectomy", *Annals of the Academy of Medicine*, vol. 25, no. 5, pp. 744-747, 1996.
- [21] I. JOCOBY, "Proceedings of the NIH Consensus Development Conference on Gallstones and Laparoscopic Cholecystectomy. Bethesda, Maryland, 1992", *American Journal of Surgery*, vol. 165, no. 4, pp. 387-548, 1993.
- [22] S. SCHWARTZ, J. HUNTER, *Principios de Cirugía*, 7a ed, Vol. II. México: McGrawHill Interamericana, 1999.
- [23] DIMEDA, "Página de Inicio – Dimeda Surgical Instruments". Disponible: <http://www.dimeda.de>. [Consultado: Agosto 2011].
- [24] G. JIMÉNEZ, "Colecistectomía laparoscópica cirugía de la vesícula biliar". http://www.susmedicos.com/0_Articulos_General/art_cirugia_esicula_biliar.htm. [Consultado Julio del 2011].
- [25] LatinSalud, "Hombre vs La Maquina", *Latín Salud* 2010, Disponible: <http://www.latinsaude.com/articulos/00625.asp>. [Consultado Febrero 2011].
- [26] J. ACUÑA, "Nueva era de la cirugía", Curso de cirugía laparoscópica, Universidad Del Sinu Seccional Cartagena, Enero de 2005.
- [27] B. CAREN, "Robots assist in common surgery", Disponible en: <http://www.photonics.com/Article.aspx?AID=36781>. [Consultado Marzo 2011].
- [28] J JHON, C EDWARD. "Robotics and Computers in Minimally Invasive Spine Surgery", Disponible en: <http://www.spineuniverse.com/exams->

tests/devices/robotics-computers-minimally-invasive-spine. [Consultado Marzo 2011]

- [29] S. MEJÍA, A. ARANA, C. ARANGO, V. PÉREZ, H. POSADA, Y A. TORRES, “Kirubot: brazo robótico ayudante en cirugía”, vol.14, no.4, p.254, 2001.
- [30] C. MUÑOZ, E. LASSO, C. TOSS, F. ORDÑES, A. VIVAS, “Sistema virtual para el posicionamiento del robot porta endoscopio Hibou a través de un casco” Artículo Departamento de electrónica, Instrumentación y control Universidad del Cauca. Noviembre 2011.
- [31] L. W. TSAI, (1999),”Robot Analysis”, The Mechanics of Serial And Parallel Manipulators. Wiley-Interscience.
- [32] R. ARACIL. R. J. SALTAREN. J. M. SABATER. O. REINOSO, ”Robots Paralelos: Maquinas con un pasado para una robótica del futuro”. Revista Iberoamericana de Automática e Información Industrial, Vol.3, Núm. Q, 2006.
- [33] Y. VALDEZ, “Resolución de Mecanismos paralelos Planar 3RRR impulsado por Actuadores Eléctricos”. Tesis de Maestría, CICATA-QRO IPN.
- [34] G. YANG. W. CHEN AND I. CHEN, “A geometrical method for the singularity analysis of 3 GDL RRR planar parallel robots with different actuation schemes”. Singapore Institute of Manufacturing Tecnology, Singapore 2002.
- [35] Z. SHOHAMC, M. MILGROMB, C. JOSKOWICZA, L. YANIVA y A. SIMKINB, A. “Robot-assisted system for long bone intra medullary distal locking: concept and preliminary results”. International congress computer assisted radiologic and surgery. Pp. 495-491. Marzo 2003.
- [36] Y. K. KWOH, J. HOU, E. A. JONCKHEERE, AND S. HAYATI. A robot with improved absolute positioning accuracy for stereotactic brain surgery. IEEE Transactions on Biomedical Engineering, 35(2):153–160, February1988.

- [37] D. CAMARILLO, T. KRUMMEL and K. SALISBURY, "Robotic Technology in Surgery: Past, Present, and Future", *The American Journal of Surgery*, vol. 188, pp. 2-14, 2004.
- [38] W. KORB, R. MARMULLA, J. RACZKOWSKY, J. MÜHLING and S, "Menschliche Zuverlässigkeitsanalyse (HRA) für die Anwendung von Robotersystemen in der Chirurgie." Curac, 2004.
- [39] HASSFELD, "Robots in the Operating Theatre—Chances and Challenges", *International journal of Oral and Maxillofacial Surgeons*, vol. 33, pp. 721-732, 2004.
- [40] C. ARROYO, "Cirugía Robótica", *Revista: Elementos: Ciencia y Cultura*, vol. 12, pp. 13-17, 2005.
- [41] RESEARCH INSTITUTE SRI INTERNATIONAL, "Ultrasound and Robotics Surgery Pioneer to Receive High Award from SRI International", 2003. www.sri.com/news/releases. [Consultado: abril de 2007].
- [42] INTUITIVE SURGICAL, "The Da Vinci Surgical System", 2005. www.intuitivesurgical.com/products/davinci_surgicalsistem/index.htm. [Consultado: octubre de 2006].
- [43] INTUITIVE SURGICAL, "In Vivo Accessories for Minimally Invasive Robotic Surgery", U. S. Patent 7125403B2, 2006.
- [44] COMPUTER MOTION Inc, "Method and Apparatus for Performing Minimally Invasive Surgical Procedures", U. S. Patent 6436107B1, 2002.
- [45] M. HADI, H. ZAKARIA, A. ALMULHIM, A. ALGHAMDI, H. MOWAFI and A. HUSSIEN, "Gas or Gas-Less Laparoscopic Cholecystectomy?", *Kuwait Medical Journal*, vol. 34, pp. 292-295, 2002.

- [46] NARANG MEDICAL LIMITED, "Operating Room Tables". www.narang.com/surgical_medical_hospital_laboratory_products/index.php. [Consultado: febrero de 2008].
- [47] J. MOSSO, A. MINOR, V. LARA y E. MAYA, "Brazo Robótico para Sujetar y Posicionar Laparoscopios", *Revista Cirugía y Cirujanos*, vol. 69, pp. 295-299, 2001.
- [48] MAXON, "Hoja de Datos del Motor Maxon EC-Powermax 30". www.maxonmotor.com. [Consultado: junio 2008].
- [49] FAICO, "Instrumental Quirúrgico". www.faico.com.ar/servicio.htm. [Consultado: marzo 2008].
- [50] E. DOMBRE, P. POIGNET, G. DUCHEMIN et F. PIERROT, "Quelques Problemes de Conception et Commande de Robots Medicaux", *Mécanique & Industries*, vol. 4, pp. 559–563, 2003.
- [51] R. FRANZINO, "The Laprotek Surgical System and the Next Generation of Robotics", *Surgical Clinics of North America*, vol. 83, no. 6, pp. 1317-1320, 2003.
- [52] ENDOVIA MEDICAL Inc, "Articulated Apparatus for Telemanipulator System", U. S. Patent 6692485B1, 2004.
- [53] M. MICHELIN, "Contribution a la Commande de Robots pour la Chirurgie Mini-Invasive", PhD. thesis, Université de Montpellier II, France, 2004.
- [54] R. BARLOW, "Up Close with InTouch Health's Yulun Wang: Healthcare Robotics Guru Plugs Man into Machine", *Healthcare Purchasing News*, 2005. www.highbeam.com/Healthcare+Purchasing+News/publications.aspx. Consultado: mayo de 2008.

- [55] J. MARESCAUX, J. LEROY and M. GAGNER, “Transatlantic Robot-Assisted Telesurgery”, *Nature*, vol. 413, pp. 379-380, 2001.
- [56] H. KANG and J. WEN, “Endobot: A Robotic Assistant in Minimally Invasive Surgeries”, *IEEE International Conference on Robotics and Automation*, pp. 2031-2036, Korea, 2001.
- [57] A. J. MADHANI, “Design of Teleoperated Surgical Instruments for Minimally Invasive Surgery”, PhD. thesis, Massachusetts Institute of Technology (MIT), United States, 1998.
- [58] O. SCHNEIDER, J. TROCCAZ, O. CHAVANON and D. BLIN, “Padyc: A Synergistic Robot for Cardiac Puncturing”, *2000IEEE International Conference on Robotics and Automation*, pp. 2883-2888, 2000.
- [59] P. BERKELMAN, E. BOIDARD and P. CINQUIN, “Automatic Instrument Tracking with a Compact Laparoscopic Endoscope Robot Using an External Optical Localizer”, *1st European Conference on Computer Assisted Surgery*, France, 2002.
- [60] K. CLEARY and C. NGUYEN, “State of the Art in Surgical Robotics: Clinical Applications and Technology Challenges”, *Computer Aided Surgery*, vol. 6, pp. 312-328, 2001.
- [61] K. CLEARY AND C. NGUYEN. *Medical Robotics*, pages 32.1–32.26. CRC Press LLC, 2004.
- [62] K. CLEARY AND C. NGUYEN. State of the art in surgical robotics: Clinical applications and technology challenges. *Computer Aided Surgery*, 6(6):312–328, 2001.
- [63] L.W. Tsai, *Kinematics of a Three-DOF Platform Manipulator with Three Extensible Limbs*, *Recent Advances in Robot Kinematics*, pp. 401–410, Kluwer Academic Publishers, 1996.

- [64] A. VIVAS. "Diseño y Control de Robots Industriales: Teoria y Practica". Buenos Aires: Elaleph 2010.
- [65] J.P. MERLET. "Les robots parallèles". pp. 27-28. Hermes. Paris 1997.
- [66] S. GROSSMAN. "Algebra lineal", Quinta edición. México: Mc Graw Hill, 1996, página 252.
- [67] W. Khalil y E. Dombre, *Modeling, Identification and Control of Robots*. London: Kogan Page Science, 2002. [E-Book]. Disponible: google-books.
- [68] J.P. MERLET. "Parallel Robots". pp. 120. Springer. Paris 2000.
- [69] S. GUEGAN. "Contribution a la modelisation et l'identification dynamique des robots paralleles". Université de Nantes. 2003.
- [70] FIRGELLI. "Miniature Linear Motion Series L12". [Online]. Disponible: http://firgelli.com/Uploads/L12_datasheet.pdf. [Consultado: Octubre 2012].
- [71] O. KHATIB. "A unified approach for motion and force control of robot manipulators: the operational space formulation", *IEEE J. of Robotics and Automation*, Vol. RA-3(1), pp. 43-53, février 1987.
- [72] J. MULIER, B DILLEMANS Y S. VAN CAUWENBERGE, "Impact of the patient's Body Position on The Intra-abdominal Workspace During Laparoscopic Surgery", *National Center for Biotechnology Information*, 2009. [Online]. Disponible: <http://www.ncbi.nlm.nih.gov/pmc/articles/PMC2869437/>. [Consultado: Agosto 2010].
- [73] GOSSELIN, C., ANGELES, J. (1990), Singularity analysis of closed-loop kinematic chains, Int. Conf. on Robotics and Automation, Cincinnati, USA.
- [74] W. KHALIL Y D. CREUSOT, "Symoro+: A System for the Symbolic Modelling of Robots", *Robotica*, vol. 15, pp. 153-161, 1997.

- [75] L. SCIAVICCO, Y B. SICILIANO, Modeling and control of robot manipulators 6th ed. Londres: Springer, 2005. [E-Book]. Disponible: google-books.
- [76] C. SMITH, "Fundamentals of Control Theory", *Chemical Engineering*, vol. 86, pp. 11-39. 1979.

ANEXOS

Anexo A. Programas implementados en Matlab.

A.1. Función MGI.m

Esta función calcula las longitudes de los actuadores en función de la posición del centro de la plataforma, para este cálculo es necesario definir los radios de la base y la plataforma del robot.

```
function Q =MGI(PX,PY,PZ)
    rb=0.06;%radio de la base
    rp=0.05;%radio de la plataforma
    %Se definen las longitudes de los actuadores
    Q31=sqrt(PX^2+PY^2+PZ^2+2*(rb-rp)*(PX*cosd(30)+PY*sind(30))+(rb-rp)^2);
    Q32=sqrt(PX^2+PY^2+PZ^2+2*(rb-rp)*(PY*sind(30)-PX*cosd(30))+(rb-rp)^2);
    Q33=sqrt(PY^2+PZ^2-2*(rb-rp)*PY+(rb-rp)^2);
    Q=[Q31 Q32 Q33]; %Retorna Q
```

A.2. Función MGD.m

Esta función calcula la posición del centro de la plataforma en función de las longitudes de los actuadores, para este cálculo es necesario definir los radios de la base y la plataforma del robot.

```
function P=MGD(Q31,Q32,Q33)
    rb=0.06;%radio de la base
    rp=0.05;%radio de la plataforma
    %Se define la posición del centro de la plataforma
    PX=(Q31^2-Q32^2)/(4*(rb-rp)*cosd(30));
```



```

PY=((Q31^2-Q33^2-PX^2-2*(rb-rp)*PX*cosd(30))/(2*(rb-rp)*sind(30)+2*(rb-rp)));
PZ=sqrt(Q33^2-PY^2+2*(rb-rp)*PY-(rb-rp)^2);
P= [PX PY PZ]; %Retorna P

```

A.3. Función visualización.m.

Esta función grafica el robot a partir del estado de los actuadores. Como primera medida se obtiene la posición del centro de la plataforma y el extremo del endoscopio, seguidamente se definen los puntos de conexión de cada brazo a la base y a la plataforma, estos últimos dependen del centro de la plataforma, cuyas coordenadas ya se han obtenido, finalmente se grafica cada brazo, incluido el endoscopio, para una mejor apreciación se diferencia entre la parte del endoscopio que está fuera del abdomen (color azul) y la que se encuentra inmersa (color amarillo).

```

function visualizacion(Q31,Q32,Q33)
    %Esta función grafica el robot, a partir del estado de los actuadores
    %Se obtiene la posición del centro de la plataforma
    P=MGD(Q31,Q32,Q33);
    x=P(1);
    y=P(2);
    z=P(3);

    %Se calcula la posición de la punta del endoscopio
    L1=sqrt(x^2+y^2+z^2);
    L2=0.15-L1;
    Xc=-L2*x/L1;
    Yc=-L2*y/L1;
    Zc=-L2*z/L1;

    %Se definen vectores necesarios para realizar la grafica
    piernas5 = zeros(2,3);
    piernas4 = zeros(2,3);
    piernas1 = zeros(2,3);

```

```
piernas2 = zeros(2,3);
piernas3 = zeros(2,3);
base1= zeros(2,3);
base2= zeros(2,3);
base3= zeros(2,3);
plataforma1= zeros(2,3);
plataforma2= zeros(2,3);
plataforma3= zeros(2,3);
```

%Los vectores base contienen la información de las coordenadas de los puntos A1, A2, A3 (puntos de conexión de los brazos a la base).

```
base1(1,1)=0.02598;
base1(1,2)=-0.015;
base1(2,1)=-0.02598;
base1(2,2)=-0.015;
```

```
base2(1,1)=0.02598;
base2(1,2)=-0.015;
base2(2,1)=0;
base2(2,2)=0.03;
```

```
base3(1,1)=-0.02598;
base3(1,2)=-0.015;
base3(2,1)=0;
base3(2,2)=0.03;
```

%Los vectores plataforma contienen la información de las coordenadas de los puntos P1, P2, P3 (puntos de conexión de los brazos a la plataforma)

```
plataforma1(1,1)=x+0.02165;
plataforma1(1,2)=y-0.0125;
plataforma1(1,3)=z;
plataforma1(2,1)=x-0.02165;
```

```
plataforma1(2,2)=y-0.0125;  
plataforma1(2,3)=z;
```

```
plataforma2(1,1)=x+0.02165;  
plataforma2(1,2)=y-0.0125;  
plataforma2(1,3)=z;  
plataforma2(2,1)=x;  
plataforma2(2,2)=y+0.025;  
plataforma2(2,3)=z;
```

```
plataforma3(1,1)=x-0.02165;  
plataforma3(1,2)=y-0.0125;  
plataforma3(1,3)=z;  
plataforma3(2,1)=x;  
plataforma3(2,2)=y+0.025;  
plataforma3(2,3)=z;
```

%Los vectores piernas son necesarios para graficar los brazos del robot

```
piernas4(2,1)=x;  
piernas4(2,2)=y;  
piernas4(2,3)=z;
```

```
piernas5(2,1)=Xc;  
piernas5(2,2)=Yc;  
piernas5(2,3)=Zc;
```

```
piernas1(1,1)=0.02598;  
piernas1(1,2)=-0.015;  
piernas1(2,1)=x+0.02165;  
piernas1(2,2)=y-0.0125;  
piernas1(2,3)=z;
```

```
piernas2(1,1)=-0.02598;  
piernas2(1,2)=-0.015;
```

```
piernas2(2,1)=x-0.02165;  
piernas2(2,2)=y-0.0125;  
piernas2(2,3)=z;
```

```
piernas3(1,2)=0.03;  
piernas3(2,1)=x;  
piernas3(2,2)=y+0.025;  
piernas3(2,3)=z;
```

%Se grafican los brazos del robot

```
plot3(piernas4(:,1),piernas4(:,2),piernas4(:,3),'b','linewidth',3);  
hold on  
plot3(piernas5(:,1),piernas5(:,2),piernas5(:,3),'y','linewidth',3);  
plot3(piernas1(:,1),piernas1(:,2),piernas1(:,3),'b','linewidth',3);  
plot3(piernas2(:,1),piernas2(:,2),piernas2(:,3),'b','linewidth',3);  
plot3(piernas3(:,1),piernas3(:,2),piernas3(:,3),'b','linewidth',3);
```

%Se grafica la base y la plataforma del robot, en forma triangular

```
plot3(base1(:,1),base1(:,2),base1(:,3),'r','linewidth',3);  
plot3(base2(:,1),base2(:,2),base2(:,3),'r','linewidth',3);  
plot3(base3(:,1),base3(:,2),base3(:,3),'r','linewidth',3);  
plot3(plataforma1(:,1),plataforma1(:,2),plataforma1(:,3),'r','linewidth',3);  
plot3(plataforma2(:,1),plataforma2(:,2),plataforma2(:,3),'r','linewidth',3);  
plot3(plataforma3(:,1),plataforma3(:,2),plataforma3(:,3),'r','linewidth',3);
```

%Se establecen las características del gráfico.

```
ejex = 0.10;  
ejez = 0.20;  
axis([-ejex ejex -ejex ejex -ejez/2 ejez]);  
xlabel('X');  
ylabel('Y');  
zlabel('Z');  
daspect([1 1 1]);  
grid on;
```

```
hold off
drawnow;
return
```

A.4. Programa circular.m.

Este programa se utiliza para calcular las coordenadas de una trayectoria circular.

```
%----Movimiento circular-----
%-----
% Tiempo de muestreo y duración final de la trayectoria:
Tfinal=3.0;
Tem=0.01;
% Cálculo del número de muestras:
nbech=(Tfinal/Tem)+1;
if ((round(nbech)-nbech) == 0)
    instant=[0:Tem:Tfinal]';
else
    nbech=nbech+1;
    instant=[0:Tem:Tfinal+Tem]';
end
t=0;
for h=1:1:nbech
    t=t+Tem;
    x1(h)=0.05*sin(2*pi*1/Tfinal*t);
    y1(h)=0.05*cos(2*pi*1/Tfinal*t);
end
x1=x1';
y1=y1';
%-----
cons1= 0 + x1;
cons2= 0 + y1;
cons3=0.08*ones(301,1);
```

A.5. Programa lineal.m.

```
%----Movimiento lineal-----  
%-----  
Tfinal=3.0;  
%Tfinal=1.0;%para cambiar la velocidad de la trayectoria  
Tem=0.01;  
% Calculo del numero de muestras %  
nbech=(Tfinal/Tem)+1;  
if ((round(nbech)-nbech) == 0)  
    instant=[0:Tem:Tfinal]';  
else  
    nbech=nbech+1;  
    instant=[0:Tem:Tfinal+Tem]';  
end  
instant = instant(1:300,:);  
%instant = instant(1:1000,:);% aquí también se cambia para cambiar la velocidad  
de la trayectoria.  
t=0;  
for h=1:1:150  
    t=t+Tem;  
    x1(h)=t;  
    y1(h)=t;  
end  
t=1.5;  
for h=1:1:150  
%for h=1:1:500%para cambiar velocidad  
    t=t+Tem;  
    x2(h)=-t + 3.0;  
    %x2(h)=-t + 1.0;%para cambiar velocidad  
    y2(h)= t;  
end  
xx = [x1 x2];
```

```

yy = [y1 y2];
xx = xx';
yy = yy';
%-----
cons1= 0.01*xx;
cons2= 0.01*yy;
cons3 = 0.08*ones(300,1);

```

A.6. Programa paso_incisión.m.

Esta función grafica el endoscopio para todos los puntos de las trayectorias definidas anteriormente. Como primera medida se definen las coordenadas de la trayectoria, circular o lineal, luego se crea un ciclo donde se calculan las posiciones del centro de la plataforma y del extremo del endoscopio, con estas coordenadas se grafica la posición del endoscopio, el ciclo se repite para todos los puntos que forman la trayectoria, seguido a esto se marcan las coordenadas del centro de la plataforma y del extremo del endoscopio en la grafica y finalmente para una mejor visualización, se establecen las características del grafico.

```

clear all
clc
piernas=zeros (2,3); %vector necesario para graficar el endoscopio
circular; %lineal; se escoge cualquiera de las dos trayectorias.
zd=cons3;
xd=cons1;
yd=cons2;
for i=1:150 %cuando se selecciona la trayectoria circular, este valor va hasta 301.
    Trayectoria lineal 150
    Q=MGI (xd (i),yd (i),zd (i));
    xc (i)=P (1);
    yc (i)=P (2);
    zc (i)=P (3);
    %Se obtienen las coordenadas del extremo del endoscopio.
    L1 (i)=sqrt (xc (i)^2+yc (i)^2+zc (i)^2);

```

```

L2 (i)=0.15-L1 (i);
x (i)=-L2(i)*xc (i)/L1(i);
y (i)=-L2(i)*yc (i)/L1(i);
z (i)=-L2(i)*zc (i)/L1(i);
%Se llenan los vectores piernas, con las coordenadas obtenidas.
piernas (1,1)=xc (i);
piernas (1,2)=yc (i);
piernas (1,3)=zc (i);
piernas (2,1)=x (i);
piernas (2,2)=y (i);
piernas (2,3)=z (i);
% Se grafican las posiciones del endoscopio durante toda la trayectoria.
plot3 (piernas (:,1),piernas (:,2),piernas (:,3),'g');
hold on;
piernas (1,2)=-yc(i);
piernas (2,2)=-y(i);
plot3 (piernas (:,1),piernas (:,2),piernas (:,3),'g');
hold on;
end
% Se marcan las coordenadas del centro de la plataforma y el extremo del
laparoscopio durante toda la trayectoria.
plot3 (xc,yc,zc,'-sb','MarkerEdgeColor','k','MarkerFaceColor','b','MarkerSize',2)
hold on
plot3 (xc,-yc,zc,'-sb','MarkerEdgeColor','k','MarkerFaceColor','b','MarkerSize',2)
hold on
plot3 (x,y,z,'-sb','MarkerEdgeColor','k','MarkerFaceColor','b','MarkerSize',2)
hold on
plot3 (x,-y,z,'-sb','MarkerEdgeColor','k','MarkerFaceColor','b','MarkerSize',2)
%Se establecen las características del grafico
xlabel ('X');
ylabel ('Y');
zlabel ('Z');
grid on
ejex = 0.015;

```



```
ejez = 0.10; %
axis ([-ejez ejez -ejez ejez -ejez ejez]);
```

A.7. Programa espaciot.m.

En este programa se suponen una serie de coordenadas que podría alcanzar la plataforma y con las restricciones dadas por las longitudes de los actuadores se dibujan las posiciones a las que verdaderamente llega la plataforma.

```
for z=7:0.5:15
    for y=-15:0.5:15
        for x=-15:0.5:15
            Q=MGI(x,y,z)
            l1=sqrt(x^2+y^2+z^2);
            l2=15-l1;
            x1=-x*l2/l1;
            y1=-y*l2/l1;
            z1=-z*l2/l1;
            if((Q(1)>15)||((Q(2)>15)||((Q(3)>15)||((Q(1)<7)||((Q(2)<7)||((Q(3)<7)))
                a=1;
            else
                %plot3(x,y,z,'r')
                plot(x,z,'r')
                hold on
                plot(x1,z1,'b')
                %hold on
                %plot3(x1,y1,z1,'b')
            end
        end
    end
end
xlabel('X')
ylabel('Z')
```

```

xlabel('u')
grid on
h = legend('Plataforma','Camara',2);
set(h,'Interpreter','none')

```

A.8. Funciones para los modelos dinámicos inversos.

A.8.1. Función mdi_f

Esta función calcula el modelo dinámico inverso del brazo 1, sus parámetros son las posiciones, velocidades y aceleraciones articulares del brazo 1 y retorna los pares de cada articulación que forman el brazo 1.

```

function GAM1=mdi_f(Q11,Q21,Q31,QP11,QP21,QP31,QDP11,QDP21,QDP31)
p_iner; %Se definen los parámetros inerciales de base.

```

%Se determina el vector de gravedad del brazo 1

```

Q1=[(981*MX2*cosd(Q11)*cosd(Q21))/100 + (981*MZ3*cosd(Q11)*sind(Q21))/100
    - (981*M3*Q31*cosd(Q11)*sind(Q21))/100;

```

```

    (981*MZ3*cosd(Q21)*sind(Q11))/100 - (981*MX2*sind(Q11)*sind(Q21))/100 -
    (981*M3*Q31*cosd(Q21)*sind(Q11))/100;

```

```

    -(981*M3*sind(Q11)*sind(Q21))/100];

```

%Se obtiene la matriz de inercia del brazo 1.

```

A1=[ZZ1R-
    2*MZ3*Q31*(sind(Q21))^2+XX2R*(sind(Q21))^2+M3*Q31^2*(sind(Q21))^2    0
    0;

```

```

    0 ZZ2R-2*MZ3*Q31+M3*Q31^2 0;

```

```

    0 0 M3+IA3];

```

%Se calculan los pares del brazo 1

```
GAM1=A1*[QDP11;QDP21;QDP31]+Q1;
```

A.8.2. Función mdi_f2

Esta función calcula el modelo dinámico inverso del brazo 2, sus parámetros son las posiciones, velocidades y aceleraciones articulares del brazo 2 y retorna los pares de cada articulación que forman el brazo 2.

```
function GAM2=mdi_f2(Q12,Q22,Q32,QP12,QP22,QP32,QDP12,QDP22,QDP32)
```

p_iner; %Se definen los parámetros inerciales de base.

%Se determina el vector de gravedad del brazo 2.

```
Q2=[(981*Mx2*cosd(Q12)*cosd(Q22))/100 + (981*Mz3*cosd(Q12)*sind(Q22))/100  
+(981*M3*Q32*cosd(Q12)*sind(Q22))/100;
```

```
(981*Mz3*cosd(Q22)*sind(Q12))/100 - (981*Mx2*sind(Q12)*sind(Q22))/100  
+ (981*M3*Q32*cosd(Q22)*sind(Q12))/100;
```

```
(981*M3*sind(Q12)*sind(Q22))/100];
```

%Se obtiene la matriz de inercia del brazo 2.

```
A2=[ZZ1R+2*Mz3*Q32*(sind(Q22))^2+XX2R*(sind(Q22))^2+M3*Q32^2*(sind(Q22)  
))^2 0 0;
```

```
0 ZZ2R+2*Mz3*Q32+M3*Q32^2 0;
```

```
0 0 M3+IA3];
```

%Se calculan los pares del brazo 2.

GAM2=A2*[QDP12;QDP22;QDP32]+Q2;

A.8.3. Función mdi_f3

Esta función calcula el modelo dinámico inverso del brazo 3, sus parámetros son las posiciones, velocidades y aceleraciones articulares del brazo 3 y retorna los pares de cada articulación que forman el brazo 3.

`function` GAM3=mdi_f3(Q13,Q23,Q33,QP13,QP23,QP33,QDP13,QDP23,QDP33)
p_iner; %Se definen los parámetros inerciales de base.

%Se obtiene la matriz de inercia del brazo 2.

A3=[(MZ3*Q33*(sind(Q23))^2)+sqrt(3)*MZ3*Q33*(sind(Q23))*(cosd(Q23))+((M3*Q33^2*(sind(Q23))^2)/4)+((sqrt(3)*M3*Q33^2*(sind(Q23))*(cosd(Q23)))/2)+((3*M3*Q33^2*(cosd(Q23))^2)/4)+ZZ1R+XX2R*(sind(Q23))^2 0 0;

0 ((3*M3*Q33^2)/4)+MZ3*Q33+ZZ2R+((M3*Q33^2)/4) sqrt(3)*M3*Q33;

0 sqrt(3)*M3*Q33 M3+IA3];

%Se determina el vector de gravedad del brazo 3.

Q3=[M3*((981*Q33*cosd(Q13)*sind(Q23))/200+(981*3^(1/2)*Q33*cosd(Q13)*cosd(Q23))/200)+(981*MX2*cosd(Q13)*cosd(Q23))/100+(981*MZ3*cosd(Q13)*sind(Q23))/100;

M3*((981*Q33*cosd(Q23)*sind(Q13))/200-981*3^(1/2)*Q33*sind(Q13)*sind(Q23))/200)+(981*MZ3*cosd(Q23)*sind(Q13))/100 - (981*MX2*sind(Q13)*sind(Q23))/100;

M3*((981*sind(Q13)*sind(Q23))/200+(981*3^(1/2)*cosd(Q23)*sind(Q13))/200)];

%Se calculan los pares del brazo 3.

GAM3=A3*[QDP13;QDP23;QDP33]+Q3;

A.9. Funciones para los modelos dinámicos directos.

A.9.1. Función mdd_f

Esta función calcula el modelo dinámico directo del brazo 1, sus parámetros son las posiciones, velocidades y pares de las articulaciones del brazo 1 y retorna las aceleraciones de cada articulación que forman el brazo 1.

```
function QDP1=mdd_f(Q11,Q21,Q31,QP11,QP21,QP31,GAM11,GAM21,GAM31)
p_iner; %Se definen los parámetros inerciales de base.
```

```
%Se determina el vector de gravedad del brazo 1.
```

```
Q1=[(981*MZ3*cosd(Q11)*cosd(Q21))/100 + (981*MZ3*cosd(Q11)*sind(Q21))/100
    - (981*M3*Q31*cosd(Q11)*sind(Q21))/100;
```

```
    (981*MZ3*cosd(Q21)*sind(Q11))/100 - (981*MZ3*sind(Q11)*sind(Q21))/100 -
    (981*M3*Q31*cosd(Q21)*sind(Q11))/100;
```

```
    -(981*M3*sind(Q11)*sind(Q21))/100];
```

```
%Se obtiene la matriz de inercia del brazo 1.
```

```
A1=[ZZ1R-
    2*MZ3*Q31*(sind(Q21))^2+XX2R*(sind(Q21))^2+M3*Q31^2*(sind(Q21))^2    0
    0;
    0 ZZ2R-2*MZ3*Q31+M3*Q31^2 0;
    0 0 M3+IA3];
```

```
A11=inv(A1); %Se calcula la inversa de la matriz de inercia.
```

```
%Se calculan las aceleraciones de las articulaciones del brazo 1.
```

QDP1=AI1*([GAM11;GAM21;GAM31]-Q1);

A.9.2. Función mdd_f2

Esta función calcula el modelo dinámico directo del brazo 2, sus parámetros son las posiciones, velocidades y pares de las articulaciones del brazo 2 y retorna las aceleraciones de cada articulación que forman el brazo 2.

```
function QDP2=mdd_f2(Q12,Q22,Q32,QP12,QP22,QP32,GAM12,GAM22,GAM32)
```

```
p_iner; %Se definen los parámetros inerciales de base.
```

```
%Se determina el vector de gravedad del brazo 2.
```

```
Q2=[(981*Mx2*cosd(Q12)*cosd(Q22))/100+(981*Mz3*cosd(Q12)*sind(Q22))/100  
+(981*M3*Q32*cosd(Q12)*sind(Q22))/100;
```

```
(981*Mz3*cosd(Q22)*sind(Q12))/100 - (981*Mx2*sind(Q12)*sind(Q22))/100  
+ (981*M3*Q32*cosd(Q22)*sind(Q12))/100;
```

```
(981*M3*sind(Q12)*sind(Q22))/100];
```

```
%Se obtiene la matriz de inercia del brazo 2.
```

```
A2=[ZZ1R+2*Mz3*Q32*(sind(Q22))^2+XX2R*(sind(Q22))^2+M3*Q32^2*(sind(Q22)  
)^2 0 0;  
0 ZZ2R+2*Mz3*Q32+M3*Q32^2 0;  
0 0 M3+IA3];
```

```
AI2=inv(A2); %Se calcula la inversa de la matriz de inercia.
```

```
%Se calculan las aceleraciones de las articulaciones del brazo 2.
```

QDP2=AI2*([GAM12;GAM22;GAM32]-Q2);

A.9.3. Función mdd_f3.

Esta función calcula el modelo dinámico directo del brazo 3, sus parámetros son las posiciones, velocidades y pares de las articulaciones del brazo 32 y retorna las aceleraciones de cada articulación que forman el brazo 3.

```
function QDP3=mdd_f3(Q13,Q23,Q33,QP13,QP23,QP33,GAM13,GAM23,GAM33)
```

```
p_iner; %Se definen los parámetros inerciales de base.
```

```
%Se obtiene la matriz de inercia del brazo 2.
```

```
A3=[(MZ3*Q33*(sind(Q23))^2)+sqrt(3)*MZ3*Q33*(sind(Q23))*(cosd(Q23))+((M3*Q33^2*(sind(Q23))^2)/4)+((sqrt(3)*M3*Q33^2*(sind(Q23))*(cosd(Q23)))/2)+((3*M3*Q33^2*(cosd(Q23))^2)/4)+ZZ1R+XX2R*(sind(Q23))^2 0 0;
```

```
0 ((3*M3*Q33^2)/4)+MZ3*Q33+ZZ2R+((M3*Q33^2)/4) sqrt(3)*M3*Q33;
```

```
0 sqrt(3)*M3*Q33 M3+IA3];
```

```
AI3=inv(A3); %Se calcula la inversa de la matriz de inercia.
```

```
%Se determina el vector de gravedad del brazo 2.
```

```
Q3=[M3*((981*Q33*cosd(Q13)*sind(Q23))/200+(981*3^(1/2)*Q33*cosd(Q13)*cosd(Q23))/200)+(981*MX2*cosd(Q13)*cosd(Q23))/100+(981*MZ3*cosd(Q13)*sind(Q23))/100;
```

```
M3*((981*Q33*cosd(Q23)*sind(Q13))/200-(981*3^(1/2)*Q33*sind(Q13)*sind(Q23))/200)+(981*MZ3*cosd(Q23)*sind(Q13))/100 - (981*MX2*sind(Q13)*sind(Q23))/100;
```

$M3*((981*\sin(Q13)*\sin(Q23))/200+(981*3^{1/2}*\cos(Q23)*\sin(Q13))/200)];$

%Se calculan las aceleraciones de las articulaciones del brazo 2.

QDP3=A13*([GAM13;GAM23;GAM33]-Q3);

A.10. Programa p_iner.

Se definen los parámetros inerciales de base.

xx1=0.08;xx2=8.19;xx3=1225.8;xxp=378.46;
yy1=0.04;yy2=1268.42;yy3=1225.59;ryp=378.46;
zz1=0.04;zz2=1268.21;zz3=1.17;zrp=746;
xy1=0;xy2=0;xy3=0;xyp=0;
xz1=0;xz2=0;xz3=0;xzp=0;
yz1=0;yz2=0;yz3=0;yzp=0;
M1=0.00012;
M2=0.007;
M3=0.004;
X1=0;X2=0.03428;X3=0;
Y1=0; Y2=0;Y3=0;
Z1=0; Z2=0;Z3=0.04812;

MX1=M1*X1;MX2=M2*X2;MX3=M3*X3;
MY1=M1*Y1;MY2=M2*Y2;MY3=M3*Y3;
MZ1=M1*Z1;MZ2=M2*Z2;MZ3=M3*Z3;

ZZ1R=yy2+zz1+zz3;
MY1R=MY1-MZ2;
XX2R=xx2+xx3-yy2-zz3;
XY2R=xy2-xz2;
XZ2R=xy3+xz2;
YZ2R=yz2-yz3;
ZZ2R=yy3+zz2;


```
IA3=0.0000278;
```

A.11. Programa Inicio.m

```
%INICIAL
```

```
clear all;
```

```
close all;
```

```
clc;
```

```
Tem=0.01;
```

```
circular;
```

```
%linealre;
```

```
%QI=MGI(cons1(1),cons2(1),cons3(1));
```

```
QI=MGI(0,0,0.08);
```

```
Qi1=[QI(1) QI(2) QI(3)];
```

```
QPi1=[0 0 0];
```

```
QPi2=[0 0 0];
```

```
QPi3=[0 0 0];
```

```
Qi2=[QI(4) QI(5) QI(6)];
```

```
Qi3=[QI(7) QI(8) QI(9)];
```

```
kp1=30000;
```

```
kp2=30000;
```

```
kp3=30000;
```

```
kv1=200;
```

```
kv2=200;
```

```
kv3=200;
```

Anexo B. Simulación 3D

Al realizar el trabajo de simulación 3D del robot paralelo en V-RealmBuilder, se logró un avance significativo, pero no se pudo completar la simulación, ya que al momento de ensamblar la plataforma fue imposible lograrlo, en términos de V-RealmBuilder, no fue posible lograr que un hijo tenga 3 padres, por lo que se decidió cambiar, a OGRE obteniendo el mismo resultado. En los Anexos C y D se muestran paso a paso como se exportaron las piezas de Solid Edge a V-RealmBuilder y Ogre, respectivamente.

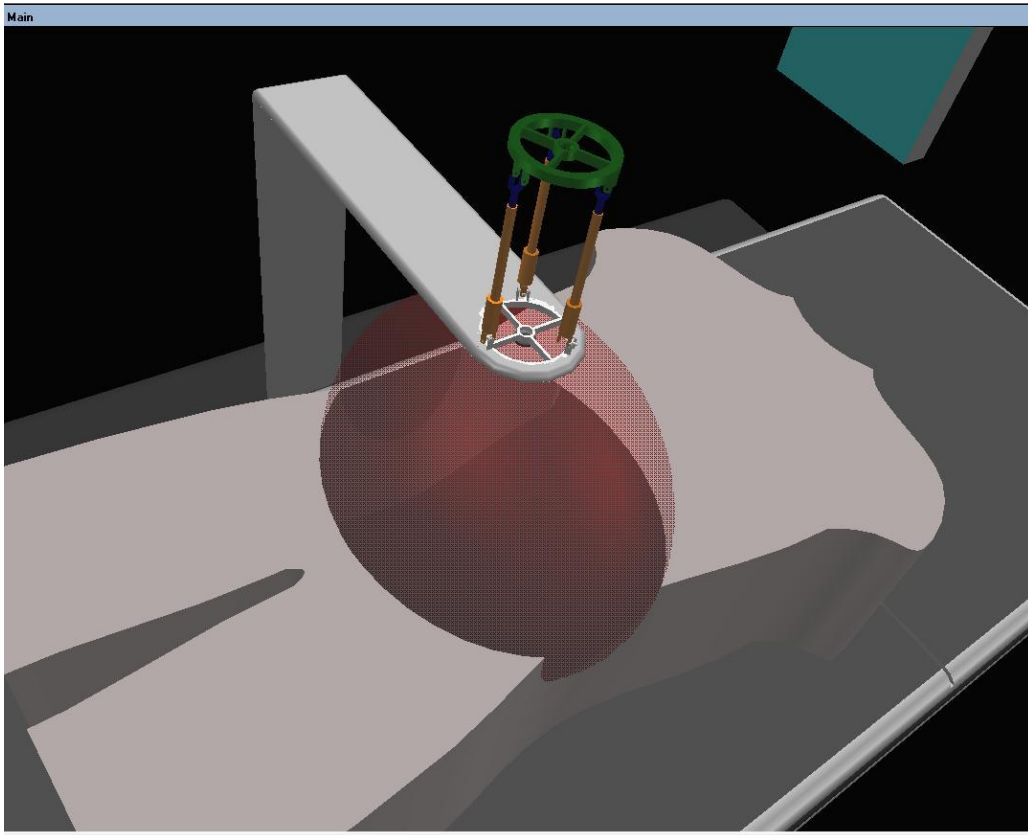


Figura B-1: Robot paralelo en V-RealmBuilder.

Es fácil darse cuenta en la gráfica, que no se pudo obtener un perfecto acoplamiento de la plataforma a los brazos, ya que genera un error, por lo que no se pudo seguir trabajando en esta herramienta.

A continuación se mostrará detalladamente cómo se realizó el ensamble de cada una de las piezas en V-RealmBuilder

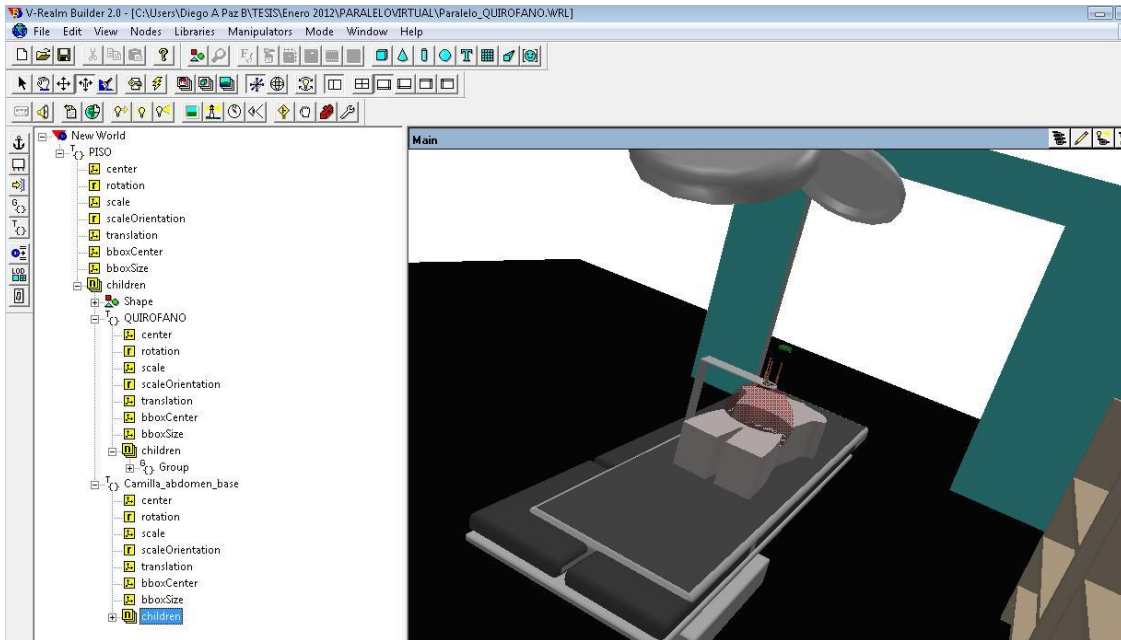


Figura B-2: Árbol de hijos (Izquierda) y padres en V-RealmBuilder (Derecha). (Piso, Camilla, base)

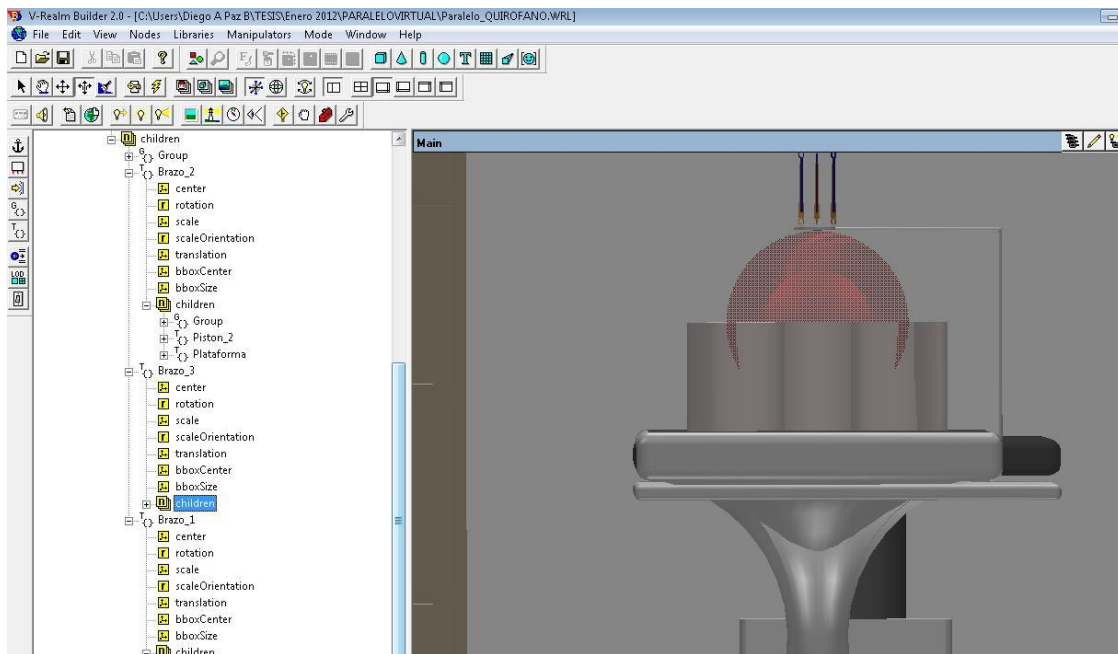


Figura B-3: Árbol de hijos y padres en V-RealmBuilder. (Brazos, pistones).

La figura B.2 muestra como se realizaron los ensambles cuando ya se tenía el piso puesto. En primera lugar se agregó, como hijo del piso al quirófano, posteriormente, como hijo del piso, la camilla, abdomen y base. Y como hijo de la camilla, cada uno de los brazos del robot, luego como hijo de cada uno de los brazos, sus respectivos, pistones. Por último se agregó como hijo de uno de los pistones la plataforma y en este caso el ensamble se logra. El problema radica cuando queremos que la plataforma sea hijo de otros pistones pues la herramienta V-RealmBuilder no lo permite siendo esto lo que necesitamos.

Anexo C. Pazos para trabajar en Virtual Reality, con piezas hechas en Solid Edge.

Como primera medida se deben tener las piezas del robot en Solid Edge o cualquier programa CAD.

Se graban dichas piezas en una extensión llamada wrl, como se muestra en la figura C.1.

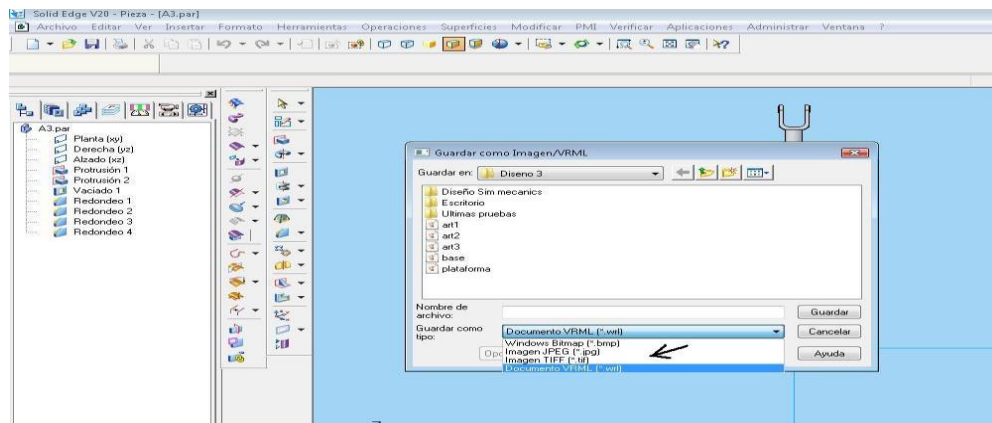


Figura C-1: Grabar piezas en CAD a VRML.

Luego se abre Simulink en MATLAB, posteriormente, se busca el Toolbox Simulink 3D Animation, en él se busca el cuadro de VR sink, y se inserta en un archivo de Simulink en blanco, ya realizado este paso, se procede dar doble clic al bloque insertado para que abra el cuadro de parámetros "Parameters: VR Sink".

En el recuadro Parameters: VR Sink, se da clic en el botón New, se abre una nueva ventana con el programa V-Realm Builder 2.0, como se ve en la figura C.2.

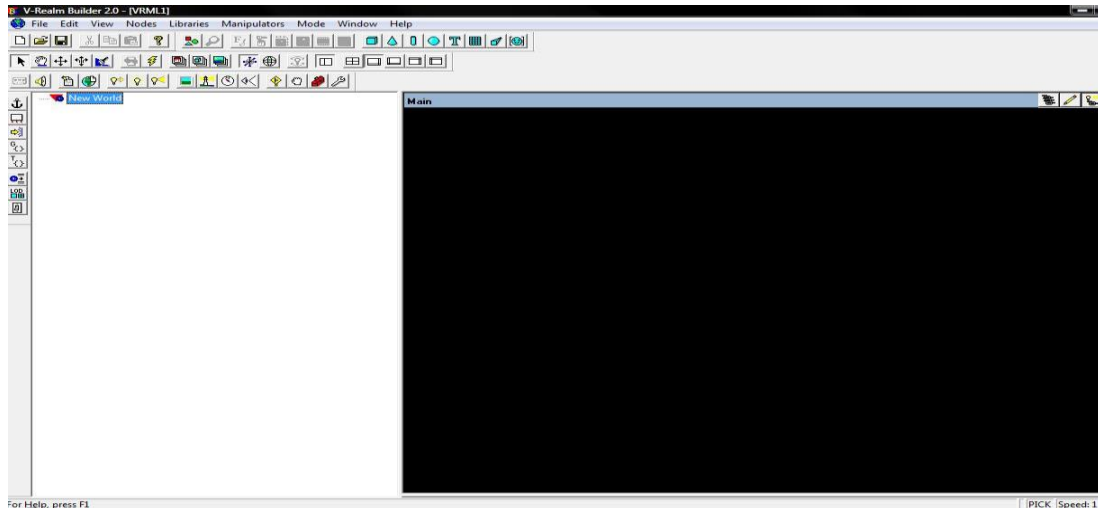


Figura C-2:V-Realm Builder.

Para empezar a crear el entorno virtual se debe crear el piso, este es muy importante ya que a partir de él se desarrollará todo el entorno, se le da clic, como muestra la figura C.3, en el lugar que indica la flecha.

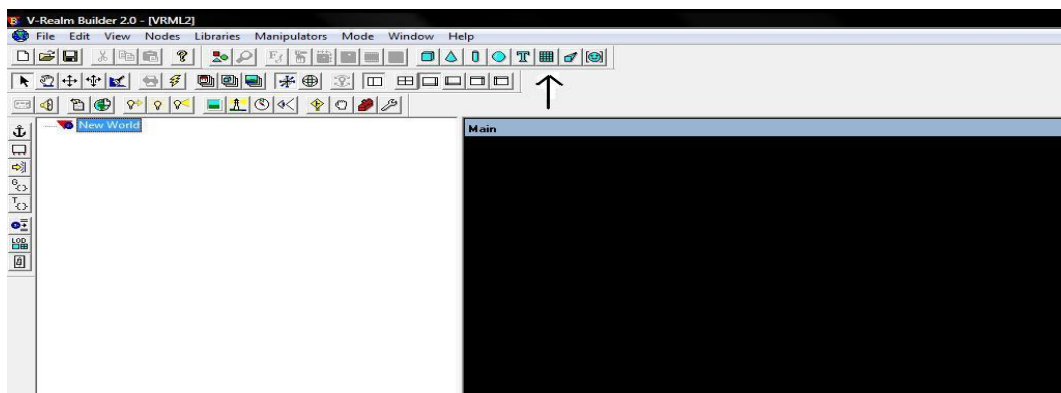


Figura C-3: Creando piso en V-Realm Builder.

Luego en el lado izquierdo, aparece la siguiente información, y en el derecho el piso, ver figura C.4.

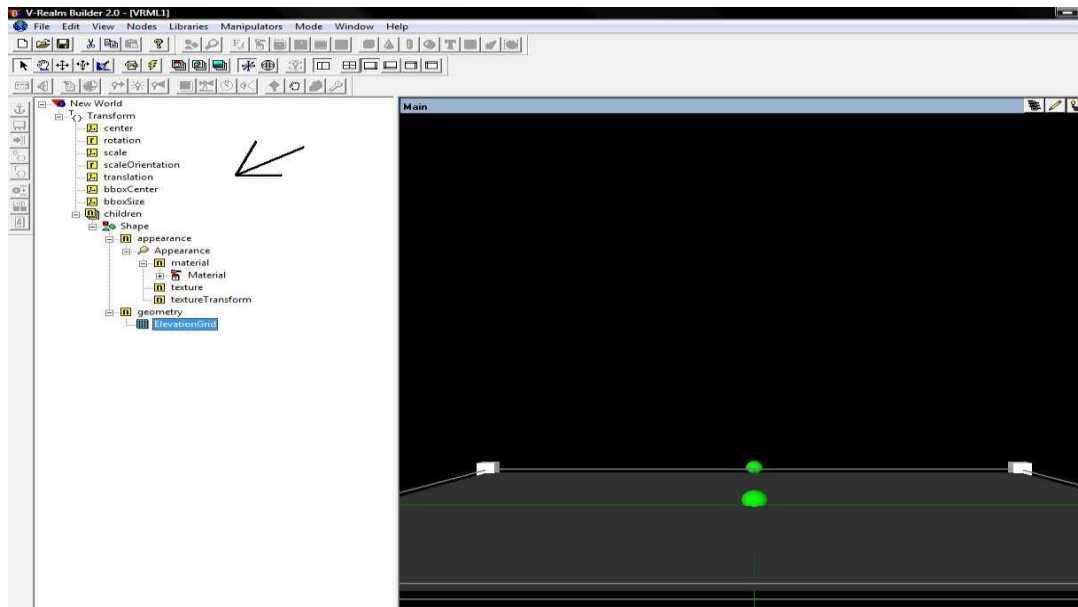


Figura C-4: Piso e información del mismo en V-Realm Builder.

Luego se procede a abrir las piezas guardadas en VRML, como se ve en la figura C.5.

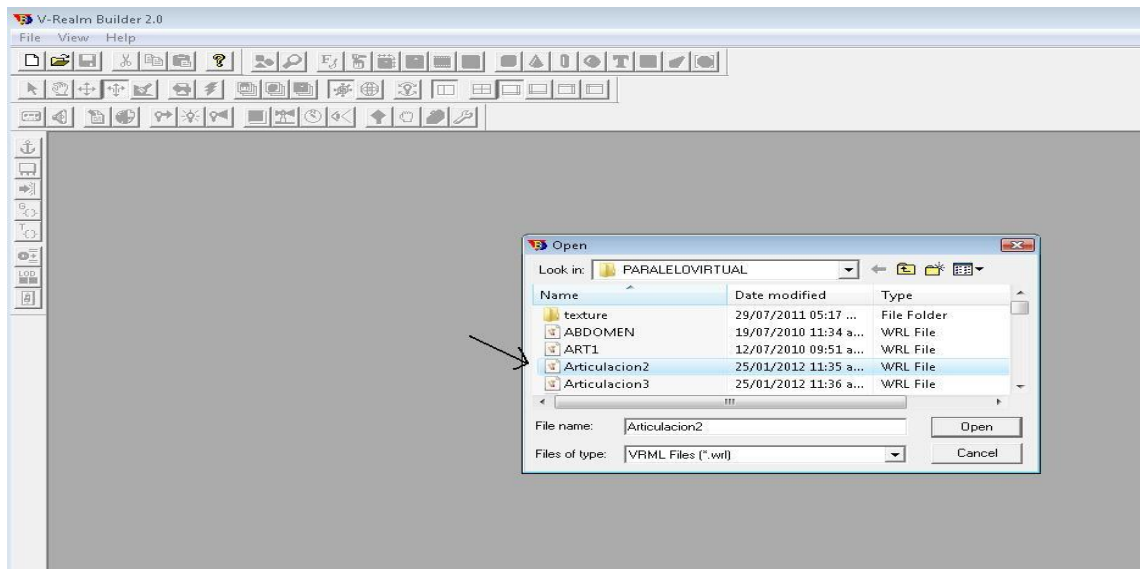


Figura C-5: Abriendo piezas guardadas en VRML.

Una vez abierta la pieza se le da copiar en el recuadro señalado Grup ver figura C.6, es decir se copia todo el árbol de propiedades de la imagen.

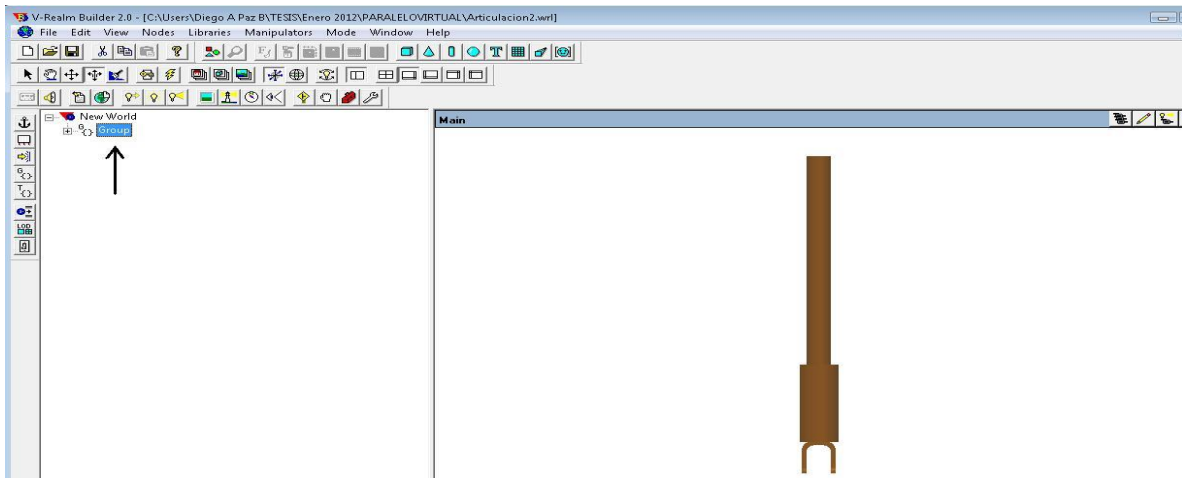


Figura C-6: Copiando el grupo.

Se regresa al archivo donde se inició el proyecto y se crea una nueva transformación y dentro de ella se pega el grupo que se copió en la imagen anterior, ver figura C.7.

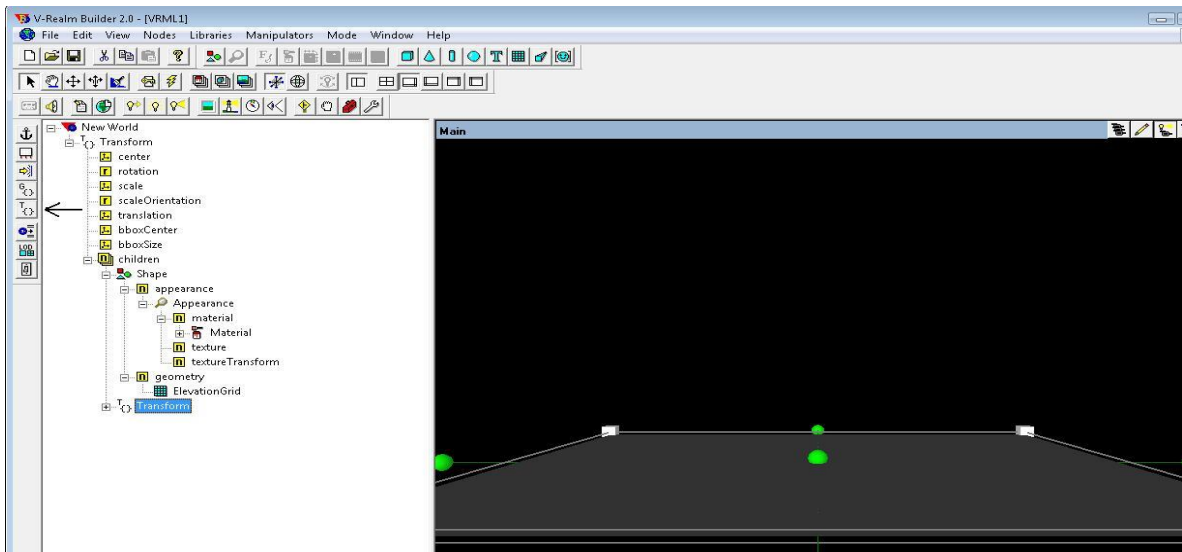


Figura C-7: Transformación creada

Realizado lo anterior se procede a copiar en la parte que dice children, el grupo figura C.8, donde se nota que se ha agregado una pieza nueva al proyecto.

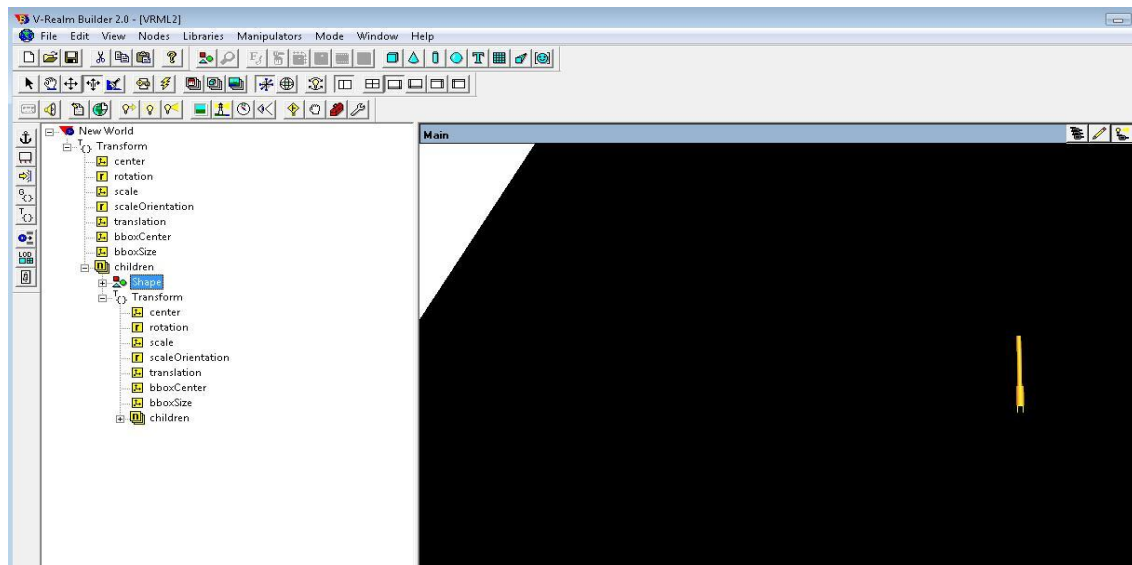


Figura C-8: Grupo (Pieza) copiado en el proyecto.

El mismo procedimiento se sigue para insertar todas las piezas que forman el robot, es necesario trasladar y rotar las piezas para que queden bien ensambladas y en el lugar indicado.

A continuación se muestran diferentes imágenes del avance obtenido en V-Realm Builder 2.0

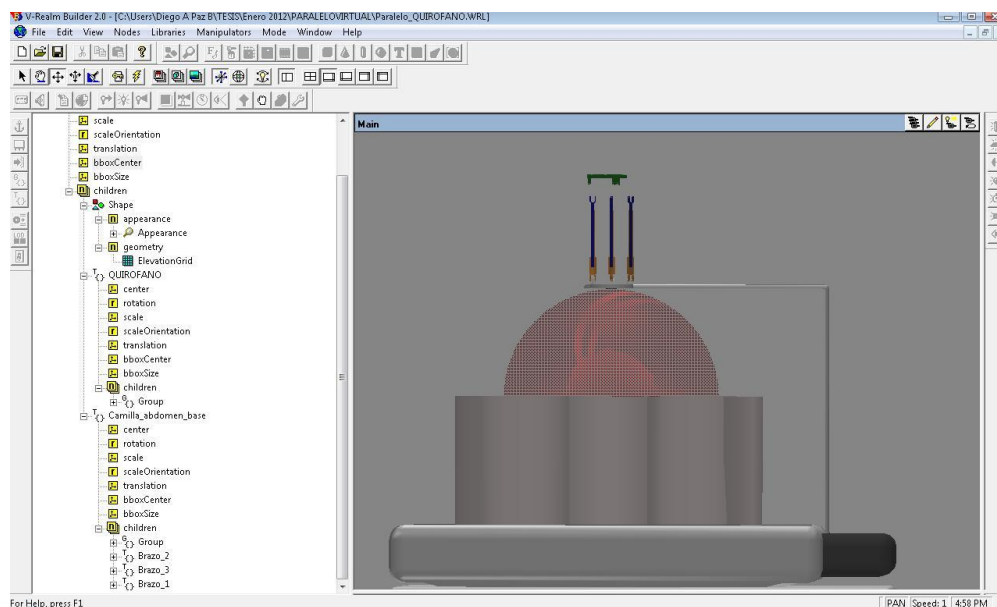


Figura C-9: Camilla, soporte, robot en V-Realm Builder 2.0.

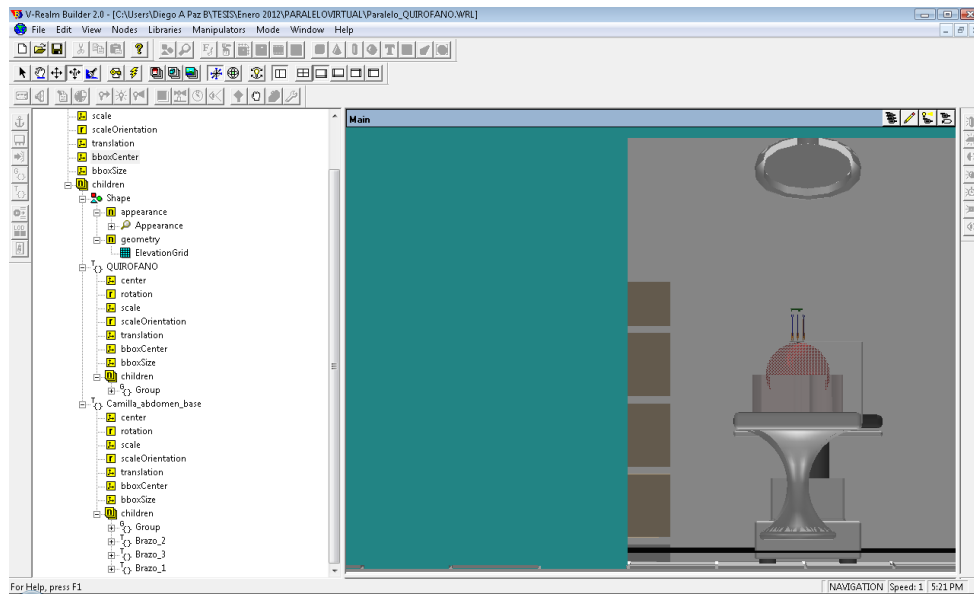


Figura C-10: Quirófano en V-Realm Builder 2.0.

Anexo D. Trabajo realizado en OGRE.

Como primera medida se debe tener las piezas del robot en Solid Edge o cualquier programa CAD.

Se graban dichas piezas en una extensión llamada .STL, como se muestra en la siguiente figura D.1.

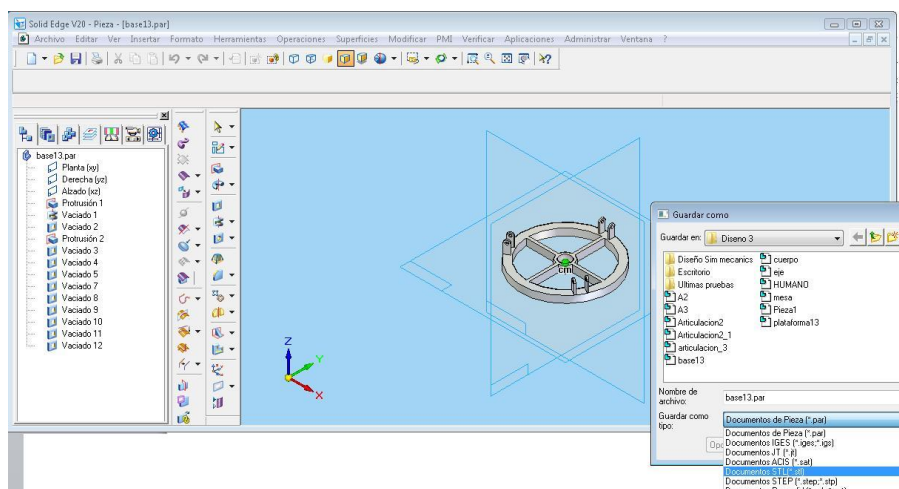


Figura D-1: Como guardar piezas en STL.

Con Blender se importan las piezas, como se muestra en la figura D.2.

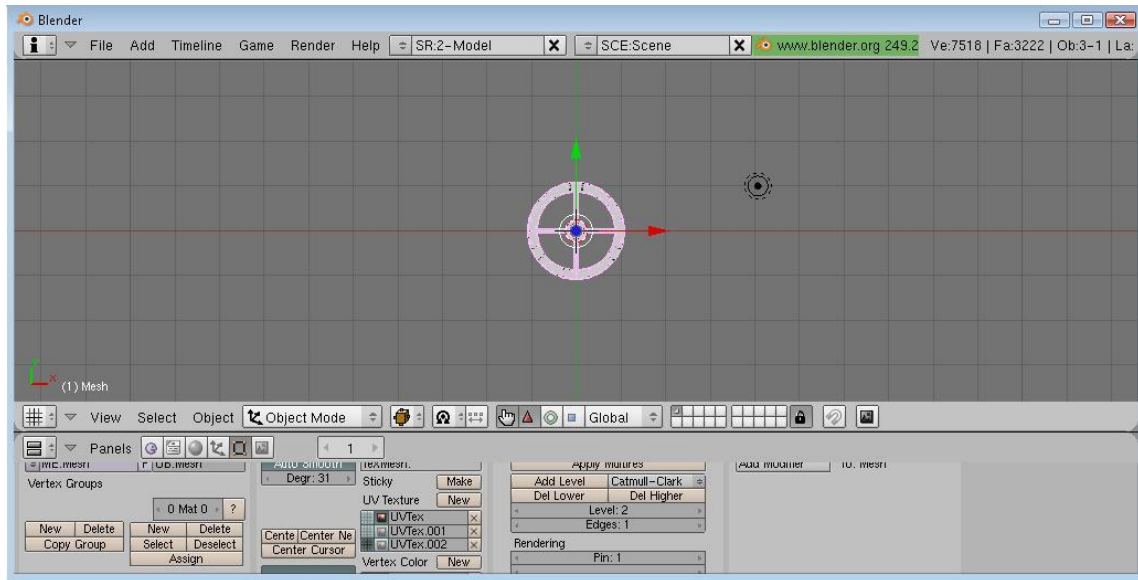


Figura D-2: Importación de piezas utilizando la herramienta Blender.

Una vez importadas las piezas se procede a exportarlas a OGRE ver figura D.3.

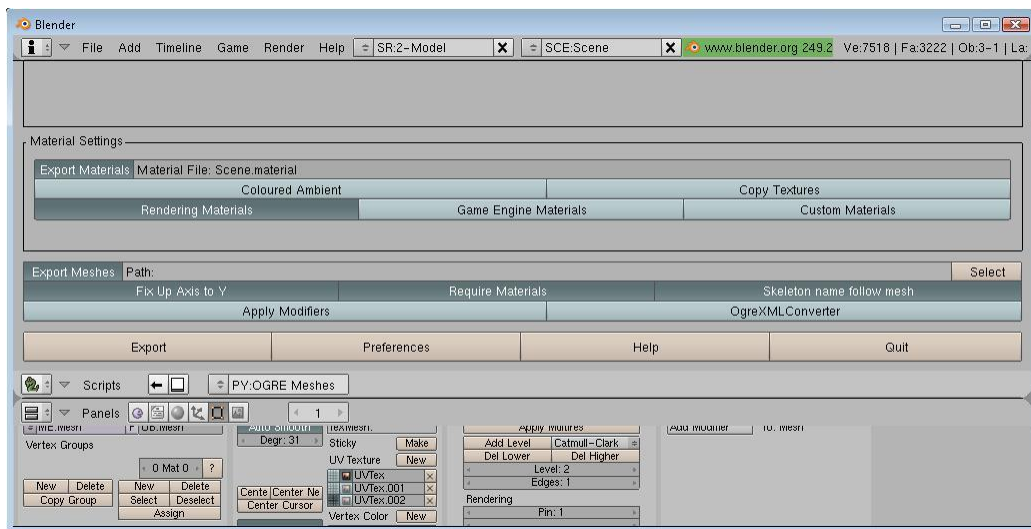


Figura D-3: Exportando piezas a OGRE.

Cuando se tengan todas las piezas exportadas se procede a realizar el armado del robot, Ogre trabaja en código C++, con la ayuda de Microsoft Visual Studio esto se

puede ver en la figura D.4.; el código completo del ensamblado del robot se muestra en el anexo E.

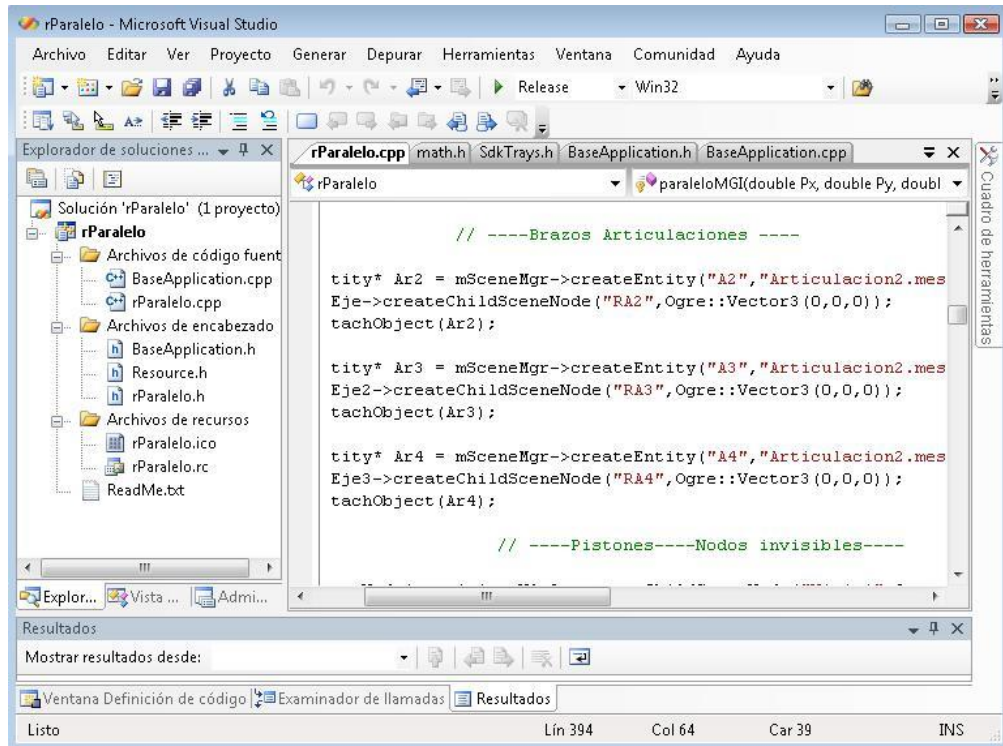


Figura D-4: Herramienta para realizar el armado del Robot.

Cuando se terminó el ensamblado del robot, se tuvo el mismo inconveniente que en Virtual Reality, ya que no se pudo tener un hijo con varios padres, es decir, no se logró que el movimiento de la plataforma dependiera de los tres brazos al mismo tiempo, se realizaron preguntas directamente en OGRE, pero el emulador nos respondió que no se podía realizar lo que queríamos, en la figura D.5 se puede observar la pregunta y respuesta.

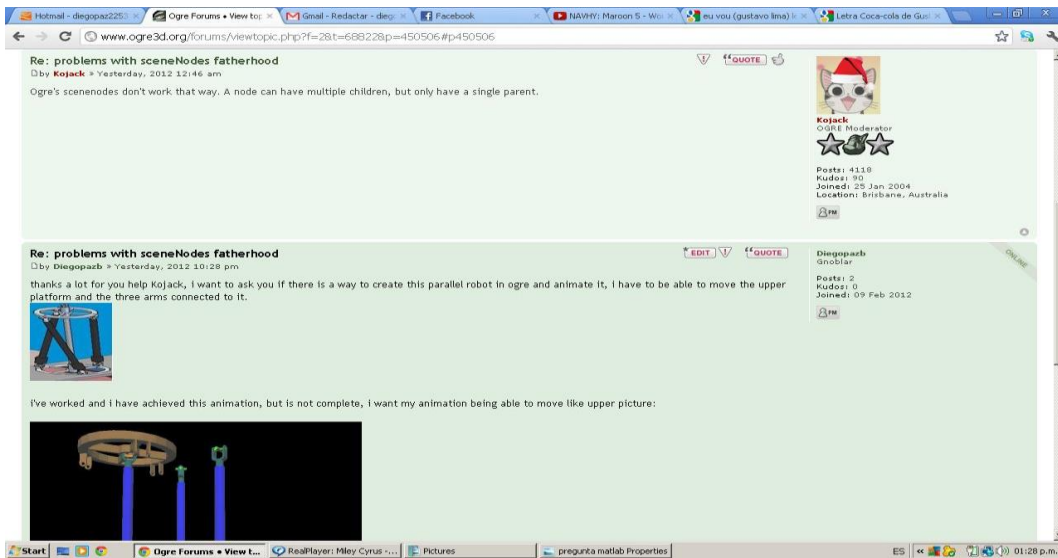


Figura D-5: Pregunta y respuesta a emulador OGRE.

A continuación se muestra detalladamente cómo se realizó el armado del robot.

El primer paso fue ubicar la base y los ejes de las juntas universales, ver figura D.6.



Figura D-6: Ubicación de la plataforma y ejes.

Acoplamiento de las articulaciones 2 a la base, figura D.7.

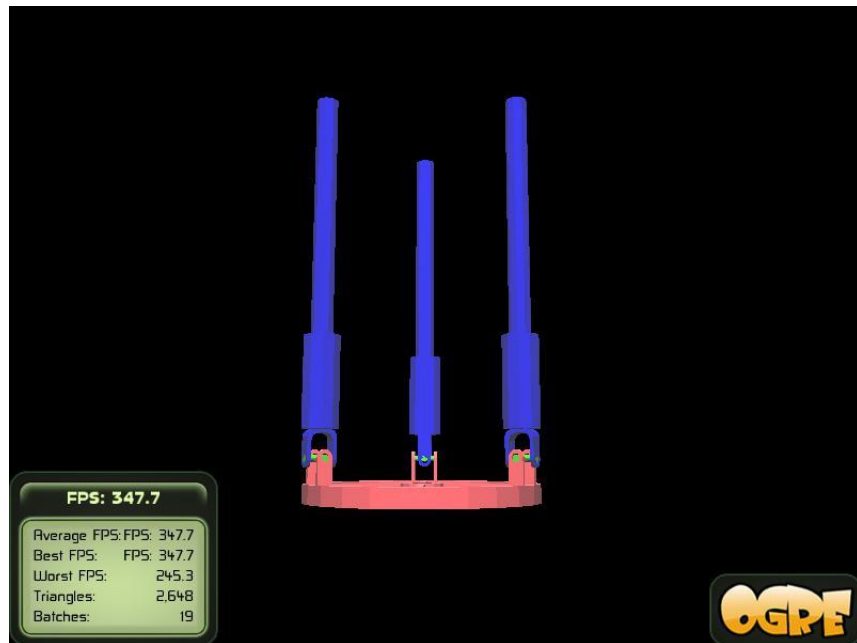


Figura D-7: Acoplamiento de la articulación 2.

Ajustar pistones a cada una de las articulaciones 2, figura D.8.

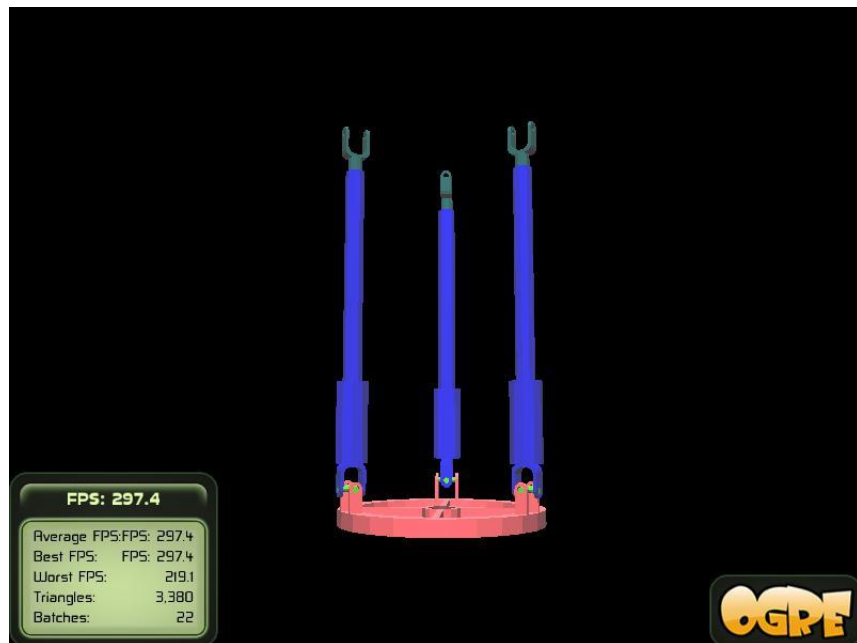


Figura D-8: Ajuste de pistones a la articulación 2.

Unir ejes a los pistones y por último se enlaza la plataforma a los 3 brazos ya armados. Figura D.9.

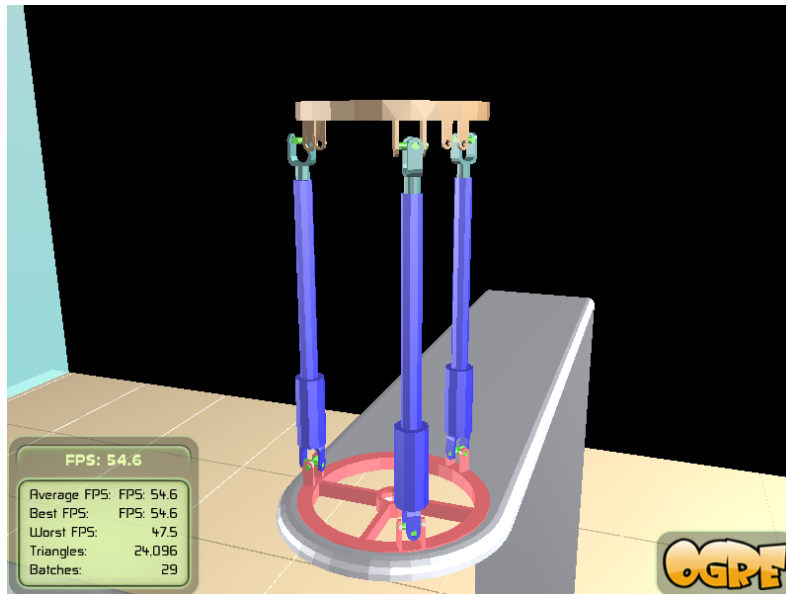


Figura D-9: Unión de ejes a pistones y plataforma a los brazos.

A continuación se muestran diferentes imágenes de los avances obtenidos en OGRE del robot PREH.

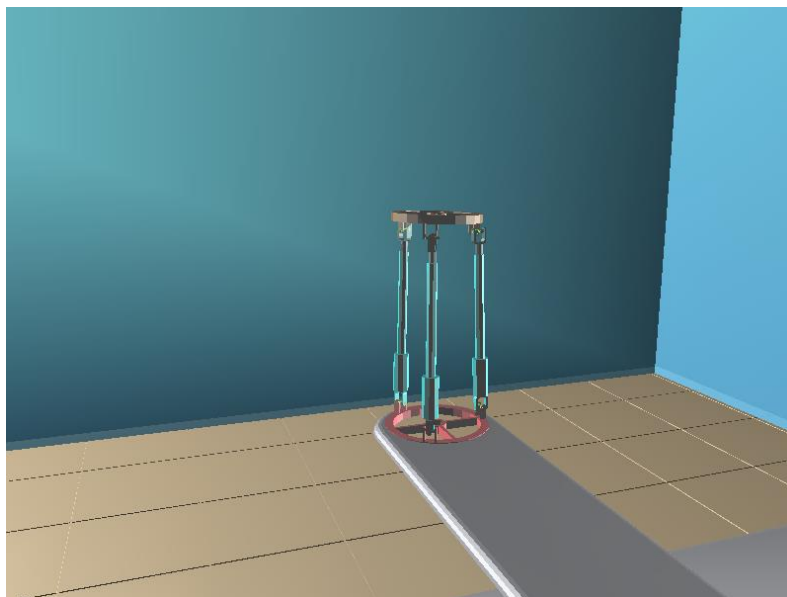


Figura D-10: Robot PREH armado, vista de las paredes y piso del quirófano.

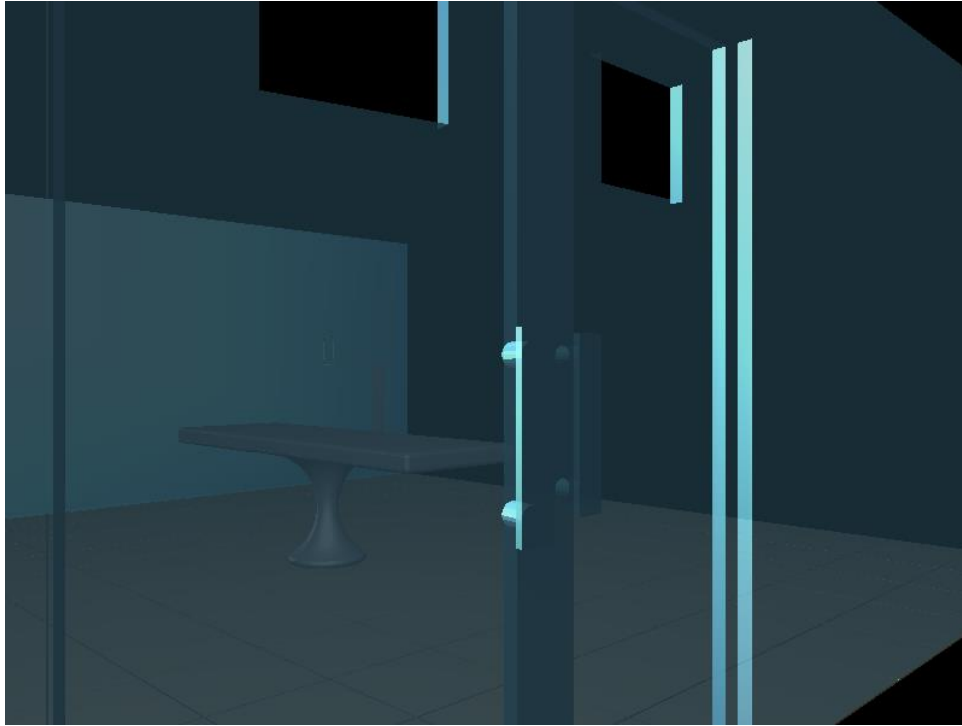


Figura D-11: Toma del Quirófano.

Anexo E. Código en C++ empleado para realizar el armado y movimientos del robot.

```
//-----Ambiente quirófano e iluminación-----  
void rParalelo::createScene(void)  
Datos = fopen("DATOS.txt","wt");  
Datos = fopen("DATOS.txt","wt");  
mSceneMgr->setAmbientLight(Ogre::ColourValue(0.25, 0.25, 0.25));  
mSceneMgr->setAmbientLight(Ogre::ColourValue(0.5, 0.5, 0.5));  
Ogre::Light* pointLight = mSceneMgr->createLight("pointLight");  
pointLight->setType(Ogre::Light::LT_DIRECTIONAL);  
pointLight->setPosition(Ogre::Vector3(250, 150, 250));  
pointLight->setDiffuseColour(Ogre::ColourValue::White);  
pointLight->setSpecularColour(Ogre::ColourValue::White);
```

```

Ogre::Light* pointLight2 = mSceneMgr->createLight("pointLight2");
pointLight2->setType(Ogre::Light::LT_POINT);
pointLight2->setPosition(Ogre::Vector3(-250, 150, 250));
pointLight2->setDiffuseColour(Ogre::ColourValue::White);
pointLight2->setSpecularColour(Ogre::ColourValue::White);
Ogre::Light* pointLight3 = mSceneMgr->createLight("pointLight3");
pointLight3->setType(Ogre::Light::LT_SPOTLIGHT);
pointLight3->setPosition(Ogre::Vector3(3, 100, 0));
pointLight3->setDiffuseColour(0, 0, 1.0);
pointLight3->setSpecularColour(0, 0, 1.0);
pointLight3->setSpotlightRange(Ogre::Degree(35), Ogre::Degree(50));
//----Base ---- Camilla ---- Pared ----Suelo----Techo
Ogre::Entity* Base = mSceneMgr->createEntity("base", "Base.mesh");
Ogre::SceneNode* NBase = mSceneMgr->getRootSceneNode()-
>createChildSceneNode("Base",Ogre::Vector3(-4.7,-0.7,-40));
NBase->attachObject(Base);
NBase->scale(3,3,3);
Ogre::Entity* Camilla = mSceneMgr->createEntity("camilla", "Camilla.mesh");
Ogre::SceneNode* NCamilla = mSceneMgr->getRootSceneNode()-
>createChildSceneNode("Camilla",Ogre::Vector3(-40,-40,-40));
NCamilla->attachObject(Camilla);
NCamilla->scale(3,3,3);
Ogre::Entity* pared = mSceneMgr->createEntity("Pared", "pared.mesh");
Ogre::SceneNode* Npared = mSceneMgr->getRootSceneNode()-
>createChildSceneNode("Par",Ogre::Vector3(-5,-113,-40));
Npared->attachObject(pared);
Npared->scale(3,3,3);
Ogre::Entity* suelo= mSceneMgr->createEntity("suelo", "suelo.mesh");
Ogre::SceneNode* Nsuelo = mSceneMgr->getRootSceneNode()-
>createChildSceneNode("sue",Ogre::Vector3(-5,-113,-40));
Nsuelo->attachObject(suelo);
Nsuelo->scale(3,3,3);
//----Ejes ---- Nodos invisibles----
noEje1 = NBase->createChildSceneNode("NEje1",Ogre::Vector3(0,0,0));

```



```

noEje1->yaw(Ogre::Degree(-90));
//noEje1->roll(Ogre::Degree(-6.5));
Ogre::Entity* Eje = mSceneMgr->createEntity("eje", "Eje.mesh");
NEje = noEje1->createChildSceneNode("REje",Ogre::Vector3(-1.185,0.462,0));
NEje->attachObject(Eje);
Ogre::SceneNode*          noEje2          =          NBase-
>createChildSceneNode("NEje2",Ogre::Vector3(0,0,0));
noEje2->yaw(Ogre::Degree(30));
Ogre::Entity* Eje2 = mSceneMgr->createEntity("eje2", "Eje.mesh");
NEje2 = noEje2->createChildSceneNode("REje2",Ogre::Vector3(-1.185,0.462,0));
NEje2->attachObject(Eje2);
Ogre::SceneNode*          noEje3          =          NBase-
>createChildSceneNode("NEje3",Ogre::Vector3(0,0,0));
noEje3->yaw(Ogre::Degree(-30));
Ogre::Entity* Eje3 = mSceneMgr->createEntity("eje3", "Eje.mesh");
NEje3 = noEje3->createChildSceneNode("REje3",Ogre::Vector3(1.185,0.462,0));
NEje3->attachObject(Eje3);
// ----Brazos Articulaciones ----
Ogre::Entity* Ar2 = mSceneMgr->createEntity("A2", "Articulacion2.mesh");
NAr2 = NEje->createChildSceneNode("RA2",Ogre::Vector3(0,0,0));
NAr2->attachObject(Ar2);
Ogre::Entity* Ar3 = mSceneMgr->createEntity("A3", "Articulacion2.mesh");
NAr3 = NEje2->createChildSceneNode("RA3",Ogre::Vector3(0,0,0));
NAr3->attachObject(Ar3);
Ogre::Entity* Ar4 = mSceneMgr->createEntity("A4", "Articulacion2.mesh");
NAr4 = NEje3->createChildSceneNode("RA4",Ogre::Vector3(0,0,0));
NAr4->attachObject(Ar4);
// ----Pistones----Nodos invisibles----
Ogre::SceneNode*          nopis1          =          NAr2-
>createChildSceneNode("Nipis1",Ogre::Vector3(0,0,0));
nopis1->pitch(Ogre::Degree(180));
Ogre::Entity* pis1 = mSceneMgr->createEntity("pis1", "Articulacion3.mesh");
Npis1 = nopis1->createChildSceneNode("Rp1",Ogre::Vector3(0,-4.5,0));
Npis1->attachObject(pis1);

```

```

Ogre::SceneNode*          nopis2          =          NAr3-
>createChildSceneNode("Nipis2",Ogre::Vector3(0,0,0));
nopis2->pitch(Ogre::Degree(180));
Ogre::Entity* pis2 = mSceneMgr->createEntity("pis2", "Articulacion3.mesh");
Npis2 = nopis2->createChildSceneNode("Rp2",Ogre::Vector3(0,-4.5,0));
Npis2->attachObject(pis2);
Ogre::SceneNode*          nopis3          =          NAr4-
>createChildSceneNode("Nipis3",Ogre::Vector3(0,0,0));
nopis3->pitch(Ogre::Degree(180))
Ogre::Entity* pis3 = mSceneMgr->createEntity("pis3", "Articulacion3.mesh");
Npis3 = nopis3->createChildSceneNode("Rp3",Ogre::Vector3(0,-4.5,0));
Npis3->attachObject(pis3);
Ogre::Entity* Epis1 = mSceneMgr->createEntity("Epis1", "Eje.mesh");
Ogre::SceneNode*          Nepis1          =          Npis1-
>createChildSceneNode("ERp1",Ogre::Vector3(0,0.09,0));
Nepis1->attachObject(Epis1);
Ogre::Entity* Epis2 = mSceneMgr->createEntity("Epis2", "Eje.mesh");
Ogre::SceneNode*          Nepis2          =          Npis2-
>createChildSceneNode("ERp2",Ogre::Vector3(0,0.09,0));
Nepis2->attachObject(Epis2);
Ogre::Entity* Epis3 = mSceneMgr->createEntity("Epis3", "Eje.mesh");
Ogre::SceneNode*          Nepis3          =          Npis3-
>createChildSceneNode("ERp3",Ogre::Vector3(0,0.09,0));
Nepis3->attachObject(Epis3);
Ogre::SceneNode*          nopis6          =          Npis2-
>createChildSceneNode("Nipis6",Ogre::Vector3(0,0,0));
nopis6->yaw(Ogre::Degree(-30));
//----Plataforma----
Ogre::Entity* P = mSceneMgr->createEntity("plataforma", "Plataforma.mesh");
Ogre::SceneNode*          Npla           =          nopis6-
>createChildSceneNode("plataforma1",Ogre::Vector3(0.85,-0.35,-0.45));
Npla->attachObject(P);
//Create a light
// create the light

```

```

// Create the scene node
NBase = mSceneMgr->getRootSceneNode()-
>createChildSceneNode("CamNode1", Ogre::Vector3(0,0,0));
// Make it look towards the ninja
NBase->yaw(Ogre::Degree(-45));
// Create the pitch node
NBase = NBase->createChildSceneNode("PitchNode1");
NBase->attachObject(mCamera);
// create the second camera node/pitch node
NBase = mSceneMgr->getRootSceneNode()-
>createChildSceneNode("CamNode2", Ogre::Vector3(0,0,0));
NBase = NBase->createChildSceneNode("PitchNode2");
mCamNode = mCamera->getParentSceneNode();
}
void rParalelo::createFrameListener(void)
{
BaseApplication::createFrameListener();
// Populate the camera container
    // ---- Establecer la rotación y la velocidad de movimiento de cada uno se los
                brazos ----

mRotate = 0.13;
mMove = 25;
g1 = 0.0361111;
g2 = -0.0361111;
g3 = 0.0361111;
g4 = -4.5;
g5 = -4.5;
g6 = -4.5;
g7= 0;
g8= 0;
g9= 0;
mDirection = Ogre::Vector3::ZERO
}
bool rParalelo::frameRenderingQueued(const Ogre::FrameEvent& evt){

```

```

if(mWindow->isClosed())
return false;
if (mShutDown)
return false;
// Need to capture/update each device
mKeyboard->capture();
mMouse->capture();
mTrayMgr->frameRenderingQueued(evt);
mCamNode->translate(mDirection * evt.timeSinceLastFrame,
Ogre::Node::TS_LOCAL);
NEje->setOrientation(Ogre::Quaternion(Ogre::Radian(g7),Ogre::Vector3(0,0,1)));
NEje2->setOrientation(Ogre::Quaternion(Ogre::Radian(g9),Ogre::Vector3(1,0,0)));
NEje3-
>setOrientation(Ogre::Quaternion(Ogre::Radian(g11),Ogre::Vector3(0,0,1)));
NAr2->setOrientation(Ogre::Quaternion(Ogre::Radian(g8),Ogre::Vector3(0,0,1)));
NAr3->setOrientation(Ogre::Quaternion(Ogre::Radian(g10),Ogre::Vector3(0,0,1)));
NAr4->setOrientation(Ogre::Quaternion(Ogre::Radian(g12),Ogre::Vector3(0,0,1)));
Npis1->setPosition(0,g4,0);
Npis2->setPosition(0,g5,0);
Npis3->setPosition(0,g6,0);
return true;
}
// OIS::KeyListener
bool rParalelo::keyPressed( const OIS::KeyEvent &arg ){
switch (arg.key)
{
case OIS::KC_1:
mCamera->getParentSceneNode()->detachObject(mCamera);
mCamNode = mSceneMgr->getSceneNode("CamNode1");
mCamNode->attachObject(mCamera);
break;
case OIS::KC_2:
mCamera->getParentSceneNode()->detachObject(mCamera);
mCamNode = mSceneMgr->getSceneNode("CamNode2");

```

```
mCamNode->attachObject(mCamera);
break;
case OIS::KC_UP:
case OIS::KC_W:
mDirection.z = -mMove;
break;
case OIS::KC_DOWN:
case OIS::KC_S:
mDirection.z = mMove;
break;
case OIS::KC_LEFT:
case OIS::KC_A:
mDirection.x = -mMove;
break;
case OIS::KC_RIGHT:
case OIS::KC_D:
mDirection.x = mMove;
break;
case OIS::KC_PGDOWN:
case OIS::KC_E:
mDirection.y = -mMove;
break;
case OIS::KC_PGUP:
case OIS::KC_Q:
mDirection.y = mMove;
break;
case OIS::KC_H:
g1 = g1 - 0.05;
break;
case OIS::KC_J:
g1 = g1 + 0.05;
break;
case OIS::KC_ESCAPE:
mShutDown = true;
```

```

fclose(Datos);
break;
default:
break;
}
if (arg.key == OIS::KC_SYSRQ) // take a screenshot
{
mWindow->writeContentsToTimestampedFile("screenshot", ".bmp");
}
return true;
}
bool rParalelo::keyReleased( const OIS::KeyEvent &arg ){
switch (arg.key)
{
case OIS::KC_UP:
case OIS::KC_W:
mDirection.z = 0;
break;
case OIS::KC_DOWN:
case OIS::KC_S:
mDirection.z = 0;
break;
case OIS::KC_LEFT:
case OIS::KC_A:
mDirection.x = 0;
break;
case OIS::KC_RIGHT:
case OIS::KC_D:
mDirection.x = 0;
break;
case OIS::KC_PGDOWN:
case OIS::KC_E:
mDirection.y = 0;
break;

```

```

case OIS::KC_PGUP:
case OIS::KC_Q:
mDirection.y = 0;
break;

default:
break;
}
return true;
}
// OIS::MouseListener
bool rParalelo::mouseMoved( const OIS::MouseEvent &arg )
{
if (arg.state.buttonDown(OIS::MB_Right))
{
mCamNode->yaw(Ogre::Degree(mRotate * arg.state.X.rel),
Ogre::Node::TS_WORLD);
mCamNode->pitch(Ogre::Degree(-mRotate * arg.state.Y.rel),
Ogre::Node::TS_LOCAL);
}
if (arg.state.buttonDown(OIS::MB_Left))
{
X = X - arg.state.X.rel*0.01;
Y = Y - arg.state.Y.rel*0.01;
Z = 1;
paraleloMGI(X,Y,Z);
}
return true;

}
bool rParalelo::mousePressed( const OIS::MouseEvent &arg, OIS::MouseButtonID
id )
{
//Ogre::Light *light = mSceneMgr->getLight("Light1");

```

```

switch (id)
{
case OIS::MB_Left:
//    light->setVisible(! light->isVisible());
break;
default:
break;
}
return true;
}
bool    rParalelo::mouseReleased(    const    OIS::MouseEvent    &arg,
OIS::MouseButtonID id ){return true;}
void rParalelo::paraleloMGI(double x,double y,double z)
{
double ra=5;
double rb=6;
double L=15;
double L2=sqrt(pow(x,2)+pow(y,2)+pow(z,2));
double L1=L-L2;
double PX=-x*L1/L2;
double PY=-y*L1/L2;
double PZ=-z*L1/L2;
double K = pow(PX,2)+pow(PY,2)+pow(PZ,2)+pow(ra,2)+pow(rb,2)-2*ra*rb;
g4 = sqrt(K+2*(rb-ra)*PX); //Longitud del brazo 1
g5 = sqrt(K-(rb-ra)*PX+(1.7321*rb-1.7321*ra)*PY); //Longitud del brazo 2
g6 = sqrt(K-(rb-ra)*PX-(1.7321*rb-1.7321*ra)*PY); //Longitud del brazo 3
g7= acos(((PX-0.5)/g4))-1.5708;
//Angulo de giro en x de la junta universal 1
g8= acos(PY/g4)-1.5708;
//Angulo de giro en y de la junta universal 1
g9= acos(((PY*sin(1.04719)+PX*cos(1.04719)+0.5)/g5))-1.5708;
//Angulo de giro en x de la junta universal 2
g10=acos((PX*sin(1.0471666)-PY*cos(1.0471666))/g5)-1.5708;
//Angulo de giro en y de la junta universal 2

```



```

g11=acos(((PY*sin(1.0471666)-PX*cos(1.047166)-0.5)/g6))-1.5708;
//Angulo de giro en x de la junta universal 3
g12=acos((PX*sin(1.0471666)+PY*cos(1.0471666))/g6)-1.5708;
//Angulo de giro en y de la junta universal 3
fprintf(Datos," %.6lf %.6lf %.6lf\n",g7,g9,g11);
g4 = (((g4-7.99) * 2.7)/65.6)-4.5;
g5 = (((g5-7.35) * 2.7)/65.6)-4.5;
g6 = (((g6-8.57) * 2.7)/65.6)-4.5;
}
#if OGRE_PLATFORM == OGRE_PLATFORM_WIN32
#define WIN32_LEAN_AND_MEAN
#include "windows.h"
#endif
#ifdef __cplusplus
extern "C" {
#endif
#if OGRE_PLATFORM == OGRE_PLATFORM_WIN32
INT WINAPI WinMain( HINSTANCE hInst, HINSTANCE, LPSTR strCmdLine, INT )
#else
int main(int argc, char *argv[])
#endif
{
// Create application object
rParalelo app;
try {
app.go();
} catch( Ogre::Exception& e ) {
#if OGRE_PLATFORM == OGRE_PLATFORM_WIN32
MessageBox( NULL, e.getFullDescription().c_str(), "An exception has occurred!",
MB_OK | MB_ICONERROR | MB_TASKMODAL);
#else
std::cerr << "An exception has occurred: " <<
e.getFullDescription().c_str() << std::endl;
#endif
}
}

```

```

}
return 0;
}
#ifdef __cplusplus
}
#endif

```

Anexo F. Obtención de los modelos dinámicos del brazo 1 del robot.

F.1. Modelo Dinámico Inverso del brazo 1

Como primera medida se deben obtener los parámetros geométricos de todas las articulaciones que forman el brazo 1, estos se muestran en la Tabla F.1.

Tabla F-1: Parámetros geométricos del brazo 1

J_1	a_{i1}	u_{i1}	σ_{i1}	γ_{i1}	b_{i1}	α_{i1}	d_{i1}	θ_{i1}	r_{i1}
1 ₁	0	0	0	0	0	$-\pi/2$	0	q_{11}	0
2 ₁	1 ₁	0	0	0	0	$\pi/2$	0	q_{21}	0
3 ₁	2 ₁	1	1	0	0	$\pi/2$	0	0	q_{31}

Con estos parámetros se procede a calcular las matrices de orientación, posición y transformación de acuerdo a la ecuación F.1, F.2 y F.3.

La matriz de orientación se define como:

$${}^i A_j = \begin{bmatrix} C\gamma_i C\theta_i - S\gamma_i C\alpha_i S\theta_i & -C\gamma_i S\theta_i - S\gamma_i C\alpha_i C\theta_i & S\gamma_i S\alpha_i \\ S\gamma_i C\theta_i + C\gamma_i C\alpha_i S\theta_i & -S\gamma_i S\theta_i + C\gamma_i C\alpha_i C\theta_i & -C\gamma_i S\alpha_i \\ S\alpha_i S\theta_i & S\alpha_i C\theta_i & C\alpha_i \end{bmatrix} \quad (F-1)$$

$${}^0 A_1 = \begin{bmatrix} Cq_{11} & -Sq_{11} & 0 \\ 0 & 0 & 1 \\ -Sq_{11} & -Cq_{11} & 0 \end{bmatrix}$$

$${}^1A_2 = \begin{bmatrix} Cq_{21} & -Sq_{21} & 0 \\ 0 & 0 & -1 \\ Sq_{21} & Cq_{21} & 0 \end{bmatrix}$$

$${}^2A_3 = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 0 & -1 \\ 0 & 1 & 0 \end{bmatrix}$$

El vector de posición se define como:

$${}^iP_j = \begin{bmatrix} d_i C\gamma_i + r_i S\gamma_i S\alpha_i \\ d_i S\gamma_i - r_i C\gamma_i S\alpha_i \\ r_i C\alpha_i + B_i \end{bmatrix} \quad (\text{F-2})$$

$${}^0P_1 = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix}$$

$${}^1P_2 = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix}$$

$${}^2P_3 = \begin{bmatrix} 0 \\ -q_{31} \\ 0 \end{bmatrix}$$

La matriz de transformación del brazo se define por:

$${}^iT_j = \begin{bmatrix} {}^iA_j & {}^iP_j \\ 0 & 1 \end{bmatrix} \quad (\text{F-3})$$

$${}^0T_1 = \begin{bmatrix} Cq_{11} & -Sq_{11} & 0 & 0 \\ 0 & 0 & 1 & 0 \\ -Sq_{11} & -Cq_{11} & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$${}^1T_2 = \begin{bmatrix} Cq_{21} & -Sq_{21} & 0 & 0 \\ 0 & 0 & -1 & 0 \\ Sq_{21} & Cq_{21} & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$${}^2T_3 = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & -1 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Con la ayuda del software Symoro, se reducen los parámetros inerciales de base de cada articulación.

Tabla F-2: Parámetros inerciales de base del brazo 1

J	XXj	XYj	XZj	YYj	YZj	ZZj	MXj	MYj	MZj	Mj	Iaj
1	0	0	0	0	0	ZZIR	MX1	MY1R	0	0	0
2	XX2R	XY2R	XZ2R	0	YZ2R	ZZ2R	MX2	MY2	0	0	0
3	0	0	0	0	0	0	MX3	MY3	MZ3	M3	IA3

Tabla F-3: Parámetros inerciales esenciales del brazo 1

J	XXj	XYj	XZj	YYj	YZj	ZZj	MXj	MYj	MZj	Mj	Iaj
1	0	0	0	0	0	ZZIR	0	0	0	0	0
2	XX2R	0	0	0	0	ZZ2R	MX2	0	0	0	0
3	0	0	0	0	0	0	0	0	MZ3	M3	IA3

Según la tabla de parámetros de base hallada anteriormente, las matrices del tensor de inercia y del primer momento de inercia pueden organizarse así:

$${}^1J_1 = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & ZZR1 \end{bmatrix} \quad {}^2J_2 = \begin{bmatrix} XXR2 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & ZZR2 \end{bmatrix}$$

$${}^3J_3 = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} \quad I_a = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & IA3 \end{bmatrix}$$

$${}^1MS_1 = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix} \quad {}^2MS_2 = \begin{bmatrix} MX2 \\ 0 \\ 0 \end{bmatrix} \quad {}^3MS_3 = \begin{bmatrix} 0 \\ 0 \\ MZ3 \end{bmatrix}$$

Cálculo de las velocidades de rotación según la ecuación:

$${}^j w_j = {}^j A_{j-1} {}^{j-1} w_{j-1} + \bar{\sigma}_j \dot{q}_j {}^j a_j \quad (\text{F-4})$$

$${}^j a_j = \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}$$

$${}^0 w_0 = 0$$

$${}^1 w_1 = \begin{bmatrix} 0 \\ 0 \\ \dot{q}_1 \end{bmatrix}$$

$${}^2 w_2 = \begin{bmatrix} Cq_{21} & 0 & Sq_{21} \\ -Sq_{21} & 0 & Cq_{21} \\ 0 & -1 & 0 \end{bmatrix} \begin{bmatrix} 0 \\ 0 \\ q_{11} \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ \dot{q}_{21} \end{bmatrix} = \begin{bmatrix} Sq_{21}\dot{q}_1 \\ Cq_{21}\dot{q}_1 \\ \dot{q}_{21} \end{bmatrix}$$

$${}^3 w_3 = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & -1 & 0 \end{bmatrix} \begin{bmatrix} Sq_{21}\dot{q}_{11} \\ Cq_{21}\dot{q}_{11} \\ \dot{q}_{21} \end{bmatrix} = \begin{bmatrix} Sq_{21}\dot{q}_{11} \\ \dot{q}_{21} \\ -Cq_{21}\dot{q}_{11} \end{bmatrix}$$

Cálculo de las velocidades de translación según la ecuación:

$${}^j V_j = {}^j A_{j-1} [{}^{j-1} V_{j-1} + {}^{j-1} w_{j-1} \times {}^{j-1} P_j] + \sigma_j \dot{q}_j {}^j a_j \quad (\text{F-5})$$

$${}^0 V_0 = 0$$

$${}^1 V_1 = 0$$

$${}^2 V_2 = 0$$

$${}^3 V_3 = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & -1 & 0 \end{bmatrix} \left[\begin{bmatrix} Sq_{21}\dot{q}_1 \\ Cq_{21}\dot{q}_1 \\ \dot{q}_{21} \end{bmatrix} \times \begin{bmatrix} 0 \\ -q_{31} \\ 0 \end{bmatrix} \right] + \begin{bmatrix} 0 \\ 0 \\ \dot{q}_{31} \end{bmatrix}$$

$${}^3 V_3 = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & -1 & 0 \end{bmatrix} \begin{bmatrix} q_{31}\dot{q}_{21} \\ 0 \\ -q_{31}Sq_{21}\dot{q}_{11} \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ \dot{q}_{31} \end{bmatrix}$$

$${}^3V_3 = \begin{bmatrix} q_{31}q_{21} \\ -q_{31}Sq_{21}q_{11} \\ q_{31} \end{bmatrix}$$

Cálculo de los elementos de la matriz de inercia \mathbf{A} , utilizando la ecuación de la energía cinética.

$$\mathbf{E}_j = \frac{1}{2} \left[{}^j\boldsymbol{\omega}_j^T {}^j\mathbf{J}_j {}^j\boldsymbol{\omega}_j + M_j {}^j\mathbf{V}_j^T {}^j\mathbf{V}_j + 2 {}^j\mathbf{M}\mathbf{S}_j^T ({}^j\mathbf{V}_j \times {}^j\boldsymbol{\omega}_j) \right] \quad (\text{F-6})$$

Energía cinética del cuerpo 1

$$E_{11} = \frac{1}{2} \begin{bmatrix} 0 & 0 & q_{11} \end{bmatrix} \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & ZZR1 \end{bmatrix} \begin{bmatrix} 0 \\ 0 \\ q_{11} \end{bmatrix}$$

$$E_{11} = \frac{1}{2} \begin{bmatrix} 0 & 0 & q_{11} \end{bmatrix} \begin{bmatrix} 0 \\ 0 \\ ZZR1q_{11} \end{bmatrix}$$

$$E_{11} = \frac{1}{2} ZZR1 \dot{q}_{11}^2$$

Energía cinética del cuerpo 2

$$E_{21} = \frac{1}{2} \begin{bmatrix} Sq_{21}q_{11} & CSq_{21}q_{11} & q_{21} \end{bmatrix} \begin{bmatrix} XXR2 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & ZZR2 \end{bmatrix} \begin{bmatrix} Sq_{21}q_{11} \\ Cq_{21}q_{11} \\ q_{21} \end{bmatrix}$$

$$E_{21} = \frac{1}{2} \begin{bmatrix} Sq_{21}q_{11} & Cq_{21}q_{11} & q_{21} \end{bmatrix} \begin{bmatrix} XXR2Sq_{21}q_{11} \\ 0 \\ ZZR2q_{21} \end{bmatrix}$$

$$E_{21} = \frac{1}{2} [Sq_{21}^2 XXR2 \dot{q}_{11}^2 + ZZR2 \dot{q}_{21}^2]$$

Energía cinética del cuerpo 3

$$E_{31} = \frac{1}{2} \left[M_3 [q_{31} \dot{q}_{21} \quad -q_{31} S q_{21} \dot{q}_{11} \quad \dot{q}_{31}] \begin{bmatrix} q_{31} \dot{q}_{21} \\ -q_{31} S q_{21} \dot{q}_{11} \\ \dot{q}_{31} \end{bmatrix} \right. \\ \left. + 2[0 \quad 0 \quad MZ3] \left(\begin{bmatrix} q_{31} \dot{q}_{21} \\ -q_{31} S q_{21} \dot{q}_{11} \\ \dot{q}_{31} \end{bmatrix} \times \begin{bmatrix} S q_{21} \dot{q}_{11} \\ \dot{q}_{21} \\ -C q_{21} \dot{q}_{11} \end{bmatrix} \right) \right]$$

$$E_{31} = \frac{1}{2} \left[M_3 (q_{31}^2 \dot{q}_{21}^2 + q_{31}^2 S q_{21}^2 \dot{q}_{11}^2 + \dot{q}_{31}^2) \right. \\ \left. + 2[0 \quad 0 \quad MZ3] \left(\begin{bmatrix} q_{31} S q_{21} C q_{21} \dot{q}_{11}^2 - \dot{q}_{21} \dot{q}_{31} \\ C q_{21} \dot{q}_{11} q_{31} \dot{q}_{21} + \dot{q}_{31} S q_{21} \dot{q}_{11} \\ q_{31} \dot{q}_{21}^2 + q_{31} S q_{21}^2 \dot{q}_{11}^2 \end{bmatrix} \right) \right]$$

$$E_{31} = \frac{1}{2} [M_3 q_{31}^2 \dot{q}_{21}^2 + M_3 q_{31}^2 S q_{21}^2 \dot{q}_{11}^2 + M_3 \dot{q}_{31}^2 + 2MZ3 q_{31} \dot{q}_{21}^2 \\ + 2MZ3 q_{31} S q_{21}^2 \dot{q}_{11}^2]$$

$$E_{31} = \frac{1}{2} [(M_3 q_{31}^2 S q_{21}^2 + 2MZ3 q_{31} S q_{21}^2) \dot{q}_{11}^2 + (M_3 q_{31}^2 + 2MZ3 q_{31}) \dot{q}_{21}^2 + M_3 \dot{q}_{31}^2]$$

Una vez obtenidas las expresiones de las tres energías cinéticas se procede a armar la matriz de inercia

$$A_{11} = ZZR1 + S q_{21}^2 XXR2 + M_3 q_{31}^2 S q_{21}^2 + 2MZ3 q_{31} S q_{21}^2$$

$$A_{22} = ZZR2 + M_3 q_{31}^2 + 2MZ3 q_{31}$$

$$A_{33} = M_3$$

Cálculo del vector de gravedad utilizando la ecuación:

$$\mathbf{U}_j = -[{}^0\mathbf{g}^T \quad 0] {}^0\mathbf{T}_j \begin{bmatrix} {}^j\mathbf{MS}_j \\ M_j \end{bmatrix} \quad (\text{F-7})$$

$${}^0T_1 = \begin{bmatrix} Cq_{11} & -Sq_{11} & 0 & 0 \\ 0 & 0 & 1 & 0 \\ -Sq_{11} & -Cq_{11} & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$${}^0T_2 = \begin{bmatrix} Cq_{11}Cq_{21} & -Cq_{11}Sq_{21} & Sq_{11} & 0 \\ Sq_{21} & Cq_{21} & 0 & 0 \\ -Sq_{11}Cq_{21} & Sq_{11}Sq_{21} & Cq_{11} & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$${}^0T_3 = \begin{bmatrix} Cq_{11}Cq_{21} & Sq_{11} & Cq_{11}Sq_{21} & -q_{31}Cq_{11}Sq_{21} \\ Sq_{21} & 0 & -Cq_{21} & q_{31}Cq_{21} \\ -Sq_{11}Cq_{21} & Cq_{11} & -Sq_{11}Sq_{21} & q_{31}Sq_{11}Sq_{21} \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Energía potencial del cuerpo 1:

$$U_1 = -[0 \ 0 \ G3 \ 0] \begin{bmatrix} Cq_{11} & -Sq_{11} & 0 & 0 \\ 0 & 0 & 1 & 0 \\ -Sq_{11} & -Cq_{11} & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 0 \\ 0 \\ 0 \\ M1 \end{bmatrix}$$

$$U_1 = 0$$

Energía potencial del cuerpo 2:

$$U_2 = -[0 \ 0 \ G3 \ 0] \begin{bmatrix} Cq_{11}Cq_{21} & -Cq_{11}Sq_{21} & Sq_{11} & 0 \\ Sq_{21} & Cq_{21} & 0 & 0 \\ -Sq_{11}Cq_{21} & Sq_{11}Sq_{21} & Cq_{11} & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} MX2 \\ 0 \\ 0 \\ M2 \end{bmatrix}$$

$$U_2 = -[0 \ 0 \ G3 \ 0] \begin{bmatrix} Cq_{11}Cq_{21}MX2 \\ Sq_{21}MX2 \\ -Sq_{11}Cq_{21}MX2 \\ M2 \end{bmatrix}$$

$$U_2 = Sq_{11}Cq_{21}MX2G3$$

Energía potencial del cuerpo 3:

$$U_3 = -[0 \quad 0 \quad G3 \quad 0] \begin{bmatrix} Cq_{11}Cq_{21} & Sq_{11} & Cq_{11}Sq_{21} & -q_{31}Cq_{11}Sq_{21} \\ Sq_{21} & 0 & -Cq_{21} & q_{31}Cq_{21} \\ -Sq_{11}Cq_{21} & Cq_{11} & -Sq_{11}Sq_{21} & q_{31}Sq_{11}Sq_{21} \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 0 \\ 0 \\ MZ3 \\ M3 \end{bmatrix}$$

$$U_3 = -[0 \quad 0 \quad G3 \quad 0] \begin{bmatrix} X \\ X \\ -Sq_{11}Sq_{21}MZ3 + q_{31}Sq_{11}Sq_{21}M3 \\ X \end{bmatrix}$$

$$U_3 = Sq_{11}Sq_{21}MZ3G3 - q_{31}Sq_{11}Sq_{21}G3M3$$

$$U = Sq_{11}Cq_{21}MX2G3 + Sq_{11}Sq_{21}MZ3G3 - q_{31}Sq_{11}Sq_{21}G3M3$$

$$Q_1 = \frac{\partial U}{\partial q_{11}} = Cq_{11}Cq_{21}MX2G3 + Cq_{11}Sq_{21}MZ3G3 - q_{31}Cq_{11}Sq_{21}G3M3$$

$$Q_2 = \frac{\partial U}{\partial q_{21}} = -Sq_{11}Sq_{21}MX2G3 + Sq_{11}Cq_{21}MZ3G3 - q_{31}Sq_{11}Cq_{21}G3M3$$

$$Q_3 = \frac{\partial U}{\partial q_{31}} = -Sq_{11}Sq_{21}G3M3$$

El modelo dinámico inverso del brazo 1 es:

$$\Gamma_1 = A(q)\ddot{q}_1 + Q(q)$$

$$\Gamma_1 = \begin{bmatrix} ZZR1 + Sq_{21}^2 XXR2 + M_3 q_{31}^2 Sq_{21}^2 + 2MZ3q_{31}Sq_{21}^2 & 0 & 0 \\ 0 & ZZR2 + M_3 q_{31}^2 + 2MZ3q_{31} & 0 \\ 0 & 0 & M_3 \end{bmatrix} \begin{bmatrix} \ddot{q}_{11} \\ \ddot{q}_{21} \\ \ddot{q}_{31} \end{bmatrix} + \begin{bmatrix} Cq_{11}Cq_{21}MX2G3 + Cq_{11}Sq_{21}MZ3G3 - q_{31}Cq_{11}Sq_{21}G3M3 \\ -Sq_{11}Sq_{21}MX2G3 + Sq_{11}Cq_{21}MZ3G3 - q_{31}Sq_{11}Cq_{21}G3M3 \\ -Sq_{11}Sq_{21}G3M3 \end{bmatrix}$$

F.2. Modelo Dinámico Directo del brazo 1

$$A^{-1}(q)(\Gamma_1 - Q(q)) = \ddot{q}_1 \quad (F-8)$$

Para los modelos dinámicos de los otros dos brazos se sigue el mismo procedimiento, los parámetros geométricos se muestran en las tablas siguientes.

Tabla F-4: Parámetros geométricos del brazo 2

J_2	a_{j2}	u_{j2}	σ_{j2}	γ_{j2}	b_{j2}	α_{j2}	d_{j2}	θ_{j2}	r_{j2}
1 ₂	0	0	0	0	0	$-\pi/2$	d_{12}	q_{12}	0
2 ₂	1 ₂	0	0	0	0	$\pi/2$	0	q_{22}	0
3 ₂	2 ₂	1	1	0	0	$\pi/2$	0	0	q_{32}

Tabla F-5: Parámetros geométricos del brazo 3

J_3	a_{j3}	u_{j3}	σ_{j3}	γ_{j3}	b_{j3}	α_{j3}	d_{j3}	θ_{j3}	r_{j3}
1 ₃	0	0	0	γ_{13}	0	$-\pi/2$	d_{13}	q_{13}	0
2 ₃	1 ₃	0	0	0	0	$\pi/2$	0	q_{23}	0
3 ₃	2 ₃	1	1	0	0	$\pi/2$	0	0	q_{33}

Los parámetros Inerciales de base para los brazos 2 y 3 son los mismos que los del brazo 1. El brazo 4 o brazo central no es necesario modelarlo ya que todas sus articulaciones son pasivas.

Anexo G. Obtención de los parámetros inerciales en Solid Edge.

En el presente anexo muestra los diferentes pasos para realizar la obtención de los parámetros inerciales de cada uno de los brazos; explicando cada una de las piezas que lo componen.

Para la pieza eje.par, el primer paso a realizar es la obtención de los parámetros inerciales. Teniendo el diseño de la pieza culminado con su respectivo material como se ve en la figura G-1.

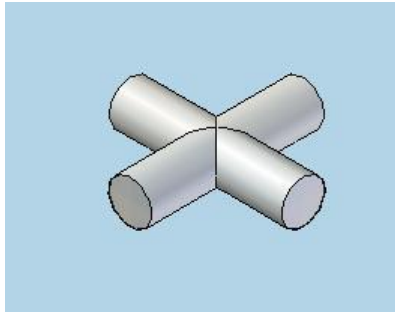


Figura G-1: Pieza eje.par.

Se procede a incluir el eje de coordenadas, recordando que es de vital importancia para la obtención de los parámetros inerciales que el eje Z coincida con el eje de la articulación. Para ubicar los ejes en la ventana operaciones le damos clic y él nos abre una ventana, luego buscamos donde dice sistema de coordenadas le damos clic y listo queda añadido el sistema como de observa en la siguiente figura G-2.

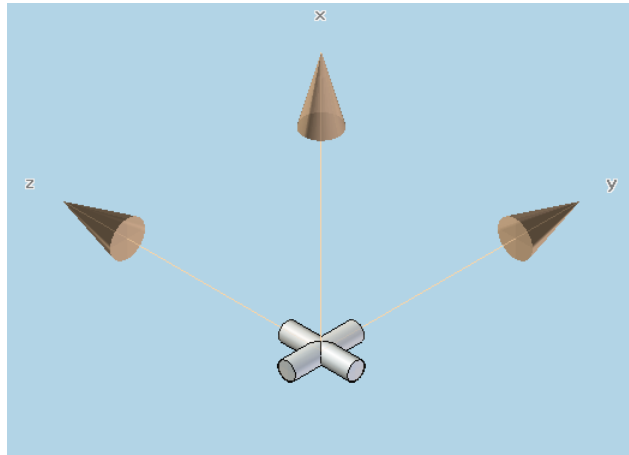


Figura G-2: Pieza eje.par con eje de coordenadas.

Ahora se procede a visualizar los parámetros inerciales de base, por lo que se dirige a la ventana verificar, ella abre una ventana y se le da click donde dice propiedades físicas. Ya realizados estos pasos se visualiza una la tabla de parámetros inerciales como se ve en la figura G-3

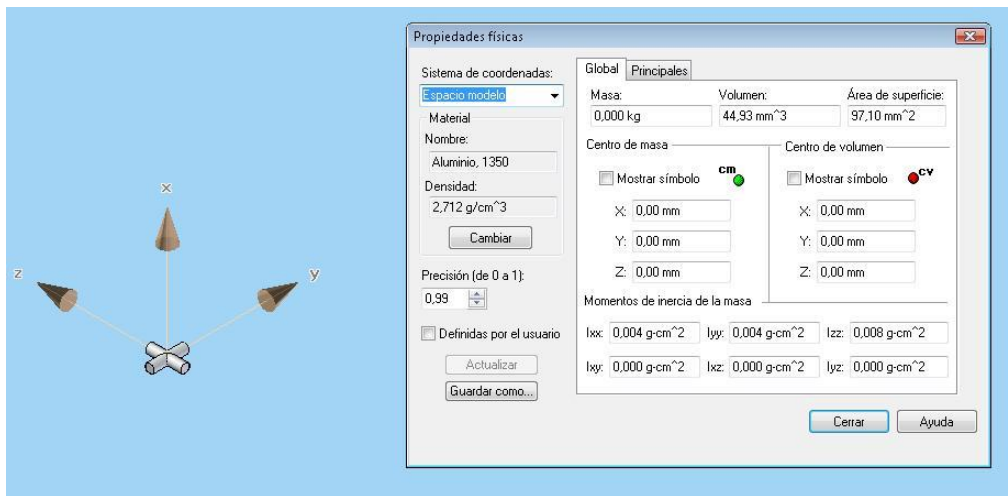


Figura G-3: eje.par Primer momento de inercia.

La figura G-3 visualiza una tabla en la cual se muestra el primer momento de inercia de la pieza eje.paz, las cuales se encuentran simbolizadas de la siguiente manera.

En esta figura se observa a además de el primer momento de inercia el material con el que fue construida, densidad, masa, volumen, área de superficie.

En ella se ve el primer momento de inercia de la pieza eje.par, las cuales se encuentran simbolizadas de la siguiente manera:

$$\begin{aligned} I_{xx} &= xx1; & I_{yy} &= yy1; \\ I_{zz} &= zz1; & I_{xy} &= xy1; \\ I_{xz} &= xz1; & I_{yz} &= yz1. \end{aligned}$$

En el mismo recuadro se dirige a la ventana que dice principales y arroja los siguientes valores ver figura G-4.

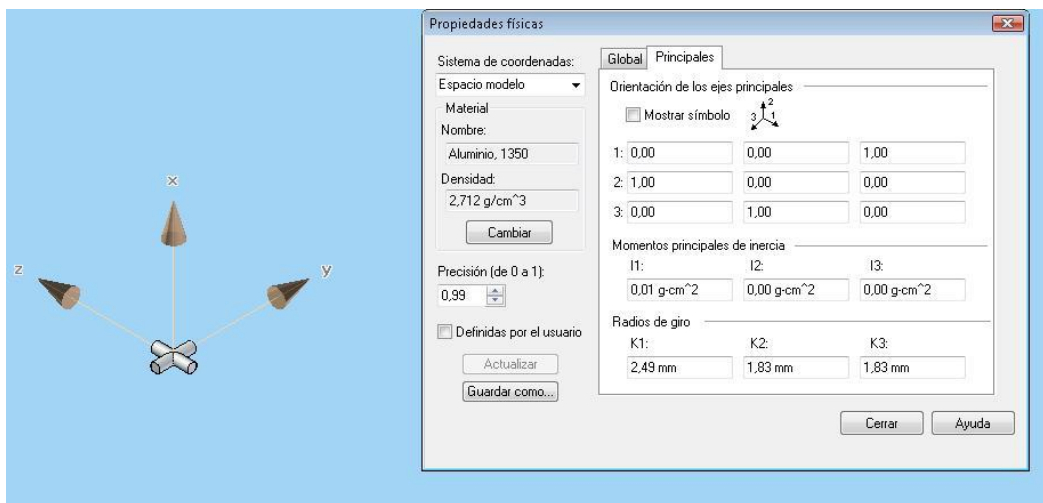


Figura G-4: Segundo momento de inercia.

Para la pieza articulación 2, el primer paso a realizar es la obtención de los parámetros inerciales. Teniendo el diseño de la pieza culminado con su respectivo material como se ve en la figura G-5

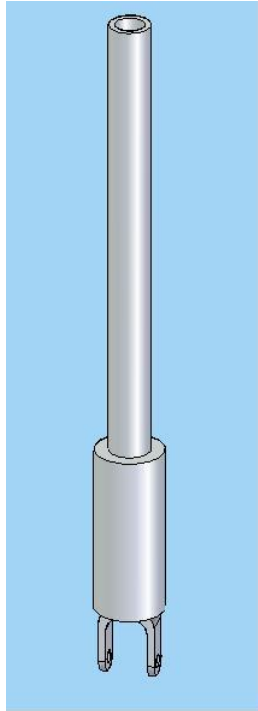


Figura G-5: Pieza articulación 2.

Se procede a incluir el eje de coordenadas, recordando que es de vital importancia para la obtención de los parámetros inerciales que el eje Z coincida con el eje de la articulación. Para ubicar los ejes en la ventana operaciones le damos clic y él nos abre una ventana, luego buscamos donde dice sistema de coordenadas le damos clic y listo queda añadido el sistema como de observa en la siguiente figura G-6.

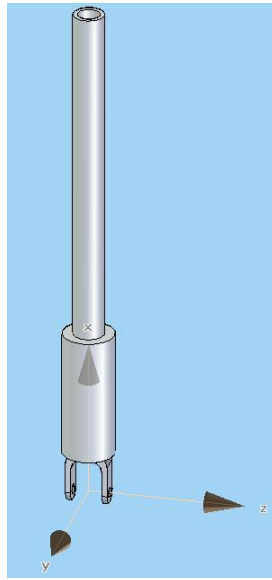


Figura G-6: Pieza articulación 2 con sistema de coordenadas

Ahora se procede a visualizar los parámetros inerciales de base, por lo que se dirige a la ventana verificar, ella abre una ventana y se le da click donde dice propiedades físicas. Ya realizados estos pasos se visualiza una la tabla de parámetros inerciales como se ve en la figura G-7

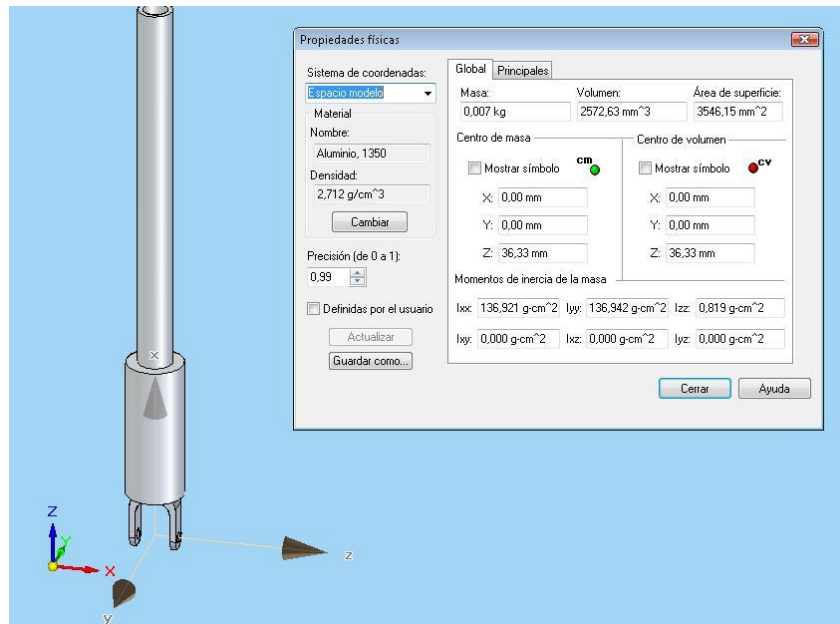


Figura G-7: Primer momento de inercia.

La figura G-7 visualiza una tabla en la cual se muestra el primer momento de inercia de la pieza articulacion 2.par, las cuales se encuentran simbolizadas de la siguiente manera.

En esta figura se observa a además de el primer momento de inercia el material con el que fue construida, densidad, masa, volumen, área de superficie.

En ella se ve el primer momento de inercia de la pieza articulacion 2.par, las cuales se encuentran simbolizadas de la siguiente manera:

$$\begin{array}{ll} I_{xx}=xx1; & I_{yy}=yy1; \\ I_{zz}=zz1; & I_{xy}=xy1; \\ I_{xz}=xz1; & I_{yz}=yz1. \end{array}$$

En el miso recuadro se dirige a la ventana que dice principales y arroja los siguientes valores ver figura G-8.

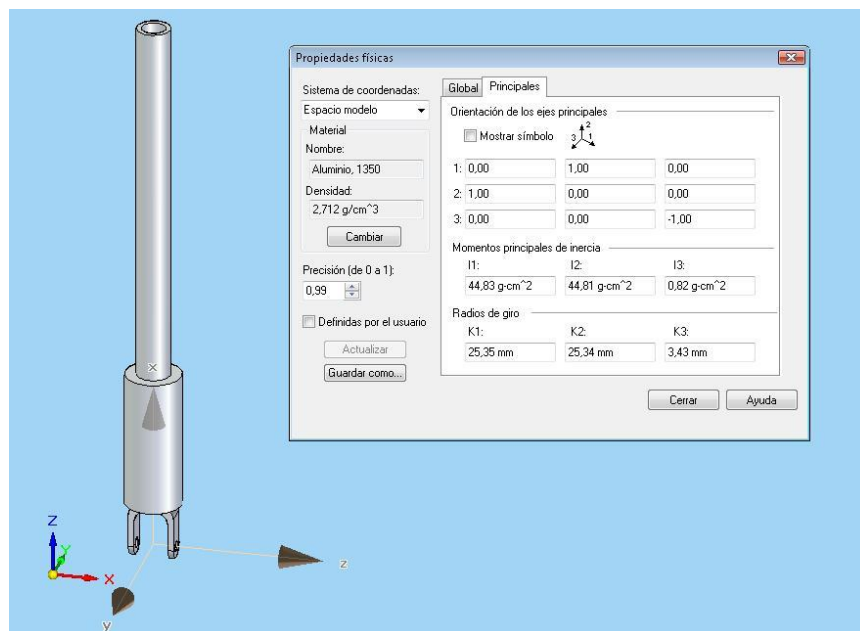


Figura G-8: Segundo momento de inercia

Para la pieza articulación_3, el primer paso a realizar es la obtención de los parámetros inerciales. Teniendo el diseño de la pieza culminado con su respectivo material como se ve en la figura G-8.

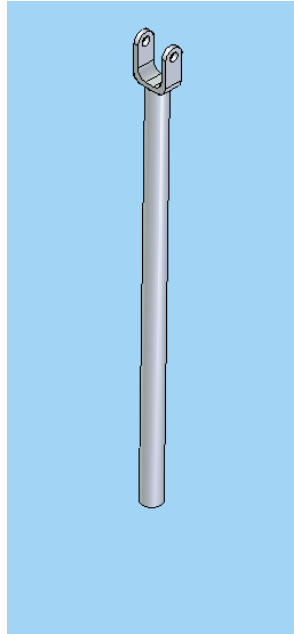


Figura G-9:Pieza articulación_3.

Se procede a incluir el eje de coordenadas, recordando que es de vital importancia para la obtención de los parámetros inerciales que el eje Z coincida con el eje de la articulación. Para ubicar los ejes en la ventana operaciones le damos clic y él nos abre una ventana, luego buscamos donde dice sistema de coordenadas le damos clic y listo queda añadido el sistema como de observa en la siguiente figura G-10.



Figura G-10: Pieza articulación_3 con sistema de coordenadas.

Ahora se procede a visualizar los parámetros inerciales de base, por lo que se dirige a la ventana verificar, ella abre una ventana y se le da click donde dice propiedades físicas. Ya realizados estos pasos se visualiza una la tabla de parámetros inerciales como se ve en la figura G-11.

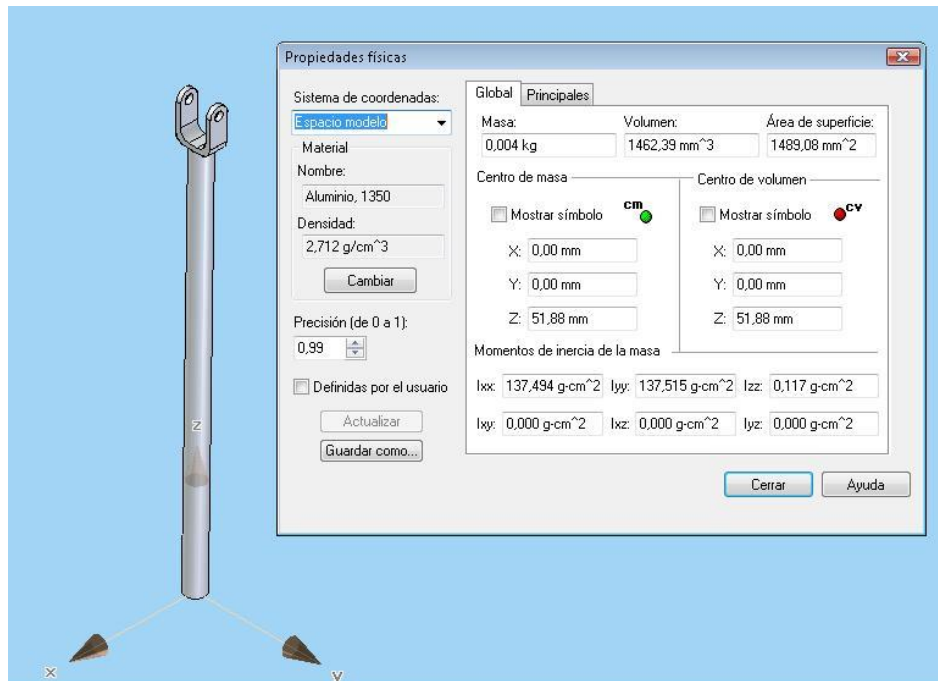


Figura G-11: Primer momento de inercia.

La figura G-11 visualiza una tabla en la cual se muestra el primer momento de inercia de la pieza articulación_3.par, las cuales se encuentran simbolizadas de la siguiente manera.

En esta figura se observa a además de el primer momento de inercia el material con el que fue construida, densidad, masa, volumen, área de superficie.

En ella se ve el primer momento de inercia de la pieza articulación_3.par, las cuales se encuentran simbolizadas de la siguiente manera:

$$\begin{array}{ll} I_{xx}=xx1; & I_{yy}=yy1; \\ I_{zz}=zz1; & I_{xy}=xy1; \\ I_{xz}=xz1; & I_{yz}=yz1. \end{array}$$

En el mismo recuadro se dirige a la ventana que dice principales y arroja los siguientes valores ver figura G-12.

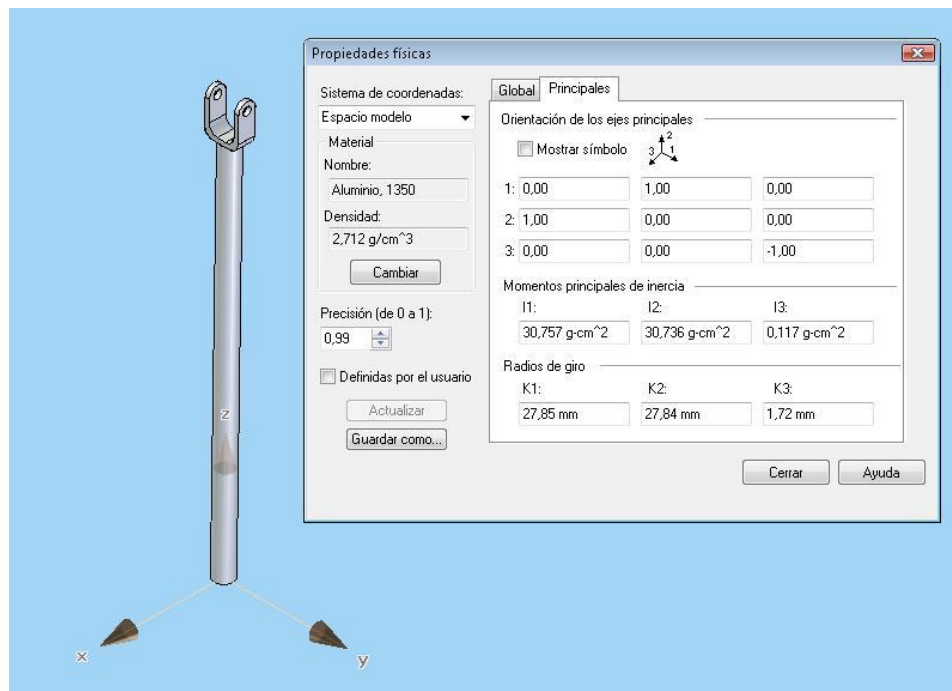


Figura G-12: Segundo momento de inercia.