

SERGIO ALEXANDER SALINAS



**MODELADO, SIMULACIÓN EN 3D Y CONTROL DE UN ROBOT PARA
CIRUGÍA LAPAROSCÓPICA**

**Universidad del Cauca
Facultad de Ingeniería Electrónica y Telecomunicaciones
Maestría en Electrónica y Telecomunicaciones**

**Popayán
2009**

SERGIO ALEXANDER SALINAS

**MODELADO, SIMULACIÓN EN 3D Y CONTROL DE UN ROBOT PARA
CIRUGÍA LAPAROSCÓPICA**

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

**Magíster en Ingeniería, Área
Electrónica y Telecomunicaciones**

**Director:
Oscar Andrés Vivas Albán
PhD. Robótica**

**Popayán
2009**

Hoja de Aprobación

Director

PhD. Oscar Andrés Vivas Albán

Jurado

PhD. Carlos Alberto Gaviria López

Jurado

PhD. Vera Zasúlich Pérez Ariza

Fecha de sustentación: Popayán, 24 de Abril de 2009

*A las diosas que impulsan
los motores de mi existir.
Teresa, Valentina y Nidia.*

Agradecimientos

A Dios por permitirme vivir y disfrutar la realización y culminación de mis estudios de postgrado.

A mi esposa por su constancia y continua compañía, sin temor a entregarlo todo.

A mi hermana y mi madre por su amor impulsor alimentado constantemente.

A Andrés Vivas por su paciencia y colaboración en todo momento como director de este proyecto.

A Diego Aguilar por su gran ayuda en el desarrollo de las trayectorias de la colecistectomía y el ambiente tridimensional.

A mis compañeros de estudio y de trabajo que ayudaron con sus consejos y correcciones.

A los evaluadores de este proyecto que con su visión engrandecen el aporte científico del mismo.

A la Universidad del Cauca y el Instituto de Postgrados en Electrónica y Telecomunicaciones por su colaboración administrativa y económica.

Resumen

El proyecto “Modelado, Simulación en 3D y Control de un Robot para Cirugía Laparoscópica” es desarrollado en la Universidad del Cauca (Colombia) por el grupo de investigación en Automática Industrial, representado por el autor y el director del mismo.

Se han desarrollado diferentes soluciones para aplicar la robótica en la medicina con gran éxito, logrando igualar y algunas veces mejorar los procedimientos médicos sin robots. Específicamente en cirugía mini-invasiva a nivel abdominal (laparoscopia), se han diseñado robots como Da Vinci, Zeus, Endobot, Black Falcon, Endovia, entre otros, que tienen estructuras seriales abiertas o cerradas, con ventajas y desventajas particulares; sin embargo lo que tienen en común es que han sido diseñados y contruidos en países desarrollados.

LapBot (Robot para Laparoscopia) es el robot diseñado en este proyecto, basado en el análisis de las estructuras de los robots antes mencionados. Se modela matemáticamente a nivel geométrico y dinámico para comprobar que puede desempeñarse en cirugías laparoscópicas, pasando a través de un punto estático en el espacio cartesiano que representa la incisión realizada al paciente en el abdomen.

Para verificar la dinámica de LapBot y demostrar su utilidad, se diseña y sintoniza un controlador por par calculado (CTC) que permita obtener un error cartesiano menor a un milímetro en el seguimiento de trayectorias de prueba y de procesos quirúrgicos de una colecistectomía (cirugía mini-invasiva para extraer una vesícula biliar enferma).

La simulación tridimensional de un sistema electromecánico es importante para observar el comportamiento del mismo ante diferentes situaciones reales o no reales, e identificar posibles mejoras o soluciones a problemas que pudiesen presentarse al construir un prototipo físico de dicho sistema.

Para el sistema robótico LapBot se desarrolla un ambiente tridimensional usando el programa de representación gráfica por computador Ogre3D, que permite la comunicación en modo cliente-servidor con Matlab, programa en el cual se realizaron todas las pruebas con el controlador.

El ambiente 3D muestra dos brazos de LapBot realizando los movimientos de una colecistectomía real, a través de dos orificios de una caja que representan las incisiones realizadas en el abdomen del paciente. El simulador usa los modelos matemáticos del robot, permite ver la escena desde diferentes puntos de vista y funciona con las dos principales interfaces de programación de aplicaciones gráficas (OpenGL y Direct3D).

Palabras Clave: Robótica quirúrgica, colecistectomía, simulación tridimensional, modelado de robots, control de robots.

Abstract

The “Modeling, 3D Simulation and Control of a Laparoscopic Surgery Robot” project is developed in the Cauca University (Colombia) by the Automatic Industrial investigation group. In this project, the author and director are representing the group.

The robotics has been applied in medicine with excellent results, it has obtained equalize and sometimes improve medical procedures without robots. To specify, the Da Vinci, Zeus, Endobot, Black Falcon and Endovia robots are designed for mini-invasive surgery and they can to do abdominal surgery (laparoscopy). These robots contain open or close serial structures, with their advantage and disadvantage. But these robots have been designed and constructed by development countries.

LapBot (Laparoscopic Robot) is designed in this project; the design is based in a structural analysis of the above-mentioned robots. LapBot was modeled (geometric and dynamic models) to study its performance in laparoscopic surgeries, the robot should cross over a static point in the cartesian space. This point represents the abdominal patient incision.

We have designed and tuned a Computed Torque Control (CTC), the purpose is to verify the LapBot dynamic and to demonstrate its utility. The robot tracks probe trajectories and cholecystectomic processes (mini-invasive surgery did to reject a gall bladder sick). It permits to obtain a cartesian error less to one millimeter.

The three-dimensional simulation of an electro-mechanic system is important to regard its performance in view of real or unreal situations and to detect possible improvements or problems solutions. Therefore, its possible constructs a good physic prototype without said problems.

For the LapBot system, we have developed a three-dimensional environment using Ogre3D (graphic computer representation software), it permits the server-client communication with Matlab software. The control system was probed totally in Matlab.

The 3D environment has two LapBot arms doing cholecystectomic real movements across two orifices in a box, this orifices represents the patient abdomen incisions.

The simulator uses the robot mathematical models, permits different view points of the scene and runs with two graphical Application Programming Interfaces (OpenGL and Direct3D).

Key Words: Surgery robotics, cholecystectomy, three-dimensional simulation, modeling of robots, control of robots.

Contenido

	Pág.
<u>Lista de Tablas</u>	<u>xvii</u>
<u>Lista de Figuras</u>	<u>xix</u>
<u>Lista de Abreviaturas</u>	<u>xxiii</u>
<u>Lista de Símbolos</u>	<u>xxv</u>
<u>1. Introducción</u>	<u>27</u>
<u>2. Robótica Aplicada a la Cirugía Mini-Invasiva</u>	<u>29</u>
2.1 Conceptos de Manipuladores Robotizados	29
2.2 Cirugía Mini-Invasiva (MIS)	30
2.2.1 Laparoscopia	31
2.3 Robots en Cirugía	33
2.3.1 Clasificación	34
2.3.2 Robots más Usados en Cirugía	34
2.4 Simulación en 3D	37
2.4.1 Ejemplos para Laparoscopia	37
<u>3. Modelado del Robot para Laparoscopia ‘LapBot’</u>	<u>41</u>
3.1 Estructuras Funcionales de Robots para Laparoscopia	41
3.1.1 Da Vinci	41
3.1.2 Black Falcon	42
3.1.3 Zeus	43
3.1.4 Laprotek	44
3.1.5 Endobot	44
3.2 Estructura Funcional de LapBot	45
3.3 Modelado Matemático de LapBot	47
3.3.1 Modelo Geométrico Directo (MGD)	47
3.3.2 Modelo Geométrico Inverso (MGI) y Restricción por Incisión	49
3.3.3 Prueba de los Modelos Geométricos	54

3.3.4	Modelos Dinámicos	55
3.3.5	Diseño General de los Cuerpos de LapBot	57
4.	Control de LapBot para el Seguimiento de una Colecistectomía	61
4.1	Colecistectomía Laparoscópica	61
4.1.1	Etapas Quirúrgicas	63
4.1.2	Trayectorias para Simulación	65
4.2	Control por Par Calculado (CTC)	69
4.3	Seguimiento de Trayectorias	71
4.3.1	Pruebas Iniciales	71
4.3.2	Procesos de la Colecistectomía	74
5.	Simulación Tridimensional de la Colecistectomía con LapBot	79
5.1	Simulador Tridimensional (3D)	79
5.2	Construcción del Ambiente de Simulación en 3D	81
5.2.1	Enlace de los Cuerpos de LapBot	81
5.2.2	Enlace de LapBot con la Habitación y la Camilla	82
5.2.3	Interfaz Gráfica de Usuario (GUI) y Ambiente 3D	83
5.3	Programación del Ambiente 3D	86
6.	Conclusiones	91
	Referencias Bibliográficas	93
Anexo A.	Parámetros Inerciales de Base para LapBot	101
A.1.	Expresiones Simbólicas	101
A.2.	Valores Numéricos	102
Anexo B.	Manual de Usuario de la Aplicación 3D para LapBot	105
B.1.	Requerimientos	105
B.2.	Instalación	105
B.3.	Manejo	107

Lista de Tablas

	Pág.
Tabla 2.1. Algunas diferencias entre cirugía convencional y MIS.	31
Tabla 2.2. Comparación y clasificación de robots aplicados en cirugía.	35
Tabla 3.1. Parámetros geométricos de LapBot.	47
Tabla 3.2. Parámetros del tensor de inercia de LapBot.	55
Tabla 3.3. Parámetros del primer momento de inercia, masas e inercia de motores.	56
Tabla 3.4. Valores de los parámetros inerciales de base de LapBot.	60
Tabla 4.1. Ganancias para el controlador PD.	71
Tabla 5.1. Comparación de motores de simulación 3D.	80
Tabla A.1. Valores de los parámetros inerciales para cada cuerpo de LapBot.	103
Tabla B.1. Funciones de las teclas y el ratón en la aplicación 3D.	107

Lista de Figuras

	Pág.
Figura 2.1. Aguja de Veress (izquierda) y trocar (derecha) [27].	32
Figura 2.2. Configuración de elementos básicos para laparoscopia.	32
Figura 2.3. Da Vinci (izquierda) y su sistema de visión (derecha) [45].	35
Figura 2.4. Manipuladores (izquierda) y consola de mando (derecha) del Zeus [14].	36
Figura 2.5. Control manual de Endobot [59].	36
Figura 2.6. Black Falcon (izquierda) e interfaz de control Phantom (derecha) [60].	37
Figura 2.7. Simulador para laparoscopia ProMIS [10].	38
Figura 2.8. Simulador de entrenamiento [12].	38
Figura 2.9. Simulador con realimentación de tacto [11].	38
Figura 2.10. Ambiente virtual de un manipulador para laparoscopia [64].	39
Figura 2.11. Simulador 3D para el robot DaVinci [13].	39
Figura 3.1. Estructura cinemática de doble paralelogramo [46].	42
Figura 3.2. Robot Da Vinci con 3 brazos [46].	42
Figura 3.3. Diagrama estructural del robot Black Falcon [60].	43
Figura 3.4. Estructura cinemática del robot Zeus [47].	43
Figura 3.5. Movimiento de las articulaciones pasivas del robot Zeus [53].	44
Figura 3.6. Diagramas estructural y cinemático de Laprotek [55].	44
Figura 3.7. Endobot (izquierda) y la estructura cinemática (derecha) [59].	45
Figura 3.8. Estructura cinemática del robot LapBot.	46
Figura 3.9. Movimiento alrededor del punto de incisión.	50
Figura 3.10. Algoritmo para hallar el MGI por métodos numéricos.	52

Figura 3.11. Diagrama de prueba de los modelos geométricos.	54
Figura 3.12. Seguimiento de trayectoria circular usando los modelos geométricos.	55
Figura 3.13. Máxima elongación para LapBot.	58
Figura 3.14. Diseños en SolidEdge® de la base y los dos primeros cuerpos de LapBot.....	58
Figura 3.15. Diseños en SolidEdge® de los cuerpos para θ_3 , θ_4 y θ_5 de LapBot.	59
Figura 3.16. Diseños en SolidEdge® de los cuerpos para θ_6 , θ_7 y θ_8 de LapBot.	59
Figura 3.17. Diseños en SolidEdge® del cuerpo para θ_9 , la pinza y el cauterizador.	60
Figura 4.1. Órganos biliares (izquierda) y vesícula biliar (derecha) [67].	61
Figura 4.2. Distribución de elementos en el quirófano [68].	62
Figura 4.3. Ubicación de instrumentos quirúrgicos.	62
Figura 4.4. Exposición del conducto y arteria císticos [68].	63
Figura 4.5. Disección del conducto cístico [68].	64
Figura 4.6. Sección del conducto y arteria císticos [68].	64
Figura 4.7. Disección de la vesícula del lecho hepático [68].	65
Figura 4.8. Entorno de trabajo en Blender [70].	66
Figura 4.9. Trayectoria del instrumento 1 para la disección del triángulo de Calot [70].	66
Figura 4.10. Trayectoria del instrumento 2 para la disección del triángulo de Calot [70].	67
Figura 4.11. Trayectoria del instrumento 1 para el corte del conducto cístico [70].	68
Figura 4.12. Trayectoria del instrumento 1 para el corte de la arteria cística [70].	68
Figura 4.13. Trayectoria del instrumento 1 para la disección de la vesícula del lecho hepático [70].	69
Figura 4.14. Esquema general del control CTC.....	70
Figura 4.15. Control CTC con modificaciones para LapBot.	71
Figura 4.16. Trayectoria circular (izquierda) y lineal (derecha).	72
Figura 4.17. Trayectoria tridimensional.	72
Figura 4.18. Seguimiento del círculo (izquierda) y del cambio de dirección (derecha).	73
Figura 4.19. Seguimiento de la trayectoria tridimensional.	73

Figura 4.20. Error de seguimiento cartesiano para el círculo (izquierda) y el cambio de dirección (derecha).....	73
Figura 4.21. Error de seguimiento cartesiano para la trayectoria tridimensional.	74
Figura 4.22. Modelo estilo caja del abdomen del paciente.....	74
Figura 4.23. Seguimiento y error cartesiano del instrumento I1 en la disección del triangulo de Calot.....	75
Figura 4.24. Seguimiento y error cartesiano del instrumento I2 en la disección del triangulo de Calot.....	76
Figura 4.25. Seguimiento y error cartesiano para la sección del conducto cístico. ...	76
Figura 4.26. Seguimiento y error cartesiano para la sección de la artería cística.	77
Figura 4.27. Seguimiento y error cartesiano para la disección de la vesícula.....	77
Figura 5.1. Interfaz de usuario para el ambiente 3D.	84
Figura 5.2. Ambiente 3D con dos LapBot y la interfaz de usuario.....	85
Figura 5.3. Seguimiento 3D de la trayectoria Disección de Calot.....	85
Figura 5.4. Diagrama de flujo del programa principal.....	86
Figura 5.5. Pantalla de opciones de Ogre.	87
Figura B.1. Ventana para adicionar un <i>Path</i> en Matlab.....	106
Figura B.2. Ventana del Símbolo del sistema en Windows.	106

Lista de Abreviaturas

3D:	Tridimensional.
API:	(<i>Application Programming Interface</i>) Interfaz de Programación de Aplicaciones.
CEGUI:	(<i>Crazy Eddie's GUI System</i>) Sistema para crear GUI con Ogre.
CO ₂ :	Dióxido de Carbono.
CTC:	(<i>Computed Torque Control</i>) Control por Par Calculado.
GUI:	(<i>Graphical User Interface</i>) Interfaz Gráfica de Usuario.
LapBot:	Robot para Laparoscopia.
MDD:	Modelo Dinámico Directo.
MDI:	Modelo Dinámico Inverso.
MGD:	Modelo Geométrico Directo.
MGI:	Modelo Geométrico Inverso.
MIS:	(<i>Minimally Invasive Surgery</i>) Cirugía mini-invasiva.
OGRE:	(<i>Object-Oriented Graphics Rendering Engine</i>) Librería Orientada a Objetos de Representación Gráfica.
OpenGL:	(<i>Open Graphics Library</i>) Librería Gráfica de Libre Uso.

Lista de Símbolos

A	Matriz de inercia.
\hat{A}	Matriz de inercia aproximada.
iA_j	Matriz de orientación.
$A_i, B_i, E_i, F_i, M, K, N$	Variables auxiliares para algunas ecuaciones.
C	Vector de fuerzas de Coriolis y centrífugas.
C_i	$\cos(\theta_i)$
C_{ij}	$\cos(\theta_i + \theta_j)$
D, R, d	Distancias fijas [m].
H	Matriz de fuerzas de Coriolis, centrífugas y la gravedad.
\hat{H}	Matriz aproximada de Coriolis, centrífugas y la gravedad.
I_a	Inercia de motor [$\text{kg}\cdot\text{m}^2$].
j	Número de la articulación.
K_p	Vector de ganancias proporcionales.
K_v	Vector de ganancias derivativas.
M_j	Masa de la articulación j [kg].
$MS = [MX \ MY \ MZ]^T$	Primer momento de inercia [$\text{kg}\cdot\text{m}$].
P_i	Vector de posición.
q	Vector de variables articulares.

$\dot{\mathbf{q}}$	Vector de velocidades de las variables articulares.
$\ddot{\mathbf{q}}$	Vector de aceleraciones de las variables articulares.
\mathbf{q}^d	Vector de variables articulares deseadas.
\mathbf{Q}	Vector de fuerzas gravitacionales.
r	Variable articular de traslación [m].
s, n, a	Parámetros de la matriz de orientación.
S_i	$\text{sen}(\theta_i)$
S_{ij}	$\text{sen}(\theta_i + \theta_j)$
t	Trocar
t_x, t_y, t_z	Coordenada cartesiana del trocar [m].
${}^i\mathbf{T}_j$	Matriz de transformación.
\mathbf{U}_o	Matriz de transformación deseada.
\mathbf{w}	Vector de señales de control.
x, y, z	Variables cartesianas [m].
\mathbf{X}	Vector de posición cartesiana real.
\mathbf{X}^d	Vector de posición cartesiana deseada.
XX, XY, XZ, YY, YZ, ZZ	Elementos del tensor de inercias [kg*m ²].
α	Angulo fijo [°].
θ	Variable articular de rotación [rad].
$\mathbf{\Gamma}$	Vector de pares mecánicos [N*m].
μ, σ	Parámetros binarios de las articulaciones.

1. Introducción

El origen de la cirugía se remonta a la Grecia antigua, fue una ciencia artesanal hasta el siglo XIX cuando William Halsted estableció los cimientos de la cirugía moderna [1]. En 1901 Kelling [2] reportó el primer caso de inspección abdominal en un perro de forma mini-invasiva (laparoscopia) y desde entonces se han desarrollado técnicas para mejorar este tipo de procedimientos y ampliar su aplicación a varias clases de cirugía. En la actualidad “representa una nueva teoría quirúrgica” [1].

En la cirugía mini-invasiva, los instrumentos y un sistema de visión son introducidos a través de pequeñas incisiones, esto implica que el paciente se recupera más rápido después de la cirugía, con menos traumas y menores cicatrices respecto a una cirugía convencional [3]. Sin embargo, tiene algunas dificultades tales como: el control de la cámara de video, la dificultad ergonómica en el procedimiento quirúrgico, el entrenamiento del cirujano, la pérdida de sensación táctil sobre el paciente, la precisión, la pérdida de visión tridimensional que se tiene respecto a las cirugías abiertas y la falta de rotación de la articulación de la muñeca del médico [4].

Para resolver las dificultades descritas, se han desarrollado diferentes sistemas robotizados. Camarillo [3] plantea que hasta hoy se han hecho progresos favorables en la integración de tecnologías automatizadas con instrumentos quirúrgicos y la robótica quirúrgica ha mostrado claros beneficios. Sin embargo, este es un campo de investigación aún en desarrollo y como tal, posee ámbitos por mejorar [5].

Las principales investigaciones, desarrollos y aplicaciones sobre robótica quirúrgica se han realizado en países como Estados Unidos, Japón, Francia, Alemania y Europa en general, con resultados similares y algunas veces mejores que los obtenidos en procedimientos de cirugía mini-invasiva sin asistencia robotizada [6], [7]. Pero, para países en desarrollo es más difícil adquirir y aplicar esta tecnología principalmente por su costo, por ejemplo el robot “Da Vinci” cuesta aproximadamente 1 millón de dólares y cada instrumento 2.000 dólares [6].

Aun así, se han hecho proyectos en Colombia de prototipos para cirugía robotizada, que se encuentran aún en proceso de investigación: Kirubot, diseñado por el Grupo de Investigación en Bioingeniería de la Universidad Pontificia Bolivariana que es controlado desde LabView [8] y, un manipulador desarrollado por el Grupo de Automática y Robótica de la Pontificia Universidad Javeriana de Cali para cirugías de

columna vertebral [9]; hechos que demuestran el esfuerzo de los investigadores por incursionar en el campo de la robótica quirúrgica y estar a la vanguardia mundial.

Para este proyecto se tuvo en cuenta que un medio económico para simular equipos costosos es el desarrollo de ambientes computacionales en tres dimensiones (3D), que además permiten observar el comportamiento del equipo en diferentes situaciones [5], siendo el primer paso en el diseño de prototipos reales o en el desarrollo de entrenadores para cirujanos. Ejemplos de simuladores 3D para entrenamiento o planeación de cirugías son los desarrollados por Haptica [10], Delingette y Ayache [11] para simulación sin robots, o el de Suzuki y otros [12] y Adhami y otros [13] que usan el robot Da Vinci en la simulación.

Sin embargo, para simular un robot es necesario conocer su modelo matemático con todos sus parámetros, y dado que los robots para cirugía aún se encuentran en investigación, no es posible obtener del fabricante los modelos y parámetros que los describen. Por ello se decidió diseñar un robot, basado en estructuras conocidas para robots similares y probarlo en simulación para comprobar su posible aplicación en cirugías laparoscópicas.

Los robots más conocidos para cirugía laparoscópica son ARTEMIS, Endobot, Black Falcon, Zeus y Da Vinci [14], [15]. Estos dos últimos son controlados por un cirujano desde una consola de control y se utilizan en gran cantidad de procedimientos quirúrgicos, como por ejemplo la colecistectomía (extracción de una vesícula biliar enferma) [16].

En este proyecto se realiza el modelado matemático y estructural de dos brazos robóticos llamados LapBot, se diseña un controlador por par calculado (CTC) para que el robot siga las trayectorias deseadas de una colecistectomía, teniendo en cuenta la restricción física generada por el punto de incisión en el abdomen del paciente y, finalmente, se construye y usa un ambiente de simulación 3D para observar los movimientos del robot en conjunto con el controlador.

Se ha escogido la colecistectomía por ser una cirugía estandarizada a nivel mundial y una de las más practicadas en Colombia y el mundo [17] - [19]. Es de aclarar que no se usan modelos exactos de la anatomía humana para realizar la simulación, dada la complejidad matemática que tal tarea conlleva, lo cual sobrepasa los objetivos de este estudio.

En el segundo capítulo se presentan algunos conceptos básicos sobre robótica, cirugía mini-invasiva y simulación 3D, que permiten exponer con mayor claridad el resto del documento. En el capítulo tres se modela estructural y matemáticamente a LapBot, además de diseñar los cuerpos que lo conforman. El controlador que permite seguir las trayectorias de la colecistectomía y las pruebas correspondientes con los modelos de LapBot se pueden observar en el capítulo cuatro, el quinto muestra cómo se seleccionó el software para simulación 3D y cómo se construyó el ambiente para que fuese compatible con lo anteriormente desarrollado (modelos y cuerpos de LapBot), y por último se presentan las conclusiones del proyecto.

2. Robótica Aplicada a la Cirugía Mini-Invasiva

A continuación se presentan conceptos básicos sobre robótica y cirugía mini-invasiva, además de un resumido estado del arte sobre cómo ha incursionado la robótica en laparoscopia, y el uso de la simulación tridimensional (3D) como una herramienta de apoyo.

2.1 Conceptos de Manipuladores Robotizados

Ollero [20] define los robots como “máquinas en las que se integran componentes mecánicos, eléctricos, electrónicos y de comunicaciones, que son dotadas de un sistema informático para su control en tiempo real, percepción del entorno y programación”. Un manipulador es esencialmente un brazo articulado, una cadena cinemática abierta formada por un conjunto de elementos interrelacionados mediante articulaciones que permiten el movimiento [20]. Los siguientes son algunos conceptos sobre robots [20], [21]:

- **Tipos de articulaciones:**
 - ♦ **Rotoide:** establece un grado de libertad de rotación alrededor del eje de la articulación.
 - ♦ **Prismática:** establece un grado de libertad de traslación a lo largo del eje de la articulación.
- **Grados de libertad:** definen el número de parámetros independientes que fijan la ubicación y orientación de la herramienta que porta el robot.
- **Espacio articular:** espacio vectorial que representa los valores de las variables asociadas a las articulaciones del robot.
- **Espacio cartesiano o espacio operacional:** representa la localización, posición y orientación de la herramienta. Generalmente se usan coordenadas cartesianas.

- **Modelado de robots:** se usa la dinámica, cinemática y geometría del robot teniendo en cuenta distancias, masas, inercias y frotamientos, para simularlo y controlarlo.
- ♦ **Modelos geométrico directo e inverso:** el Modelo Geométrico Directo (MGD) permite calcular la posición y orientación del robot en el espacio cartesiano, conociendo las variables articulares. Con el Modelo Geométrico Inverso (MGI) se calculan las variables articulares correspondientes a una localización del robot en el espacio cartesiano.
- ♦ **Modelos cinemático directo e inverso:** el Modelo Cinemático Directo (MCD) establece las velocidades cartesianas a partir de las articulares, por medio de una matriz llamada Jacobiana. El Modelo Cinemático Inverso (MCI) relaciona las velocidades de las articulaciones a partir de las velocidades cartesianas deseadas, por medio de la inversa de la matriz Jacobiana.
- ♦ **Modelos dinámico directo e inverso:** el Modelo Dinámico Inverso (MDI) provee los pares articulares y fuerzas en términos de las posiciones, velocidades y aceleraciones articulares. El Modelo Dinámico Directo (MDD) describe las aceleraciones articulares en términos de las posiciones, velocidades y pares. Los modelos dinámicos del robot son importantes para el diseño, control, simulación y operación real. Por ejemplo, el MDD es empleado para llevar a cabo simulaciones con el propósito de examinar el desempeño del robot y estudiar posibles esquemas de control; el MDI, es utilizado en los sistemas de control por modelo de referencia, es decir, aquellos que incluyen el modelo matemático del robot.

2.2 Cirugía Mini-Invasiva (MIS*)

La cirugía mini-invasiva revolucionó el concepto de los procedimientos quirúrgicos. En ésta, los instrumentos quirúrgicos y el equipo de visión son insertados en el cuerpo del paciente a través de pequeñas incisiones. El gran desarrollo de la MIS comenzó a finales de la década de 1980 [3], y desde entonces se ha podido determinar algunas diferencias importantes entre la cirugía convencional y la MIS [1], [3], que se resumen en la Tabla 2.1.

La interfaz de video es una ventaja en cuanto permite visualizar el interior del paciente sin necesidad de realizar grandes incisiones, sin embargo es también una desventaja en la medida que limita la realimentación de fuerzas, visión 3D y ergonomía [3].

* MIS: *Minimally Invasive Surgery*

Cirugía Convencional	Cirugía Mini-Invasiva
Basada en la anatomía.	Basada también en simulación virtual.
Interfaz sensorial directa.	Interfaz video-endoscópica.
Depende de la habilidad del cirujano.	Depende también de la tecnología.
Gran cantidad de herramientas.	Cantidad limitada de herramientas.
Más tiempo de recuperación del paciente.	Menos tiempo de recuperación del paciente.
Grandes marcas estéticas.	Pequeñas marcas estéticas.

Tabla 2.1. Algunas diferencias entre cirugía convencional y MIS.

La MIS se puede clasificar según el área del cuerpo a operar, así [22]:

- ♦ Laparoscopia: área abdominal o cavidad peritoneal.
- ♦ Toracoscopia: área torácica.
- ♦ Extracavitaria: en regiones diferentes al peritoneo y al tórax.

La laparoscopia es la cirugía mini-invasiva que se ha escogido para trabajar en este proyecto porque es una de las más desarrolladas a nivel mundial y nacional [18], [19], la cual permite el acceso a mayor cantidad de órganos internos del paciente y tiene menos complicaciones quirúrgicas que las demás MIS [1].

2.2.1 Laparoscopia

En 1901, Kelling [2] reportó el primer caso de inspección de la cavidad abdominal en un perro, con la ayuda de un cistoscopio (instrumento médico que sirve para explorar el interior de la vejiga, introducido a través de un orificio cutáneo). Jacobaeus [23], en 1912, reportó el empleo de un cistoscopio para el examen de las cavidades abdominal y pleural, siendo el primero en realizar laparoscopias en humanos. Un problema de la laparoscopia era 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 [24].

Mühe [25], en 1985, realizó la primera colecistectomía laparoscópica, dando inicio a la laparoscopia moderna [1], y en 1987 Mouret [26] desarrolló un estudio completo sobre la misma cirugía, de tal magnitud que en 1992 se convirtió en un estándar [17].

En laparoscopia es necesario 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 realizó en un principio, con insuflación por aire, después se usó óxido nítrico y actualmente se emplea 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 [27]. En la Figura 2.1 se puede observar una aguja de Veress y un trocar.



Figura 2.1. Aguja de Veress (izquierda) y trocar (derecha) [27].

Por medio de los trocares se insertan pinzas, tijeras, cauterizadores o cualquier otro instrumento que se requiera, además del sistema de visión (endoscopio) [22]. La Figura 2.2 muestra la configuración de los elementos, los trocares aparecen de color violeta y el sistema de visión posee el cable de luz óptica.

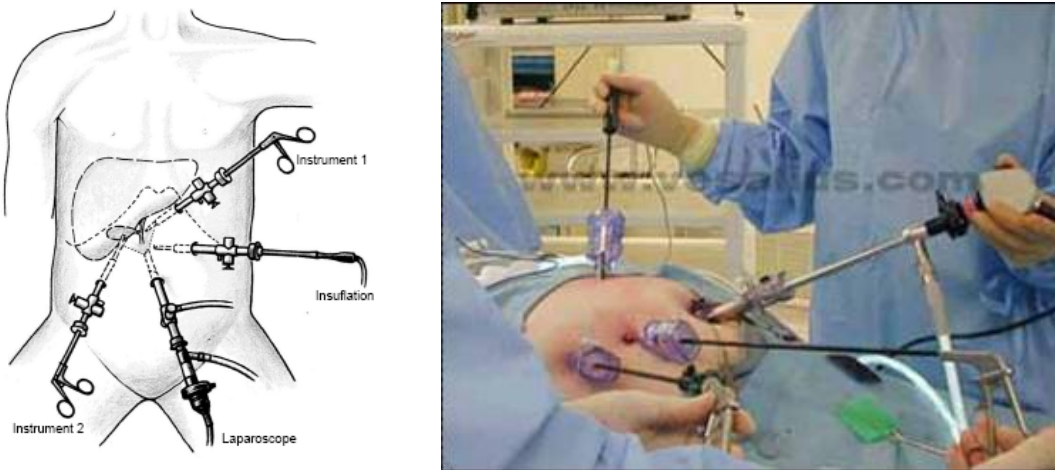


Figura 2.2. Configuración de elementos básicos para laparoscopia†.

La cirugía laparoscópica puede ser clasificada como funcional o ablativa. La primera incluye procedimientos reparativos y la segunda supone la extirpación de órganos o algunas de sus partes [28]. Actualmente, los procedimientos quirúrgicos laparoscópicos más comunes son:

- ♦ **Colecistectomía:** intervención que se realiza para quitar una vesícula biliar enferma, que está inflamada u obstruida por cálculos biliares [29].

† www.versatius.com

- ♦ **Apendicectomía:** extirpación de un apéndice inflamado o infectado (apendicitis). Se hace una incisión pequeña en el cuadrante inferior derecho del abdomen y se extirpa el apéndice [30].
- ♦ **Histerectomía:** cirugía para extirpar el útero (matriz). El cirujano realiza una incisión por la cual corta las trompas de Falopio y separa el útero en su unión con el cuello [31].
- ♦ **Colectomía asistida:** extirpación o resección de una parte enferma del intestino grueso (colon) [22].
- ♦ **Ligadura de trompas:** es una forma permanente de evitar el embarazo cerrando las trompas de Falopio de una mujer [32].
- ♦ **Gastrectomía:** extirpación parcial o total del estómago, en caso de presentarse problemas gástricos crónicos como úlceras o cáncer [33].
- ♦ **Prostatectomía:** intervención para extraer la totalidad o parte de la glándula prostática (próstata), para la curación del cáncer, la preservación de la continencia o la función sexual [34].
- ♦ **Hepatectomía:** procedimiento para diagnosticar, tratar o evacuar las lesiones y tumores en el hígado [35].
- ♦ **Vasectomía:** procedimiento para lograr la esterilización masculina. Consiste en cortar y ligar los conductos deferentes, impidiendo el recorrido del semen [36].

2.3 Robots en Cirugía

El desarrollo de robots prácticos es un fenómeno reciente, los primeros manipuladores industriales programables fueron desarrollados a finales de la década de 1940. Ya en 1961, Engelberger formó una compañía llamada Unimation y empezó la producción de robots comerciales para la industria [37].

En 1988, Kwoh y otros [38] aplicaron por primera vez un robot en cirugía, usando un robot industrial Puma 560 en una cirugía de cerebro con el fin de insertar una guía e introducir un instrumento que extrajera una muestra para biopsia. Sin embargo en 1989, Davies y otros [39] fueron los primeros en usar un robot para actividades invasivas en una cirugía de próstata. Otros hitos históricos importantes son [4], [40]:

1992 Aparición del robot ROBODOC, usado para cirugía ortopédica.
 1994 Creación del robot AESOP para MIS.
 1996 Primera laparoscopia usando un robot asistencial.
 1998 Lanzamiento del Robot Zeus aplicado a la MIS.
 1999 Aparición del robot Da Vinci para MIS.
 2001 Tele-cirugía realizada desde New York hasta Strasbourg, Francia.

2.3.1 Clasificación

En el campo quirúrgico existen robots para: urología, radiocirugía, neurocirugía, laparoscopia, ortopedia, cirugía reconstructiva (maxilofacial), cardiología, toracoscopia, otorrinolaringología, endoscopia, terapias percutáneas, cirugía de retina y microcirugía [3], [4], [41]. Pero, Taylor [42] menciona que la clasificación de estos robots se puede basar en la tecnología, la aplicación o el rol del robot en la sala quirúrgica. Según la tecnología existen: teleguiados y autónomos; según el rol pueden existir tres tipos de robots [3]:

- ♦ **Rol Pasivo:** robot limitado en alcance y usado en situaciones de poco riesgo.
- ♦ **Rol Restringido:** robot responsable de tareas invasivas y de riesgo, pero todavía está restringido a partes esenciales del procedimiento quirúrgico.
- ♦ **Rol Activo:** robot íntimamente involucrado en el procedimiento quirúrgico con responsabilidad en situaciones de alto riesgo.

2.3.2 Robots más Usados en Cirugía

En la Tabla 2.2 se presenta un resumen de los robots más usados en cirugía [3], [41], [43] - [63]. Los robots importantes en laparoscopia actualmente, tanto en investigación como en aplicación se resaltan.

Robot	Tipo	Aplicación	Fabricante/Desarrollador
Padyc	Restringido y algo autónomo	Cardiología	Schneider y otros
Ler	Activo y teleguiado	Endoscopia	Berkelman y otros
EndoAssist	Restringido y dirigido	Endoscopia	Armstrong Healthcare
AESOP	Restringido y algo autónomo	Endoscopia	Intuitive Surgical Systems
Da Vinci	Activo y teleguiado	Laparoscopia, Cardiología, Urología.	Intuitive Surgical Systems
Zeus	Activo y teleguiado	Laparoscopia, Cardiología, Urología.	Intuitive Surgical Systems
Endobot	Activo y algo autónomo	Laparoscopia	Kang y Wen
Black Falcon	Activo y teleguiado	Laparoscopia	Madhani

ARTEMIS	Activo y teleguiado	Laparoscopia	Karlsruhe Nuclear Research Center and The University of Tuebingen
Pathfinder	Restringido y algo autónomo	Neurocirugía	Armstrong Healthcare
Minerva	Activo	Neurocirugía	Universidad de Lausanne
NeuroMate	Activo y poco autónomo	Neurocirugía	Integrated Surgical Systems
Neurobot	Restringido	Neurocirugía	Colegio Imperial de Londres
Acrobot	Activo y algo autónomo	Ortopedia	Acrobot Company
RoboDoc	Restringido y algo autónomo	Ortopedia	Integrated Surgical Systems
Caspar	Restringido y autónomo	Ortopedia	URS Ortho
CyberKnife	Pasivo y Autónomo	Radiocirugía	Accuray
Probot	Restringido y autónomo	Urología	Imperial Collage of UK

Tabla 2.2. Comparación y clasificación de robots aplicados en cirugía.

En los años 1980's Phil Green desarrolló en el Instituto de Investigación de *Stanford* (SRI) el sistema quirúrgico que actualmente se conoce como **Da Vinci** [44], el cual consta de una consola de control ergonómica, una camilla con 4 brazos robóticos interactivos, el sistema de visión *InSite®* de gran rendimiento y los instrumentos propietarios de *EndoWrist®*. Los movimientos del cirujano son escalados y filtrados (para eliminar el temblor) y luego transmitidos con precisión a los instrumentos. En la Figura 2.3, se puede observar el sistema completo a la izquierda y el sistema de visión 3D de alta resolución con imágenes reales a la derecha [36], [45].



Figura 2.3. Da Vinci (izquierda) y su sistema de visión (derecha) [45].

En 1989, Yulun Wang fundó la empresa *Computer Motion* que desarrolló los sistemas robotizados AESOP, Hermes y **Zeus** [57]. Este último es una plataforma modular (Figura 2.4) que contiene a AESOP para el manejo de la cámara, una mesa de cirugía con brazos independientes que emplean instrumentos miniaturizados de 5 grados de libertad y una consola de mando para el cirujano.



Figura 2.4. Manipuladores (izquierda) y consola de mando (derecha) del Zeus [14].

Puede ser usado en tele-cirugía a grandes distancias, demostrado en el año 2001 por el grupo de trabajo del profesor Marescaux [58], el cual realizó la primera laparoscopia transatlántica robotizada desde Nueva York hasta Strasbourg, en Francia. Un defecto de este robot es la falta de integración del sistema: sus brazos trabajan por separado y no "saben" dónde están en el espacio, limitando el potencial de integración del robot con datos para imágenes tridimensionales [26].

El sistema **Endobot** mostrado en la Figura 2.5, es un prototipo que permite establecer control sobre el brazo esclavo usando restricciones artificiales, con el objetivo de limitar los movimientos del instrumento quirúrgico [59]. Con este sistema se ha experimentado en la automatización de la sutura quirúrgica donde se necesita la realización de nudos [16].

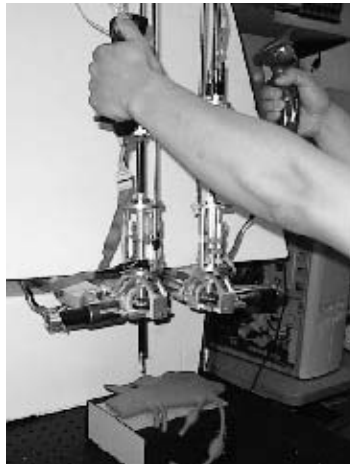


Figura 2.5. Control manual de Endobot [59].

Black Falcon es el manipulador que se presenta en la Figura 2.6, se caracteriza por ser eficiente en la realización de suturas complejas. Tiene adaptada una interfaz háptica Phantom modificada que permite controlar los movimientos del robot y realimentar al cirujano los esfuerzos entre el instrumento y los tejidos [60].

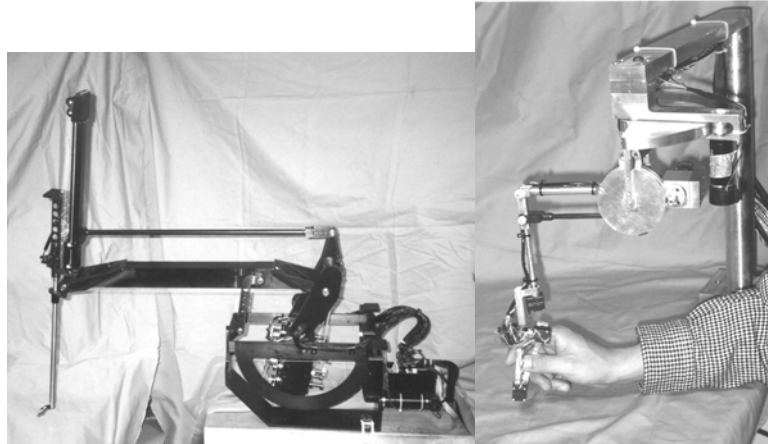


Figura 2.6. Black Falcon (izquierda) e interfaz de control Phantom (derecha) [60].

ARTEMIS (*Advanced Robot and Telem manipulator System for Minimally Invasive Surgery*) fue un proyecto desarrollado en Alemania por Hermann Rinnsland, del Centro de Investigación Nuclear de Karlsruhe y Gerhard Buess de la Universidad de Tuebingen. Es un sistema similar al del SRI pero con algunas diferencias en la consola de control, donde los manipuladores pasan por encima del hombro para dar un grado de libertad adicional. El sistema fue eficiente pero el proyecto no se renovó y por tanto sólo quedó en prototipo [15].

2.4 Simulación en 3D

La simulación tridimensional (3D) de sistemas reales permite observar el comportamiento de éstos en diferentes situaciones sin tener que construir costosos prototipos físicos, visualizar con mayor claridad las respuestas del sistema a cualquier evento (así el evento no suceda en la realidad), y realizar cambios al modelo del sistema hasta obtener los mejores resultados de modelado y/o diseño [5].

2.4.1 Ejemplos para Laparoscopia

En laparoscopia se han desarrollado ambientes virtuales principalmente para simular la anatomía del paciente y son muy usados en el entrenamiento de cirujanos. Algunos ejemplos de estos simuladores son:

- ♦ El simulador quirúrgico ProMIS para entrenamiento de cirujanos, desarrollado por la empresa Haptica, que posee cinco módulos: para laparoscopia, manejo de instrumentos, disección, diatermia y suturado. En la Figura 2.7 se muestra el simulador ProMIS en el módulo para laparoscopia [10].



Figura 2.7. Simulador para laparoscopia ProMIS [10].

- ♦ El simulador para entrenamiento en Internet desarrollado por Suzuki y otros [12]. La Figura 2.8 muestra la interfaz gráfica del simulador donde se observa la extirpación de un órgano y el porcentaje de precisión del cirujano. Las pinzas que se emplean en la simulación son las que usa el robot Da Vinci.

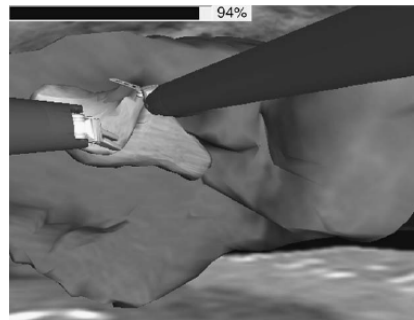


Figura 2.8. Simulador de entrenamiento [12].

- ♦ Sistema para entrenamiento de cirujanos desarrollado por Delingette y Ayache [11], en el cual se simula una cirugía hepática y permite al cirujano tener la sensación de estar tocando el órgano por medio de una interfaz háptica, como se observa en la Figura 2.9.



Figura 2.9. Simulador con realimentación de tacto [11].

También se han desarrollado simuladores que no utilizan los modelos de la anatomía humana o que son sólo para planificación de cirugías, pero no para entrenamiento.

- ♦ Un ejemplo es el simulador de un manipulador para laparoscopia citado por Ortmaier [64], que se muestra en la Figura 2.10.

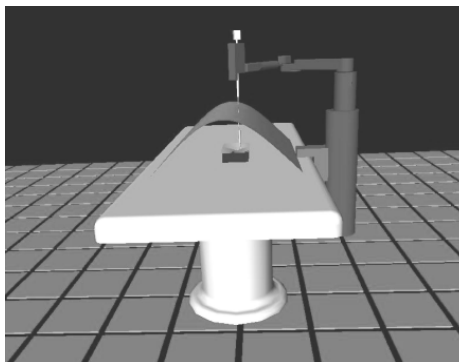


Figura 2.10. Ambiente virtual de un manipulador para laparoscopia [64].

- ♦ El robot Da Vinci también ha sido simulado en un ambiente tridimensional [13], con el fin de realizar la planificación de cirugías, especialmente de corazón. La interfaz del simulador se muestra en la Figura 2.11.

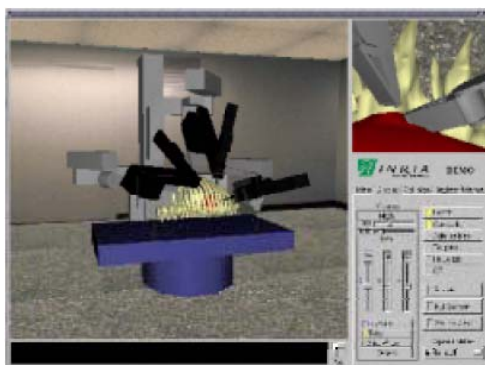


Figura 2.11. Simulador 3D para el robot DaVinci [13].

3. Modelado del Robot para Laparoscopia ‘LapBot’

Este capítulo muestra las estructuras funcionales de robots importantes en MIS y cómo a partir de éstas se definió la estructura de LapBot. Además de presentar la obtención de los modelos matemáticos teniendo en cuenta la restricción del paso por el trocar y cómo al final se diseñan los cuerpos que componen al robot para extraer los valores numéricos de parámetros dinámicos.

3.1 Estructuras Funcionales de Robots para Laparoscopia

A continuación se presenta un resumen de las diferentes topologías de los robots más sobresalientes que han sido diseñados como asistentes en operaciones de laparoscopia.

3.1.1 Da Vinci

Es comercializado por la empresa *Intuitive Surgical* [45], y tiene licencia de la FDA (*Food and Drug Administration*) de los Estados Unidos para ser empleado en pacientes humanos desde 1999 [40]. Se trata de un sistema teleoperado, donde el cirujano está sentado frente a una consola observando en una pantalla la imagen tridimensional de los órganos, y manejando dos instrumentos maestros hápticos que mueven los respectivos instrumentos quirúrgicos esclavos ubicados al interior del paciente.

El portador (Figura 3.1) posee una cinemática cerrada que le proporciona un centro de rotación remoto, de tal forma que el robot siempre se posiciona a través del punto de incisión por medio de una estructura en doble paralelogramo.

El robot posee 13 grados de libertad en total, 7 activos o motorizados, de los cuales se representan 4 en color rojo (2 con el punto y 2 con el rectángulo) en la Figura 3.1, los otros 3 están dentro del paciente y controlan el movimiento de la pinza. Otros 6

son pasivos o no motorizados, hacen parte de la estructura de doble paralelogramo y aparecen en color verde. Los dos primeros grados de libertad (punto rojo en la figura) deben estar alineados con el punto de incisión para que el movimiento mantenga el punto de pivote sobre dicha incisión.

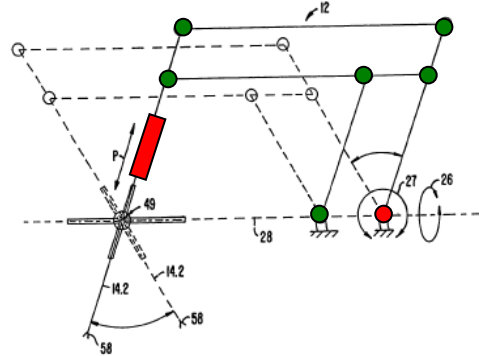


Figura 3.1. Estructura cinemática de doble paralelogramo [46].

La base de este robot soporta tres manipuladores de doble paralelogramo (un porta endoscopio y dos brazos quirúrgicos). En la Figura 3.2 se observa la estructura de soporte y los tres brazos del robot.

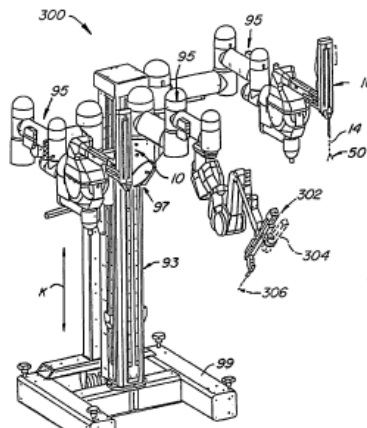


Figura 3.2. Robot Da Vinci con 3 brazos [46].

3.1.2 Black Falcon

Desarrollado a manera experimental por Madhani en 1998 en el MIT (*Massachusetts Institute of Technology*) [60]. Posee la misma cinemática de paralelogramo que se muestra en la Figura 3.1 para el robot Da Vinci, pero es balanceado con un contrapeso gracias a la masa misma de sus motores, permitiendo que la estructura ubicada sobre el paciente sea mínima (Ver Figura 2.6). Tiene 8 grados de libertad activos, los 4 primeros se localizan fuera del paciente y los otros 4 controlan el movimiento fino de la herramienta, los demás grados de libertad son pasivos como lo

muestra la Figura 3.3. Alcanza una profundidad entre 0.038 y 0.24 m después de atravesar la incisión, aunque la barra del brazo es de 0.38m.

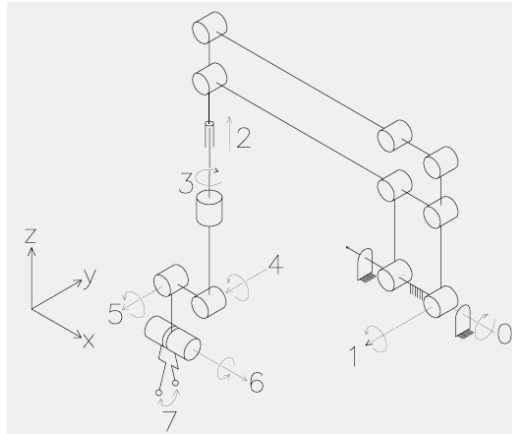


Figura 3.3. Diagrama estructural del robot Black Falcon [60].

3.1.3 Zeus

Fue desarrollado y comercializado por la empresa *Computer Motion*, actualmente fusionada con *Intuitive Surgical*. Igual que el sistema Da Vinci, tiene licencia de la FDA para ser empleado en pacientes humanos desde 1999 [40] y también es un sistema teleoperado. La cadena cinemática del robot se presenta en la Figura 3.4.

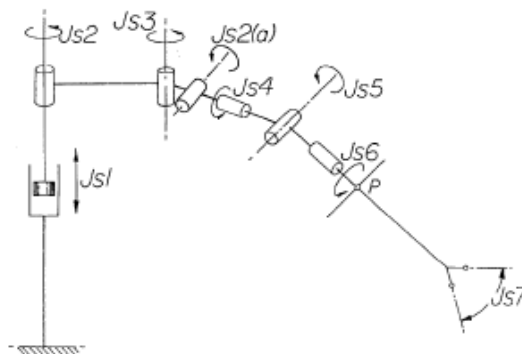


Figura 3.4. Estructura cinemática del robot Zeus [47].

Tiene una cinemática serial con 6 grados de libertad activos ($Js1$, $Js2$, $Js3$, $Js6$, $Js7$ y otra de la muñeca del instrumento) y 3 pasivos ($Js2a$, $Js4$ y $Js5$) de los cuales $Js2a$ tiene valor de ángulo fijo. Las articulaciones activas permiten orientar y posicionar al robot en cualquier punto dentro del abdomen del paciente y las pasivas permiten mantener un movimiento de pivote en el punto de incisión como se puede observar en la Figura 3.5.

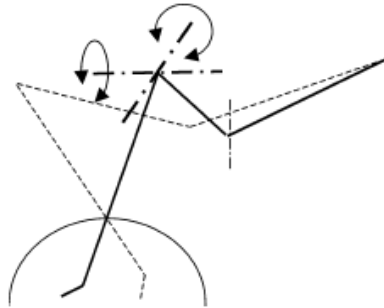


Figura 3.5. Movimiento de las articulaciones pasivas del robot Zeus [53].

3.1.4 Laprotek

Es un sistema robotizado que está siendo elaborado por la empresa *EndoVia Medical* en Boston (Estados Unidos) y evaluado por la FDA para su licenciamiento en el uso con humanos [54].

La Figura 3.6 presenta el diagrama estructural del robot, donde la letra P indica el punto de incisión en el paciente, mostrándose además el diagrama cinemático correspondiente [55].

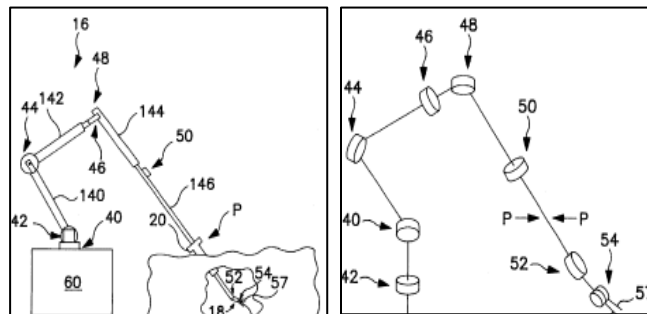


Figura 3.6. Diagramas estructural y cinemático de Laprotek [55].

Este robot tiene 2 articulaciones pasivas que permiten mantener un movimiento libre a través del punto P, dichas articulaciones se muestran en el diagrama con los números 46 y 48. En total tiene 8 articulaciones que le permiten posicionar y orientar la herramienta quirúrgica en un espacio tridimensional manteniendo un punto fijo P.

3.1.5 Endobot

Fue diseñado con propósitos de investigación por Kang y Wen en el *Rensselaer Polytechnic Institute* de Nueva York [59]. En la Figura 3.7 se pueden observar los dos instrumentos del robot y su estructura cinemática.

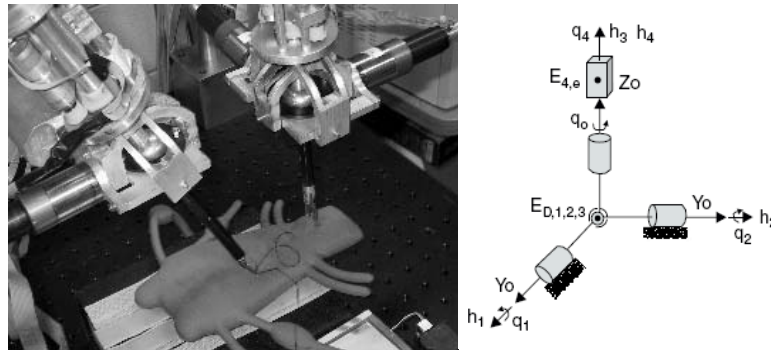


Figura 3.7. Endobot (izquierda) y la estructura cinemática (derecha) [59].

Este robot está basado en una articulación esférica y dos arcos semicirculares que se localizarían sobre el punto de incisión, es un diseño compacto y liviano que ofrece el mínimo de articulaciones. Está provisto de 4 grados de libertad en el manipulador (3 de rotación y uno de translación) y 2 grados de libertad en la herramienta, para un total de 6. Sin embargo la estructura física del robot quedaría sobre el paciente, dificultando el desarrollo de una cirugía real.

3.2 Estructura Funcional de LapBot

Después del estudio de las diferentes topologías estructurales presentadas, se ha definido que la estructura de un robot asistente para cirugía laparoscópica debe cumplir con los siguientes requisitos mínimos:

- Seis grados de libertad activos que le proporcionen la posibilidad de orientar y posicionar la pinza o herramienta en cualquier lugar dentro de un espacio tridimensional, en este caso la cavidad abdominal del paciente.
- Dos grados de libertad de los seis activos deben ubicarse al interior del abdomen con el fin de orientar de forma fina la herramienta, quedando la mayor parte del robot por fuera del cuerpo del paciente.
- Por lo menos dos grados de libertad pasivos que le permitan mantener un centro de rotación remoto donde se ubicaría el trocar, de tal forma que el robot no tenga la posibilidad de perder esa referencia. Es de anotar que otro enfoque normalmente utilizado es no utilizar grados de libertad pasivos para respetar el paso por el trocar, sino algoritmos presentes en el modelo geométrico que tengan en cuenta estas restricciones [56], la desventaja de esta solución es que el sistema no es intrínsecamente seguro, dado que tiene la posibilidad motora de perder el centro de rotación remoto (la ubicación del trocar) y por tanto podría lesionar al paciente.

Teniendo en cuenta las condiciones anteriores, para el diseño de este robot (LapBot) se escogió una estructura de nueve grados de libertad, con seis articulaciones activas que permiten posicionar y orientar el instrumento quirúrgico en un espacio tridimensional; y tres pasivas, también muy importantes, que permiten mantener el punto fijo de movimiento sobre el punto de incisión donde se encuentra el trocar. Con estos 3 grados de libertad se garantiza que el robot es intrínsecamente seguro y que no dañará al paciente por un esfuerzo agresivo en la incisión.

Se descartó la estructura cerrada de doble paralelogramo por tener demasiadas articulaciones pasivas que complican en exceso el análisis e incrementan el peso y el costo del robot.

En la Figura 3.8 se puede observar la estructura cinemática que se escogió para LapBot, las distancias del robot se han etiquetado con letras D y R (según la Tabla 3.1, que muestra los valores de sus parámetros geométricos), la articulación de traslación se representa con un prisma y las de rotación con cilindros, cada una con su respectivo número.

El punto de incisión por donde cruza el trocar en la cirugía se ha representado con un anillo que se encuentra entre las articulaciones 7 y 8, de tal forma que sólo dos articulaciones quedan dentro de la cavidad abdominal del paciente.

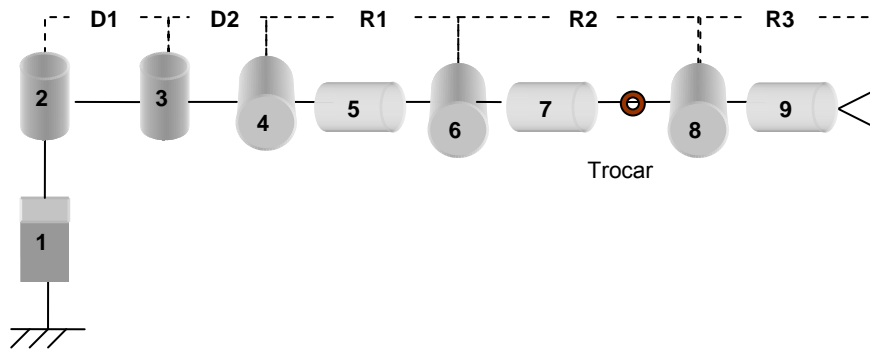


Figura 3.8. Estructura cinemática del robot LapBot.

Las tres primeras articulaciones permiten posicionar a LapBot en un espacio tridimensional, la cuarta articulación se fija en un valor determinado para inclinar al robot y ubicarlo por encima del abdomen del paciente, las articulaciones quinta y sexta son pasivas y mantienen el movimiento del cuerpo de longitud $R2$ a través del trocar, y las últimas tres articulaciones forman una estructura en forma de muñeca que permite la orientación de la herramienta.

3.3 Modelado Matemático de LapBot

Después de escoger la estructura se procede a calcular los modelos matemáticos del robot para verificar que puede realizar movimientos quirúrgicos respetando el punto de incisión, además estos modelos son útiles para realizar el control y la simulación por computador.

3.3.1 Modelo Geométrico Directo (MGD)

Se analiza la estructura del robot, se colocan ejes coordenados en cada articulación y se obtiene la tabla de parámetros geométricos (Tabla 3.1) siguiendo la metodología descrita por Khalil y Dombre [21].

j	μ_j	σ_j	α_j	d_j	θ_j	r_j
1	1	1	0	0	0	r_1
2	1	0	0	0	θ_2	0
3	1	0	0	$D1$	θ_3	0
4	0	0	90°	$D2$	θ_4	0
5	0	0	90°	0	θ_5	$R1$
6	0	0	-90°	0	θ_6	0
7	1	0	90°	0	θ_7	$R2$
8	1	0	-90°	0	θ_8	0
9	1	0	90°	0	θ_9	0
10	0	1	0	0	0	$R3$

Tabla 3.1. Parámetros geométricos de LapBot.

Donde:

j representa el número de la articulación o cuerpo.

μ_j indica si la articulación es activa (con motor) (1) o pasiva (sin motor) (0).

σ_j indica si la articulación es de traslación (1) o de rotación (0).

α_j , θ_j son ángulos que dependen de los ejes de las articulaciones o de la rotación de una articulación rotoide.

d_j , r_j son distancias entre los ejes de las articulaciones o el desplazamiento de una articulación prismática.

A partir de esta tabla se puede encontrar el Modelo Geométrico Directo (MGD), el cual permite calcular la posición y orientación del efector final del robot en el espacio cartesiano, conociendo los valores θ_j y r_j asociados a las articulaciones, haciendo uso de las matrices de transformación entre los sistemas coordenados de cada articulación. La ecuación (3.1) muestra la matriz de transformación del sistema j al sistema i , \mathbf{A} es una matriz 3x3 que representa la orientación y \mathbf{P} (x , y , z) es un vector columna que representa la posición del instrumento.

$${}^i T_j = \begin{bmatrix} {}^i A_j & {}^i P_j \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (3.1)$$

Dado que la punta del efector final de LapBot está ubicada en el sistema coordenado 10 (ver Tabla 3.1) y la base se estipula como el sistema coordenado 0, el MGD está determinado por la matriz de transformación ${}^0 T_{10}$ de la cual se pueden obtener las siguientes ecuaciones, que relacionan la posición del instrumento con las variables articulares:

$$x = ((((C23C4C5 + S23S5)C6 - C23S4S6)C7 + (-C23C4S5 + S23C5)S7)S8 + ((C23C4C5 + S23S5)S6 + C23S4C6)C8)R3 - (-(C23C4C5 + S23S5)S6 - C23S4C6)R2 + C23S4R1 + C23D2 + C2D1 \quad (3.2)$$

$$y = ((((S23C4C5 - C23S5)C6 - S23S4S6)C7 + (-S23C4S5 - C23C5)S7)S8 + ((S23C4C5 - C23S5)S6 + S23S4C6)C8)R3 - (-(S23C4C5 - C23S5)S6 - S23S4C6)R2 + S23S4R1 + S23D2 + S2D1 \quad (3.3)$$

$$z = (((S4C5C6 + C4S6)C7 - S4S5S7)S8 + (S4C5S6 - C4C6)C8)R3 - (-S4C5S6 + C4C6)R2 - C4R1 + r1 \quad (3.4)$$

Donde: $C_i = \cos(\theta_i)$
 $S_i = \sin(\theta_i)$
 $C_{ij} = \cos(\theta_i + \theta_j)$
 $S_{ij} = \sin(\theta_i + \theta_j)$

Al comparar la matriz de orientación ${}^0 A_{10}$ con la forma general que se presenta en la ecuación (3.5),

$${}^i A_j = \begin{bmatrix} s_x & n_x & a_x \\ s_y & n_y & a_y \\ s_z & n_z & a_z \end{bmatrix}, \quad (3.5)$$

se encuentran las ecuaciones que representan la orientación del efector final en cada eje coordenado:

$$s_x = (((((C23C4C5 + S23S5)C6 - C23S4S6)C7 + (-C23C4S5 + S23C5)S7)C8 - ((C23C4C5 + S23S5)S6 + C23S4C6)S8)C9 + (-(C23C4C5 + S23S5)C6 - C23S4S6)S7 + (-C23C4S5 + S23C5)C7)S9 \quad (3.6)$$

$$s_y = (((((S23C4C5 - C23S5)C6 - S23S4S6)C7 + (-S23C4S5 - C23C5)S7)C8 - ((S23C4C5 - C23S5)S6 + S23S4C6)S8)C9 + (-(S23C4C5 - C23S5)C6 - S23S4S6)S7 + (-S23C4S5 - C23C5)C7)S9 \quad (3.7)$$

$$s_z = (((S4C5C6 + C4S6)C7 - S4S5S7)C8 - (S4C5S6 - C4C6)S8)C9 + (- (S4C5C6 + C4S6)S7 - S4S5C7)S9 \quad (3.8)$$

$$n_x = - (((C23C4C5 + S23S5)C6 - C23S4S6)C7 + (-C23C4S5 + S23C5)S7)C8 - ((C23C4C5 + S23S5)S6 + C23S4C6)S8)S9 + (-((C23C4C5 + S23S5)C6 - C23S4S6)S7 + (-C23C4S5 + S23C5)C7)C9 \quad (3.9)$$

$$n_y = - (((S23C4C5 - C23S5)C6 - S23S4S6)C7 + (-S23C4S5 - C23C5)S7)C8 - ((S23C4C5 - C23S5)S6 + S23S4C6)S8)S9 + (-((S23C4C5 - C23S5)C6 - S23S4S6)S7 + (-S23C4S5 - C23C5)C7)C9 \quad (3.10)$$

$$n_z = - (((S4C5C6 + C4S6)C7 - S4S5S7)C8 - (S4C5S6 - C4C6)S8)S9 + (- (S4C5C6 + C4S6)S7 - S4S5C7)C9 \quad (3.11)$$

$$a_x = (((C23C4C5 + S23S5)C6 - C23S4S6)C7 + (-C23C4S5 + S23C5)S7)S8 + ((C23C4C5 + S23S5)S6 + C23S4C6)C8 \quad (3.12)$$

$$a_y = (((S23C4C5 - C23S5)C6 - S23S4S6)C7 + (-S23C4S5 - C23C5)S7)S8 + ((S23C4C5 - C23S5)S6 + S23S4C6)C8 \quad (3.13)$$

$$a_z = ((S4C5C6 + C4S6)C7 - S4S5S7)S8 + (S4C5S6 - C4C6)C8. \quad (3.14)$$

3.3.2 Modelo Geométrico Inverso (MGI) y Restricción por Incisión

El Modelo Geométrico Inverso (MGI) permite calcular los valores de las variables θ_j y r_j asociadas a las articulaciones, dependiendo de la orientación y localización deseada del efector final del robot en el espacio cartesiano (x, y, z). Es decir este modelo calcula lo contrario al MGD.

Para hallar el MGI de un robot con seis grados de libertad se puede usar el método de Paul [65], pero dado que LapBot posee ocho grados de libertad móviles y uno fijo, se hace necesario encontrar otras ecuaciones linealmente independientes adicionales a las que ofrece el método de Paul para hallar el MGI completo. Estas ecuaciones se pueden hallar por la restricción espacial que aparece cuando el robot debe pasar a través de una pequeña incisión y moverse sobre ésta (ver Figura 3.9).

Las articulaciones 6 y 8 se representan por las posiciones cartesianas $P_6(x_6, y_6, z_6)$ y $P_8(x_8, y_8, z_8)$ respectivamente, y el punto de incisión (trocar) está ubicado en $P_t(x_t, y_t, z_t)$. El vector desde P_6 hasta P_8 debe ser colineal con el vector entre P_6 y P_t , lo cual se puede expresar por medio del siguiente producto vectorial:

$$(P_t - P_6) \times (P_8 - P_t) = 0 \quad (3.15)$$

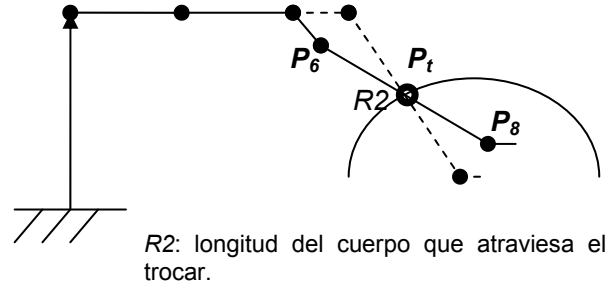


Figura 3.9. Movimiento alrededor del punto de incisión.

De aquí aparecen 2 ecuaciones linealmente independientes. Una ecuación adicional se puede definir teniendo en cuenta que LapBot debe mantener parte del cuerpo de longitud $R2$ dentro del abdomen del paciente, por lo tanto la distancia entre la posición de P_t y P_6 más la distancia entre la posición de P_8 y P_t debe ser igual a dicha longitud. Esta restricción se modela por medio de la ecuación (3.16).

$$|P_t - P_6| + |P_8 - P_t| = R2 \quad (3.16)$$

Una vez definidas las ecuaciones necesarias para resolver el problema del MGI, se hallan las variables articulares más lejanas a la base del robot. En este caso, para las que están relacionadas con la orientación de la herramienta (θ_9 , θ_8 , y θ_7), se utiliza parte del método de Paul, el cual define una ecuación de igualdad entre el MGD y una matriz Uo deseada (3.17).

$$MGD = {}^0T_{10} = {}^0T_6 {}^6T_{10} = Uo = \begin{bmatrix} s_x & n_x & a_x & x \\ s_y & n_y & a_y & y \\ s_z & n_z & a_z & z \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (3.17)$$

Dado que las matrices 0T_6 y ${}^6T_{10}$ se pueden hallar usando la Tabla 3.1 y suponiendo que se conocen los componentes de Uo , se multiplica la ecuación (3.17) por la inversa de 0T_6 , como muestra la ecuación (3.18).

$${}^6T_{10} = inv({}^0T_6) * Uo \quad (3.18)$$

$$\begin{bmatrix} C7C8C9-S7S9 & -C7C8S9-S7C9 & C7S8 & C7S8R3 \\ S8C9 & -S8S9 & -C8 & -C8R3-R2 \\ S7C8C9+C7S9 & -S7C8S9+C7C9 & S7S8 & S7S8R3 \\ 0 & 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} s_{x6} & n_{x6} & a_{x6} & x_6 \\ s_{y6} & n_{y6} & a_{y6} & y_6 \\ s_{z6} & n_{z6} & a_{z6} & z_6 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Donde los valores de la matriz de la derecha están en función de las articulaciones r_1 , θ_2 , θ_3 , θ_4 , θ_5 , θ_6 . Se despejan las variables articulares de interés (θ_9 , θ_8 , y θ_7), de tal forma que:

$$\theta_9 = \arctan\left(\frac{-n_{y6}}{s_{y6}}\right) \quad (3.19)$$

$$\theta_8 = \arctan\left(\frac{\sqrt{a_{x6}^2 + a_{z6}^2}}{-a_{y6}}\right) \quad (3.20)$$

$$\theta_7 = \arctan\left(\frac{a_{z6}}{a_{x6}}\right) \quad (3.21)$$

Para hallar las demás variables articulares se probaron dos métodos diferentes, que se presentan a continuación.

♦ Por Métodos Numéricos.

Es posible encontrar los puntos \mathbf{P}_6 y \mathbf{P}_8 en función de variables articulares usando las matrices de transformación 0T_6 y 0T_8 y conociendo el punto \mathbf{P}_t :

$$\begin{aligned} \mathbf{P}_6 &= \begin{bmatrix} C23S4R1 + C23D2 + C2D1 \\ S23S4R1 + S23D2 + S2D1 \\ -C4R1 + r1 \end{bmatrix} \\ \mathbf{P}_8 &= \begin{bmatrix} -(C23C4C5 + S23S5)S6 - C23S4C6R2 + C23S4R1 + C23D2 + C2D1 \\ -(S23C4C5 - C23S5)S6 - S23S4C6R2 + S23S4R1 + S23D2 + S2D1 \\ -(S4C5S6 + C4C6)R2 - C4R1 + r1 \end{bmatrix} \quad (3.22) \\ \mathbf{P}_t &= \begin{bmatrix} tx \\ ty \\ tz \end{bmatrix} \end{aligned}$$

Los vectores de (3.22) se usan en la ecuación de restricción por incisión (3.15) para hallar las variables θ_5 y θ_6 en función de las variables r_1 , θ_2 , θ_3 , θ_4 . Se observó que surgen tres ecuaciones de las cuales una es linealmente dependiente de las otras dos, y que una de las ecuaciones tiene la forma $A_1S6+B_1C6=0$, tal que:

$$\theta_6 = \arctan\left(\frac{-B_1}{A_1}\right) \quad (3.23)$$

Luego se reemplazó θ_6 en la siguiente ecuación linealmente independiente y ésta tomó la forma $A_2S5+B_2C5=0$, por tanto:

$$\theta_5 = \arctan\left(\frac{-B_2}{A_2}\right) \quad (3.24)$$

Ahora, para hallar las soluciones de r_1 , θ_2 , θ_3 , se reemplazan las ecuaciones (3.19), (3.20) y (3.21) en las ecuaciones (3.2), (3.3) y (3.4), dando como resultado:

$$\begin{aligned} x &= C23(S4R1 + S4C6R2 + D2 + C4S6C5R2) + a_x R3 + S23S5S6R2 + C2D1 \\ y &= S23(S4R1 + S4C6R2 + D2 + C4S6C5R2) + a_y R3 - C23S5S6R2 + S2D1 \\ z &= r1 + S4C5S6R2 - C4C6R2 + a_z R3 - C4R1 \end{aligned} \quad (3.25)$$

pero aún quedan en función de θ_5 y θ_6 . Se intentó reemplazar las soluciones (3.23) y (3.24), pero se volvían demasiado complejas las ecuaciones (3.25), por lo tanto se aplicó un método numérico para resolverlas y después comprobar su solución empleando la ecuación de restricción (3.16) y conociendo que r_1 tiene que ser positiva para que el robot quede por encima de la camilla donde está el paciente. El algoritmo se muestra en la Figura 3.10.

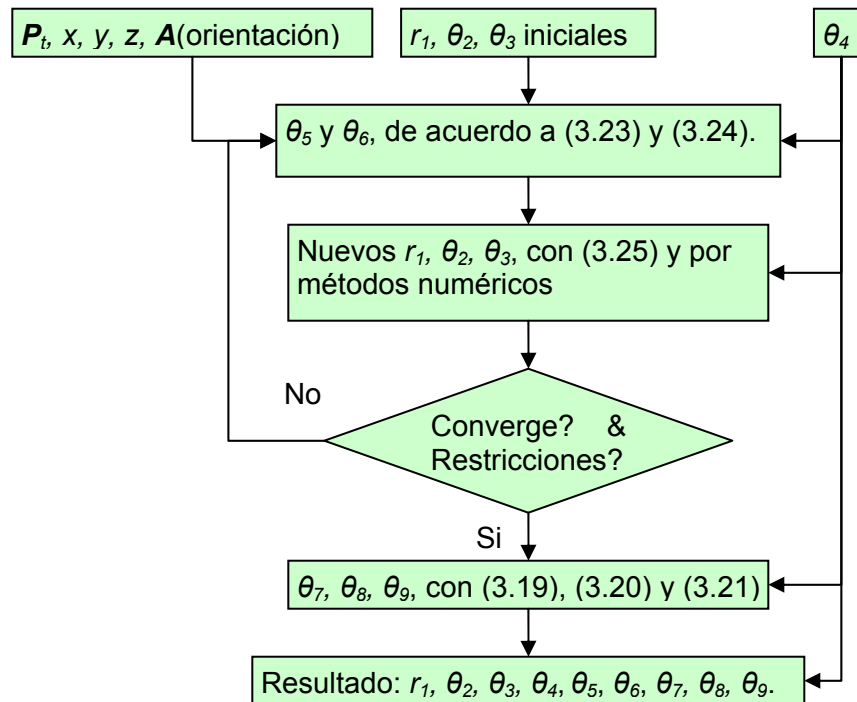


Figura 3.10. Algoritmo para hallar el MGI por métodos numéricos.

♦ Por Resolución de Ecuaciones.

Se modifica el modo de calcular el punto P_8 de tal forma que estuviese involucrada la posición de la herramienta del robot y que el punto fuese estático, es decir, sin dependencia de las variables articulares. Esto se logra usando de nuevo el MGD y calculando P_8 usando la expresión derecha de la ecuación (3.26), contrario a lo realizado en la ecuación (3.22):

$$\begin{aligned} MGD = {}^0T_{10} &= {}^0T_8 {}^8T_{10} = Uo \\ {}^0T_8 &= Uo * inv({}^8T_{10}) \\ P_8 &= Uo * {}^{10}P_8 = \begin{bmatrix} x - a_x R3 \\ y - a_y R3 \\ z - a_z R3 \end{bmatrix} \end{aligned} \quad (3.26)$$

Esta ecuación de P_8 se usa en conjunto con los vectores P_6 y P_t de (3.22) para reemplazarlos en la ecuación de restricción por incisión (3.15), dando como resultado dos ecuaciones linealmente independientes que se presentan en (3.27).

$$\begin{aligned} A_3 S23 + B_3 S2 &= E_{31} \\ A_3 C23 + B_3 C2 &= E_{32} \end{aligned}$$

Donde:

$$\begin{aligned} A_3 &= (-S4R1 - D2)(z - a_z R3 - t_z) \\ B_3 &= -D1(z - a_z R3 - t_z) \\ E_{31} &= -t_y(z - a_z R3 - t_z) + (t_z + C4R1 - r1)(y - a_y R3 - t_y) \\ E_{32} &= (t_z + C4R1 - r1)(x - a_x R3 - t_x) - t_x(z - a_z R3 - t_z) \end{aligned} \quad (3.27)$$

Se pueden despejar θ_2 y θ_3 en función de r_1 , pero siendo independientes de θ_5 y θ_6 . Las soluciones se presentan en la ecuación (3.28).

$$\begin{aligned} \theta_2 &= \arctan \left(\frac{E_{31}F + E_{32}\sqrt{E_{31}^2 + E_{32}^2 - F^2}}{E_{32}F - E_{31}\sqrt{E_{31}^2 + E_{32}^2 - F^2}} \right) \\ \text{Con: } F &= \frac{E_{31}^2 + E_{32}^2 + B_3^2 - A_3^2}{2B_3} \\ \theta_3 &= \arctan \left(\frac{E_{31} - B_3 \sin(\theta_2)}{E_{32} - B_3 \cos(\theta_2)} \right) - \theta_2 \end{aligned} \quad (3.28)$$

Ahora, para hallar la solución de r_1 , se usa la restricción de longitud planteada en la ecuación (3.16) y se reemplazan las soluciones de θ_2 y θ_3 de tal forma que la nueva ecuación sólo tenga como incógnita a r_1 . La forma de la ecuación resultante y su solución se presentan en (3.29).

$$\frac{M(K + r1)^2}{N} = (R2)^2 \quad (3.29)$$

$$r1 = \sqrt{\frac{N(R2)^2}{M}} - K$$

Las variables M , K y N son independientes de las demás articulaciones activas del robot, por lo que es la primera en calcularse. Luego se hallan θ_2 y θ_3 usando (3.28), para hallar θ_5 y θ_6 se usan (3.23) y (3.24), y por último las ecuaciones (3.19), (3.20) y (3.21) permiten encontrar los valores de θ_7 , θ_8 y θ_9 .

3.3.3 Prueba de los Modelos Geométricos

Antes de continuar con el proyecto se realiza una prueba con los modelos geométricos para verificar que es posible seguir trayectorias cartesianas pasando a través de un punto de incisión. Para esto se usó el diagrama que se presenta en la Figura 3.11.

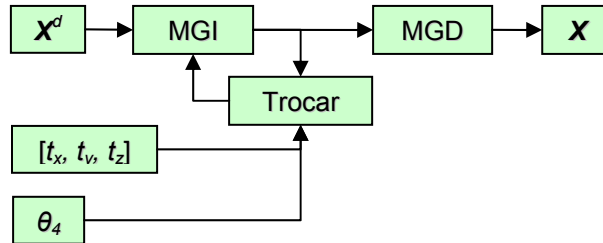


Figura 3.11. Diagrama de prueba de los modelos geométricos.

Donde:

X^d representa las posiciones cartesianas deseadas.

t_x , t_y , t_z define la posición de la incisión en el espacio cartesiano (trocar).

θ_4 es el ángulo fijo de la cuarta articulación.

Trocar contiene las ecuaciones de restricción espacial o las soluciones de las mismas.

MGI contiene una de las dos soluciones planteadas anteriormente.

X representa las posiciones cartesianas de salida.

Al realizar la simulación del comportamiento de LapBot para seguir diferentes trayectorias se obtiene un buen seguimiento manteniendo el punto de incisión. Una de las trayectorias se puede observar en la Figura 3.12, es un círculo realizado sobre el plano xy con radio de 0.15 m, centro en (0.25 m, 0.25 m, 0 m), $\theta_4 = \pi/4$ radianes, $t_x = 0.2$ m, $t_y = 0.2$ m, $t_z = 0.1$ m, y un tiempo de simulación de 3 segundos. Se ha graficado la ubicación del robot en diferentes instantes de tiempo para observar su movimiento.

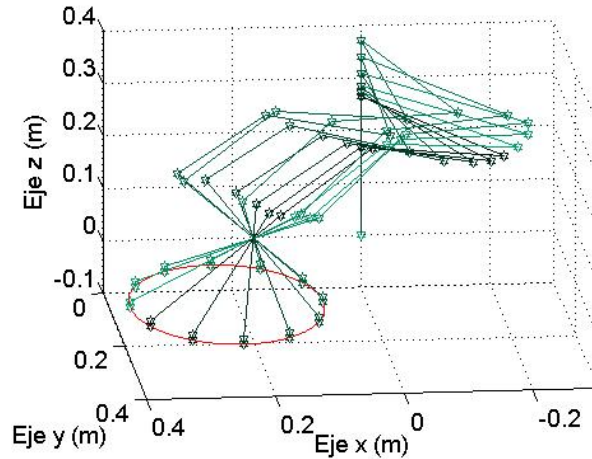


Figura 3.12. Seguimiento de trayectoria circular usando los modelos geométricos.

Así mismo, se probaron otras trayectorias y se obtuvieron resultados igual de satisfactorios, con lo cual se pudo continuar con el trabajo propuesto.

3.3.4 Modelos Dinámicos

Por su parte, los modelos dinámicos, que sirven para simular y controlar el robot, relacionan los pares o fuerzas proporcionadas a los motores con las posiciones, velocidades y aceleraciones articulares, para su cálculo es necesario inicialmente calcular los parámetros inerciales de base que constituyen el conjunto mínimo de valores de inercias y masas a tener en cuenta en el modelo dinámico del robot [21]. Los parámetros de base permiten eliminar aquellos parámetros que no tienen efecto sobre el modelo y agrupar otros con el fin de simplificar las ecuaciones finales del modelo, así como el tiempo computacional de cálculo.

Siguiendo esta metodología, el programa SYMORO (*SYmbolic MOdelling of RObots*) [66] proporciona la tabla de parámetros mínimos de LapBot, que se muestran en las tablas 3.2 y 3.3. Las ecuaciones de reagrupamiento se encuentran en el Anexo A.

j	XX_j	XY_j	XZ_j	YY_j	YZ_j	ZZ_j
1	0	0	0	0	0	0
2	0	0	0	0	0	$ZZ2R$
3	0	0	0	0	0	$ZZ3R$
4	0	0	0	0	0	0
5	$XX5R$	0	0	0	0	$ZZ5R$
6	$XX6R$	0	0	0	0	$ZZ6R$
7	$XX7R$	0	0	0	0	$ZZ7R$
8	$XX8R$	0	0	0	0	$ZZ8R$
9	$XX9R$	0	0	0	0	$ZZ9R$

Tabla 3.2. Parámetros del tensor de inercia de LapBot.

j	MX_j	MY_j	MZ_j	M_j	la_j
1	0	0	0	$M1R$	$la1$
2	0	0	0	0	0
3	$MX3R$	$MY3R$	0	0	$la3$
4	0	0	0	0	0
5	$MX5$	$MY5R$	0	0	0
6	$MX6$	$MY6R$	0	0	0
7	$MX7$	$MY7R$	0	0	$la7$
8	$MX8$	$MY8R$	0	0	$la8$
9	$MX9R$	$MY9R$	0	0	$la9$

Tabla 3.3. Parámetros del primer momento de inercia, masas e inercia de motores.

Donde:

M_j : masa del cuerpo de la articulación j .

${}^j\mathbf{MS}_j = [MX_j \ MY_j \ MZ_j]^T$: vector columna del primer momento de inercia de la articulación j .

$XX_j, XY_j, XZ_j, YY_j, YZ_j, ZZ_j$: elementos que representan el tensor de inercia de la articulación j .

la_j : inercia del motor de la articulación j .

La letra R al final de algunos elementos significa que dicho parámetro agrupa en él a otros que en las tablas aparecen como nulos. Esta reducción de parámetros permite trabajar con sólo 30 parámetros inerciales de base y no con el total de 99 parámetros, lo cual simplifica enormemente los cálculos del modelo en simulación.

Usando las dos tablas anteriores y de nuevo con la ayuda del software SYMORO se obtiene el Modelo Dinámico Inverso (MDI), que permite calcular los pares mecánicos a aplicar en los motores en términos de la posición, velocidad y aceleración de cada articulación (ecuación (3.30)), de acuerdo al método de Newton-Euler [21]:

$$\mathbf{\Gamma} = \mathbf{A}(\mathbf{q})\ddot{\mathbf{q}} + \mathbf{C}(\mathbf{q}, \dot{\mathbf{q}})\dot{\mathbf{q}} + \mathbf{Q}(\mathbf{q}) \quad (3.30)$$

Donde:

$\mathbf{\Gamma}$: pares aplicados a los motores.

$\mathbf{A}(\mathbf{q})$: matriz de inercias, simétrica y definida positiva.

$\mathbf{C}(\mathbf{q}, \dot{\mathbf{q}})$: matriz que incluye las fuerzas de Coriolis y centrífugas.

$\mathbf{Q}(\mathbf{q})$: vector de fuerzas gravitacionales.

Las posiciones articulares son \mathbf{q} , las velocidades $\dot{\mathbf{q}}$ y las aceleraciones $\ddot{\mathbf{q}}$.

Despejando las aceleraciones articulares de la ecuación (3.30), se calcula el Modelo Dinámico Directo (MDD) que permite simular el robot en un software como Matlab/Simulink®. La expresión de este modelo es mostrada en la ecuación (3.31).

$$\ddot{\mathbf{q}} = \text{inv}(\mathbf{A}(\mathbf{q})) * (\mathbf{\Gamma} - \mathbf{C}(\mathbf{q}, \dot{\mathbf{q}})\dot{\mathbf{q}} - \mathbf{Q}(\mathbf{q})) \quad (3.31)$$

Los programas que simulan estos modelos y contienen las ecuaciones optimizadas, se pueden examinar en los archivos de computador que se entregan con este documento.

3.3.5 Diseño General de los Cuerpos de LapBot

Para obtener los valores aproximados de los parámetros de base y de las dimensiones geométricas de LapBot se tienen en cuenta los siguientes aspectos:

- ♦ La pared abdominal sube entre 0.15 y 0.2 m debido al gas de CO_2 que se inyecta al paciente [48].
- ♦ Una camilla estándar grande tiene un ancho de 0.6 m y una altura mínima respecto al piso de 0.5 m [49].
- ♦ La parte del robot que se introduce en la cavidad abdominal debe alcanzar una distancia máxima entre 0.24 y 0.30 m desde el punto de incisión [50], [60].
- ♦ Emplear motores especiales para robots usados en cirugía, dado que deben ser livianos y pequeños pero con suficiente par mecánico, confiables, de baja fricción interna, con compatibilidad electromagnética y la posibilidad de control de posición [51], [60].
- ♦ Los materiales de las herramientas y los cuerpos del robot que están dentro del paciente deben ser de acero quirúrgico inoxidable 410 [52], para poderlos esterilizar y reutilizar. Pero para los cuerpos externos se puede usar aluminio [50], [60], que es más liviano y económico.
- ♦ El diámetro del cuerpo que pasa a través del punto de incisión es de entre 0.005 y 0.01 m [46], [60].

Por lo tanto, se escogen las siguientes dimensiones: $R3=0.01$ m. (tamaño de la punta de la pinza), $R2=0.3$ m de longitud y 0.008 m de diámetro (cuerpo que pasa por el trocar). Como el cuerpo de longitud $R1$ es aquel que permite posicionar al robot por encima del abdomen del paciente, para que el resto de los cuerpos no colisionen, la mínima dimensión de $R1$ será de 0.2 m.

Ahora, teniendo en cuenta que se colocarán dos brazos de LapBot a lado y lado de la camilla y a no más de 0.05 m de su orilla, y suponiendo que la incisión al paciente no se realizará a más de 0.3 m de la base del robot, la posición que se presenta en la Figura 3.13 es la que exige la mayor elongación de LapBot. Como se observa, el cuerpo $R2$ queda casi totalmente retraído lo cual indica que la suma de $D1$, $D2$ y $R1$ es aproximadamente 0.65 m. Como $R1 = 0.2$ m, entonces $D1+D2 = 0.45$ m. Luego,

para darle mayor versatilidad y espacio de trabajo sobre la camilla se escoge $D1 = 0.25$ m y $D2 = 0.20$ m.

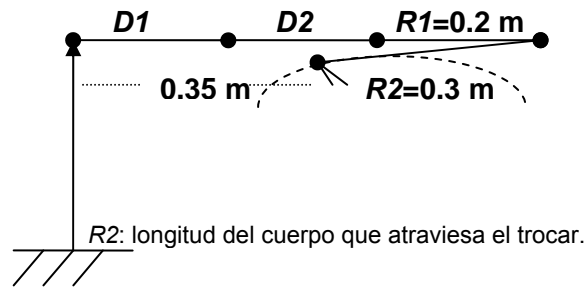


Figura 3.13. Máxima elongación para LapBot.

En cuanto a los parámetros inerciales mostrados en las tablas 3.2 y 3.3, se realizó el cálculo construyendo los cuerpos del robot en el software CAD SolidEdge®, usando aluminio con 2712 kg/m^3 de densidad para los cuerpos que están fuera del paciente y acero inoxidable 410 de 7750 kg/m^3 de densidad para los cuerpos que están dentro del paciente. La base tiene una altura de 0.5 m y un espacio interno donde se ubicarán algunos motores y sistemas electrónicos; el cuerpo de movimiento traslacional para r_1 , de 0.8 m de longitud y 0.08 m de diámetro, y el cuerpo para la rotación θ_2 se pueden observar en la Figura 3.14.

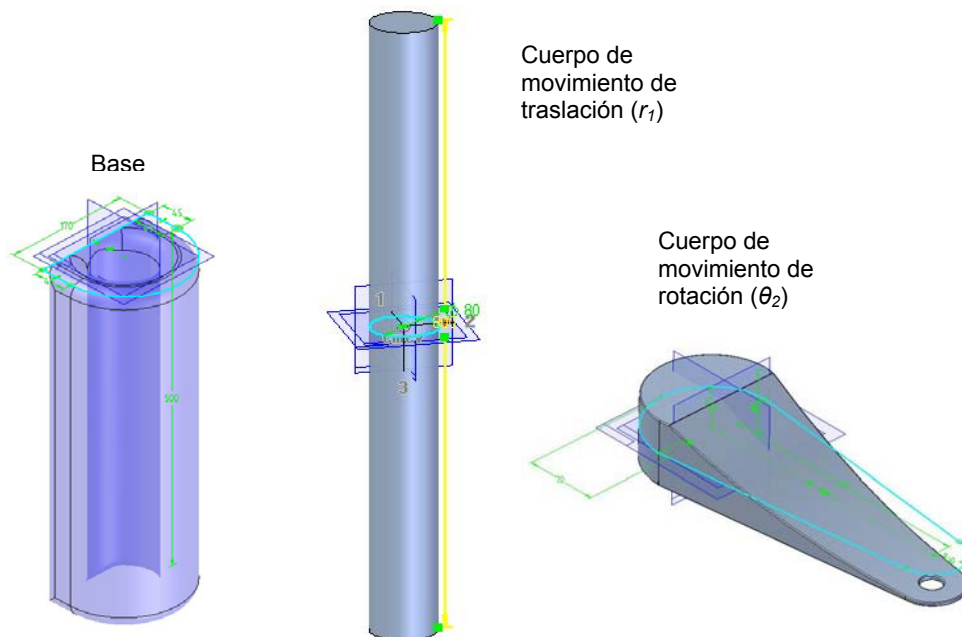


Figura 3.14. Diseños en SolidEdge® de la base y los dos primeros cuerpos de LapBot.

Los siguientes tres cuerpos correspondientes a los movimientos de las articulaciones θ_3 , θ_4 y θ_5 se presentan en la Figura 3.15.

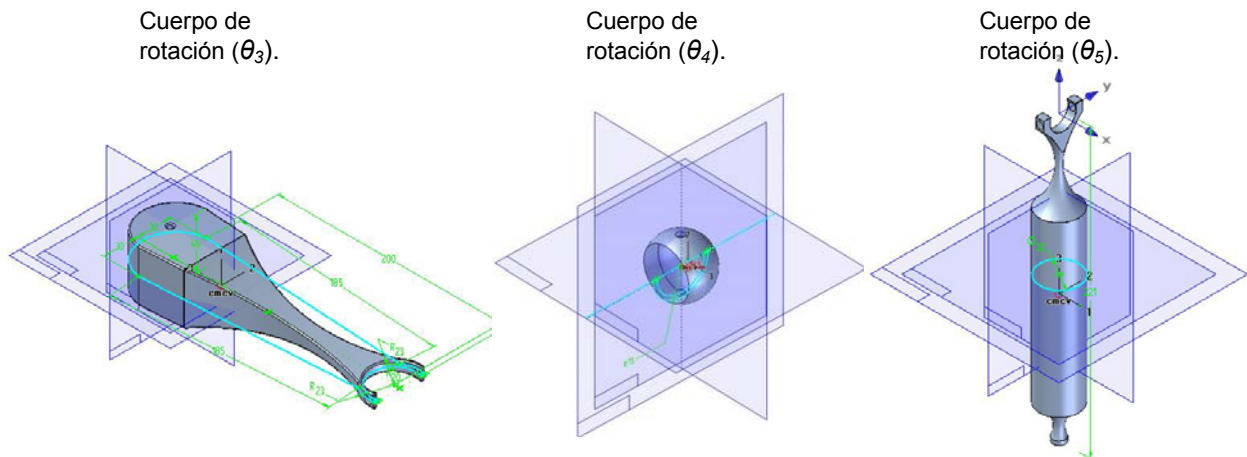


Figura 3.15. Diseños en SolidEdge® de los cuerpos para θ_3 , θ_4 y θ_5 de LapBot.

Para θ_6 , θ_7 y θ_8 también se construyeron cuerpos en el software de tal forma que se pudiesen acoplar a las demás partes, especialmente el cuerpo para θ_7 , que atraviesa la incisión, tiene una longitud de 0.3 m, diámetro de 0.008 m y es de acero inoxidable. Los diseños se muestran en la Figura 3.16.

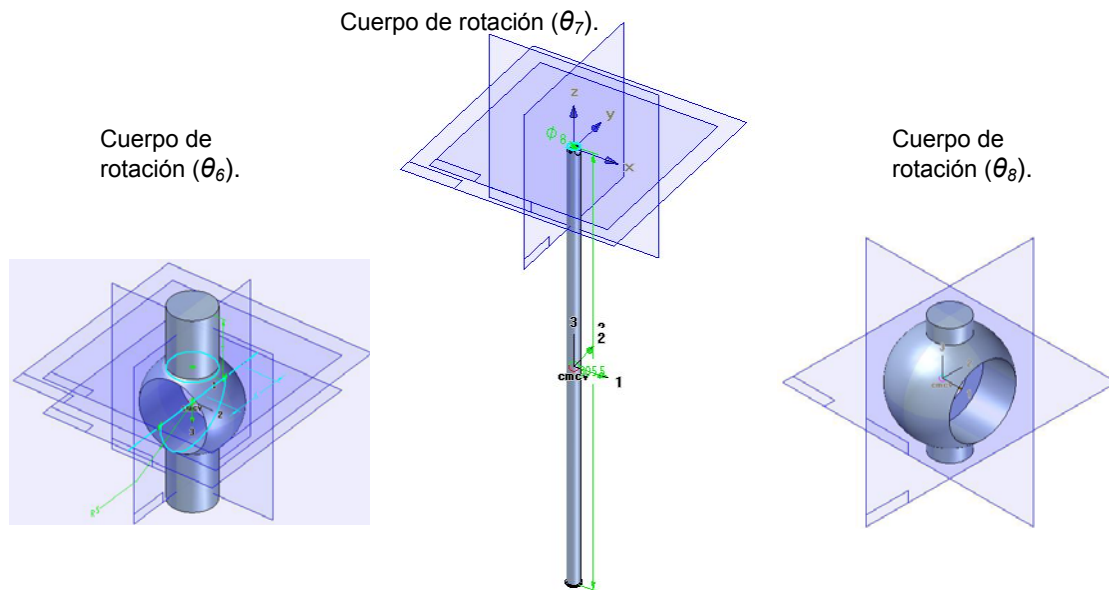


Figura 3.16. Diseños en SolidEdge® de los cuerpos para θ_6 , θ_7 y θ_8 de LapBot.

Por último se diseñaron: el cuerpo para la rotación θ_9 , una pinza y un gancho cauterizador de 0.001 m de longitud, los cuales pueden intercambiarse dependiendo

del procedimiento quirúrgico que se esté realizando. El cuerpo respectivo y las herramientas se muestran en la Figura 3.17.

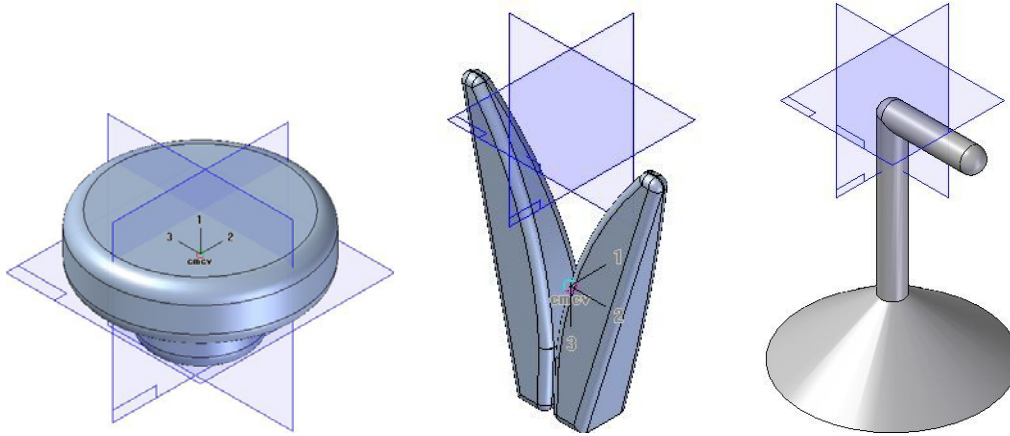


Figura 3.17. Diseños en SolidEdge® del cuerpo para θ_9 , la pinza y el cauterizador.

Además, se utilizan motores Maxon EC Powermax 30 especialmente diseñados para robots quirúrgicos, los cuales pesan aproximadamente 0.165 Kg [51]. En el Anexo A se pueden observar las ecuaciones de los parámetros inerciales de base, los valores dinámicos físicos obtenidos de SolidEdge® y la forma en la cual se reagruparon. En la Tabla 3.4 se muestran los valores finales calculados de los parámetros para LapBot, con $\theta_4 = \pi/4$ radianes.

Parámetro	Valor	Parámetro	Valor
XX5R	5.45e-7	MX7	0
XX6R	2.8e-3	MX8	0
XX7R	2.1e-9	MX9R	0
XX8R	1.31e-7	MY3R	6e-9
XX9R	-1.6e-9	MY5R	0
ZZ2R	5.06e-2	MY6R	-1.34e-2
ZZ3R	1.8e-2	MY7R	0
ZZ5R	1.34e-5	MY8R	2.215e-6
ZZ6R	2.8e-3	MY9R	0
ZZ7R	8.92e-7	M1R	5.98
ZZ8R	1.327e-7	la1	2e-6
ZZ9R	2.1e-9	la3	2e-6
MX3R	6.87e-3	la7	2e-6
MX5	0	la8	2e-6
MX6	0	la9	2e-6

Tabla 3.4. Valores de los parámetros inerciales de base de LapBot.

Las unidades de los elementos del tensor de inercia son $\text{kg}\cdot\text{m}^2$, para el primer momento de inercia son $\text{kg}\cdot\text{m}$, para la inercia del accionador son $\text{kg}\cdot\text{m}^2$ y para la masa $M1R$ es kg.

4. Control de LapBot para el Seguimiento de una Colecistectomía

Después de modelar matemáticamente el robot y tener los valores de los parámetros del mismo, se procede a realizar el seguimiento de trayectorias correspondientes a procedimientos quirúrgicos que hacen parte de una colecistectomía laparoscópica, usando un controlador por par calculado que incluye la dinámica del robot.

4.1 Colecistectomía Laparoscópica

Es la intervención quirúrgica que se realiza para extraer una vesícula biliar enferma, que esté inflamada u obstruida por cálculos biliares [29], por medio de pequeñas incisiones en el abdomen del paciente.

La bilis es una secreción elaborada en el hígado y evacuada por el conducto hepático hacia el duodeno, pasando por un elemento biliar llamado vesícula que está conectada al conducto hepático por el conducto cístico (Ver la parte izquierda de la Figura 4.1). La excreción de bilis es discontinua y depende del tránsito digestivo, así que se acumula en la vesícula cuando no hay alimento pasando por el duodeno [67].

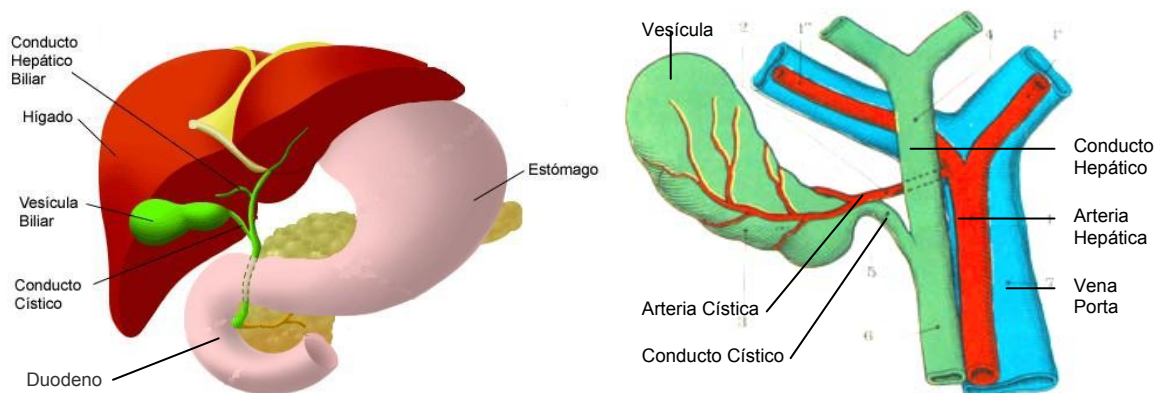


Figura 4.1. Órganos biliares[‡] (izquierda) y vesícula biliar (derecha) [67].

[‡] www.facmed.unam.mx/deptos/anatomia/computo/higado.htm

En la parte derecha de la Figura 4.1 se observa la vesícula biliar de aspecto periforme, que tiene una longitud de 0.08 a 0.1 m y 0.03 a 0.04 m de ancho en una persona adulta, está conectada a la arteria hepática por medio de la arteria cística que atraviesa el triángulo de Calot, formado entre el conducto cístico, el hepático y el hígado [67].

La presencia de partículas sólidas en la vesícula, llamadas cálculos, obstaculiza el paso de la bilis generando dolor e inflamación en el paciente. La colecistectomía es un método frecuente para tratar esta enfermedad ya que la extirpación de la vesícula biliar perturba poco la excreción biliar [67].

En el quirófano, el cirujano se ubica a la izquierda del paciente, a su lado el auxiliar encargado del manejo de la cámara, y al frente de ellos o a la derecha del paciente se ubican el asistente principal y la enfermera (Figura 4.2).

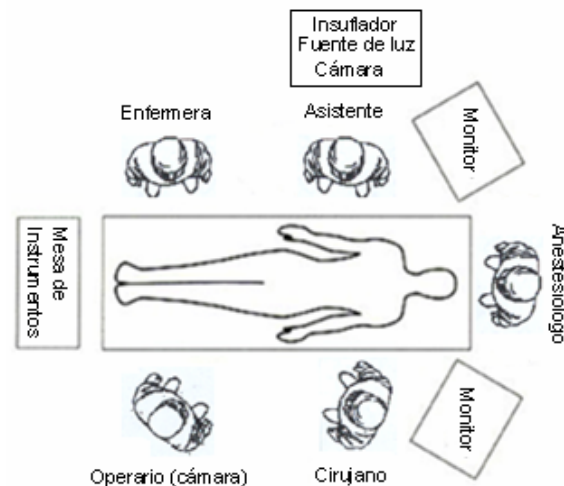


Figura 4.2. Distribución de elementos en el quirófano [68].

Según [69], se hacen tres pequeñas incisiones de acuerdo a la distribución que se presenta en la Figura 4.3.

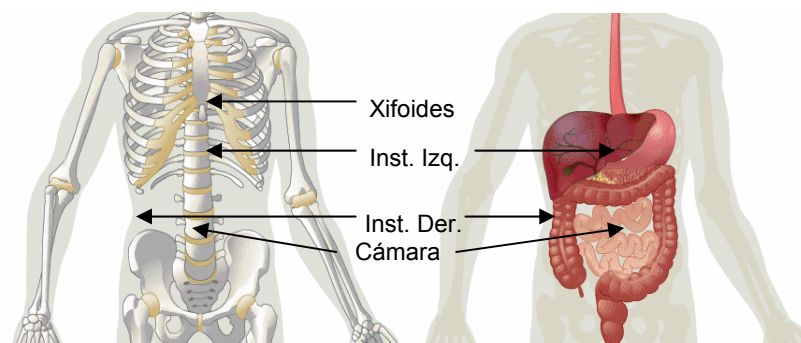


Figura 4.3. Ubicación de instrumentos quirúrgicos.

1. La primera de 0.01 m (10 mm) se hace cercana al ombligo o sobre éste, para introducir el sistema de visión.
2. La segunda de 0.01 m (10 mm) se realiza 0.05 m por debajo del xifoides en la parte superior del abdomen, por la cual se introducen los instrumentos para cortar, disecar y coagular durante la intervención.
3. La tercera de entre 0.005 m (5 mm) y 0.01 m (10 mm) se realiza 0.05 m debajo de las costillas del lado derecho del paciente, ésta se utiliza para introducir pinzas que mueven el hígado, la vesícula y sus conductos.

Es importante mencionar que se realizan colecistectomías con otra incisión del lado derecho del abdomen para mejorar la maniobrabilidad de los órganos [68], pero en este proyecto se simulan y controlan con LapBot los movimientos quirúrgicos de los instrumentos correspondientes a las incisiones 2 y 3.

4.1.1 Etapas Quirúrgicas

Generalmente la colecistectomía se divide en cinco pasos o etapas que se describen a continuación [68]:

1. Exposición del conducto y arteria císticos.

Para esta etapa se utilizan dos pinzas de retracción (instrumentos de la izquierda de la Figura 4.4) introducidas por la parte derecha del abdomen. Una pinza se usa para agarrar y mover la vesícula hacia arriba, en dirección del hombro derecho del paciente y la otra hala hacia la región llamada *Bolsa de Hartman*.

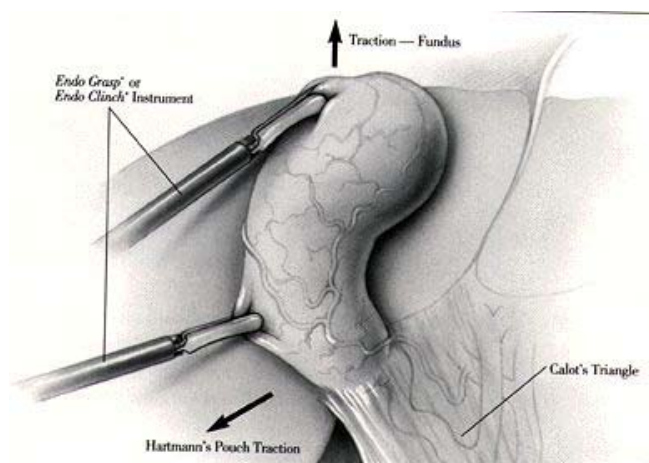


Figura 4.4. Exposición del conducto y arteria císticos [68].

2. Disección del triángulo de Calot.

Se introduce una pinza de retracción adicional por la incisión realizada en la parte superior del abdomen, con la cual se expone el conducto y la arteria císticos por fuera del tejido como lo muestra la Figura 4.5, para esta exposición es necesario realizar movimientos de apertura y cierre sobre los conductos tratando de dilatar la zona y mejorar la visualización en el área de trabajo.

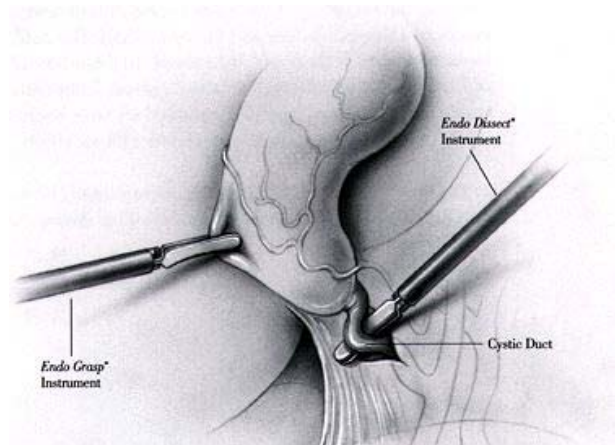


Figura 4.5. Disección del conducto cístico [68].

3. Sección del conducto y la arteria císticos.

Se introduce por la incisión superior una pistola de endoclips que grapa con tres clips el conducto cístico, colocando dos a la derecha del conducto y uno a la izquierda (cercano a la vesícula). Luego se reemplaza la pistola por una tijera quirúrgica que corta en medio de los clips. Y se repite el procedimiento para la sección de la arteria (Figura 4.6).

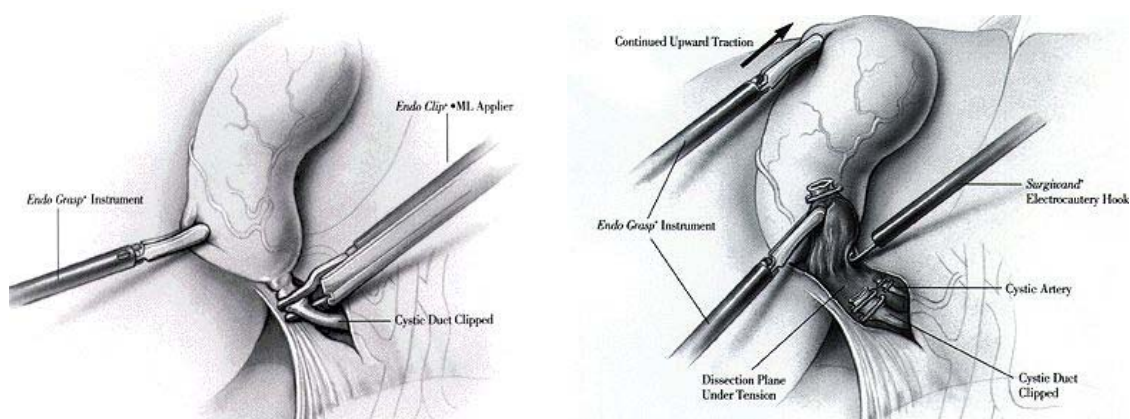


Figura 4.6. Sección del conducto y arteria císticos [68].

4. Disección de la vesícula del lecho hepático.

Como la vesícula se encuentra ligada al lecho hepático por una serie de adherencias, se procede a disecar la vesícula del lecho hepático (despegar de la membrana del hígado) utilizando un instrumento de electrocauterio (instrumento a la derecha de la Figura 4.7) que se introduce por la incisión superior y que tiene la capacidad de disecar y cauterizar la sangre producida.

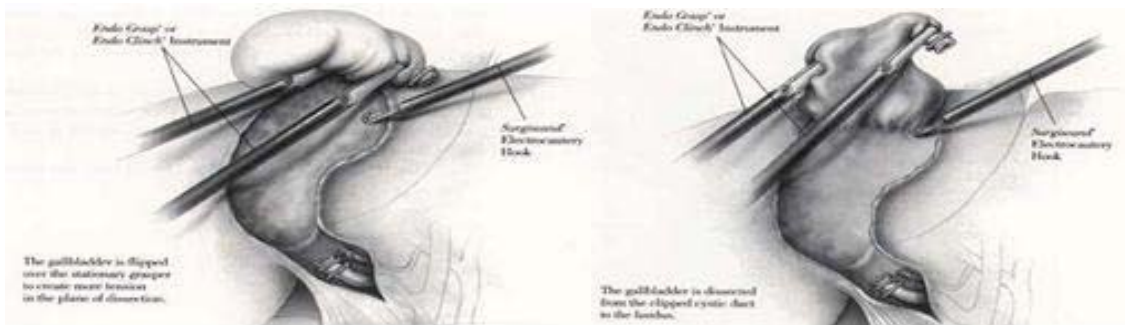


Figura 4.7. Disección de la vesícula del lecho hepático [68].

5. Extracción de la vesícula biliar.

Finalmente se extrae la vesícula, generalmente por el ombligo, pero esta etapa no se seleccionó para generar trayectorias dado que el procedimiento varía dependiendo de factores como la cantidad de cálculos o tamaño de la inflamación, la experiencia del cirujano y las técnicas del grupo medico.

4.1.2 Trayectorias para Simulación

Mora y Garcés [70] realizaron la construcción de las trayectorias cartesianas para las etapas 2, 3 y 4, utilizando videos proporcionados por la Fundación Clínica Valle del Lili de la ciudad de Cali (Colombia), por medio del Doctor José Pablo Vélez Londoño, jefe del Servicio de Cirugía Mínimamente Invasiva.

Enlazaron los videos con el software Blender para capturar la posición de los instrumentos en cada instante de tiempo, tomando como referencia central la vesícula. En la Figura 4.8 se puede observar el ambiente tridimensional desarrollado, el video y el código para exportar los datos de Blender a un archivo de texto.

El entorno de trabajo en Blender tiene la vesícula en el centro (0, 0, 0), con 0.08 m en el eje X, 0.04 m en el eje Y, 0.05 m en el eje Z; y los conductos biliares con las siguientes dimensiones: 0.04 m en el eje X, 0.005 m en el eje Y, 0.005 m en el eje Z. Con esto, se realiza la obtención de las trayectorias que se describen a continuación.

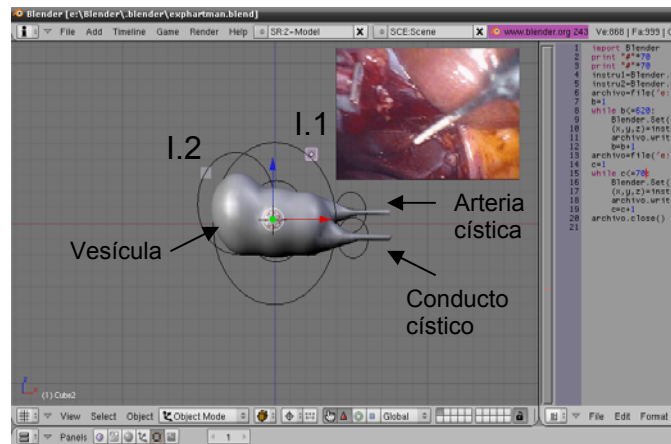


Figura 4.8. Entorno de trabajo en Blender [70].

1. Disección del triángulo de Calot.

Se requieren dos instrumentos quirúrgicos, uno encargado del agarre de la vesícula que se denota como instrumento número 2 (I2) y otro destinado a la disección del triángulo etiquetado como instrumento número 1 (I1).

Para I1 la trayectoria consta de 150 puntos, inicia en el punto con coordenadas (4, -0.6, -1.5), lugar más próximo al conducto cístico, luego se dirige en busca de la parte superior del conducto cístico en las coordenadas (6, 0.9, -0.63). A continuación el instrumento se traslada hacia el punto superior del triángulo de Calot (4.31, 0.44, -0.09) y finalmente el recorrido llega a las coordenadas del último punto (6, 1.4, 1). La Figura 4.9 muestra la trayectoria.

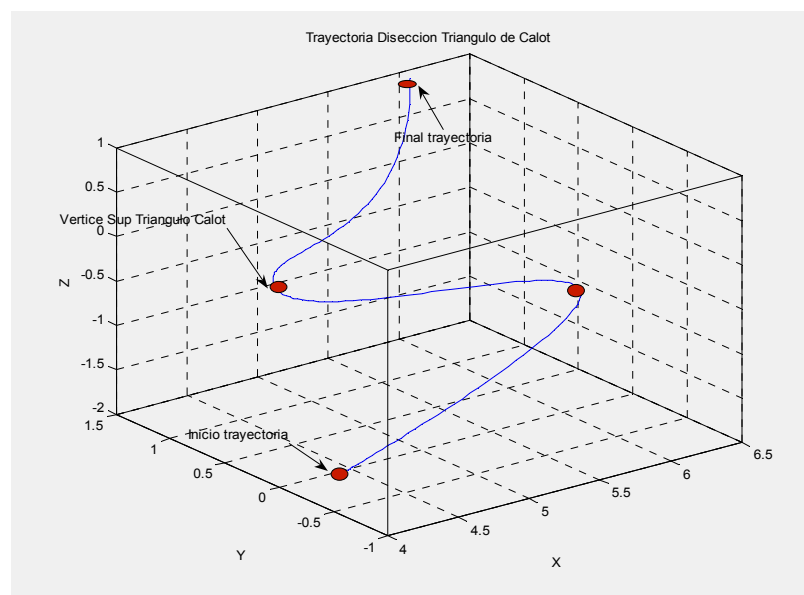


Figura 4.9. Trayectoria del instrumento 1 para la disección del triángulo de Calot [70].

Para I2, el movimiento está definido por la tracción de la vesícula, consta de 340 puntos e inicia en el punto con coordenadas (2.07, -1.85, -0.15), lugar en el cual se agarra una fracción de vesícula y se tracciona (empuja) hasta el punto (1.19, -1.85, -2.01), después se ubica la bolsa de Hartman en el punto (2.19, -1, -2.01). A partir de esta zona se realiza la segunda y ultima tracción hasta el punto final con coordenadas (-0.65, -2.40, -2.5). Es importante resaltar que durante todo el procedimiento quirúrgico este movimiento puede repetirse varias veces. El movimiento se muestra en la Figura 4.10.

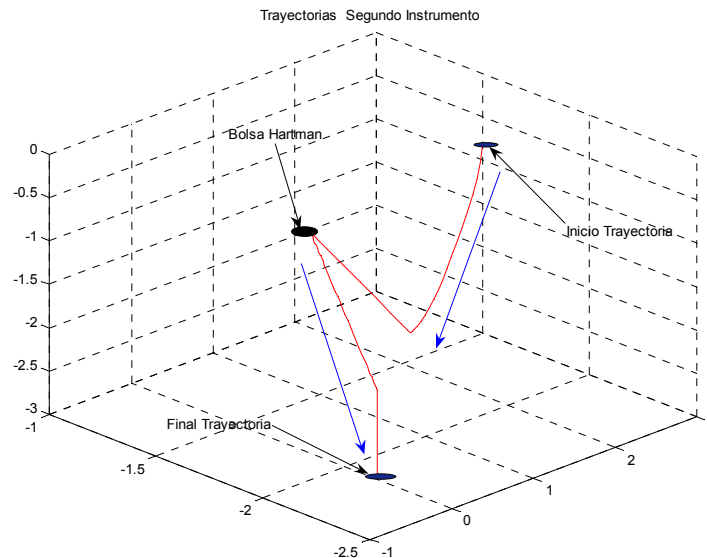


Figura 4.10. Trayectoria del instrumento 2 para la disección del triángulo de Calot [70].

2. Sección del conducto cístico.

En esta etapa, I2 repite el movimiento de la Figura 4.10, pero I1 tiene la trayectoria de 70 puntos que se presenta en la Figura 4.11 y que inicia en el punto de referencia (6, 2, 2).

El primer clip se ubica sobre el conducto cístico en la posición (6.7, -0.25, -0.85), se extrae el instrumento para recargarse con otro clip y vuelve al punto de referencia, luego instala el segundo clip en (5.15, -0.245, -0.85), el instrumento regresa al punto de referencia, se recarga y pasa a colocar el tercer clip en (7, -0.25, -0.85). Luego, para cortar el conducto, I1 se extrae del paciente y se le coloca una tijera que se lleva al punto de corte (6.1, -0.25, -0.85).

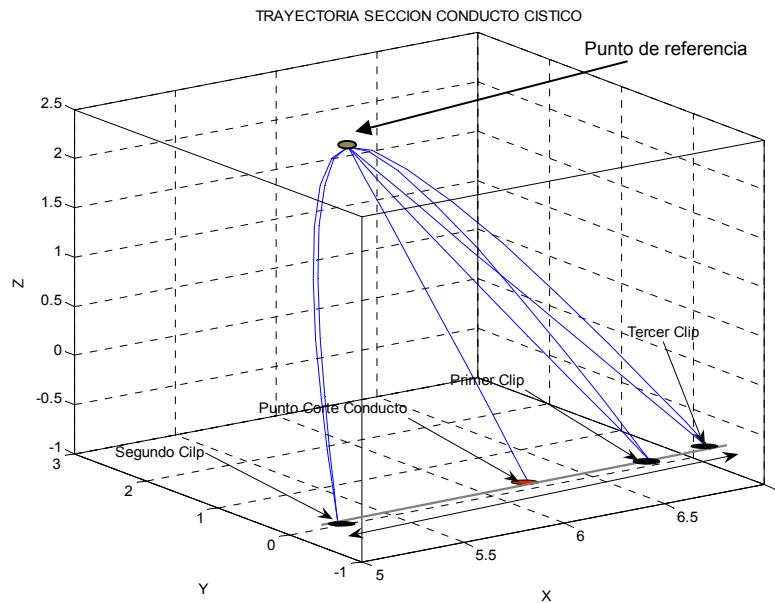


Figura 4.11. Trayectoria del instrumento 1 para el corte del conducto cístico [70].

3. Sección de la arteria cística.

El procedimiento para el corte de la arteria cística es similar al corte del conducto. El primer clip es colocado en la posición (6.7, 1.2, 0.7), el segundo en (5.5, 1.2, 0.7), el tercero en (7, 1.2, 0.7) y se realiza el corte en el punto (6, 1.2, 0.7). La Figura 4.12 muestra en rojo el punto de corte y en azul los clips.

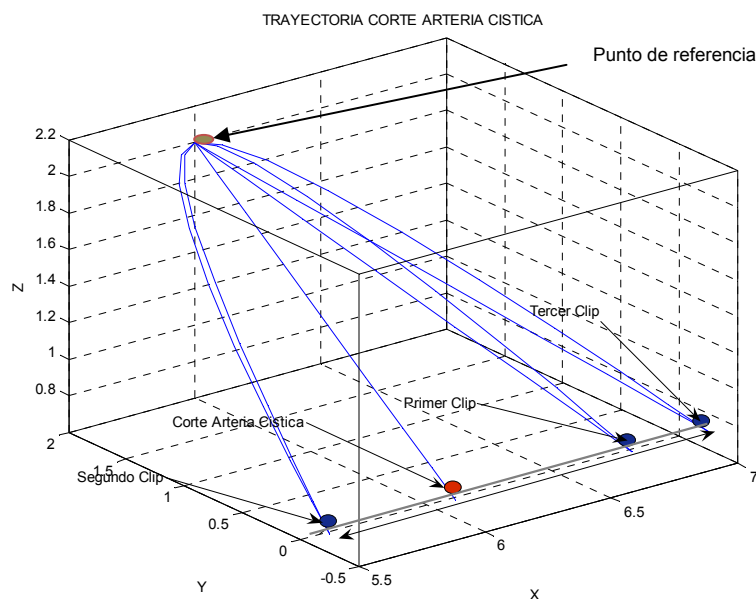


Figura 4.12. Trayectoria del instrumento 1 para el corte de la arteria cística [70].

4. Disección de la vesícula del lecho hepático.

Se ingresa I1 con un electrocauterio al punto (3.76, 1.53, 1.362) y simultáneamente I2 realiza la tracción de la Figura 4.10. Luego se realizan 480 puntos de movimiento en zigzag para ubicarse en la posición (3.4, -0.6, -0.75), de tal forma que los $16 \times 10^{-4} \text{ m}^2$ de vesícula adheridos al hígado quedan separados, concluyendo la trayectoria (Figura 4.13).

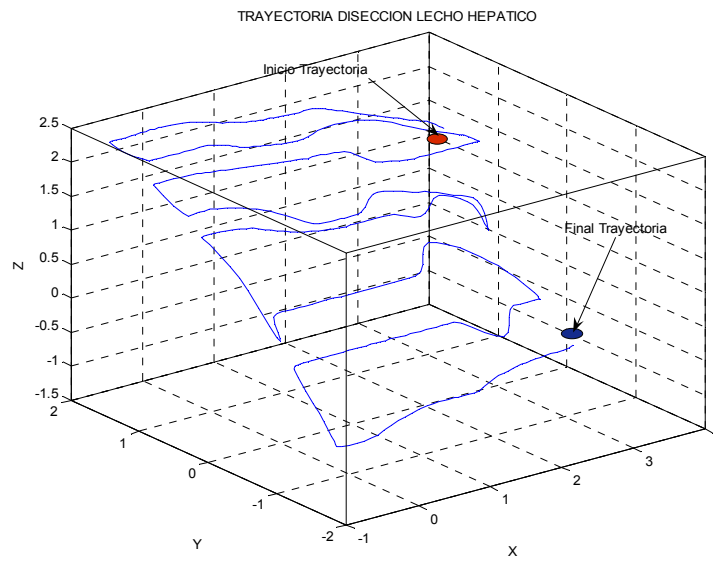


Figura 4.13. Trayectoria del instrumento 1 para la disección de la vesícula del lecho hepático [70].

4.2 Control por Par Calculado (CTC[§])

Después de definir las trayectorias que se deben seguir con el robot en el espacio cartesiano para la realización de una colecistectomía, se procede a diseñar un controlador que permita obtener errores pequeños teniendo en cuenta los modelos dinámicos que representan al robot real en la simulación.

El control por par calculado (CTC) utiliza el modelo dinámico inverso para linealizar y desacoplar la compleja dinámica de un robot, asegurando teóricamente un comportamiento uniforme en cualquier configuración de manera que es posible aplicar una técnica de control lineal [21], [71], [72].

La ecuación (3.30) puede ser rescrita como en la ecuación (4.1), donde la matriz H incluirá los términos de Coriolis, gravitacionales y las fuerzas centrífugas.

[§] Por sus siglas en inglés *Computed Torque Control*

$$\Gamma = A(q)\ddot{q} + H(q, \dot{q}) \quad (4.1)$$

Asumiendo que las matrices A y H se pueden estimar por medio de \hat{A} y \hat{H} , midiendo las posiciones articulares q y velocidades articulares \dot{q} , se escoge una señal de entrada al robot Γ como aparece en la ecuación (4.2), que será el vector de pares a aplicar en los motores del robot.

$$\Gamma = \hat{A}(q)w(t) + \hat{H}(q, \dot{q}) \quad (4.2)$$

Donde $w(t)$ representa la nueva señal de control. Si el sistema es conocido con cierta precisión, se puede suponer que $\hat{A} = A$ y $\hat{H} = H$, de tal forma que al conectarlo con el robot se logra hacer un desacoplamiento de las dependencias de las articulaciones y una compensación de las altas no linealidades del robot. Permitiéndose así aplicar un controlador lineal para obtener la señal de control $w(t)$, usando las posiciones articulares deseadas q^d y las velocidades articulares deseadas \dot{q}^d , como aparece en la ecuación (4.3) y en la Figura 4.14.

$$w(t) = K_p(q^d - q) + K_v(\dot{q}^d - \dot{q}) \quad (4.3)$$

Donde K_p y K_v son las ganancias proporcional y derivativa del controlador lineal respectivamente. En la Figura 4.14, el bloque del modelo geométrico inverso (MGI) se encarga de convertir la consigna cartesiana deseada X^d en consigna articular deseada q^d .

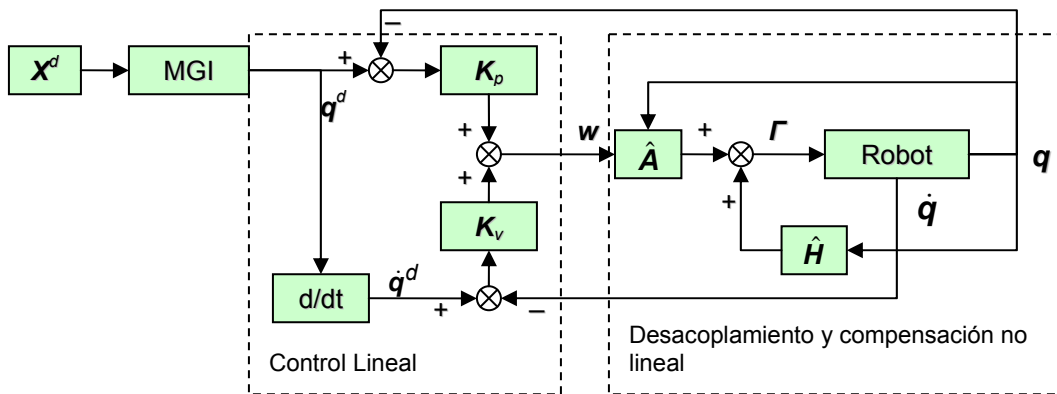


Figura 4.14. Esquema general del control CTC.

Sin embargo, para aplicar la estrategia de control CTC a LapBot es necesario adicionar las restricciones debidas al punto de incisión como se muestra en la Figura

4.15. Obsérvese que se utiliza también el modelo geométrico directo (MGD) para obtener el movimiento del robot en el espacio cartesiano \mathbf{X} y verificar que está siguiéndose la trayectoria deseada \mathbf{X}^d .

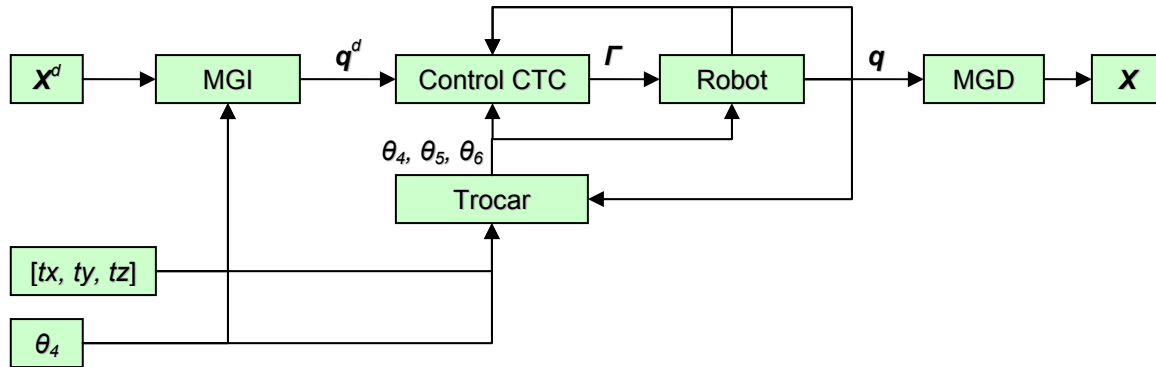


Figura 4.15. Control CTC con modificaciones para LapBot.

La posición cartesiana deseada (\mathbf{X}^d) del efector final pasa por el bloque de ecuaciones del modelo geométrico inverso (MGI), con el fin de obtener las variables articulares deseadas (\mathbf{q}^d), de tal forma que el controlador CTC garantice un mínimo error entre \mathbf{q} y \mathbf{q}^d , así como un mínimo error entre \mathbf{X} y \mathbf{X}^d . El bloque Trocar contiene las ecuaciones para obtener los valores de las articulaciones pasivas (θ_5, θ_6), usando las variables articulares reales, la posición cartesiana del trocar y la articulación fija θ_4 .

4.3 Seguimiento de Trayectorias

Se realiza la sintonización del controlador PD por el método manual planteado en [73], obteniéndose los valores de ganancias presentados en la Tabla 4.1, los cuales se refieren sólo a las articulaciones activas (motorizadas), dado que las pasivas (no motorizadas) son independientes del controlador.

j	1	2	3	7	8	9
K_p	80000	200000	150000	50000	200000	140000
K_v	200	350	270	150	300	300

Tabla 4.1. Ganancias para el controlador PD.

4.3.1 Pruebas Iniciales

Se prueba el correcto funcionamiento de los modelos y el controlador usando trayectorias simples, todas con el punto de incisión ubicado en $t_x=0.2$ m, $t_y=0.2$ m,

$t_z=0.15$ m, la articulación fija en $\theta_4=\pi/4$ radianes, y un período de muestreo de 0.001 s. Las trayectorias simples de prueba son:

1. Círculo sobre el plano xy con radio de 0.05 m, centro en (0.3, 0.3, 0.7) m y un tiempo de simulación de 3 s, se presenta al lado izquierdo de la Figura 4.16.

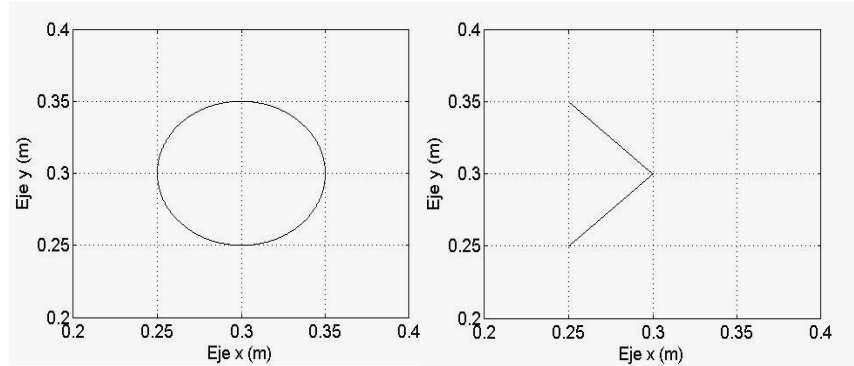


Figura 4.16. Trayectoria circular (izquierda) y lineal (derecha).

2. Trayectoria lineal con cambio brusco de dirección sobre el plano xy, que comienza en (0.25, 0.35, 0.7) m y termina en (0.25, 0.25, 0.7) m, con tiempo de simulación de 1 s. La trayectoria está en el lado derecho de la Figura 4.16.
3. Trayectoria tridimensional que pretende simular una sutura simple realizada en 3 s (Figura 4.17).

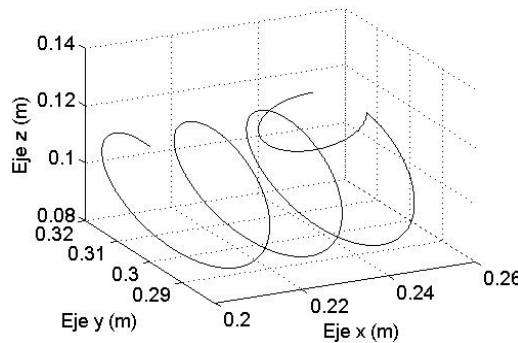


Figura 4.17. Trayectoria tridimensional.

El seguimiento de estas trayectorias se presenta en las figuras 4.18 y 4.19, donde se puede observar que LapBot respeta el punto de incisión. Se ha graficado la ubicación del robot en diferentes instantes de tiempo para representar el movimiento.

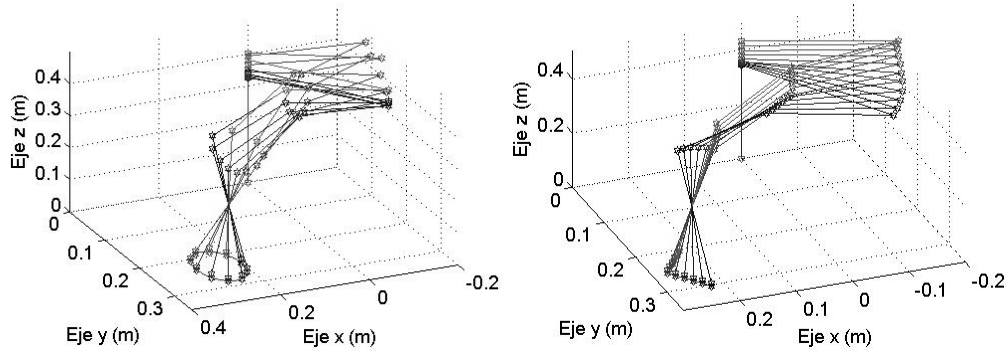


Figura 4.18. Seguimiento del círculo (izquierda) y del cambio de dirección (derecha).

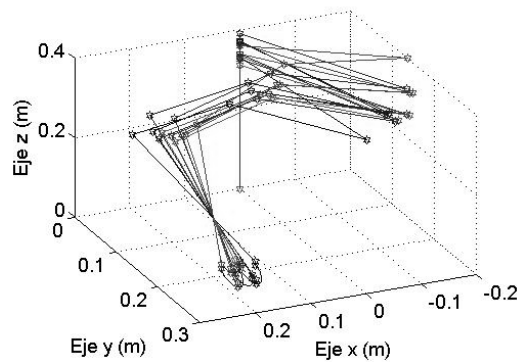


Figura 4.19. Seguimiento de la trayectoria tridimensional.

Los errores de posición obtenidos en el seguimiento de las trayectorias cartesianas se pueden visualizar en las figuras 4.20 y 4.21, donde se observa que para el círculo el error máximo es de 0.00027 m, para el cambio de dirección es de 0.00046 m, para el movimiento de sutura es de 0.00048 m con un sobrepico en 2.25s debido al cambio brusco que se observa en la trayectoria de la Figura 4.17. En todos los casos el error de estado estacionario tiende a ser menor a 0.000007 m ($7\mu\text{m}$).

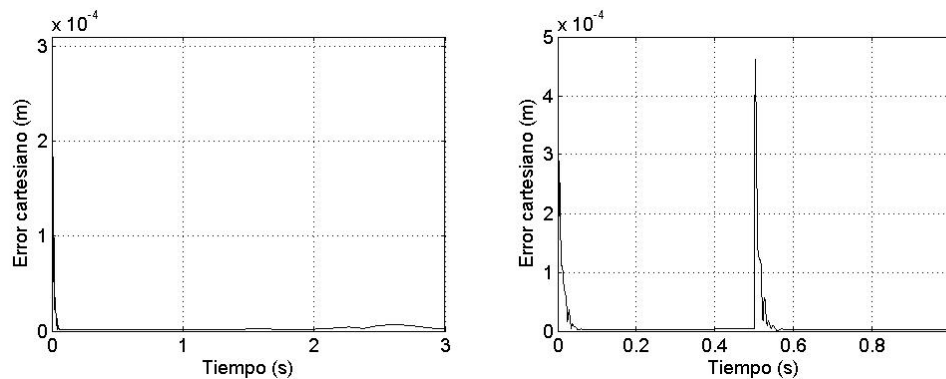


Figura 4.20. Error de seguimiento cartesiano para el círculo (izquierda) y el cambio de dirección (derecha).

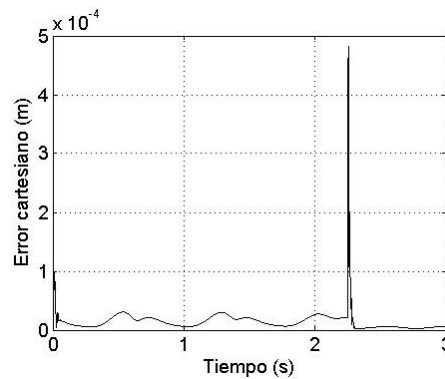


Figura 4.21. Error de seguimiento cartesiano para la trayectoria tridimensional.

4.3.2 Procesos de la Colecistectomía

Después de demostrar que con LapBot se pueden realizar movimientos complejos dentro del paciente, con un error muy pequeño de seguimiento y respetando el paso a través del trocar, se inicia la comprobación de las trayectorias necesarias para la realización de una colecistectomía, pero modificando el punto de incisión de acuerdo al modelo matemático del cuerpo humano desarrollado por Hanavan [74]. Este modelo determina que el abdomen de un hombre adulto promedio de 1,60 m de estatura y 60 Kg de peso, es similar a un paralelepípedo de 0,164 m x 0,224 m x 0,274 m.

También se tiene en cuenta los 0,15 m que se eleva el abdomen debido al gas de CO_2 [48]. La Figura 4.22 muestra un esquema aproximado del abdomen insuflado y las ubicaciones de las dos incisiones realizadas para introducir los instrumentos quirúrgicos.

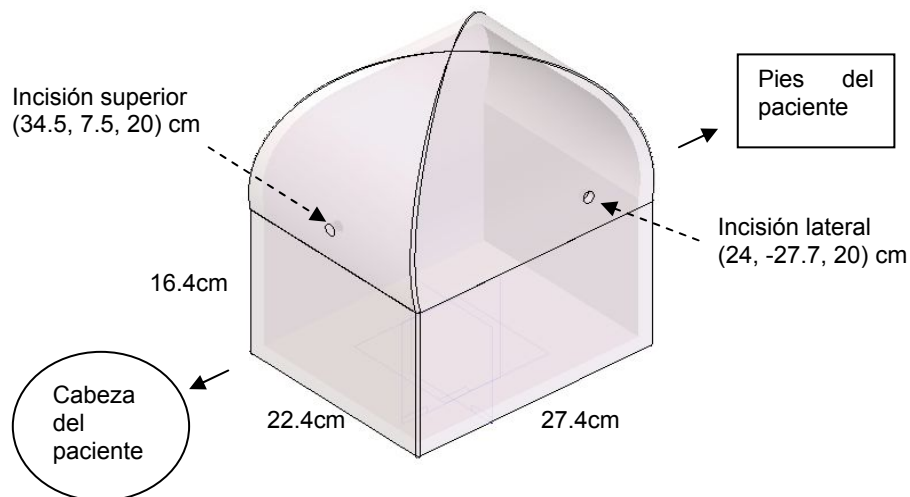


Figura 4.22. Modelo estilo caja del abdomen del paciente.

Las coordenadas de los puntos de incisión se calculan a partir de las bases de los robots que se colocan en los bordes de la camilla, en las mismas ubicaciones donde se encuentran el cirujano y el asistente de la Figura 4.2.

Para realizar el seguimiento de las trayectorias descritas en el ítem 4.1.2 usando el controlador CTC diseñado para un tiempo de muestreo de 0.001 s, fue necesario incrementar el número de muestras de dichas trayectorias empleando una interpolación lineal para predecir las muestras intermedias y de esta forma permitir que tuviesen más tiempo de simulación, acordes a los videos provistos por la Fundación Clínica Valle del Lili [70]. A continuación se describe el seguimiento de los procesos quirúrgicos de la colecistectomía.

1. Disección del triángulo de Calot.

- ♦ Para el instrumento I1, que es introducido por la incisión superior, se tiene un tiempo de simulación de 1.2 s. El seguimiento y el error cartesiano con máximo en 1×10^{-5} m se presentan en la Figura 4.23.

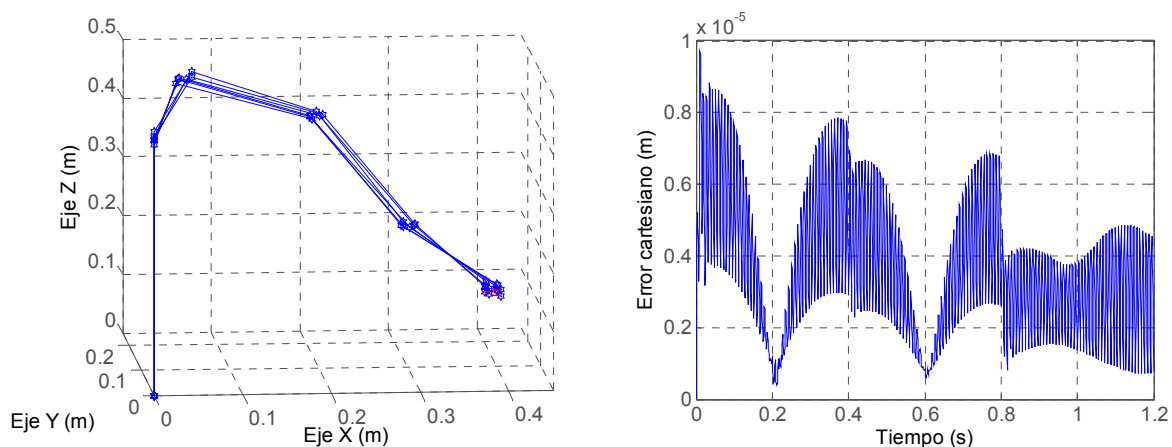


Figura 4.23. Seguimiento y error cartesiano del instrumento I1 en la disección del triángulo de Calot.

- ♦ Para el instrumento I2, introducido por la incisión lateral, se tiene un tiempo de simulación de 2.712 s, el error cartesiano máximo (4.8×10^{-4} m) y el seguimiento es mostrado en la Figura 4.24. Este movimiento del instrumento se repite en las otras trayectorias, así que solamente se mostrarán los movimientos del instrumento I1 en las demás trayectorias.

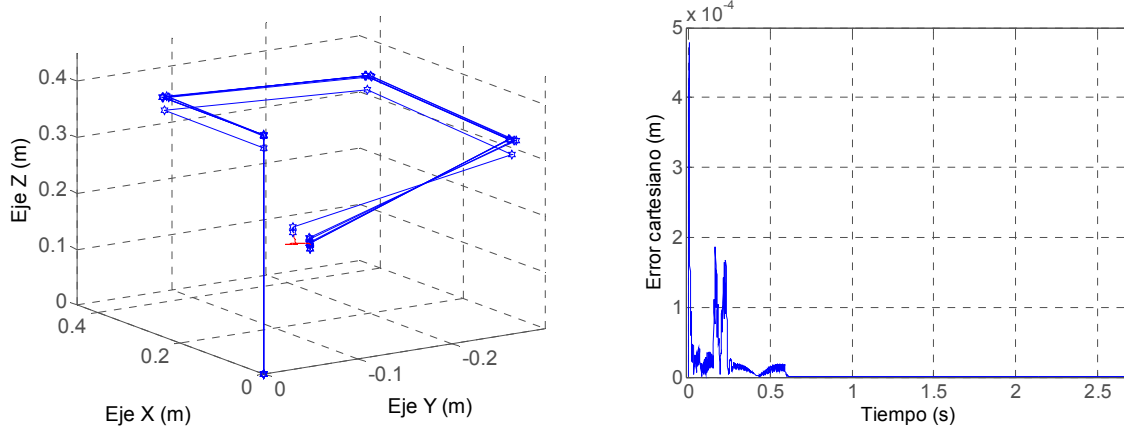


Figura 4.24. Seguimiento y error cartesiano del instrumento I2 en la disección del triángulo de Calot.

2. Sección del conducto cístico.

Para I1, se tiene un tiempo de simulación de 0.552 s y un error cartesiano máximo de 4.5×10^{-4} m (Figura 4.25).

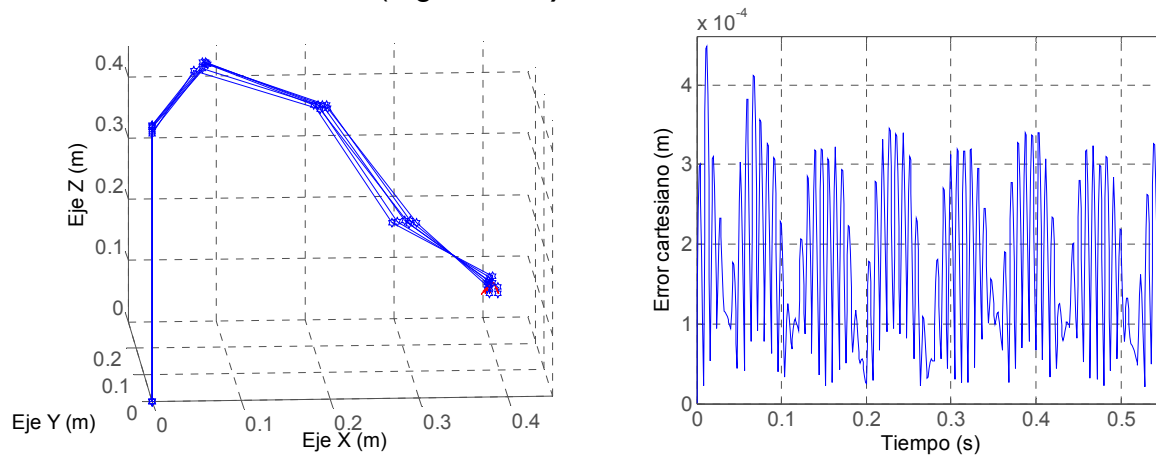


Figura 4.25. Seguimiento y error cartesiano para la sección del conducto cístico.

3. Sección de la arteria cística.

El tiempo de simulación también es de 0.552 s, el error cartesiano tiene su máximo en 1.85×10^{-4} m. El seguimiento y el error se presentan en la Figura 4.26.

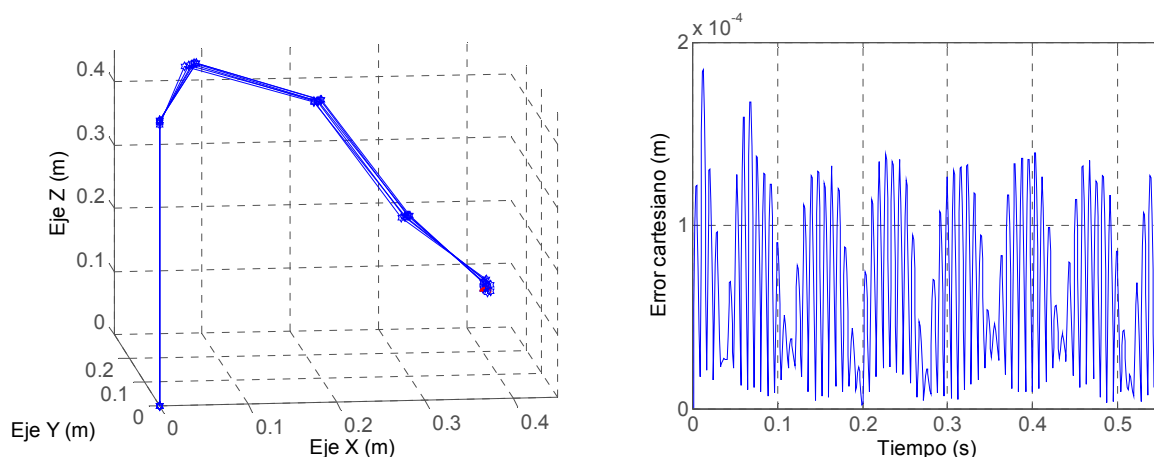


Figura 4.26. Seguimiento y error cartesiano para la sección de la arteria cística.

4. Disección de la vesícula del lecho hepático.

El tiempo de simulación es de 3.832 s y el error cartesiano tiene su máximo en 2.2×10^{-4} m. El seguimiento y el error se presentan en la Figura 4.27.

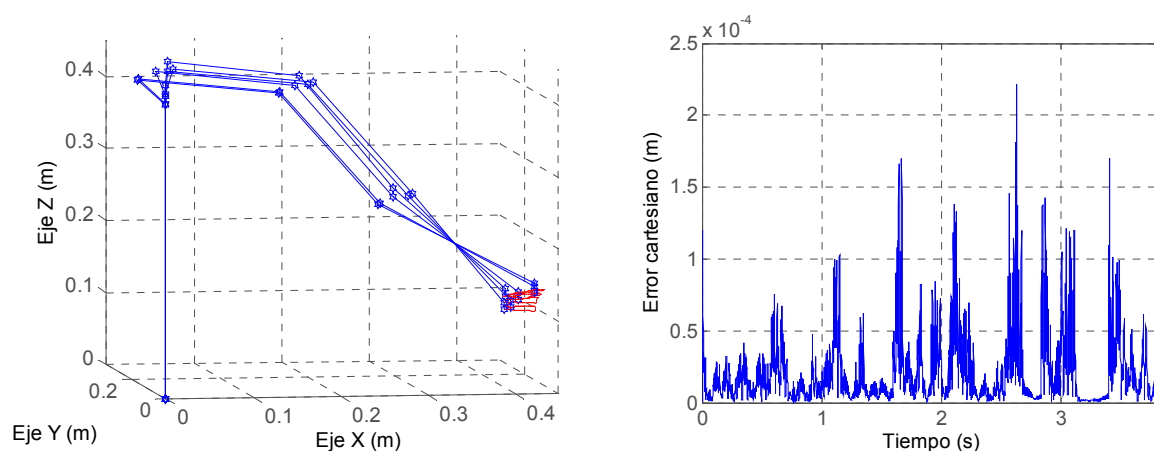


Figura 4.27. Seguimiento y error cartesiano para la disección de la vesícula.

Con estas pruebas se demuestra la posibilidad de realizar el seguimiento de trayectorias quirúrgicas de una colecistectomía laparoscópica con LapBot. De otra parte, al aplicar el control por par calculado a la dinámica de LapBot se logran errores cartesianos de seguimiento menores a medio milímetro, respetando el paso por un punto inmóvil que representa la incisión en el abdomen del paciente. Las oscilaciones en las figuras del error son despreciables comparadas con los movimientos y el temblor de un cirujano humano real.

5. Simulación Tridimensional de la Colecistectomía con LapBot

Para enlazar las simulaciones hechas en Matlab/Simulink®, los cuerpos 3D para LapBot, los modelos matemáticos y lograr una comprobación visual tridimensional de la funcionalidad del robot para realizar una colecistectomía laparoscópica, se desarrolló un ambiente de simulación 3D que usa los modelos y el controlador CTC diseñado para seguir las trayectorias de la cirugía.

5.1 Simulador Tridimensional (3D)

Al realizar el análisis de los requerimientos del proyecto y de posibles proyectos posteriores, se determina que la herramienta computacional de simulación en 3D debe cumplir con las siguientes características mínimas:

- De distribución libre, que permita usarlo sin pagar licencia, de tal forma que el sistema a desarrollar quede libre de compromisos con empresas desarrolladoras de este tipo de software.
- Que soporte modelos matemáticos no lineales tanto del robot como de las restricciones espaciales así como del controlador, los cuales han sido implementados todos en el paquete computacional Matlab®.
- Que permita generar un archivo ejecutable para que el ambiente virtual de Lapbot se pueda instalar y utilizar en cualquier computador sin necesidad de un programa 3D base especial.
- Que tenga la opción de crear una interfaz de usuario amigable para que sea comprensible y manipulable por cualquier profesional en el área de la salud o de la ingeniería.
- Que genere imágenes de buen realismo que permita incorporar diferentes puntos de vista de la cirugía, imágenes de órganos humanos, camillas y estructuras articuladas como la de Lapbot.

- Con la posibilidad de importar los cuerpos ya construidos en el programa CAD SolidEdge®.
- Que soporte las 2 principales API (Interfaz de Programación de Aplicaciones) gráficas que son OpenGL (*Open Graphics Library*) y Direct3D, de tal forma que pueda ser usado en ambiente Windows o Linux.
- Suficiente soporte bibliográfico, tutoriales e información para desarrollar una aplicación de simulación en 3D.

Se hizo un análisis de algunos motores de simulación 3D libres sugeridos en la base de datos [75], para escoger uno que cumpliera con las características ya mencionadas. En la Tabla 5.1 se presenta la comparación usando información de [75] - [82].

Motor 3D	Interactúa con Matlab	Uso de cuerpos de SolidEdge	Soporta las 2 API	Interfaz realista/amigable	Lenguajes de programación	Ayuda bibliográfica
Ogre3D	Sí	Sí	Sí	Muy Buena	C++, VB.NET, C#, Phyton, Java.	Muy Buena
Open Sim	Sí	No es claro	OpenGL	Baja	C++	Baja
Open Scene Graph	Sí	No es claro	OpenGL	Buena	C++, Java, Python.	Regular
Blender Game	Posible	Sí	OpenGL	Muy Buena	C++ o Python en Blender.	Muy Buena
Globe3D	Posible	Posible	OpenGL	Buena	Ada	Regular
Java3D	Sí	Posible	No, usa Java API	Regular	Java	Buena
Irrlicht	Sí	Posible	Sí	Buena	C++, C#, VB.NET.	Baja

Tabla 5.1. Comparación de motores de simulación 3D.

Todos los motores para manejo de imágenes 3D permiten realizar un archivo ejecutable por medio de su lenguaje de programación. Aquellos lugares de la tabla donde aparece la palabra *posible* se refieren a que la documentación así lo hace entender, pero no lo puede asegurar el autor. Algunos ítems de la tabla se colocaron *no es claro* porque así se aprecia en la bibliografía.

Se escogió Ogre3D (*Object-Oriented Graphics Rendering Engine*) [78], que se programa en Visual C++, porque presenta las mejores características para este proyecto respecto a los otros motores analizados.

Ogre3D es un motor, es decir, un conjunto de librerías y utilidades, para realizar manipulación y renderizado gráfico (representación de una imagen en una pantalla de computador), por medio de un lenguaje de programación. Soporta diferentes formatos de imágenes, lenguajes de programación y ofrece un buen soporte bibliográfico. Se han desarrollado más de 40 juegos de video con Ogre y utilidades para generar pequeños proyectos de simulación [78].

5.2 Construcción del Ambiente de Simulación en 3D

Para realizar la construcción del ambiente tridimensional usando Ogre3D y Visual C++, primero se exportaron los cuerpos construidos en SolidEdge® en formato STL y se pasaron a formato *mesh* (necesario para imágenes 3D en Ogre), luego se enlazaron por programación para formar el robot, se construyó una camilla, la caja que simula el abdomen insuflado del paciente de la Figura 4.22, se diseñó una habitación con distribución de luces en Ogre y finalmente se diseñó una pequeña interfaz gráfica de usuario (GUI**).

5.2.1 Enlace de los Cuerpos de LapBot

Los cuerpos descritos en el ítem 3.3.5, se exportaron al formato *.mesh*, usando el software Blender y un importador adicional que se instala sobre éste. Así los cuerpos se vuelven imágenes manejables en Ogre, el enlace de las imágenes se hace usando nodos como se muestra en los siguientes comandos:

```
Entity *r1 = mSceneMgr->createEntity("r1", "r1.mesh");
r1 ->setMaterialName("MatExterno");
nodo1 = r1->createChildSceneNode(Vector3( 0, 0, 0));
nodo1 ->attachObject(r1);

Entity *Brazo = mSceneMgr->createEntity("Brazo", "Brazo.mesh");
Brazo ->setMaterialName("MatExterno");
nodoBrazo = nodo1->createChildSceneNode(Vector3( 0,0,17.12));
nodoBrazo ->attachObject(Brazo);

Entity *Antebrazo = mSceneMgr->createEntity("Antebrazo",
"Antebrazo.mesh");
Antebrazo ->setMaterialName("MatExterno");
nodoAntebrazo = nodoBrazo->createChildSceneNode(Vector3(9.84,0,-0.3));
nodoAntebrazo->attachObject(Antebrazo);
```

Primero se crea una entidad (`createEntity`) que contiene la imagen del cuerpo (`r1.mesh`), luego se asigna el material que se desea para dicha imagen (`setMaterialName`), y se crea un nodo de movimiento que tiene una relación padre-hijo con el nodo del cuerpo anterior (`createChildSceneNode`). Se tiene la posibilidad

** Por sus siglas en inglés *Graphical User Interface*.

de ubicar el nuevo nodo en una posición específica deseada (`Vector3`) respecto al nodo padre. Por último se enlaza la entidad con el nodo para que se muevan juntos en la simulación (`attachObject`).

Es necesario enlazar los cuerpos de la forma padre-hijo para que al mover una articulación padre las hijas también se muevan, como sucede en la realidad, de esta forma se logra que el robot sea una pieza completa, conjunta y articulada.

Este procedimiento se realiza con cada cuerpo hasta armar completamente el robot. En algunos cuerpos también es necesario rotar un nodo usando los comandos `pitch`, `yaw`, `roll`, dependiendo del eje con respecto al cual se hace la rotación.

5.2.2 Enlace de LapBot con la Habitación y la Camilla

Se construye en Ogre una habitación modular que contendrá al robot, la camilla y la caja abdomen, con las siguientes líneas de comando:

```
//HABITACION
Entity *piso = mSceneMgr->createEntity("piso",mSceneMgr->PT_CUBE);
piso ->setMaterialName("Mpiso");
SceneNode *npiso = nodoRoom->createChildSceneNode(Vector3(0,-5.1,0));
npiso->attachObject(piso);
npiso->scale(0.2, 0.0015, 0.2);

Entity *pared = mSceneMgr->createEntity("pared",mSceneMgr->PT_CUBE);
pared->setMaterialName("mpared");
SceneNode *npared = npiso->createChildSceneNode(Vector3(50,3000,0));
npared->attachObject(pared);
npared->roll(Degree(90));
npared->scale(0.6,1,1);

Entity *pared2 = pared->clone("pared2");
SceneNode *npared2=npiso->createChildSceneNode(Vector3(-50,3000,0));
npared2->attachObject(pared2);
npared2->roll(Degree(90));
npared2->scale(0.6,1,1);

Entity *pared3 = pared->clone("pared3");
SceneNode *npared3=npiso->createChildSceneNode(Vector3(0,3000,-50));
npared3->attachObject(pared3);
npared3->pitch(Degree(90));
npared3->scale(1,1,0.6);

Entity *pared4 = pared->clone("pared4");
SceneNode *npared4 = npiso->createChildSceneNode(Vector3(0,3000,50));
npared4->attachObject(pared4);
npared4->pitch(Degree(90));
npared4->scale(1,1,0.6);

Entity *techo = pared->clone("techo");
SceneNode *ntecho = npiso->createChildSceneNode(Vector3(0,6000,0));
ntecho->attachObject(techo);
```

Como se observa se realiza primero el piso, luego las paredes y por último el techo; a cada entidad se le puede asignar un material diferente para formar el diseño de habitación que se desee, en el caso de este proyecto se seleccionó piso de madera y lo demás color mármol. La creación de materiales se realiza con estos comandos:

```
MaterialPtr mpared = MaterialManager::getSingleton().create("mpared",
ResourceGroupManager::DEFAULT_RESOURCE_GROUP_NAME);
TextureUnitState* tpared = mpared->getTechnique(0)->getPass(0)->
createTextureUnitState("marble_7.jpg");
```

También se colocaron tres luces, de tal forma que se puedan observar los objetos, de lo contrario todo estaría negro. El comando para crear una luz es:

```
Light *l1 = mSceneMgr->createLight("Luz1");
```

Luego, se construyó en SolidEdge® una camilla grande de dimensiones estándar con 0.6 m de ancho, 2 m de largo, altura respecto al piso de 0.6 m, y un cuerpo estilo caja para representar el abdomen insuflado del paciente con las respectivas incisiones. Estos cuerpos también fueron exportados a Ogre para complementar el ambiente.

El código en C++ se muestra a continuación:

```
//Mesa
Entity *Mesa = mSceneMgr->createEntity("Mesa", "mesa.mesh");
Mesa ->setMaterialName("MatMesa");
SceneNode *nodoMesa = nodoRoom->
createChildSceneNode( Vector3(-1.325,-2.65,0));
nodoMesa->attachObject(Mesa);
nodoMesa->pitch( Degree(-90));
nodoMesa->scale(escuerpo, escuerpo, escuerpo);

//Abdomen
Entity *Abdomen = mSceneMgr->createEntity("Abdomen", "abdomen.mesh");
Abdomen ->setMaterialName("MatAbdomen");
SceneNode *nodoAbdomen = nodoMesa->
createChildSceneNode(Vector3( 11.8, -8.8, 0));
nodoAbdomen->attachObject(Abdomen);
nodoAbdomen->roll( Degree(90));
```

Los robots se adicionan al final como hijos de la mesa y se les adiciona los ejes coordenados como puede verse en la Figura 5.1.

5.2.3 Interfaz Gráfica de Usuario (GUI) y Ambiente 3D

Para que el usuario tenga control sobre algunas características del ambiente 3D se desarrolla una pequeña interfaz gráfica en 2D usando el *Crazy Eddie's GUI System* (CEGUI), que viene integrado con Ogre3D como una librería adicional. Esta librería está principalmente diseñada para capturar las interacciones del usuario con el GUI.

Pero para diseñar los botones, listas y demás objetos del GUI es mejor usar el *CELayoutEditor* y luego programar los objetos en C++, que fue lo que se hizo. La Figura 5.1 muestra el GUI desarrollado.



Figura 5.1. Interfaz de usuario para el ambiente 3D.

La interfaz permite seleccionar entre las cuatro trayectorias mencionadas en el capítulo anterior que son: Disección de Calot, Conducto Cístico, Arteria Cística, Lecho Hepático, y se adicionó otra que es la Colecistectomía y contiene a las demás en secuencia.

Se puede escoger realizar la simulación teniendo en cuenta sólo los modelos geométricos (lo cual es más veloz en la simulación pero no representa la realidad) o usar los modelos dinámicos en conjunto con el controlador CTC diseñado (más lento en simulación pero con la dinámica real del robot).

Se pueden tener diferentes puntos de vista predeterminados del ambiente 3D, como son: Vista Superior, Vista Frontal, Vista Lateral y Vista Interna, en una mini-imagen.

El botón “Aceptar” se activa cuando el usuario ha escogido una trayectoria, mientras se desarrolla la simulación se vuelve a desactivar para no crear conflictos internos en la simulación o bloqueos de la aplicación.

El botón “Salir” permite cerrar la ventana que contiene el ambiente, el servidor de Matlab® y el GUI. El ambiente 3D con los dos LapBot, el abdomen, el GUI y los demás elementos se presentan en la Figura 5.2.

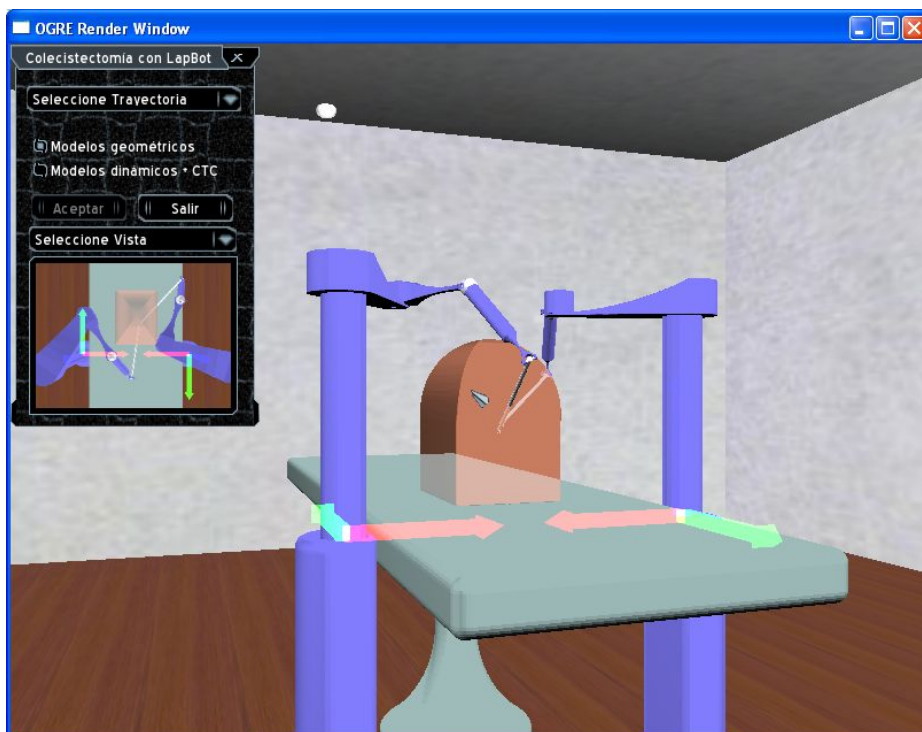


Figura 5.2. Ambiente 3D con dos LapBot y la interfaz de usuario.

Cuando el robot se comienza a mover, se puede observar la trayectoria que está siguiendo dentro del abdomen porque se remarca con un color azul oscuro, como se observa en la Figura 5.3.

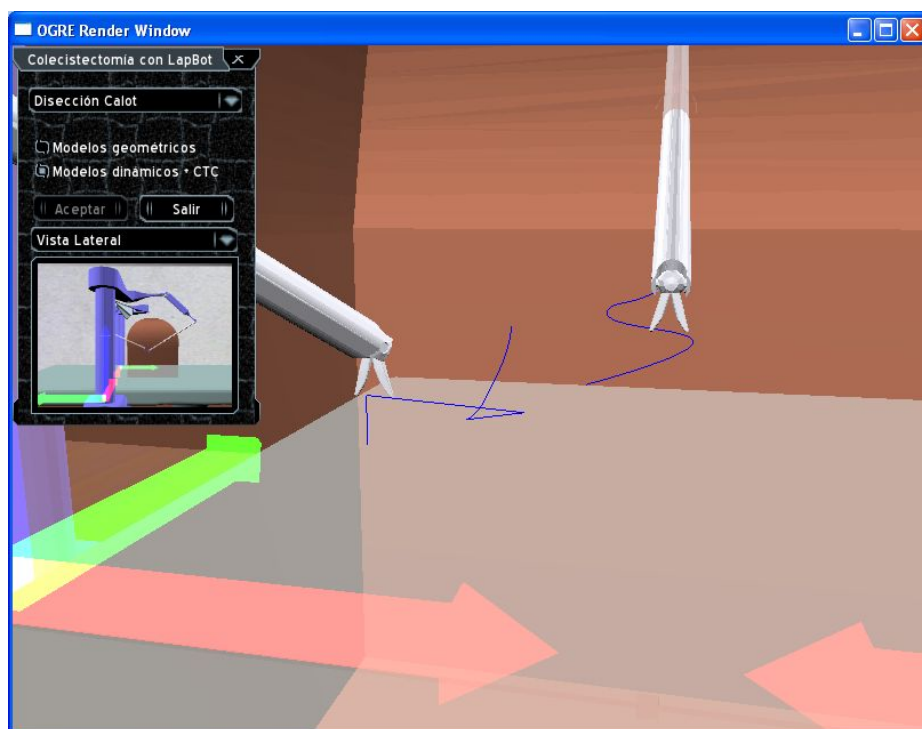


Figura 5.3. Seguimiento 3D de la trayectoria Disección de Calot.

Además al dar clic derecho en el ratón del computador se puede rotar el punto de vista principal y mover con las teclas a, s, d, w (éstas son las que vienen por defecto programadas en Ogre) que permiten mayor versatilidad en el movimiento de la cámara. Para más detalles sobre la instalación y el manejo de la aplicación leer el Anexo B.

5.3 Programación del Ambiente 3D

La programación del ambiente se realiza en Visual C++, siguiendo los tutoriales y las ayudas que se obtienen de los foros que se exponen en [78]. Además se usan las funciones y los diagramas realizados en Matlab®, generando una conexión Usuario-Servidor, donde el usuario es el ambiente 3D y el servidor es Matlab®.

El diagrama de flujo general del programa se presenta en la Figura 5.4.

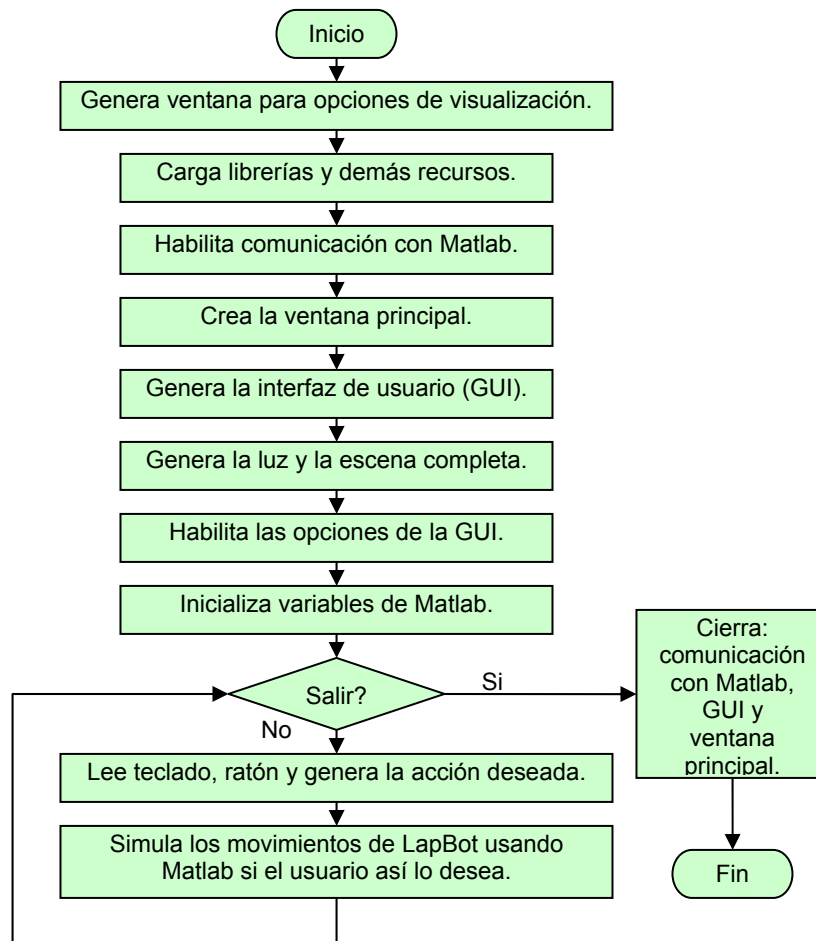


Figura 5.4. Diagrama de flujo del programa principal.

La primera ventana que abre cuando se corre el ejecutable del programa (Figura 5.5), es automáticamente generada por Ogre para que el usuario tenga la posibilidad de escoger, entre otras cosas, la API gráfica o subsistema de generación gráfica (OpenGL o Direct3D), la resolución de la ventana, el dispositivo físico para aceleración de video y si desea que el simulador se muestre en pantalla completa.

Las librerías principales que se deben cargar para que funcione el simulador son: las de Ogre, CEGUI, Matlab® y Visual C++, esta última se incluye de forma automática por Visual C++, las demás se incluyen con los comandos:

```
#include "ExampleApplication.h"           //Ogre
#include "engine.h"                       //Matlab
#include <CEGUI.h>
#include <OIS.h>
#include <CEGUISystem.h>
#include <CEGUISchemeManager.h>
#include <OgreCEGUIRenderer.h>
```

Todos los archivos necesarios están incluidos con el instalador del Simulador LapBot excepto los de Matlab®, es decir que se debe tener instalado en el computador el software Matlab 6.5 o superior para poder usarlo como servidor y emplear sus librerías.



Figura 5.5. Pantalla de opciones de Ogre.

Para utilizar a Matlab® como servidor se habilita la comunicación (Ver Anexo B) y se abre una ventana de comandos (invisible para el usuario) usando el comando:

```
ep = engOpen(NULL);
```

Luego se generan la ventana principal con la escena (que contiene el robot y demás elementos), la cámara, la luz y el GUI como se presenta en la Figura 5.2. A continuación se habilitan los botones, listas y demás opciones del GUI y se inicializan todas las variables, incluidas las usadas por Matlab®, de tal forma que el usuario se apodera de las actividades que debe desarrollar el simulador. Para capturar las opciones del usuario en cada *frame* (generación de imagen) se usa la clase `ExampleFrameListener` prediseñada en Ogre para dicha función.

Para cada opción que tiene el usuario se realiza una subrutina, por ejemplo, para la selección de vista se realiza la siguiente línea de comandos:

```
bool vista_selec(const CEGUI::EventArgs &e)
{
    CEGUI::ListboxItem *ts=vista->getSelectedItem();
    size_t indice = vista->getItemIndex(ts);

    switch (indice)
    {
        case 0: //superior
            minicam->setPosition(-8, 70, 50);
            minicam->lookAt(-8, 70, -100);
            minicam->setOrientation(Quaternion(Degree(-90),Vector3(1,0,0)));
            minicam->setFOVy(Degree(45));
            break;
        case 1: //frontal
            minicam->setPosition(-5, -70, 180);
            minicam->lookAt(-5, -70, 0);
            minicam->setFOVy(Degree(45));
            break;
        case 2: //lateral
            minicam->setPosition(160, -70, 50);
            minicam->lookAt(0, -70, 50);
            minicam->setFOVy(Degree(45));
            break;
        case 3: //interna
            minicam->setPosition(-6, -70, -80);
            minicam->lookAt(-6, -70, 80);
            minicam->setFOVy(Degree(14));
            minicam->pitch(Degree(-8));
            break;
        default:
            {}
    }
    return true;
}
```


Esta rutina asigna una nueva localización a la cámara pequeña (*minicam*) y diferente ángulo de visión dependiendo del punto de vista que se desee.

Para realizar los movimientos del robot, se deben cargar los datos de simulación de Matlab® y usar unos comandos especiales, por ejemplo:

```
engEvalString(ep, "dato=qr1(k,2);");  
movim = mxGetPr(engGetVariable(ep, "dato"));  
nodoBrazo->setOrientation(Quaternion(Radian(*movim),Vector3(0,0,1)));
```

El comando `engEvalString` hace que Matlab® corra la línea `dato=qr1(k,2);` como si se estuviese trabajando en él, luego se adquiere el resultado en la variable `movim` usando el comando `mxGetPr(engGetVariable(ep, "dato"))`; y por último se transfiere el dato al nodo que se desea mover, en este caso el `nodoBrazo`.

De esta forma se pueden mover todas las articulaciones del robot generando una dinámica conjunta en el ambiente 3D.

Cuando el usuario pulsa el botón de salir en el GUI, la ventana principal se cierra y con ella el GUI, deshabilitándose además la comunicación con Matlab® al usar el comando:

```
engClose(ep);
```

Los demás detalles de la programación se pueden observar en los archivos digitales de computador adjuntos a este documento.

Con la simulación 3D se puede comprobar que LapBot es capaz de seguir los movimientos de una colecistectomía laparoscópica respetando el punto de incisión y, que sus dimensiones son apropiadas para el ambiente de trabajo y las tareas que debe realizar. Así, se abre la posibilidad de comenzar a construir un prototipo físico y continuar con la investigación en robótica quirúrgica.

6. Conclusiones

En este proyecto se realizó el diseño estructural de un robot asistente para operaciones de laparoscopia llamado LapBot, haciendo un estudio previo de los principales robots asistentes en cirugías mini-invasivas que existen en el mundo. LapBot tiene nueve grados de libertad, seis motorizados que le permiten posicionar y orientar el instrumento quirúrgico dentro del abdomen del paciente y, tres no motorizados para respetar el paso a través de un punto fijo que representa la incisión.

Dicho punto fijo en el espacio cartesiano genera ecuaciones matemáticas a través de un análisis vectorial, que en conjunto con métodos matemáticos tradicionales en robótica, permitieron obtener el modelo geométrico inverso (MGI) y el modelo geométrico directo (MGD) de LapBot. Para el MGI se propusieron dos soluciones, una por métodos numéricos y otra por resolución de ecuaciones, esta última es más óptima a nivel computacional ya que no necesita iterar como los métodos numéricos.

También se hallaron los modelos dinámicos de LapBot, tanto directo como inverso, usando el método de Newton-Euler y la aplicación de computador Symoro+, de tal forma que las respuestas de simulación del robot fuesen más parecidas a la realidad que si solamente se usaran los modelos geométricos.

Para poder comenzar las simulaciones de LapBot y sus modelos fue necesario determinar numéricamente los parámetros geométricos, es decir las dimensiones del robot, que se logró analizando el ambiente de trabajo donde debe estar ubicado (quirófano) y la labor que debe cumplir. Los parámetros dinámicos, como inercias y masas de los cuerpos que conforman a LapBot, también son necesarios y para esto se usó el paquete de diseño asistido por computador SolidEdge®.

Después de comprobar que el robot tiene la capacidad geométrica de pasar a través de un punto fijo (incisión), debía comprobarse que también tenía la capacidad dinámica de hacerlo en una cirugía real como la colecistectomía (extracción de una vesícula biliar enferma). Para esta comprobación se obtuvieron trayectorias de procesos quirúrgicos de una colecistectomía y se sintonizó una estrategia de control por par calculado (CTC) usando diversas trayectorias cartesianas de prueba, fue simulado en Matlab® obteniéndose errores de seguimiento en la colecistectomía

menores a medio milímetro, corroborándose así las potencialidades de LapBot para realizar operaciones de laparoscopia.

Para mejorar la visualización de los resultados se construyó e implementó un ambiente tridimensional, que le permite al usuario observar los movimientos de LapBot y el trayecto que debe seguir, desde diferentes puntos de vista, usando el programa para representación gráfica Ogre3D, que interactúa con Matlab® en configuración usuario-servidor y se programa en Visual C++.

LapBot fue construido en el ambiente 3D con los cuerpos diseñados previamente en SolidEdge®, pero además se diseñaron otros elementos como la camilla, la habitación, la base del robot y la caja en forma de abdomen insuflado con incisiones, correspondiente a una persona adulta de 1,60 m de altura y 60 kg de peso.

La interfaz gráfica de usuario (GUI) que se diseñó permite al usuario tener acceso a diferentes opciones dentro del ambiente 3D, como escoger movimientos de una colecistectomía real que deba seguir el robot, puntos de vista fijos adicionales que desee observar y la forma en que Matlab® debe realizar la simulación y el análisis matemático, ya sea sólo con los modelos geométricos o con los modelos dinámicos más el control CTC. Se observó que el simulador es más veloz cuando se usan los modelos geométricos, pero más realista cuando usa los modelos dinámicos.

Los principales problemas que se tuvieron en el proyecto tienen que ver con la definición de la estructura de LapBot, el MGI y su solución, la escogencia de la herramienta computacional para desarrollar el ambiente 3D y la programación del mismo, de tal forma que permitiera interactuar con Matlab® y usar los cuerpos construidos en SolidEdge®.

Una posible continuación del proyecto sería desarrollar un simulador para el entrenamiento de cirujanos, que implemente modelos de órganos humanos y que permita incluir otras cirugías laparoscópicas, haciendo uso de interfaces hápticas. Además se podría desarrollar un robot similar a LapBot para controlar la cámara de video (porta endoscopio) necesaria en una cirugía mini-invasiva, para luego construir un primer prototipo físico y proceder a realizar pruebas reales.

Referencias Bibliográficas

- [1] J. PATIÑO, y J. CERVANTES, *Cirugía Laparoscópica y Torascópica*. México: McGrawHill Interamericana, 1997.
- [2] G. KELLING, "Mitteilung zur Benutzung des Oesophagoscops", *Allgemeine Medicinische Central-Zeitung*, 1906.
- [3] 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.
- [4] S. KRUT, "Introduction to Medical Robotics", presentado en el VI Seminario de Automática, Univesidad del Cauca, Colombia, 2006.
- [5] S. SALINAS, "Robot para Cirugía Laparoscópica", presentado en el 1er Seminario de Ingeniería Biomédica, Universidad Autónoma de Occidente, Colombia, 2006.
- [6] S. DHARIA and T. FALCONE, "Robotics in Reproductive Medicine", *Fertility and Sterility*, vol. 84, pp. 1-11, 2005.
- [7] B. SANCHEZ, C. MOHR, J. MORTON, B. SAFADI, R. ALAMI and M. CURET, "Comparison of Totally Robotic Laparoscopic Roux-en-Y Gastric Bypass and Traditional Laparoscopic Roux-en-Y Gastric Bypass", *Surgery for Obesity and Related Diseases*, vol. 1, pp. 549-554, 2005.
- [8] S. MEJÍA, A. ARANA, C. ARANGO, V. PÉREZ, H. POSADA y A. TORRES, "Kirubot: Brazo Robótico Ayudante en Cirugía", *Revista IATREIA, Resúmenes Orales*, vol. 14, no. 4, pp. 254, 2001.
- [9] A. JARAMILLO, Grupo de Automática y Robótica, Pontificia Universidad Javeriana de Cali, <http://gar.puj.edu.co/Investigacion/Grupos/GAR/TmpGar>. Consultado: diciembre de 2006.

- [10] HAPTICA, www.haptica.com. Consultado: octubre de 2006.
- [11] H. DELINGETTE and N. AYACHE, "Hepatic Surgery Simulation", *Communications of the ACM*, vol. 48, pp. 31-36, no. 2, 2005.
- [12] S. SUZUKI, N. SUZUKI, M. HASHIZUME, Y. KAKEJI, K. KONISHI, A. HATTORI and M. HAYASHIBE, "Tele-training Simulation for the Surgical Robot System "Da Vinci"", *International Congress Series*, vol. 1268, pp. 86-91, 2004.
- [13] L. ADHAMI, E. COSTE-MANIÈRE and J. BOISSONNAT, "Planning and Simulation of Robotically Assisted Minimale Invasive Surgery". *3th international conference of Medical Image Computing and Computer-Assisted Intervention*, vol. 1935, Springer, 2000.
- [14] R. GINHOUX, "Compensation des Mouvements Physiologiques en Chirurgie Robotisée par Commande Predictive", PhD thesis, Université Louis Pasteur Strasbourg I, 2003.
- [15] R. SATAVA, "History of Robotic Surgery", *World Electronic Book of Surgery*, www.websurg.com/robotics/history.php. Consultado: marzo de 2007.
- [16] H. KANG and J. T. WEN, "Robotic Knot Tying in Minimally Invasive Surgeries", *IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 1421-1426, Suiza, 2002.
- [17] "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.
- [18] C. CUELLAR, E. MORENO, C. CASAS, J. DE FRANCISCO, H. ORDOÑEZ y R. RIVEROS, "Colecistectomía Laparoscópica: Primera Experiencia en Colombia", *Revista Colombia*, vol. 6, pp. 5-12, 1991.
- [19] H. ABAÚNZA, C. CASAS y N. ZUNDEL, "Colecistectomía Laparoscópica: Trabajo Cooperativo de la Sociedad Colombiana de Cirugía", *Revista Colombia*, vol. 7, pp. 2-10, 1992.
- [20] A. OLLERO, *Robótica: Manipuladores y Robots Móviles*. Barcelona: Alfaomega, 2001.
- [21] W. KHALIL and E. DOMBRE, *Modeling, Identification and Control of Robots*. London: Kogan Page Science, 2002.

-
- [22] S. SCHWARTZ y J. HUNTER, *Principios de Cirugía*, 7a ed, Vol. II. México: McGrawHill Interamericana, 1999.
- [23] H. C. JACOBÆUS, "Laparo- und Thorakoskopie", *Beitr Klin Tuberk*, 1912.
- [24] A. KATZIR, *Lasers and Optical Fibers in Medicine*. New York: Academic Press, 1993.
- [25] E. MÜHE, "Die Erste Cholecystectomy Durch das Laparoskop", *Langenbecks Arch Klin Chir*, vol. 369, pp. 804, 1986.
- [26] P. MOURET, "How I Developed Laparoscopic Cholecystectomy", *Annals of the Academy of Medicine*, vol. 25, no. 5, pp. 744-747, 1996.
- [27] DIMEDA INSTRUMENTE, "Dimeda Surgical Instruments". <http://www.dimeda.de>. Consultado: febrero 2008.
- [28] A. CUSCHIERI, "The Spectrum of Laparoscopic Surgery", *World Journal Surgery*, vol. 16, pp. 1089-1097, 1992.
- [29] J. YARMUCH, H. SCHULTE, G. CABALLERO, C. PINTO, J. SILVA and H. SILVA, "Colecistectomía Laparoscópica, 1203 Casos con Colangiografía Ultraselectiva", *Revista Española de Enfermería*, vol. 86, no. 2, pp. 587-581, 1994.
- [30] E. VALLE, "Experiencia en Cirugía Laparoscopia en HEODRA en el Periodo de Enero de 2003 a Febrero de 2004", Esp. tesis, Universidad Nacional Autónoma de Nicaragua, León, Nicaragua, 2004.
- [31] ACOG. *Surgical Alternatives to Hysterectomy in the Management of Leiomyomas*. American College of Obstetricians and Gynecologists. Boletín no. 16, pp. 1-10, 2000.
- [32] J. TRUSSELL, E. GUILBERT and A. HEDLEY, "Sterilization Failure, Sterilization Reversal, and Pregnancy after Sterilization Reversal in Quebec", *Obstetrics and Gynecology Journal*, vol. 101, pp. 677-684, 2003.
- [33] M. MEINERO y G. MELOTTI, *Cirugía Laparoscópica*. Buenos Aires: Editorial Médica Panamericana S.A., 1994.
- [34] M. MENON, A. TEWARI, B. BAIZE, B. GUILLONNEAU and G. VALLANCIEN, "Prospective Comparison of Radical Retropubic Prostatectomy and Robot-Assisted Anatomic Prostatectomy: The Vattikuti Urology Institute Experience", *Urology Journal*, vol. 60, pp. 864-868, 2002.

- [35] R. POSSIK, E. FRANCO, D. PIRES, D. WOHNATH and E. FERREIRA, "Sensitivity, Specificity, and Predictive Value of Laparoscopy for the Staging of Gastric Cancer and for the Detection of Liver Metastases", *Cancer*, vol. 58, no. 1, pp. 1-6, 1986.
- [36] D. CRAIN, J. ROBERTS and C. AMLING, "Practice Patterns in Vasectomy Reversal Surgery: Results of a Questionnaire Study Among Practicing Urologists", *Journal Urology*, vol. 171, no. 1, 2004.
- [37] S. GIBILISCO, *The McGraw-Hill Illustrated Encyclopedia of Robotics & Artificial Intelligence*. New York: McGraw-Hill, 1994.
- [38] Y. KWON, J. HOU, E. JONCKHEERE and S. HAYALL, "A Robot with Improved Absolute Positioning Accuracy for Ct Guide, Stereoscopic Brain Surgery", *IEEE Transaction, Biomedical Engineering*, vol. 35, no. 2, pp. 153-161, 1988.
- [39] B. DAVIES, R. HIBBERD, M. COPTCOAT and J. WICKHAM, "A Surgeon Robot Prostatectomy—A Laboratory Evaluation", *Journal of Medical Engineering Technology*, vol. 13, pp. 273-277, 1989.
- [40] M. MAROHN and E. HANLY, "Twenty-first Century Surgery Using Twentyfirst Century Technology: Surgical Robotics", *Current Surgery*, vol. 61, no. 5, pp. 466-473, 2004.
- [41] W. KORB, R. MARMULLA, J. RACZKOWSKY, J. MÜHLING and S. HASSFELD, "Robots in the Operating Theatre—Chances and Challenges", *International journal of Oral and Maxillofacial Surgeons*, vol. 33, pp. 721-732, 2004.
- [42] RH. TAYLOR, "Robots as Surgical Assistants: Where We Are, With We Are Tending, and How to Get There", *Lecture Notes in Artificial Intelligence*, vol. 1211, pp. 3-11, 1997.
- [43] C. ARROYO, "Cirugía Robótica", *Revista: Elementos: Ciencia y Cultura*, vol. 12, pp. 13-17, 2005.
- [44] 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.
- [45] INTUITIVE SURGICAL, "The Da Vinci Surgical System", 2005. www.intuitivesurgical.com/products/davinci_surgicalsystm/index.htm. Consultado: octubre de 2006.

-
- [46] INTUITIVE SURGICAL, "In Vivo Accessories for Minimally Invasive Robotic Surgery", U. S. Patent 7125403B2, 2006.
- [47] COMPUTER MOTION Inc, "Method and Apparatus for Performing Minimally Invasive Surgical Procedures", U. S. Patent 6436107B1, 2002.
- [48] 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.
- [49] NARANG MEDICAL LIMITED, "Operating Room Tables". www.narang.com/surgical_medical_hospital_laboratory_products/index.php. Consultado: febrero de 2008.
- [50] 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.
- [51] MAXON, "Hoja de Datos del Motor Maxon EC-Powermax 30". www.maxonmotor.com. Consultado: junio 2008.
- [52] FAICO, "Instrumental Quirúrgico". www.faico.com.ar/servicio.htm. Consultado: marzo 2008.
- [53] 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.
- [54] 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.
- [55] ENDOVIA MEDICAL Inc, "Articulated Apparatus for Telemanipulator System", U. S. Patent 6692485B1, 2004.
- [56] M. MICHELIN, "Contribution a la Commande de Robots pour la Chirurgie Mini-Invasive", PhD. thesis, Université de Montpellier II, France, 2004.
- [57] 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.

- [58] J. MARESCAUX, J. LEROY and M. GAGNER, "Transatlantic Robot-Assisted Telesurgery", *Nature*, vol. 413, pp. 379-380, 2001.
- [59] 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.
- [60] A. J. MADHANI, "Design of Teleoperated Surgical Instruments for Minimally Invasive Surgery", PhD. thesis, Massachusetts Institute of Technology (MIT), United States, 1998.
- [61] O. SCHNEIDER, J. TROCCAZ, O. CHAVANON and D. BLIN, "Padyc: A Synergistic Robot for Cardiac Puncturing", *2000 IEEE International Conference on Robotics and Automation*, pp. 2883-2888, 2000.
- [62] 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.
- [63] 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.
- [64] T. ORTMAIER, "Motion Compensation in Minimally Invasive Robotic Surgery", Universität Manchen, PhD. thesis, Germany, 2002.
- [65] R. PAUL, *Robot Manipulators: Mathematics, Programming and Control*. Cambridge: MIT Press, 1982.
- [66] W. KHALIL and D. CREUSOT, "Symoro+: A System for the Symbolic Modelling of Robots", *Robotica*, vol. 15, pp. 153-161, 1997.
- [67] M. LATARJET y A. RUIZ, *Anatomía Humana*, 3ra ed., Vol. II. Buenos Aires: Editorial Medica Panamericana, 1996.
- [68] A. MARIN, "Técnica: La Colectectomía Laparoscópica Estándar". <http://drmarin.galeon.com/cctlap.htm>. Consultado: junio de 2008.
- [69] H. ZHOU, Y. GUO, X. YU, S. BAO, J. LIU, Y. ZHANG, Y. REN and Q. ZHENG, "Clinical Characteristics of Remote Zeus Robot-Assisted Laparoscopic Cholecystectomy: A Report of 40 Cases", *World Journal of Gastroenterology*, vol. 12, no. 16, pp. 2606-2609, 2006.

-
- [70] O. MORA y B. GARCÉS, “Estudio del Desempeño de un Robot Puma en Operaciones de Laparoscopia”, Ing. tesis, Universidad del Cauca, Popayán, Colombia, 2007.
- [71] L. SCIAVICCO and B. SICILIANO, *Modeling and Control of Robot Manipulators*. Singapore: McGrawHill, 1996.
- [72] J. SLOTIN and W. LI, *Applied Nonlinear Control*. Englewood Cliffs: Prentice Hall, 1991.
- [73] C. SMITH, “Fundamentals of Control Theory”, *Chemical Engineering*, vol. 86, pp. 11-39. 1979.
- [74] E. HANAVAN, “A Mathematical Model of the Human Body”, Aerospace Medical Research Laboratories, Wright-Patterson Air Force Base, Ohio, United States, 1964.
- [75] F. SKILTON, Ed., “DevMaster's Game and Graphics Engines Database”, www.devmaster.net. Consultado: agosto de 2008.
- [76] SILICON GRAPHICS Inc., “GLOBE_3D Stands for GL Object Based Engine for 3D”, 2008. globe3d.sourceforge.net. Consultado: agosto de 2008.
- [77] N. GEBHARDT, “Irrlicht, Lightning Fast Realtime 3D Engine”. irrlicht.sourceforge.net. Consultado: septiembre de 2008.
- [78] TORUS KNOT SOFTWARE Ltd, “OGRE: Object-Oriented Graphics Rendering Engine”. www.ogre3d.org. Consultado: enero de 2009.
- [79] D. JUNG, “OpenSim, A 3D Simulator for Autonomous Robots”. opensimulator.sourceforge.net. Consultado: septiembre de 2008.
- [80] SILICON GRAPHICS Inc., “OpenGL: Open Graphics Library”. www.opengl.org. Consultado: septiembre de 2008.
- [81] M. GERKE and A. BISCHOFF, “A VRML and Java-based Online Robot Simulation Tool”. http://prt.fernuni-hagen.de/~bischoff/research/bischoff_jb99.pdf. Consultado: septiembre de 2008.
- [82] THE MATHWORKS Inc., “External Interfaces”, Matlab Documentation, 2005.

Anexo A. Parámetros Inerciales de Base para LapBot

A continuación se describen las ecuaciones de los parámetros inerciales de base reagrupados para LapBot que se obtuvieron al usar el software Symoro y cómo se calcularon numéricamente después de construir los cuerpos del robot en el software CAD SolidEdge.

A.1. Expresiones Simbólicas

Al introducir la Tabla 3.1 de parámetros geométricos de LapBot en el software Symoro y usar las opciones del mismo para reducir el número de parámetros inerciales de base por medio de reagrupación, se obtienen las siguientes expresiones:

$$M1R = M1+M10+M2+M3+M4+M5+M6+M7+M8+M9$$

$$ZZ2R = IA2+D1^2*(M10+M3+M4+M5+M6+M7+M8+M9)+ZZ2$$

$$\begin{aligned} ZZ3R = & D2^2*(M10+M4+M5+M6+M7+M8+M9)+ZZ3+2*D2*MX4*\cos(\theta4)+ \\ & \cos(\theta4)^2*YY4+2*XY4*\cos(\theta4)*\sin(\theta4) - \\ & 2*D2*(MY4-(M10+M5+M6+M7+M8+M9)*R1-MZ5)*\sin(\theta4)+ \\ & (XX4+YY5+2*R1*MZ5+(M10+M5+M6+M7+M8+M9)*R1^2)*\sin(\theta4)^2 \end{aligned}$$

$$\begin{aligned} MX3R = & D2*(M10+M4+M5+M6+M7+M8+M9)+MX3+MX4*\cos(\theta4)- \\ & (MY4-(M10+M5+M6+M7+M8+M9)*R1-MZ5)*\sin(\theta4) \end{aligned}$$

$$MY3R = MY3-MZ4$$

$$XX5R = XX5-YY5+YY6$$

$$ZZ5R = ZZ5+YY6$$

$$MY5R = MY5+MZ6$$

$$XX6R = XX6-YY6+YY7+2*MZ7*R2+(M10+M7+M8+M9)*R2^2$$

$$ZZ6R = ZZ6 + (M10 + M7 + M8 + M9) * R2^2 + 2 * R2 * MZ7 + YY7$$

$$MY6R = MY6 - (M10 + M7 + M8 + M9) * R2 - MZ7$$

$$XX7R = XX7 - YY7 + YY8$$

$$ZZ7R = ZZ7 + YY8$$

$$MY7R = MY7 + MZ8$$

$$XX8R = 2 * MZ10 * R3 + M10 * R3^2 + XX8 + YY10 - YY8 + YY9$$

$$ZZ8R = ZZ8 + 2 * MZ10 * R3 + M10 * R3^2 + YY10 + YY9$$

$$MY8R = MY8 - MZ10 + MZ9 + M10 * R3$$

$$XX9R = XX10 + XX9 - YY10 - YY9$$

$$XY9R = XY10 + XY9$$

$$XZ9R = -MX10 * R3 + XZ10 + XZ9$$

$$YZ9R = -MY10 * R3 + YZ10 + YZ9$$

$$ZZ9R = ZZ10 + ZZ9$$

$$MX9R = MX10 + MX9$$

$$MY9R = MY10 + MY9$$

Donde los términos del lado izquierdo del igual son el resultado del reagrupamiento de los términos a la derecha. Cada término de la derecha se debe hallar por medio de un análisis físico de cada cuerpo asociado a cada articulación del robot, que depende de las dimensiones, la forma y el material con el cual está construido el cuerpo.

A.2. Valores Numéricos

El análisis físico de los cuerpos lo realiza el software SolidEdge, en el cual se diseña, construye y obtiene los valores inerciales y de masas necesarios de cada cuerpo. Por ejemplo, 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 así sucesivamente.

Para LapBot se construyeron los cuerpos que se muestran en la sección 3.3.5, de los cuales, los cuatro primeros contienen motores Maxon que cumplen la función de mover las articulaciones. Los datos de inercia (I_{a_j}) y masa de estos motores se hallaron en la hoja de datos [51]. La Tabla A.1 muestra un resumen de los valores numéricos no reagrupados que entrega SolidEdge.

j	XX_j	YY_j	ZZ_j	MX_j	MY_j	MZ_j	M_j	I_{a_j}
1	0	0	0	0	0	0	3.639	2e-6
2	0	0	1.02e-2	0	0	0	1.693	2e-6
3	0	0	1.3e-3	13.9e-3	6e-9	0	0.46	2e-6
4	2.4e-6	3e-6	2.4e-6	0	7.e-5	0	0.015	2e-6
5	13.2e-4	13.2e-4	1.3e-5	0	0	-9.32e-3	0.078	2e-6
6	1.38e-7	1.45e-7	1.63e-8	0	0	0	0.003	2e-6
7	2.8e-3	2.8e-3	8.9e-7	0	0	-13.6e-3	0.089	2e-6
8	18e-10	21e-10	14e-10	0	2e-7	0	0.45e-3	2e-6
9	2e-10	2e-10	2e-10	0	0	4.5e-8	0.13e-3	2e-6
10	21e-9	22.6e-9	1.9e-9	0	0	2.96e-6	0.49e-3	2e-6

Tabla A.1. Valores de los parámetros inerciales para cada cuerpo de LapBot.

Los valores para XY_j , XZ_j y YZ_j , no se presentan en la Tabla A.1 porque son nulos para todas las articulaciones, esto se logra construyendo los cuerpos de tal forma que tengan simetría en la mayoría de los ejes coordenados y por tanto el cuerpo esté bien balanceado.

Al usar estos valores numéricos en conjunto con las expresiones de la sección A.1 y un valor de $\theta_4 = \pi/4$, se obtienen los valores que se presentan en la Tabla 3.4 del texto. De esta forma se reduce el número de parámetros no nulos de 51 en la Tabla A.1 a 22 en la Tabla 3.4, esto reduce enormemente la carga computacional en la simulación de la dinámica del robot.

Anexo B. Manual de Usuario de la Aplicación 3D para LapBot

La aplicación que se presenta a continuación fue desarrollada para la visualización tridimensional de 2 robots para laparoscopia llamados LapBot, que siguen las trayectorias de una colecistectomía (cirugía para la extracción de una vesícula biliar enferma) pasando a través de dos puntos de incisión.

B.1. Requerimientos

Para que la aplicación funcione correctamente, el computador donde se va a instalar debe tener:

1. Windows XP, NT, 2000. (No se ha probado aun con Windows Vista ni con Linux).
2. Matlab 6.5 o superior (No necesita *toolbox*).
3. OpenGL o Direct3D (Normalmente OpenGL siempre está instalado).
4. 150MB o más de espacio en el disco duro.
5. 1GB o más de memoria RAM.
6. Procesador de 1.8GHz o mejor.
7. Preferiblemente una tarjeta aceleradora de video (funcionan mejor con Direct3D).

Los requerimientos se han determinado de acuerdo a las pruebas que se han realizado en algunos computadores, sin embargo es posible que funcione en un computador con características similares.

B.2. Instalación

Después de verificar los requerimientos del sistema, proceda a:

1. Ejecutar el archivo con el nombre Setup_LapBot y seguir las instrucciones. La aplicación y los archivos necesarios se instalarán en una carpeta creada con el nombre GUI_LapBot_V1, ubicada en la dirección de su preferencia.

2. Ejecutar Matlab, en el menú hacer clic en *File* y luego en *Set Path...*, aparecerá una ventana como la que se observa en la Figura B.1, hacer clic en *Add Folder*, se abrirá otra ventana en donde debe buscar la ubicación de la carpeta GUI_LapBot_V1 que se creó en el paso anterior, hacer clic en Aceptar, debe aparecer la dirección de la carpeta en la sección *MATLAB search path*, esto le permite a Matlab acceder a las funciones de la aplicación de LapBot. Para finalizar haga clic en *Save* y luego cierre Matlab.

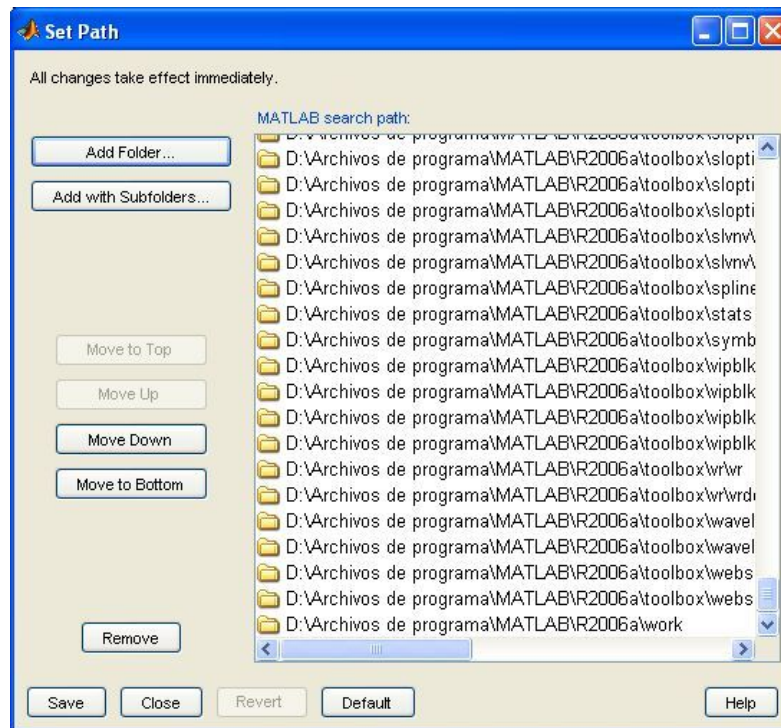


Figura B.1. Ventana para adicionar un *Path* en Matlab.

3. Para que Matlab funcione como servidor se debe registrar en el computador como tal, esto se hace en la ventana del Símbolo del Sistema en Windows, que se encuentra en Inicio → Todos los Programas → Accesorios → Símbolo del Sistema. Se abrirá una ventana negra como la de la Figura B.2.

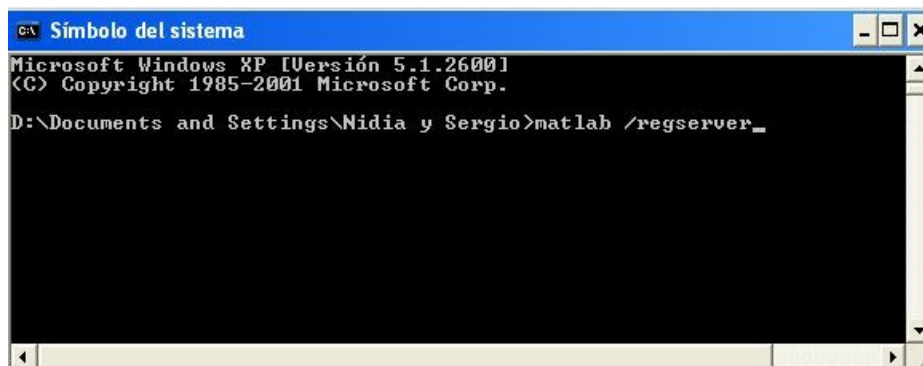


Figura B.2. Ventana del Símbolo del sistema en Windows.

En esta ventana se debe digitar el comando *matlab /regserver* y luego dar enter. Cerrar la ventana.

4. De esta forma ya debe poder ejecutar el archivo ejecutable GUI_LapBot_V1.exe que se encuentra en la carpeta instalada GUI_LapBot_V1/bin/release.

Si tiene alguna dificultad comuníquese con el autor sergiosalinas@unicauca.edu.co.

B.3. Manejo

Al ejecutar la aplicación, se abrirá la ventana de la Figura 5.5, que le permitirá seleccionar, entre otras cosas, la API gráfica o subsistema de generación gráfica (sugiero Direct3D), la resolución de la ventana, el dispositivo físico para aceleración de video y si desea que el simulador se muestre en pantalla completa.

Después de hacer clic en dicha ventana aparecerá el ambiente 3D con los robots, la camilla, un emulador de abdomen insuflado y la habitación que los contiene, como aparece en la Figura 5.2.

El sistema le permite interactuar con el ambiente 3D por medio del teclado y ratón del computador, un resumen de funciones de teclas y demás se presenta en la Tabla B.1.

<i>Tecla/ Ratón</i>	<i>Función</i>
A	Mueve la cámara hacia la izquierda
S	Mueve la cámara hacia atrás
D	Mueve la cámara hacia la derecha
W	Mueve la cámara hacia delante
Q	Sale de la aplicación
F	Muestra u oculta la velocidad de simulación
Esc	Sale de la aplicación
PgUp	Mueve la cámara hacia arriba
PgDown	Mueve la cámara hacia abajo
Flecha arriba	Mueve la cámara hacia delante
Flecha abajo	Mueve la cámara hacia atrás
Flecha derecha	Rota la cámara a la derecha
Flecha izquierda	Rota la cámara a la izquierda
Clic derecho	Activa o desactiva la rotación de la cámara con el ratón
Clic izquierdo	Activa cualquier opción que se escoja del GUI

Tabla B.1. Funciones de las teclas y el ratón en la aplicación 3D.

La interfaz de usuario (GUI), permite seleccionar cinco trayectorias: Disección de Calot, Conducto Cístico, Arteria Cística, Lecho Hepático y Colecistectomía que contiene a las demás en secuencia (Ver Figura 5.1).

Se puede escoger realizar la simulación teniendo en cuenta sólo los modelos geométricos (lo cual es más veloz en la simulación pero no representa la realidad) o usar los modelos dinámicos en conjunto con un controlador CTC (más lento en simulación pero con la dinámica real del robot).

En la parte inferior se pueden tener diferentes puntos de vista predeterminados del ambiente 3D, como son: Vista Superior, Vista Frontal, Vista Lateral y Vista Interna.

El botón “Aceptar” se activa cuando el usuario ha escogido una trayectoria para hacer la simulación, al dar clic en éste comienza el movimiento de los robots siguiendo la trayectoria que el usuario escogió. Mientras se desarrolla la simulación se desactiva.

El botón “Salir” cierra la ventana y sale de la aplicación.

Nombre de archivo: Monografía
Directorio: E:\Sergio\Proyecto_Robot para cirugía\Anteproyecto y tesis
Plantilla: D:\Documents and Settings\Nidia y Sergio\Datos de programa\Microsoft\Plantillas\Normal.dot
Título: PROPUESTA TESIS DE MAESTRIA EN INGENIERÍA
Asunto:
Autor: mosquera
Palabras clave:
Comentarios:
Fecha de creación: 06/03/2009 22:12:00
Cambio número: 55
Guardado el: 16/05/2009 22:19:00
Guardado por: Nidia y Sergio
Tiempo de edición: 2.513 minutos
Impreso el: 16/05/2009 22:29:00
Última impresión completa
Número de páginas: 108
Número de palabras: 24.534 (aprox.)
Número de caracteres: 134.937 (aprox.)